handbook of statistics 41

Conceptual Econometrics Using R

Edited by
Hrishikesh D. Vinod
C.R. Rao

Handbook of Statistics
Volume 41
Conceptual Econometrics Using R

Handbook of Statistics

Series Editor

C.R. Rao
C.R. Rao AIMSCS, University of Hyderabad Campus,
Hyderabad, India

Handbook of Statistics

Volume 41

Conceptual Econometrics Using R

Edited by

Hrishikesh D. Vinod
Fordham University, Bronx, NY, United States

C.R. Rao
AIMSCS, University of Hyderabad Campus, Hyderabad, India

North-Holland

An imprint of Elsevier

North-Holland is an imprint of Elsevier
Radarweg 29, PO Box 211, 1000 AE Amsterdam, Netherlands
The Boulevard, Langford Lane, Kidlington, Oxford OX5 1GB, United Kingdom

Notices
Knowledge and best practice in this field are constantly changing. As new research and experience broaden our understanding, changes in research methods, professional practices, or medical treatment may become necessary.

Practitioners and researchers must always rely on their own experience and knowledge in evaluating and using any information, methods, compounds, or experiments described herein. In using such information or methods they should be mindful of their own safety and the safety of others, including parties for whom they have a professional responsibility.

To the fullest extent of the law, neither the Publisher nor the authors, contributors, or editors, assume any liability for any injury and/or damage to persons or property as a matter of products liability, negligence or otherwise, or from any use or operation of any methods, products, instructions, or ideas contained in the material herein.

ISBN: 978-0-444-64311-7
ISSN: 0169-7161

For information on all North-Holland publications
visit our website at https://www.elsevier.com/books-and-journals

Publisher: Zoe Kruze
Acquisition Editor: Sam Mahfoudh
Editorial Project Manager: Peter Llewellyn
Production Project Manager: Vignesh Tamil
Cover Designer: Mark Rogers
Typeset by SPi Global, India

Contents

This page intentionally left blank

Contributors

Numbers in Parentheses indicate the pages on which the author's contributions begin.

Kris Boudt (193), Department of Economics, Ghent University, Ghent; Solvay Business School, Vrije Universiteit Brussel, Brussel, Belgium; School of Business and Economics, Vrije Universiteit Amsterdam, Amsterdam, The Netherlands

Sebastián Cano-Berlanga (281), Universitat Autònoma de Barcelona and CREIP, Catalonia, Spain

Manfred Deistler (145), TU Wien, Vienna, Austria

Jean-Marie Dufour (3), Department of Economics, McGill University, Montréal, QC, Canada

Alexios Galanos (193), Amazon, Seattle, WA, United States

José-Manuel Giménez-Gómez (281), Universitat Rovira i Virgili and CREIP, Tarragona, Spain

Chirok Han (119), Professor of Economics at Korea University, Seoul, Republic of Korea

Yong Li (81), Hanqing Advanced Institute of Economics and Finance, Renmin University of China, Beijing, China

Julien Neves (3), Cornell University, Ithaca, NY, United States

Scott Payseur (193), Amazon, Seattle, WA, United States

Peter C.B. Phillips (119), Sterling Professor of Economics and Professor of Statistics at Yale University, New Haven, CT, United States

Joaquim Jose Santos Ramalho (245), Department of Economics and BRU-IUL, Instituto Universitário de Lisboa (ISCTE-IUL), Lisboa, Portugal

Wolfgang Scherrer (145), TU Wien, Vienna, Austria

Cori Vilella (281), Universitat Rovira i Virgili and CREIP, Tarragona, Spain

Hrishikesh D. Vinod (33), Fordham University, Bronx, NY, United States

Kenneth D. West (65), Department of Economics, University of Wisconsin-Madison, Madison, WI, United States

Jun Yu (81), School of Economics and Lee Kong Chian School of Business, Singapore Management University, Singapore, Singapore

Tao Zeng (81), School of Economics, Academy of Financial Research, and Institute for Fiscal Big-Data & Policy of Zhejiang University, Zhejiang University, Zhejiang, China

Zifeng Zhao (65), Department of Information Technology, Analytics and Operations, Mendoza College of Business, University of Notre Dame, Notre Dame, IN, United States

Eric Zivot (193), Amazon; University of Washington, Seattle, WA, United States

Preface

As with earlier volumes in this series, volume 41 of *Handbook of Statistics* with the subtitle "Conceptual Econometrics Using R" and a companion volume 42 with the subtitle "Financial, Macro and Micro Econometrics Using R" provide state-of-the-art information on important topics in Econometrics, a branch of Economics concerned with quantitative methods. This handbook covers a great many conceptual topics of practical interest to quantitative scientists, especially in Economics and Finance.

The book has uniquely broad coverage with all chapter authors providing practical R software tools for implementing their research results. Despite some overlap, we divide the chapters into three parts. We list the three parts while retaining the nine chapter numbers as:

1. *Statistical Inference*

- **(1)** Jean-Marie Dufour and Julien Neves propose new simulation-based exact finite sample inference methods implemented in their R package MaxMC. Its toolkit includes overcoming nuisance parameters.
- **(2)** Hrishikesh D. Vinod discusses new tools from his R package "generalCorr" for inferring exogeneity and causal paths from passively observed data, citing applications in diverse fields.
- **(3)** Zifeng Zhao provides new tools for bias reduction in h-step ahead forecast when h is large.
- **(4)** Yong Li, Jun Yu and Tao Zeng review MCMC based frequentist inference methods which avoid using Bayes Factors.

2. *Multivariate Models*

- **(5)** Peter C. B. Phillips and Chirok Han provide efficient R tools for dynamic panel data models including difference GMM, system GMM, and within group estimation.
- **(6)** Wolfgang Scherrer and Manfred Deistler provide tools for avoiding inappropriate VAR models by using multivariate ARMA and state-space models. They consider identification issues, Hankel matrices and reduced rank regressions.
- **(7)** Kris Boudt, Alexios Galanos, Scott Payseur, and Eric Zivot survey multivariate GARCH models for large data sets and outlier-robust MGARCH and evaluations of cokurtosis and coskewness.

3. *Miscellaneous Topics*
   - **(8)** Joaquim Ramalho considers estimation and inference for direct and marginal effects in regressions where the dependent variable is restricted to the range [0,1], such as when it is a ratio.
   - **(9)** Sebastián Cano-Berlanga, José-Manuel, Giménez-Gómez and Cori Vilella discuss cooperative game theory including transferable utility, "punctual solutions," voting power index and "claims problems" while providing tools for sharing of benefits among interdependent (economic) agents.

All chapters are authored by distinguished researchers. Most senior authors have received professional honors, such as being elected "Fellows" of the *Journal of Econometrics* or of the *Econometric Society*.

The intended audience is not only students, teachers, and researchers in various industries and sciences but also profit and nonprofit business decision makers and government policymakers. The wide variety of applications of statistical methodology should be of interest to researchers in all quantitative fields in both natural and social sciences and engineering.

A unique feature of this volume is that all included chapters provide not only a review of the newer theory but also describe ways of implementing authors' new ideas using free R software. Also, the writing style is user-friendly and includes descriptions and links to resources for practical implementations on a *free* open source R, allowing readers to not only use the tools on their own data but also providing a jump start for understanding the state of the art. Open source allows reproducible research and opportunity for anyone to extend the toolbox.

According to a usage dating back to Victorian England, the phrase "The three R's" describes basic skills taught in schools: Reading, wRiting, and aRithmetic. In the 21st century, we should add R software as the *fourth R*, which is fast becoming an equally basic skill. Unfortunately, some economists are continuing to rely on expensive copyrighted commercial software which not only needs expensive updating but also hides many internal computational algorithms from critical public evaluation for robustness, speed, and accuracy. Users of open source software routinely work with the latest updated versions. This saves time, resources, and effort needed in deciding whether the improvements in the latest update are worth the price and arranging to pay for it.

In teaching undergraduate statistics classes one of us (Vinod) introduces students to R as a convenient calculator, where they can name numerical vector or matrix objects for easy manipulation by name. Starting with the convenience of not having to use Normal or Binomial tables, students begin to appreciate and enjoy the enormous power of R for learning and analyzing quantitative data.

There are over 14,686 free R packages, contributed and maintained by researchers from around the world, which can be searched at https://mran.microsoft.com/packages. In short, R has a huge and powerful ecosystem.

Students soon learn that if a statistical technique exists, there is most likely an R package which has already implemented it. The plotting functions in R are excellent and easy to use, with the ability to create animations, interactive charts and superimpose statistical information on geographical maps, including the ability to indicate dynamically changing facts. R is able to work with other programming languages including Fortran, Java, C++, and others. R is accessible in the sense that one does not need to have formal training in computer science to write R programs for general use.

For reviewing the papers we thank: Peter R. Hansen (University of North Carolina at Chapel Hill), Shujie Ma (University of California, Riverside), Aaron Smith (University of California, Davis), Tayyeb Shabbir (California State University Dominguez Hills, Carson, CA), Andreas Bauer (IMF Senior Resident Representative, New Delhi, India), José Dias Curto (ISCTE - Instituto Universitario de Lisboa, Portugal), Ruey S. Tsay (Booth School of Business, University of Chicago), Alessandro Magrini (University of Florence, Italy), Jae H. Kim (La Trobe University, Australia), In Choi (Sogang University, Korea), among others.

A common thread in all chapters in this handbook is that all authors of this volume have taken extra effort to make their research implementable in R. We are grateful to our authors as well as many anonymous researchers who have refereed the papers and made valuable suggestions to improve the chapters. We also thank Peter Llewellyn, Kari Naveen, Vignesh Tamilselvvanignesh, Arni S.R. Srinivasa Rao, Sam Mahfoudh, Alina Cleju, and others connected with Elsevier's editorial offices.

Hrishikesh D. Vinod
C.R. Rao

This page intentionally left blank

Part I

Statistical Inference

This page intentionally left blank

Chapter 1

Finite-sample inference and nonstandard asymptotics with Monte Carlo tests and R

Jean-Marie Dufour[a,*] **and Julien Neves**[b]
[a]*Department of Economics, McGill University, Montréal, QC, Canada*
[b]*Cornell University, Ithaca, NY, United States*
[*]*Corresponding author: e-mail: jean-marie.dufour@mcgill.ca*

Abstract
We review the concept of Monte Carlo test as a simulation-based inference procedure which allows one to construct tests with provably exact levels in situations where the distribution of a test statistic is difficult to establish but can be simulated. The number of simulations required can be extremely small, as low as 19 to run a test with level 0.05. We discuss three extensions of the method: (1) a randomized tie-breaking technique which allows one to use test statistics with discrete null distributions, without further information on the mass points; (2) an extension (maximized Monte Carlo tests) which yields provably valid tests when the test statistic depends on a (finite) number of nuisance parameters; (3) an asymptotic version which allows one to get asymptotically valid tests without any need to establish an asymptotic distribution. As the method is computer intensive, we describe an R package (**MaxMC**) that allows one to implement this type of procedure. A number of special cases and applications are discussed.

Keywords: R, Exact inference, Test level, Test size, Discrete distribution, Randomized tie-breaker, Nonstandard asymptotic distribution, Monte Carlo test, Maximized Monte Carlo, MMC, Simulated annealing, Genetic algorithm, Particle swarm, Bootstrap, Kolmogorov–Smirnov, Behrens–Fisher, Autoregressive model, Singular Wald test

1 Introduction

One of the central problems of statistical methodology consists in finding critical values for performing tests and building confidence sets. However, it is often the case that analytical formulae are not available. The dominant model where finite-sample methods are available is the classical linear model with fixed (or strictly exogenous) regressors and independently identically distributed

Handbook of Statistics, Vol. 41. https://doi.org/10.1016/bs.host.2019.05.001

(i.i.d.) Gaussian disturbances. As a result, statistical inference is typically based on large-sample approximations—which may be quite unreliable in finite samples—or bootstrapping. The bootstrap usually provides improvements over the use of limiting distributions, but it is also based on large-sample arguments through a demonstration that the asymptotic distribution of test statistic and the bootstrap distribution are identical in large samples; for reviews, see Efron (1982), Beran and Ducharme (1991), Efron and Tibshirani (1993), Hall (1992), Jeong and Maddala (1993), Vinod (1993), Shao and Tu (1995), Davison and Hinkley (1997), Chernick (1999), and Horowitz (1997).

In this paper, we focus on the method of *Monte Carlo tests*, which can deliver tests whose size (or level) is controlled in finite samples, without the need to establish analytically the distribution of the test statistic. The number of simulations required can be extremely small, as low as 19 to run a test with level 0.05. This feature allows one to use computationally expensive test statistics. We also emphasize that the approach can yield asymptotically valid tests in many situations where the limiting distribution of the test statistic is nonstandard or may not exist.

The technique of Monte Carlo tests actually predates bootstrapping and was originally suggested by Dwass (1957) in order to implement permutation tests. Another variant was later proposed by Barnard (1963), Hope (1968), and Birnbaum (1974), in view of performing tests based on test statistics with continuous distributions under the null hypothesis. Other early work on this method is available in Besag and Diggle (1977), Marriott (1979), Edgington (1980), Foutz (1980), Ripley (1981), Edwards (1985), Jöckel (1986), and Edwards and Berry (1987). These results typically rely on special assumptions on the form of the distributions of the test statistics (continuous or discrete in a specific way) and do not allow for the presence of nuisance parameters. A general theory of Monte Carlo tests is presented in Dufour (2006) and includes three main extensions. For other discussions and applications, see Kiviet and Dufour (1997), Dufour et al. (1998, 2003, 2010), Dufour and Kiviet (1998), Dufour and Khalaf (2001, 2002), Dufour and Farhat (2002), Dufour and Jouini (2006), Beaulieu et al. (2007, 2013), and Coudin and Dufour (2009).

The *first* extension allows for pivotal (nuisance-parameter-free) test statistics with otherwise arbitrary distributions—which may be continuous, discrete, or mixed (e.g., mixtures of continuous and discrete distributions). This is done in particular by exploiting a technique of randomized ranks for breaking ties in rank tests (see Hájek, 1969), which is simple to implement with exchangeable replications [as opposed to independent and identically distributed (i.i.d.) replications]. No information on the probabilities of mass points (if any) is needed.

The *second* extension involves test statistics whose null distribution depends on nuisance parameters. This is done by considering a simulated p -value function which depends on nuisance parameters (under the null hypothesis). Maximizing the latter with respect to the nuisance parameters then yields a test with provably exact level, irrespective of the sample size and the number replications used. We call such tests *maximized Monte Carlo* (MMC) tests.

The *third* extension is a simplified version of the latter where the nuisance-parameter space is replaced by a *consistent set estimator* of the nuisance parameters. Such set estimators can be built as soon as a consistent estimate is available, but also in other cases (e.g., when certain parameters are not identifiable under the null hypothesis). There is no need to establish the form of the asymptotic distribution or even its existence. The property of finite-sample validity is lost in this case, but the procedure remains asymptotically valid even in the presence of discontinuities in the asymptotic distribution. This includes in particular autoregressive models with unit (and explosive) roots and models where parameters may not be identified under the null hypothesis.

The method of Monte Carlo tests is intrinsically a simulation-based approach, so the use of computer-based simulation-based techniques is required. Further, MMC procedures maximize simulated p-value functions which are not smooth functions, because they are flat almost everywhere with jumps where they are not differentiable. In this chapter, we describe how this can be done using an R package called **MaxMC** (Dufour and Neves, 2019).

In Section 2, we review the theory of Monte Carlo tests for pivotal test statistics, in particular how finite-sample tests based on statistics with general (possibly discrete) distributions can be performed using this method. In Section 3, we describe an algorithm which implements it in R. Section 4 discusses the application of Monte Carlo tests to the problem of testing the equality of two distribution functions using a permutational Kolmogorov–Smirnov two-sample test, so that the null distribution is not continuous. In Section 5, we present the theory of maximized Monte Carlo tests from a finite-sample viewpoint, and in Section 6, we consider MMC tests based on consistent set estimators. The implementation of MMC tests in R is discussed in Section 7. Two examples are considered in Section 8: (1) the classic Behrens–Fisher problem of comparing the means of normal samples with different variances and (2) inference on an AR(p) model. Section 9 concludes.

2 Monte Carlo tests with continuous and discrete test statistics

In this section, we consider a test statistic $S := S(X^{(n)})$ for a null hypothesis H_0 such that the distribution of S under H_0 is uniquely determined, i.e., it does not involve unknown parameters. $X^{(n)}$ represents a sample of n observations. This distribution may not be easy to compute analytically, but can be simulated. We will now describe from a theoretical viewpoint how an exact test of H_0 based on S can be performed.

Let us denote by S_0 the statistic computed using the sample data, and by $S_1, \ldots, S_N$ a set of N i.i.d. (or exchangeable) replications of S under H_0. We consider critical regions of the form

$$R(c) = \left\{ S(X^{(n)}) \geq c \right\} \tag{1}$$

where c is a critical value for a test with level α, i.e.

$$\mathbb{P}[S(X^{(n)}) \geq c|H_0] \leq \alpha. \tag{2}$$

While useful, this separation of the sample space into a critical region and an acceptance region, requires one to find an appropriate critical value c, which furthermore only delivers a test at a given level α. It is often easier and more informative to consider the survival function

$$G[x] := \mathbb{P}[S(X^{(n)}) \geq x|H_0] \tag{3}$$

of the test statistic under H_0. If we evaluate this function at $x = S_0$, this yields the p-value

$$p(S_0) = G[S_0] \tag{4}$$

and the critical region

$$p(S_0) \leq \alpha\,. \tag{5}$$

It is then easy to see that

$$\mathbb{P}[p(S_0) \leq \alpha|H_0] \leq \alpha \tag{6}$$

with equality when the distribution of S_0 is continuous under H_0.

In many statistical and econometric applications, no analytical form is available to compute the p-value $p(S_0)$. The principle of Monte Carlo tests consists in replacing the function $p(x)$ by a simulation-based analog $\hat{p}_N(x)$. Though this may appear to be only an approximation—which may lead to level distortions—it turns out that replacing $p(x)$ by $\hat{p}_N(x)$ does allow one to perfectly control the level of the test in many situations of interest. Let $S_1, \ldots, S_N$ be i.i.d. replications of the test statistic under the null hypothesis, and set

$$\hat{p}_N(S_0) = \frac{N\hat{G}_N(S_0) + 1}{N + 1} \tag{7}$$

where $\hat{G}_N(x)$ is the sample survival function defined as

$$\hat{G}_N(x) = \frac{1}{N}\sum_{j=1}^{N} I_{[0,\,\infty)}(S_j - x) \quad \text{where} \quad I_A(z) = \begin{cases} 1 & \text{if} \quad z \in A \\ 0 & \text{if} \quad z \notin A \end{cases}. \tag{8}$$

If α is the desired level of the test, we reject the null hypothesis if $\hat{p}_N(S_0) \leq \alpha$. When the distribution of S is continuous under the null hypothesis and $\alpha(N + 1)$ is an integer, we have

$$\mathbb{P}_{H_0}[\hat{p}_N(S_0) \leq \alpha] = \alpha\,, \tag{9}$$

which means that the critical region

$$\hat{p}_N(S_0) \leq \alpha \tag{10}$$

has exact size α; for a proof, see Dufour (2006).

In addition, $N\hat{G}_N(x)$ is the number of simulated values of S larger than x. Therefore, we have the following relationship between $\hat{G}_N(S_0)$ and $\hat{R}_N(S_0)$, the sample rank of S_0 in S_0, S_1, …, S_N:

$$\hat{G}_N(S_0) = \frac{N+1-\hat{R}_N(S_0)}{N}. \tag{11}$$

We can thus rewrite the simulated p-value

$$\hat{p}_N(S_0) = \frac{N+2-\hat{R}_N(S_0)}{N+1}. \tag{12}$$

However, if the test statistic follows a discrete distribution, the presence of ties in the sample can modify the distribution of $\hat{p}_N(S_0)$ in a way that depends on the (unknown) distribution of the test statistic. Accordingly, Dufour (2006) provides a way of "breaking" ties by using randomly generated points drawn from a uniform distribution. Let $U_0, U_1, \ldots, U_N \overset{i.i.d}{\sim} U(0,1)$. Then for every S_i, we can create a pair (S_i, U_i). Using the following lexicographic ordering

$$(S_i, U_i) \leq (S_0, U_0) \Leftrightarrow \{S_i < S_0 \text{ or } (S_i = S_0 \text{ and } U_i \leq U_0)\}, \tag{13}$$

we can order the (S_i, U_i) pairs and define the randomized rank for S_0 as

$$\widetilde{R}_N(S_0) = \sum_{i=0}^{N} I[(S_i, U_i) \leq (S_0, U_0)] \tag{14}$$

where $I[(S_i, U_i) \leq (S_0, U_0)] = 1$ when the condition is satisfied, and $I[(S_i, U_i) \leq (S_0, U_0)] = 0$ otherwise. This yields a modified simulated p-value where $\hat{R}_N(S_0)$ is replaced by $\widetilde{R}_N(S_0)$, i.e.,

$$\widetilde{p}_N(S_0) = \frac{N+2-\widetilde{R}_N(S_0)}{N+1}. \tag{15}$$

If $\alpha(N+1)$ is an integer, the test is exact as with nonrandomized p -values, i.e.,

$$\mathbb{P}_{H_0}[\widetilde{p}_N(S_0) \leq \alpha] = \alpha \tag{16}$$

The function `MaxMC::pvalue` implements (15).

The Monte Carlo test procedure method based on $\widetilde{p}_N(S_0)$ can be summarized as follows.

Step 1: Compute the statistic S_0 using the observed data.
Step 2: Generate N *i.i.d.* replications S_1, …, S_N of the statistic S under H_0.
Step 3: Using S_0 and the replications S_1, …, S_N, compute the p-value

$$\widetilde{p}_N(S_0) = \frac{N+2-\widetilde{R}_N(S_0)}{N+1} \tag{17}$$

where $\widetilde{R}_N(S_0)$ is the randomized sample rank of S_0.
Step 4: Check if $\widetilde{p}_N(S_0) \leq \alpha$.

We call this procedure a MC test with *randomized tie-breaker*. The problem then consists in simulating $S_1, \ldots, S_N$. There are two basic situations. The first one consists in cases where the form of the DGP is specified, so we can simulate the data and compute the corresponding values of the test statistic. If the distribution of the test statistic does not depend on unknown parameters, these can be set at arbitrary values (compatible with the null hypothesis) in a way that makes the computation cost as small as possible. For example, the distribution of a *t*-statistic in a linear regression typically does not involve the values of the coefficients of unconstrained regressors, so these can be set to zero for the purpose of generating replications of the *t*-statistic. The second case is the one where the DGP cannot be simulated, but the test statistic can be. This happens, for example, in nonparametric setups where the DGP is incompletely specified, but signs and ranks have well defined distributions under the null hypothesis. In such cases, one can simulate the signs (or the ranks) along the corresponding values of the test statistic. Alternatively, one could also simulate any DGP compatible with null hypothesis, and proceed as in the first case.

3 Pivotal Monte Carlo tests in R

The technique of Monte Carlo tests for pivotal test statistics is implemented in the **MaxMC** package under the function name `MaxMC::mc`. The function call is reproduced here for reference.

```
mc(y, statistic, ...,
   dgp = function(y) sample(y, replace = TRUE), N = 99,
   type = c("geq", "leq", "absolute", "two-tailed"))
```

The arguments of the function are the following ones.

`y`: A vector or data frame.

`statistic`: A function or a character string which specifies how the statistic is computed. The function receives `y` as input and produces a scalar as output.

`...`: Other named arguments for the test statistic which are passed unchanged each time it is called.

`dgp`: A function. The function takes `y` as the first argument of its inputs, and produces a simulated `y` as output. It should represent the data

generating process under the null. The default value is the function `sample(y, replace = TRUE)`, i.e., the bootstrap resampling of `y`.

`N`: An atomic vector: the number of replications of the test statistic.

`type`: A character string. It specifies the type of test for which the p-value function is produced. The possible values are: `geq`, `leq`, `absolute`, and `two-tailed`. The default value is `geq`.

Four different types of p-values are allowed: `leq`, `geq`, `absolute`, and `two-tailed`. The default, `geq`, corresponds to the methods described in Section 2, i.e., the null hypothesis is rejected when S_0 is greater than some critical value. While this case is common, `MaxMC::mc` and `MaxMC::pvalue` allow for other popular types of tests. Option `leq` calculates the p-value assuming that we reject the null hypothesis when S_0 is smaller than some critical value. For two-tailed tests, `two-tailed` computes the p -value as twice the minimum of `leq` and `geq`, i.e.,

$$\widetilde{p}_N(S_0) = 2 \cdot \min\left(\frac{N\widetilde{F}_N(S_0)+1}{N+1}, \frac{N\widetilde{G}_N(S_0)+1}{N+1}\right) \tag{18}$$

where

$$\widetilde{F}_N(S_0) = 1 - \widetilde{G}_N(S_0)\,. \tag{19}$$

Finally, consider the case where we wish to perform a two-tailed test and the statistic has a symmetric null distribution. Instead of using (18), we could exploit the symmetry of the test statistic by taking the absolute value of the statistic and using `geq` to compute the p-value. This is what `absolute` performs.

The returned value of the function `MaxMC::mc` is an object of class "mc," with the following components:

`S0`: observed value of the test `statistic`;

`p.value`: Monte Carlo p-value of `statistic`;

`y`: data specified in call;

`statistic`: `statistic` function specified in call;

`dgp`: `dgp` function specified in call;

`N`: number of replications specified in call;

`type`: `type` of p-value specified in call;

`call`: original call to `mmc`;

`seed`: value of `.Random.seed` at the start of `mc` call.

4 Example: Two-sample goodness-of-fit test

In order to demonstrate how the `mc` function can be used, we look at the problem of testing the equality of the distributions of two random samples. Let $X_1, \ldots, X_n$ and $Y_1, \ldots, Y_m$ be i.i.d. observations such that $F(x) = \mathbb{P}(X_i \leq x)$ is the cumulative distribution function of X_i and $H(y) = \mathbb{P}(Y_j \leq y)$ is the cumulative distribution function of Y_j. We wish to test

$$H_0 : F = H \quad \text{against} \quad H_1 : F \neq H \,. \tag{20}$$

To test H_0, one common solution is to use the Kolmogorov–Smirnov statistic (KS) (Bulca and Arslan, 2013; Smirnov, 1948) defined as follows:

$$KS = \sup_x |\hat{F}_n(x) - \hat{G}_m(x)| \tag{21}$$

where

$$\hat{F}_n(x) = \frac{1}{n}\sum_{i=1}^{n} I_{(-\infty,\, x]}(X_i)\,, \quad \hat{H}_n(y) = \frac{1}{m}\sum_{j=1}^{m} I_{(-\infty,\, y]}(Y_j) \tag{22}$$

are the empirical distribution functions of $X_1, \ldots, X_n$ and $Y_1, \ldots, Y_m$, respectively.

The KS test is distribution-free when the observations are *i.i.d.* and follow a continuous distribution, but this property vanishes if the observations follow a discrete distribution. Therefore, in order to obtain an exact test when the X_i or Y_i can follow discrete distributions, we use a permutation test which is implemented as a MC test procedure with randomized tie-breaker (Dufour and Farhat, 2001).

To demonstrate how to use `MaxMC::mc`, we first need to generate some data. For the sake of this example, let $n = m = 8$, $X_i \sim$Poisson(10), and $Y_j \sim$Poisson(10). Then, the following code snippet will yield the desired sample data.

```
# Generate x ~ Poisson(10), y ~ Poisson(10)
x <- rpois(8, lambda = 10)
y <- rpois(8, lambda = 10)
data <- list(x = x, y = y)
```

With these randomly generated X_i and Y_j, we can compute the Kolmogorov–Smirnov statistic using the function `stats::ks.test`, i.e.

```
# Apply the test statistic
ks.test(data$x, data$y)
```

This yields the following output.

```
        Two-sample Kolmogorov-Smirnov test

data: data$x and data$y
D = 0.375, p-value = 0.6272
alternative hypothesis: two-sided

Warning message:
In ks.test(data$x, data$y) : cannot compute exact p-value with
ties
```

Next, before using `MaxMC::mc`, we need to specify both how the statistic is computed using the actual `data`, and how to generate simulated data sets under the null.

For the test statistic, we merely create a wrapper for the function `ks.test` that outputs a scalar instead of an object of class "htest." Note that we use the function `suppressWarnings` in order to prevent the generation of warning messages every time the `mc` function calls the `statistic` function. It is not required.

For the data generating process, we use a permutation of the grouped data $(X_1, \ldots, X_n, Y_1, \ldots, Y_m)$ to generate N simulated samples

$$(X_1^{(j)}, \ldots, X_n^{(j)}, Y_1^{(j)}, \ldots, Y_m^{(j)}), \quad j = 1, \ldots, N, \tag{23}$$

as described in Dufour and Farhat (2001).

```
# Set the statistic function
statistic <- function(data){
        out <- suppressWarnings(ks.test(data$x, data$y))
        return(out$statistic)
        }

# Set the DGP function
dgp <- function(data){
        perm <- sample(c(data$x, data$y))
        x <- perm[1:length(data$x)]
        y <- perm[-(1:length(data$x))]
        return(list(x = x, y = y))
        }
```

We then evaluate the function: `MaxMC::mc`.

```
library(MaxMC)
# Apply the mc procedure
mc(y = data, statistic = statistic, dgp = dgp, N = 999,
type = "absolute")
```

This yields the following output.

```
Monte Carlo with Tie-Breaker
Call:
mc(y = data, statistic = statistic, dgp = dgp, N = 999,
   type = "absolute")

D = 0.375, N = 999, p-value = 0.488
```

In theory, this p-value is exact. While the results are fairly similar to what we obtained with `stats::ks.test` directly, the point of this particular exercise is not to show an obvious failure of the ordinary test, but how effortless it is to use the MC technique with tie-breaker.

5 Maximized Monte Carlo tests

Unlike in the previous section, we now discard the assumption that $S := S(X^{(n)}, \theta)$ is a pivotal statistic, i.e., the distribution of S depends on some nuisance parameters ν under H_0. Let Ω be the parameter space of ν and ν_0 be its true value. Then we are essentially looking to test

$$H_0 : \nu_0 \in \Omega_0 \tag{24}$$

where Ω_0 is the subset of Ω consistent with the null hypothesis. To solve this type of problem and retrieve an exact test, we can now proceed as follows.

Step 1: Compute the statistic S_0 using the sample data.
Step 2: Generate N *i.i.d.* replications $S_j(\nu)$ of the statistic S for each $\nu \in \Omega_0$.
Step 3: Using the replications $S_j(\nu)$, compute the following p-value

$$\hat{p}_N(S_0|\nu) = \frac{N\hat{G}_N(S_0|\nu) + 1}{N+1} \tag{25}$$

where $\hat{G}_N(x|\nu)$ corresponds to the following survival function

$$\hat{G}_N(x|\nu) = \frac{1}{N}\sum_{j=1}^{N} I_{[0,\,\infty)}(S_j(\nu) - x)\,. \tag{26}$$

Step 4: Maximize the p-value function $\hat{p}_N(S_0|\nu)$ over the set $\nu \in \Omega_0$, i.e.,

$$\hat{Q}_N(S_0) = \sup_{\nu \in \Omega_0} \hat{p}_N(S_0|\nu)\,. \tag{27}$$

Step 5: Reject the null hypothesis if

$$\hat{Q}_N(S_0) \leq \alpha\,. \tag{28}$$

For every value of ν consistent with the null hypothesis, we find its associated p-value. Then, if and only if every p-value is smaller than the desired level,

we reject the null hypothesis. We call this procedure a maximized Monte Carlo (MMC) test, and $\hat{Q}_N(S_0)$ the MMC p-value.

If $\alpha(N+1)$ is an integer and the distribution of S is continuous, we have

$$\mathbb{P}[\hat{Q}_N(S_0) \leq \alpha] \leq \alpha \tag{29}$$

under H_0, which means that the critical region

$$\hat{Q}_N(S_0) \leq \alpha \tag{30}$$

has level α; for a proof, see Dufour (2006). Without relying on strong regularity assumptions, the MMC method provides a simple method to obtain valid tests even in the presence of nuisance parameters.

As with MC tests with a randomized tie-breaker, we can use ranks instead of directly using the survival function $\hat{G}_N(S_0|\nu)$. Therefore, as previously shown, we can write $\hat{G}_N(S_0|\nu)$ as

$$\hat{G}_N(S_0|\nu) = \frac{N+1-\hat{R}_N(S_0|\nu)}{N} \tag{31}$$

where $\hat{R}_N(S_0|\nu)$ represents the sample rank of S_0 in S_0, $S_1(\nu)$, …, $S_N(\nu)$. In the case of a discrete statistic, we can extend the discussion in the previous section to how to "break" ties in sample ranks. Hence, let $U_0, U_1, \ldots, U_N \overset{i.i.d}{\sim} U(0,1)$. Then, we can create the pairs (S_i, U_i) and use the lexicographic ordering previously described:

$$(S_i(\nu), U_i) \leq (S_0, U_0) \Leftrightarrow \{S_i(\nu) < S_0 \text{ or } (S_i(\nu) = S_0 \text{ and } U_i \leq U_0)\} \tag{32}$$

to order the (S_i, U_i) pairs and compute the randomized rank for S_0 as

$$\widetilde{R}_N(S_0|\nu) = \sum_{i=0}^{N} I[(S_i(\nu), U_i) \leq (S_0, U_0)] \tag{33}$$

where I is an indicator function for the preference relation.

We get in this way the following p-value function, using the randomized rank for S_0,

$$\widetilde{p}_N(S_0|\nu) = \frac{N+2-\widetilde{R}_N(S_0|\nu)}{N+1} \tag{34}$$

and, if $\alpha(N+1)$ is an integer, we have:

$$\mathbb{P}_{H_0}[\sup_{\nu \in \Omega_0} \widetilde{p}_N(S_0|\nu) \leq \alpha] \leq \alpha \tag{35}$$

i.e., the test is exact at level α; again for a proof, see Dufour (2006). This result holds regardless whether the distribution of S is continuous or discrete.

As discussed in Section 2, the test statistics $S_j(\nu)$ can be simulated by either simulating the restricted DGP and the corresponding statistic, or the

distribution of the test statistic. The only difference with the pivotal case is that these now depend on the nuisance parameters ν along with random disturbances. For smoothness, the same disturbances should be used for each replication, with ν viewed as a variable over which the maximization takes place. On simulating time series data, the package **meboot** may be useful; see Vinod and López-de Lacalle (2009).

The MMC procedure accompanied with the use of DGP to generate $S_j(\nu)$ is implemented in **MaxMC** under the function `MaxMC::mmc`. Before going into the details of `MaxMC::mmc`, it is important to look into some of its computational issues and how the MMC can be modified to solve them.

6 Asymptotic MMC tests

A practical difficulty inherent to the MMC method is that the computational cost typically increases with the volume.

Regardless of any past or future improvements in the field of computer science, whenever a consistent point set estimate of ν is available, we can simplify the MMC procedure by reducing the space over which the p -value is maximized, while maintaining the validity of the test asymptotically. This can be done as follows.

Step 1: Compute the statistic S_0 using the sample data.
Step 2: Let C_T be a sequence of sets such that $C_T \subset \Omega$, and

$$\lim_{T\to\infty} P[\nu_0 \in C_T] = 1 \quad \text{under } H_0\,. \tag{36}$$

Step 3: Generate N *i.i.d.* replications $S_{Tj}(\nu)$ of the statistic S for each $\nu \in C_T$.
Step 4: Using the replications $S_{Tj}(\nu)$, compute the p-value

$$\widetilde{p}_N(S_0|\nu) = \frac{N\widetilde{G}_{TN}(S_0|\nu)+1}{N+1} \tag{37}$$

where $\widetilde{G}_{TN}(x|\nu)$ is the simulated survival function

$$\hat{G}_{TN}(x|\nu) = \frac{1}{N}\sum_{j=1}^{N} I_{[0,\,\infty)}(S_{Tj}(\nu)-x) \quad \text{where} \quad I_A(z) = \begin{cases} 1 & \text{if} \quad z\in A \\ 0 & \text{if} \quad z\notin A \end{cases}. \tag{38}$$

Step 5: Maximize the p-value function $\widetilde{p}_{TN}(S_0|\nu)$ over the set $\nu \in C_T$, i.e.,

$$\widetilde{Q}_{TN}(S_0) = \sup_{\nu\in C_T} \widetilde{p}_{TN}(S_0|\nu)\,. \tag{39}$$

Step 6: Reject H_0 when

$$\lim_{T\to\infty} \widetilde{Q}_{TN}(S_0) \le \alpha\,. \tag{40}$$

We call this procedure the consistent set consistent MMC (SC-MMC) test. If $\alpha(N + 1)$ is an integer, we have:

$$\lim_{T\to\infty} \mathbb{P}_{H_0}\left[\widetilde{Q}_{TN}(S_0) \le \alpha\right] \le \alpha \tag{41}$$

i.e., the test has an asymptotic level equal to α. Finding a consistent estimator is usually pretty straightforward. Methods such as the generalized method of moments (GMM) or maximum likelihood (ML) often provide consistent estimators. Then, any set of the following form

$$C_T = \{\nu \in \Omega : \| \hat{\nu}_T - \nu \| < d\} \tag{42}$$

where d is any fixed positive constant and $\hat{\nu}_T$ is a consistent estimator, satisfies (36).

The constant d can be chosen to be arbitrarily small and will restrict the set on which $\widetilde{p}_{TN}(S_0|\nu)$ is maximized. Decreasing d normally coincides with increases in the power of the test. However, a small d might prevent us from capturing discontinuities in the distribution of statistic with respect to the nuisance parameter. We could even technically reduce the set to

$$C_T = \{\hat{\nu}_T\} \tag{43}$$

where $\hat{\nu}_T$ is a consistent point estimator of ν. Then, there is no need to maximize the p-value, since C_T contains only one point. We refer to this case as the Local Monte Carlo (LMC) which is analogous to a parametric bootstrap. However, in order to satisfy (36), we need stronger regularity assumptions on the model, which limits the cases where LMC (or parametric bootstrap) tests are asymptotically valid; for a discussion of such conditions, see Dufour (2006).

Besides, although asymptotic tests might be appealing, they fall short from one of the main advantages of familiar Monte Carlo methods, exact inference. One way to get back this property while simultaneously reducing the computational load is to use confidence sets (or intervals) for the nuisance parameters. This suggests one to employ a two-step confidence procedure of the type described in Dufour (1990) and Dufour and Kiviet (1998). When such confidence intervals are available, the MMC procedure can be modified in the following way.

Step 1: Compute the statistic S_0 using the sample data.
Step 2: Construct $C_{\nu_0}(\alpha_1)$, an exact confidence set for ν, with level α_1, i.e.,

$$P[\nu_0 \in C_{\nu_0}(\alpha_1)] = 1 - \alpha_1 \quad \text{under } H_0\,. \tag{44}$$

Step 3: Generate N *i.i.d.* replications $S_j(\nu)$ of the statistic S for each $\nu \in C_{\nu_0}(\alpha_1)$.
Step 4: Using the replications $S_j(\nu)$, compute the following p-value

$$\widetilde{p}_N(S_0|\nu) = \frac{N\widetilde{G}_N(S_0|\nu) + 1}{N+1} \tag{45}$$

where $\widetilde{G}_N(x|\nu)$ corresponds to the usual survival function.
Step 5: Maximize the p-value function $\widetilde{p}_N(S_0|\nu)$ over the set $\nu \in C_{\nu_0}(\alpha_1)$, i.e.,

$$\widetilde{Q}_N(S_0) = \sup_{\nu \in C_{\nu_0}(\alpha_1)} \widetilde{p}_N(S_0|\nu)\,. \tag{46}$$

Step 6: Reject H_0 if

$$\widetilde{Q}_N(S_0) \leq \alpha_2\,. \tag{47}$$

If $\alpha_2(N + 1)$ is an integer, we have:

$$\mathbb{P}_{H_0}[\hat{Q}_N(S_0) \leq \alpha_2] \leq \alpha_1 + \alpha_2 = \alpha \tag{48}$$

which states that the MMC procedure has exact level $\alpha_1 + \alpha_2 = \alpha$. A simple choice α_1 and α_2 consists in setting $\alpha_1 = \alpha_2 = \alpha/2$.

7 MMC tests in R

The `MaxMC::mmc` function implements the MMC technique with tie-breaker described in the previous section.

```
mmc(y, statistic, ...,
    dgp = function(y, v) sample(y, replace = TRUE),
    est = NULL, lower, upper, N = 99,
    type = c("geq", "leq", "absolute", "two-tailed"),
    method = c("GenSA", "pso", "GA", "gridSearch"),
    control = list(), alpha = NULL)
```

The arguments for the function call are the following.

`y`: A vector or data frame.

`statistic`: A function or a character string which specifies how the statistic is computed. The function takes `y` as input and produces a scalar as output.

`...`: Other named arguments for the statistic which are passed unchanged each time it is called.

`dgp`: A function. The function takes as inputs `y` and a vector of nuisance parameters `v`, and produces a simulated `y` as output. It should represent the data generating process under the null hypothesis. The default value is the function `sample(y, replace = TRUE)`, i.e., the bootstrap resampling of `y`.

`est`: A vector with the same length as `v`. It is the starting point of the algorithm. If `est` is a consistent estimate of `v`, then `mmc` returns both the MMC and Local Monte Carlo (LMC). Default is NULL, in which case, default values will be generated automatically.

`lower`: A vector with the same length as `v`. Lower bounds for nuisance parameters under the null hypothesis.

`upper`: A vector with the same length as `v`. Upper bounds for nuisance parameters under the null hypothesis.

`N`: An atomic vector: the number of replications of the test statistic.

`type`: A character string. It specifies the type of test for which the *p*-value function is computed. The possible values are: `geq`, `leq`, `absolute`, and `two-tailed`. The default is `geq`.

`method`: A character string. Type of algorithm to be used for global optimization. Four methods are available: grid search (`grid-Search`), simulated annealing (**GenSA**), genetic algorithm (**GA**), and particle swarm (**pso**). Default is **GenSA**.

`control`: A list. Arguments to be used to control the behavior of the algorithm chosen in `method`.

`alpha`: An atomic vector. If `mmc` finds a *p*-value over `alpha`, the algorithm stops. This is particularly useful if we are only looking at testing a hypothesis at a particular level. Default is NULL.

`monitor`: A logical variable. If set to TRUE, the *p*-values at every iteration and the cumulative maximum *p*-value are plotted on a graphical device. The default is FALSE.

The `dgp` function defined by the user is used to generate new observations in order to compute the simulated statistics. The only difference with the pivotal case is that the `dgp` also takes nuisance parameters among its inputs. The `statistic` and `dgp` functions are the building blocks of the procedure. It is thus essential that the functions be written efficiently in order for `mmc` to find the MMC p -value quickly.

The returned value of `mmc` is an object of class "mmc," containing the following components.

`S0`: Observed value of the `statistic`.

`pval`: Maximized Monte Carlo *p*-value of `statistic` under the null hypothesis.

`y`: Data specified in call.

`statistic`: `statistic` function specified in call.

`dgp`: `dgp` function specified in call.

`est`: `est` vector if specified in call.

`lower`: `lower` vector if specified in call.

`upper`: `upper` vector if specified in call.

`N`: Number of replications specified in call.

`type`: `type` of p-value specified in call.

`method`: `method` specified in call.

`call`: Original call to `mmc`.

`seed`: Value of `.Random.seed` at the start of `mmc` call.

`lmc`: If `par` is specified, it returns an object of class `mc` corresponding to the Local Monte Carlo test.

`opt_result`: An object returning the optimization results.

`rejection`: If `alpha` is specified, it returns a vector specifying whether the hypothesis was rejected at level `alpha`.

7.1 Global Optimization

Several methods can be used to maximize over the set of nuisance parameters. For the moment, four methods are available: grid search, simulated annealing, genetic algorithm, and particle swarm optimization. Future updates might include improved algorithms. One issue we have to face when choosing an algorithm is that the p-value function

$$\widetilde{p}_N(S_0|\nu) = \frac{N\widetilde{G}_n(S_0|\nu) + 1}{N + 1} \tag{49}$$

is not continuously differentiable. In fact, the function has derivative equal to zero everywhere except at $N + 1$ points where the derivative does not exist. Since we are trying to maximize this p-value function, any method of optimization which relies on its derivative is going to be unsuccessful (e.g., gradient descent algorithms and quasi-Newton methods).

7.1.1 gridSearch

The grid search method is the easiest to implement and understand, but sadly not efficient when the number of parameters is large and not strongly restricted under H_0. Let Ω^* be the space of nuisance parameters $\nu = (\nu_1, \nu_2, \ldots \nu_m)$ over which we maximize the p-value. A simple way to setup a grid search consists in defining a vector of lower bounds $a = (a_1, a_2, \ldots, a_m)$ and a vector of upper bounds $b = (b_1, b_2, \ldots, b_m)$ for each component of ν. Grid search involves taking n equally spaced points in each interval of the form $[a_i, b_i]$ including a_i and b_i. This creates a total of n^m possible grid points to check. Finally, once each pair of points is calculated, the maximum of these values is chosen.

The problem with this type of method is that the number of evaluations increases exponentially as n and m increase. Since we cannot really reduce m, decreasing n is the only possible way of assuring that the method stops in a reasonable time, but this decreases the validity of the solution.

The package **NMOF** (Gilli et al., 2011) provides the function `gridSearch` which implements exactly this method. It has the distinctive advantage of providing an easy way to parallelize the problem using the package **parallel** by Sasaki et al. (2005). This could resolve some of the computing issues associated with the grid search method, but we do not recommend this method in general especially for $m \geq 3$. By default, we set $n = 10$, but this can be modified to a more appropriate number.

7.1.2 GenSA

Simulated annealing is a stochastic, metaheuristic technique which can find a "good" solution to a global optimization problem, even in the presence of multiple local minima. It was originally and independently proposed by Kirkpatrick et al. (1983) and Černý (1985); see also Goffe et al. (1994).

Simulated annealing simply selects some neighboring point x' to the current position x. Then with some probability function the algorithm chooses if it will stay with x or move to x'. As the algorithm progresses, the probability of moving to a new point converges to zero. This is done through a global time-varying parameter T which follows an annealing schedule called the "temperature." Typically the definition of the algorithm sets the probability of moving equal to 1 when x' is better than x, but it is not necessary for convergence. The main feature of simulated annealing is that there is always a probability that it might move to a worse point. This usually prevents the algorithm from being stuck in a local minimum.

The simulated annealing method is implemented using the **GenSA** package described in Xiang et al. (2013). The package **stats** available in the base distribution of R also provides the function `optim` which implements simulated annealing.

As shown by Katharine (2014), the **GenSA** package outperforms the function `optim` in terms of convergence and speed. Thus, choosing which version of simulated annealing to implement in the **MaxMC** package was straightforward. The method is set by default to stop when the number of steps without improvement reaches 25. It will also stop when it finds a point where the p-value is either equal to 1 or bigger than the level `alpha`. For other details on the settings of this method, see Xiang et al. (2013) and the documentation for **MaxMC** and **GenSA**.

7.1.3 psoptim

Particle swarm optimization was introduced by Eberhart and Kennedy (1995) and Shi and Eberhart (1998). Like the simulated annealing technique, particle swarm is metaheuristic. The algorithm involves taking a set of candidate

solutions (particles) with random initial position, and the particles are set to move around the space to search for the best solution. The directions and velocities associated with the particles are guided toward both the best known position for individual particle and the best known overall position. As the number of iterations increase, so will the convergence rate to the best known position for the entire population.

Multiple packages such as **hydroPSO** (Zambrano-Bigiarini, 2013) and **pso** (Bendtsen, 2012) implement the particle swarm optimization technique in R. For the package **MaxMC**, the **pso** package and its function `psoptim` was selected.

When comparing **pso** and **hydroPSO**, Katharine (2014) found that **pso** performed a bit better. This is the main reason why it is implemented in **MaxMC**. By default, the algorithm is set to stop after no improvement to the best known location has been made in 25 steps. For more details on how the velocities, population size and other features are set, see Bendtsen (2012) and the documentation for **MaxMC** and **pso**.

7.1.4 GA

The genetic algorithm is a subclass of evolutionary algorithm techniques. The technique dates back to the 1970s (see Holland, 1992). As the name suggests, evolutionary algorithms mimic natural selection, where only the fittest individuals survive through the process of mutation, selection, and crossover. For the genetic algorithm, each candidate solution has a set of properties (or genes) which can mutate or change until we find the best solution. The following packages offer some sort of implementation of the genetic algorithm: **soma** (Clayden, 2014), **GA** (Scrucca, 2013, 2016), **genalg** (Willighagen, 2005), **mcga** (Satman, 2013), and **NMOF** (Gilli et al., 2011).

GA was chosen not only because it provides a flexible set of controls and methods for the genetic algorithm, but it also allows for easy parallelization of the problem, which as previously mentioned can be tremendously helpful.

7.2 Optimal Choice

Which global optimization method should we pick? Since each technique has its own pros and cons, the choice of methodology is not always clear-cut. For instance, we can look at the Behrens–Fisher problem shown in Section 8.1. Without going into the details of the problem, we solve it using different optimization methods to compare and contrast their speed and accuracy. In order to do so, we use **microbenchmark** (Mersmann, 2015) and its plot method based on **ggplot2** (Wickham, 2009).

As can be seen, the `pso` algorithm appears to provide the most efficient method in this case. `GenSA` takes, on average, the longest time for this example, but in some cases it finds a solution quite rapidly. Moreover, the grid search method performs well because it only evaluates the

function at 10^3 points. If we doubled the accuracy for `gridSearch`, the computation time would increase by a factor of 8, which would reduce the usefulness of this method. Further, we did not use any parallelization in the controls of the methods. This could significantly improve performance.

Finally, since the methods are metaheuristic, Fig. 1 is sadly not enough for us to make any comment on how efficient one method is compared to another

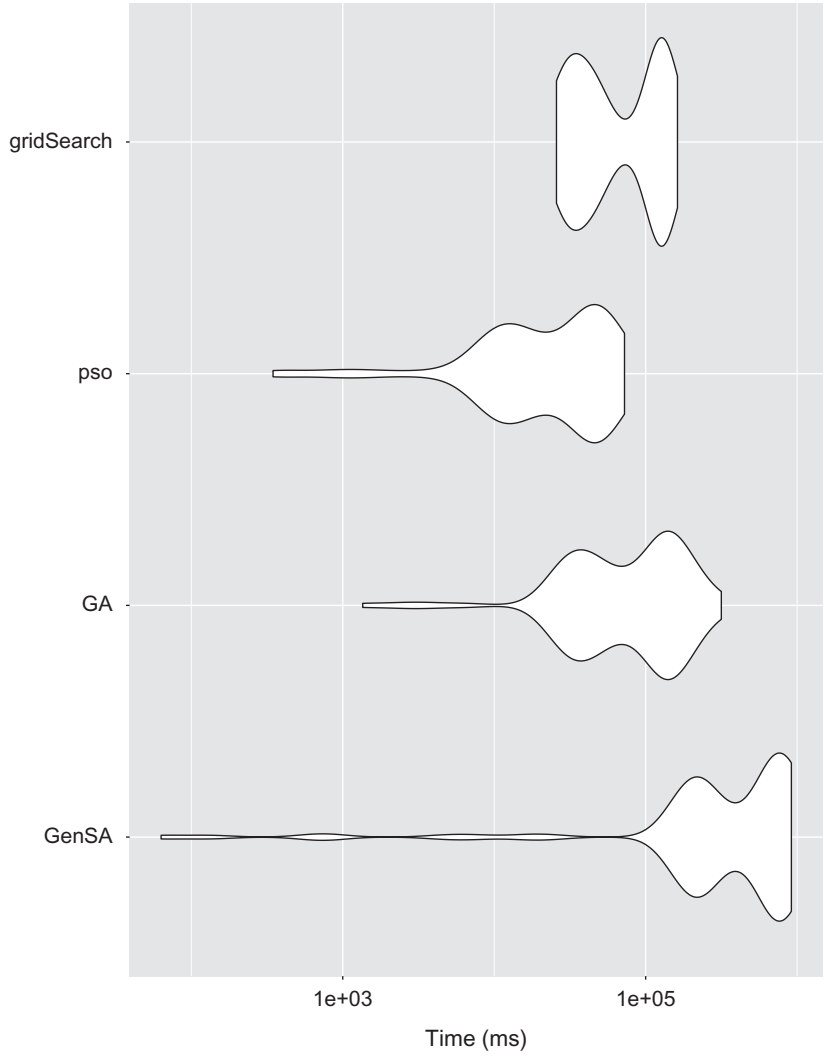

FIG. 1 Density plot for evaluation time of mmc(y = data, statistic = statistic, dgp = dgp, est = est, lower = lower, upper = upper, N = 99, type = "absolute") with different global optimization algorithms.

in finding a "good" solution. In the end, the choice of method may boil down to personal preference. What is important regarding the **MaxMC** package is that it is compatible with any of these methods and allows the user to choose.

8 MMC tests: Examples

In this section, we discuss two examples where the MMC method provides a natural solution to hypothesis testing problems: (1) the classic Behrens–Fisher problem of comparing the means of normal samples with different variances and (2) testing the unit root hypothesis in an AR(p) model. These cases are presented mainly for the purpose of showing how the MMC method can be implemented in **MaxMC**.

8.1 Behrens–Fisher problem

In order to demonstrate how to apply the `MaxMC::mmc` function, we present a simple example where we test for the equality of the means of two independent normal populations with unknown and not necessarily equal variances. This is known as the Behrens–Fisher problem; for more details, see Fisher (1935, 1941) and Behrens (1929).

Let $X_{11},\ldots,X_{1n_1} \sim N(\mu_1,\sigma_1^2)$ and $X_{21},\ldots,X_{2n_2} \sim N(\mu_2,\sigma_2^2)$. We consider the problem of testing

$$H_0 : \mu_1 - \mu_2 = 0 \text{ against } H_1 : \mu_1 - \mu_2 \neq 0. \tag{50}$$

One solution to this problem consists in using the extension of Student's t-test proposed by Welch (1947) for unequal variances with the following form

$$t = \frac{\overline{X}_1 - \overline{X}_2}{\sqrt{\frac{s_1^2}{n_1} + \frac{s_2^2}{n_2}}} \tag{51}$$

where s_i is the sample variance of X_i and n_i is the sample size of X_i. The issue with this statistic is that its finite-sample distribution under the null hypothesis depends ultimately on nuisance parameters σ_1 and σ_2, more precisely on the ratio σ_2/σ_1. Weir (1960) showed that the distribution of this statistic can be approximated with a Student's t-distribution with the following number of degrees of freedom

$$df \approx \frac{\left(\frac{s_1^2}{n_1} - \frac{s_2^2}{n_2}\right)^2}{\frac{s_1^4}{n_1^3 - n_1^2} + \frac{s_2^4}{n_2^3 - n_2^2}} \tag{52}$$

where s_i is the sample variance of X_i and n_i is the sample size of X_i.

Instead of using the t-distribution approximation, we will apply the MMC method directly. The following R code excerpt shows how the Behrens–Fisher problem can be implemented for $X_{1i} \sim N(0, 1)$ and $X_{2i} \sim N(0, 4)$ with $n_1 = n_2 = 15$.

```
library(MASS)
# Generate sample x1 ~ N(0,1) and x2 ~ N(0,4)
x1 <- rnorm(15, mean = 0, sd = 1)
x2 <- rnorm(15, mean = 0, sd = 2)
data <- list(x1 = x1, x2 = x2)

# Fit normal distributions on x1 and x2 using MLE
fit1 <- fitdistr(x1, "normal")
fit2 <- fitdistr(x2, "normal")

# Compute the estimate for the nuisance parameter
est <- fit2$estimate["sd"]/fit1$estimate["sd"]

# Set the bounds of the nuisance parameter
lower <- 0
upper <- est + 10

# Set the function for the DGP under the null
dgp <- function(data, v) {
        x1 <- rnorm(length(data$x1), mean = 0, sd = 1)
        x2 <- rnorm(length(data$x2), mean = 0, sd = v)
        return(list(x1 = x1, x2 = x2))
        }

# Set the statistic function for Welch's t-test
welch <- function(data) {
        test <- t.test(data$x2, data$x1)
        return(test$statistic)
        }

# Apply Welch's t-test
t.test(data$x2, data$x1)

# Apply the mmc procedure
mmc(y = data, statistic = welch, dgp = dgp, lower = lower,
    est = est, upper = upper, N = 99, type = "absolute",
    method = "GenSA")
```

```
        Welch Two Sample t-test

data: data$x2 and data$x1
t = -0.50217, df = 19.362, p-value = 0.6212
alternative hypothesis: true difference in means is not equal
to 0
```

```
95 percent confidence interval:
-1.6180550  0.9912269
sample estimates:
mean of x mean of y
-0.09098809 0.22242598
```

```
Maximized Monte Carlo

Call:
mmc(y = data, statistic = welch, dgp = dgp, est = est,
    lower = lower, upper = upper, N = 99, type = "absolute",
    method = "GenSA")

t = -0.50217, N = 99
Local Monte Carlo: p-value = 0.56
Maximized Monte Carlo: p-value = 0.65
```

Note that we use the **MASS** package (Venables and Ripley, 2003) to derive a point estimate $\hat{\nu}$ for the ratio of σ_2 and σ_1 which we use as a starting point for our optimization algorithm. Since our nuisance parameter can take any value on the positive real line, it is practical to restrict the optimization of the MMC to a strictly smaller subset. For instance, we can build the following interval around the estimator, $\hat{\nu}$,

$$C_{\hat{\nu}} = [0, \hat{\nu} + d] \tag{53}$$

where d is some constant. In our example, d is arbitrarily set to 10 and `MaxMC::mmc` maximizes the p-value over $C_{\hat{\nu}}$.

Note that, since the `t.test` from the **stats** package (Sasaki et al., 2005) implements by default the Welch test, so no additional arguments have to be specified in the code.

8.2 Unit root tests in autoregressive models

We now turn our attention to how **MaxMC** can be used for unit root tests in autoregressive models. In particular, we look into the details of one of the most popular class of unit root tests, the augmented Dickey–Fuller test.

8.2.1 Framework

Consider the following autoregressive model of order p

$$Y_t = \mu + \eta t + \sum_{j=1}^{p} \phi_j Y_{t-j} + u_t \tag{54}$$

where $t = p + 1, \ldots, T$, μ is the drift component, η is the time trend, and u_t is a sequence of *i.i.d.* $(0, \sigma^2)$ variables. We can rewrite the model in the following way

$$\Phi(L)Y_t = \mu + \eta t + u_t \tag{55}$$

where $t = p + 1, \ldots, T$ and

$$\Phi(L) := 1 - \sum_{j=1}^{p} \phi_j L^j \tag{56}$$

is the characteristic equation of the time process with L being the lag operator, i.e., $LY_t = Y_{t-1}$. If the characteristic polynomial has a unit root, i.e., $\Phi(1) = 0$, with multiplicity r, then Y_t is said to be an integrated process of order r, or an $I(r)$ process.

As such, tests of unit root can be easily formulated as

$$H_0 : \Phi(1) = 0 \tag{57}$$

or equivalently,

$$H_0 : \sum_{j=1}^{p} \phi_j = 1 \tag{58}$$

In order to test H_0, we can rewrite (54) in the following way

$$\Delta Y_t = \mu + \eta t + \gamma Y_{t-1} + \sum_{j=1}^{p-1} \rho_{j+1} \Delta Y_{t-j} + u_t \tag{59}$$

where $t = p + 1, \ldots, T$, $\Delta Y_t = Y_t - Y_{t-1}$, and

$$\gamma = \sum_{j=1}^{p} \phi_j - 1 \ , \ \rho_j = -\sum_{i=j}^{p} \phi_i \, . \tag{60}$$

Then, the null hypothesis can be written as

$$H_0 : \gamma = 0 \, . \tag{61}$$

To test H_0, we can simply use the usual Student t-statistic t_γ based on least-squares estimator. This is referred to as the augmented Dickey–Fuller (ADF) test statistic. The usual version of this procedure tests H_0 against the alternative hypothesis of stationarity, i.e., $H_1 : \gamma < 0$.

Since t_γ is based on (59) and not (54), the ADF test statistic does not have the usual Student's t-distribution. In fact, it is not even distributed symmetrically. But the distributional properties of t_γ have been well documented; see Fuller (1996), Banerjee et al. (1993), and MacKinnon (1999).

8.2.2 Code

To illustrate how to use **MaxMC** with the ADF test statistic, we start by examining a simple AR(2) process without drift or time trend, i.e.,

$$Y_t = \phi_1 Y_{t-1} + \phi_2 Y_{t-2} + u_t \tag{62}$$

where $t = 2, \ldots, T$ and $u_t \overset{i.i.d}{\sim} N(0, \sigma^2)$. A crude way of applying the MMC method consists in taking all parameters (e.g., σ, ϕ_1, and ϕ_2) as nuisance parameters. A better way is to take advantage of the problem setting. For example, since the distribution of the ADF statistic is independent of σ, it is not necessary to include it among the nuisance parameters. Further, if we have a unit root, the following needs to hold

$$\phi_1 + \phi_2 = 1 \tag{63}$$

While both ϕ_1 and ϕ_2 can be seen as nuisance parameters, we can exploit (63) to reduce the set of nuisance parameters: we maximize the p-value with respect to ϕ_1 (or ϕ_2) only.

For the sake of this example, we let $\phi_1 = \phi_2 = \frac{1}{2}$, i.e., we have a unit root. To generate data, we can use the function `filter` from the package **stats** as demonstrated in the following tidbit of code.

```
library(fUnitRoots)

# Generate an AR(2) process with phi = (.5,.5), and n = 25
y <- filter(rnorm(25), c(.5,.5), method = "recursive")
```

We also include a line to import **fUnitroots** (Wuertz, 2009). This is simply because **fUnitroots** provides the function `adfTest` which we are going to use to compute the ADF statistic. It is important to bear in mind that `adfTest` uses the tables provided by Banerjee et al. (1993) to compute the asymptotic p-value.

The next step is to setup the functions for the DGP and how to compute the statistic. We take ϕ_2 as the nuisance parameter, so we set $\phi_1 = 1 - \phi_2$ under the null hypothesis. This yields the following code.

```
# Set the function to generate an AR(2) integrated process
dgp <- function(y, v) {
       ran.y <- filter(rnorm(length(y)), c(1-v,v),
       method = "recursive")
       }

# Set the Augmented Dicky-Fuller statistic
statistic <- function(y){
             out <- adfTest(y, lags = 2, type = "nc")
             return(out@test$statistic)
             }
```

Note that the option `type = "nc"` for the function `adfTest` simply specifies that our model does not have a drift or a time trend.

The last step before applying `MaxMC::mmc` is to set the bounds for the nuisance parameter ϕ_2. While ϕ_2 could potentially be any number on the real line, we restrict our attention to values of ϕ_2 such that the process has no root in the interior of the unit circle. An AR(2) process whose roots do not belong to the interior of the unit circle must satisfy the following restrictions:

$$\phi_2 + \phi_1 \leq 1\,,\ \ \phi_2 - \phi_1 \leq 1\,,\ \ |\phi_2| \leq 1\,. \tag{64}$$

Hence, we can simply set our bounds to $|\phi_2| \leq 1$:

```
# Set bounds for the nuisance parameter v
lower <- -1
upper <- 1
```

Now, we are ready to apply the MMC procedure with `MaxMC::mmc` . Since we have only one nuisance parameter, we pick grid search as our method of choice for the optimization.

```
# Apply the mmc procedure
mmc(y, statistic = statistic , dgp = dgp, lower = lower,
    upper = upper, N = 99, type = "leq",
    method = "gridSearch", control = list(n = 100))
```

Hence, this `MaxMC::mmc` call generates the following output.

```
Maximized Monte Carlo

Call:
mmc(y = y, statistic = statistic, dgp = dgp, lower = lower,
    upper = upper, N = 99, type = "leq",
    method = "gridSearch", control = list(n = 100))

Dickey-Fuller = -1.0743, N = 99
Maximized Monte Carlo: p-value = 0.31
```

In this particular session, the *p*-value is equal to 0.31, which is consistent with the fact that the data generated has a unit root.

9 Conclusion

The method of Monte Carlo tests is a powerful method which allows one to perform exact tests in many situations where this is not typically feasible, as well as asymptotically valid tests in problems where the asymptotic distribution of the test statistic is nonregular and may have discontinuities. Its main

theoretical feature is *transparency*: the investigator needs little knowledge of the distributional complexities involved. The simulation takes care of these. In the case of pivotal test statistics, Monte Carlo tests implemented with the Hájek tie-breaker allow one to control size perfectly, even if no information is available on the mass points of the null distribution. This can be especially useful when analyzing qualitative data or using rank tests. For the ubiquitous problem of test statistics which depend on nuisance parameters under the null hypothesis, MMC tests do allow one to control the level as soon the test statistic can be simulated after fixing the nuisance parameters. The variant of the procedure where maximization is limited to a consistent set estimator of the nuisance parameters allows one to obtain asymptotically valid tests in cases where the asymptotic distribution is difficult to establish and may involve nuisance parameters, including discontinuities.

Clearly, this general technique is computer-intensive. The **MaxMC** package now available in R provides a systematic way of using this type of procedure.

Acknowledgments

The authors thank Nazmul Ahsan, Mathieu Blais, Russell Davidson, Tianyu He, Vinh Nguyen, Masaya Takano, and Hrishikesh Vinod for several useful comments. This work was supported by the William Dow Chair in Political Economy (McGill University), the Bank of Canada (Research Fellowship), the Toulouse School of Economics (Pierre-de-Fermat Chair of excellence), the Natural Sciences and Engineering Research Council of Canada, the Social Sciences and Humanities Research Council of Canada, and the Fonds de recherche sur la soci été et la culture (Québec).

References

Banerjee, A., Dolado, J.J., Galbraith, J.W., Hendry, D., 1993. Co-integration, Error Correction, and the Econometric Analysis of Non-Stationary Data. Oxford University Press. ISBN: 9780198288107. https://doi.org/10.1093/0198288107.001.0001. http://www.oxfordscholarship.com/view/10.1093/0198288107.001.0001/acprof-9780198288107.

Barnard, G.A., 1963. Comment on 'The spectral analysis of point processes' by M. S. Bartlett. J. R. Stat. Soc. Ser. B 25, 294.

Beaulieu, M.C., Dufour, J.M., Khalaf, L., 2007. Multivariate tests of mean-variance efficiency with possibly non-Gaussian errors: an exact simulation-based approach. J. Bus. Econ. Stat. 25 (4), 398–410.

Beaulieu, M.C., Dufour, J.M., Khalaf, L., 2013. Identification-robust estimation and testing of the zero-beta CAPM. Rev. Econ. Stud. 83 (3), 892–924.

Behrens, W.V., 1929. Ein Beitrag zur Fehlerberechnung bei wenigen Beobachtungen. Landwirtsch Jahrbucher 68, 807–837.

Bendtsen, C., 2012. PSO: Particle Swarm Optimization. https://cran.r-project.org/package=pso.

Beran, R., Ducharme, G.R., 1991. Asymptotic Theory for Bootstrap Methods in Statistics. Centre de Recherches Mathématiques, Université de Montréal, Montréal, Canada.

Besag, J., Diggle, P.J., 1977. Simple Monte Carlo tests for spatial pattern. Appl. Stat. 26, 327–333.

Birnbaum, Z.W., 1974. Computers and unconventional test-statistics. In: Proschan, F., Serfling, R.J. (Eds.), Reliability and Biometry: Statistical Analysis of Lifelength. SIAM, Philadelphia, PA, pp. 441–458.

Bulca, B., Arslan, K., 2013. Surfaces given with the Monge patch in E4. J. Math. Phys. Anal. Geom. 9 (4), 435–447. ISSN 18129471. https://doi.org/10.18287/0134-2452-2015-39-4-459-461.

Černý, V., 1985. Thermodynamical approach to the traveling salesman problem: an efficient simulation algorithm. J. Optim. Theory Appl. 45 (1), 41–51. ISSN 00223239. https://doi.org/10.1007/BF00940812. http://link.springer.com/article/10.1007/BF00940812.

Chernick, M.R., 1999. Bootstrap Methods: A Practitioner's Guide. John Wiley & Sons, New York.

Clayden, J., 2014. General-Purpose Optimisation With the Self-Organising Migrating Algorithm. https://cran.r-project.org/package=soma.

Coudin, E., Dufour, J.-M., 2009. Finite-sample distribution-free inference in linear median regressions under heteroskedasticity and nonlinear dependence of unknown form. Econ. J. 12 (S1), S19–S49. (10th anniversary special edition).

Davison, A.C., Hinkley, D.V., 1997. Bootstrap Methods and Their Application. Cambridge University Press, Cambridge (UK).

Dufour, J.-M., 1990. Exact tests and confidence sets in linear regressions with autocorrelated errors. Econometrica 58, 475–494.

Dufour, J.-M., 2006. Monte Carlo tests with nuisance parameters: a general approach to finite-sample inference and nonstandard asymptotics. J. Econ. 133 (2), 443–477. https://doi.org/10.1016/j.jeconom.2005.06.007.

Dufour, J.-M., Farhat, A., 2001. Exact Nonparametric Two-Sample Homogeneity Tests for Possibly Discrete Distributions. Département de sciences économiques, Université de Montréal. Technical Report 2001-23. https://papyrus.bib.umontreal.ca/xmlui/handle/1866/362%5Cnpapers3://publication/uuid/C96.

Dufour, J.-M., Farhat, A., 2002. Exact nonparametric two-sample homogeneity tests. In: Huber-Carol, C., Balakhrishnan, N., Nikulin, M., Mesbah, M. (Eds.), Proceedings of the 2000 International Workshop on "Goodness-of-Fit Tests and Validity of Models". Birkhaüser, Boston, MA, pp. 435–448.

Dufour, J.-M., Jouini, T., 2006. Finite-sample simulation-based tests in VAR models with applications to Granger causality testing. J. Econ. 135 (1–2), 229–254.

Dufour, J.-M., Khalaf, L., 2001. Monte Carlo test methods in econometrics. In: Baltagi, B. (Ed.), Companion to Theoretical Econometrics. Blackwell Companions to Contemporary Economics. Basil Blackwell, Oxford, UK, pp. 494–519.

Dufour, J.-M., Khalaf, L., 2002. Simulation based finite and large sample tests in multivariate regressions. J. Econ. 111 (2), 303–322.

Dufour, J.-M., Kiviet, J.F., 1998. Exact inference methods for first-order autoregressive distributed lag models. Econometrica 66, 79–104.

Dufour, J.-M., Neves, J., 2019. Package 'MaxMC': maximized Monte Carlo. CRAN Package.

Dufour, J.M., Farhat, A., Gardiol, L., Khalaf, L., 1998. Simulation-based finite sample normality tests in linear regressions. Econ. J. 1, 154–173.

Dufour, J.-M., Khalaf, L., Beaulieu, M.C., 2003. Exact skewness-kurtosis tests for multivariate normality and goodness-of-fit in multivariate regressions with application to asset pricing models. Oxf. Bull. Econ. Stat. 65, 891–906.

Dufour, J.-M., Khalaf, L., Beaulieu, M.C., 2010. Multivariate residual-based finite-sample tests for serial dependence and GARCH with applications to asset pricing models. J. Appl. Econ. 25 (2), 263–285.

Dwass, M., 1957. Modified randomization tests for nonparametric hypotheses. Ann. Math. Stat. 28, 181–187.

Eberhart, R., Kennedy, J., 1995. A new optimizer using particle swarm theory. In: MHS'95. Proceedings of the Sixth International Symposium on Micro Machine and Human Science, New York, NY, vol. 1, pp. 39–43. http://ieeexplore.ieee.org/document/494215/.

Edgington, E.S., 1980. Randomization Tests. Marcel Dekker, New York.

Edwards, D., 1985. Exact simulation-based inference: a survey, with additions. J. Stat. Comput. Simul. 22, 307–326.

Edwards, D., Berry, J.J., 1987. The efficiency of simulation-based multiple comparisons. Biometrics 43, 913–928.

Efron, B., 1982. The Jacknife, the Bootstrap and Other Resampling Plans. CBS-NSF Regional Conference Series in Applied Mathematics, Monograph No. 38, Society for Industrial and Applied Mathematics, Philadelphia, PA.

Efron, B., Tibshirani, R.J., 1993. An Introduction to the Bootstrap. Monographs on Statistics and Applied Probability, vol. 57. Chapman & Hall, New York.

Fisher, R.A., 1935. The fiducial argument in statistical inference. Ann. Eugen. 6 (4), 391–398. ISSN 20501420. https://doi.org/10.1111/j.1469-1809.1935.tb02120.x. http://doi.wiley.com/10.1111/j.1469-1809.1935.tb02120.x.

Fisher, R.A., 1941. The asymptotic approach to Behrens's integral, with further tables for the d test of significance. Ann. Eugen. 11 (1), 141–172. ISSN 20501420. https://doi.org/10.1111/j.1469-1809.1941.tb02281.x. http://doi.wiley.com/10.1111/j.1469-1809.1941.tb02281.x.

Foutz, R.V., 1980. A method for constructing exact tests from test statistics that have unknown null distributions. J. Stat. Comput. Simul. 10, 187–193.

Fuller, W.A., 1996. Introduction to Statistical Time Series. John Wiley, ISBN: 0471552399721. https://doi.org/10.1016/0167-9473(96)88922-5.

Gilli, M., Maringer, D., Schumann, E., 2011. Numerical Methods and Optimization in Finance. Academic Press, Cambridge, MA, 1–14. http://linkinghub.elsevier.com/retrieve/pii/B9780123756626000018.

Goffe, W.L., Ferrier, G.D., Rogers, J., 1994. Global optimization of statistical functions with simulated annealing. J. Econ. 60, 65–99.

Hájek, J., 1969. A Course in Nonparametric Statistics. Holden-Day, San Francisco.

Hall, P., 1992. The Bootstrap and Edgeworth Expansion. Springer-Verlag, New York.

Holland, J.H., 1992. Adaptation in Natural and Artificial Systems. MIT Press, Cambridge, MA.

Hope, A.C.A., 1968. A simplified Monte Carlo test procedure. J. R. Stat. Soc. Ser. B 30, 582–598.

Horowitz, J.L., 1997. Bootstrap methods in econometrics: theory and numerical performance. In: Kreps, D.M., Wallis, K.F. (Eds.), Advances in Economics and Econometrics: Theory and Applications. Seventh World Congress, vol. 3. Cambridge University Press, Cambridge, UK, pp. 188–222.

Jeong, J., Maddala, G.S., 1993. A perspective on application of bootstrap methods in econometrics. In: Maddala, G.S., Rao, C.R., Vinod, H.D. (Eds.), Handbook of Statistics 11: Econometrics. North-Holland, Amsterdam, pp. 573–610.

Jöckel, K.H., 1986. Finite sample properties and asymptotic efficiency of Monte Carlo tests. Ann. Stat. 14, 336–347.

Katharine, M., 2014. Continuous global optimization in R. J. Stat. Softw. 60 (6), 1–45. ISSN 19390068. https://doi.org/10.18637/jss.v060.i06.

Kirkpatrick, S., Gelatt, C.D., Vecchi, M.P., 1983. Optimization by simulated annealing. Science 220 (4598), 671–680. ISSN 00368075. https://doi.org/10.1126/science.220.4598.671.

Kiviet, J., Dufour, J.M., 1997. Exact tests in single equation autoregressive distributed lag models. J. Econ 80, 325–353.

MacKinnon, J.G., 1999. Critical values for cointegration tests in heterogeneous panels with multiple regressors. Oxf. Bull. Econ. Stat 61 (s1), 653–670. ISSN 0305-9049. https://doi.org/10.1111/1468-0084.61.s1.14. http://doi.wiley.com/10.1111/1468-0084.61.s1.14.

Marriott, F.H.C., 1979. Barnard's Monte Carlo tests: how many simulations? Appl. Stat. 28, 75–77.

Mersmann, O., 2015. microbenchmark: Accurate Timing Functions. R Package. https://cran.r-project.org/package=microbenchmark.

Ripley, B.D., 1981. Spatial Statistics. John Wiley & Sons, New York.

Sasaki, T., Massaki, N., Kubo, T., 2005. Wolbachia variant that induces two distinct reproductive phenotypes in different hosts. Heredity 95 (5), 389–393. ISSN 0018067X. https://doi.org/10.1038/sj.hdy.6800737. https://www.r-project.org/.

Satman, M.H., 2013. Machine coded genetic algorithms for real parameter optimization problems. Gazi Univ. J. Sci 26 (1), 85–95. ISSN 13039709. http://gujs.gazi.edu.tr/article/view/1060000982.

Scrucca, L., 2013. GA: a package for genetic algorithms in R. J. Stat. Softw 53 (4), 1–37. ISSN 1548-7660. https://doi.org/10.18637/jss.v053.i04. http://www.jstatsoft.org/v53/i04/.

Scrucca, L., 2016. On some extensions to GA package: hybrid optimisation, parallelisation and islands evolution. ISSN 20734859. 1605.01931. http://arxiv.org/abs/1605.01931.

Shao, S., Tu, D., 1995. The Jackknife and Bootstrap. Springer-Verlag, New York.

Shi, Y., Eberhart, R., 1998. A modified particle swarm optimizer. In: Proceedings of the IEEE Congress on Evolutionary ComputationIEEE, pp. 69–73.

Smirnov, N., 1948. Table for estimating the goodness of fit of empirical distributions. Ann. Math. Stat. 19 (2), 279–281. ISSN 0003-4851. https://doi.org/10.1214/aoms/1177730256. http://projecteuclid.org/euclid.aoms/1177730256.

Venables, W.N., Ripley, B.D., 2003. Modern Applied Statistics With S, fourth ed. Springer-Verlag, Berlin and New York.

Vinod, H.D., 1993. Bootstrap methods: applications in econometrics. In: Maddala, G.S., Rao, C.R., Vinod, H.D. (Eds.), Handbook of Statistics 11: Econometrics. North-Holland, Amsterdam, pp. 629–661.

Vinod, H.D., López-de Lacalle, J., 2009. Maximum entropy bootstrap for time series: the meboot R package. J. Stat. Softw. 29 (5), 1–19.

Weir, J.B.V., 1960. Significance of the difference between two means when the population variances may be unequal. Nature 187 (4735), 438. ISSN 00280836. https://doi.org/10.1038/187438a0.

Welch, B.L., 1947. The generalization of 'student's' problem when several different population variances are involved. Biometrika 34 (1/2), 28. ISSN 00063444. https://doi.org/10.2307/2332510. http://www.jstor.org/stable/2332510?origin=crossref.

Wickham, H., 2009. Elegant Graphics for Data Analysis. Springer-Verlag, New York.

Willighagen, E., 2005. Genalg: R Based Genetic Algorithm. https://cran.r-project.org/package=genalg.

Wuertz, D., 2009. \pkg{fUnitRoots}: Trends and Unit Roots. http://cran.r-project.org/package=fUnitRoots.

Xiang, Y., Gubian, S., Suomela, B., Hoeng, J., 2013. Generalized simulated annealing for global optimization: the GenSA package. R J. 5, 13–28. ISSN 20734859. http://rjournal.github.io/archive/2013-1/xiang-gubian-suomela-etal.pdf.

Zambrano-Bigiarini, M., 2013. Particle Swarm Optimisation, With Focus on Environmental Models. http://www.rforge.net/hydroPSO,%5Cnhttp://cran.r-project.org/web/packages/hydroPSO.

This page intentionally left blank

Chapter 2

New exogeneity tests and causal paths

Hrishikesh D. Vinod*
Fordham University, Bronx, NY, United States
**Corresponding author: e-mail: vinod@fordham.edu*

Abstract
We modify Suppes' probabilistic causality theory by replacing inequalities among probabilities of events by unequal residuals of flipped kernel regression conditional expectation functions, $E\,f(X_j|X_i, X_k)$ and $E\,f(X_i|X_j, X_k)$, allowing asymmetry. Using three criteria we aggregate evidence from four orders of stochastic dominance and new asymmetric partial correlation coefficients to develop a unanimity index: *UI*. The index yields a decision rule for X_i to be self-driven or exogenous, based on confirming the causal paths $X_i \rightarrow X_j$. Thus *UI* can replace Hausman–Wu's indirect exogeneity test which diagnoses endogeneity (a disease) by showing that instrumental variables (IV) estimator remedy "works." A simulation supports our decision rules. An illustration identifies exogenous variables which can help predict US economic recession.

Keywords: Kernel regression, Stochastic dominance, Bootstrap inference, Instrumental variables, Directed acyclic graphs

1 Introduction

Estimation and inference regarding causal directions is a fundamental challenge in many sciences, which is linked to a long-standing need in econometrics for testing exogeneity without exclusively relying on instrumental variables (IV). This chapter develops new empirical approaches for causal paths and exogeneity testing using only passively observed data within econometrics paradigm by extending the Cowles commission simultaneous equation models (CC-SEM) and Koopmans (1950).

Consider a set of p random variables $V = (X_1, \ldots, X_p)$, with subscripts from the index set $V_I = \{1, \ldots, p\}$ and joint density $f_{jt}(V)$. The marginal density $f(X_i)$ is the Radon–Nikodym derivative of the joint density, either with respect to the Lebesgue or the counting measure. This chapter provides new

Handbook of Statistics, Vol. 41. https://doi.org/10.1016/bs.host.2018.11.011

tools for exploratory assessment of causal relations, which can be described by regression equations in CC-SEM between passively observed data generating processes (DGPs) in V.

Remark 1. (Limited scope of our causality discussion). Philosophers have debated causality concepts for over a millennium. The reader can check philosophical references in Reiss (2016) and the Internet Stanford Encyclopedia, Zalta (2018). Deterministic causal relations expressed as functional relations without random components are outside the scope of this chapter. For example, Boyle's law (pressure * volume = a constant) where all component variables (pressure and volume) can be independently controlled in a laboratory is considered deterministic causality here. Situations where a known treatment causes a known outcome (effect), and the problem is assessing the size of treatment effects, such as those considered in Imbens and Rubin (2015), are also outside the scope of this chapter.

An alternative to deterministic causality which admits noise implying rare violations was proposed by Suppes (1970). The intuitive idea behind Suppes' "probabilistic theory of causation," is that if the event X_i causes the event X_j (implying the causal path $X_i \to X_j$) we should have

$$P(X_j|X_i) > P(X_j) \quad a.e., \tag{1}$$

where *a.e.* denotes "almost everywhere" in a relevant measure space. If the number of violations of the inequality (1) is too many, it will not be a set of measure zero, in terms of the Lebesgue measure. Then the inequality does not hold *a.e.* Some authors including Salmon (1977) have long known serious limitations of Suppes' theory, which are summarized in the following Lemma, with a new proof.

Lemma 1. *Suppes' condition* (1) *is neither necessary nor sufficient for causality.*

Proof. Let X_k denote an additional omitted cause which might be a confounder. It is possible to construct counter examples where the true causal paths are $(X_k \to X_i)$ and $(X_k \to X_j)$. For example, let X_i denote the event of atmospheric barometer falling sharply and let X_j denote a weather storm event. These X_i, X_j satisfy Eq. (1). However, the barometer itself does not cause the storm! The true cause X_k "falling atmospheric pressure" is hidden from (1). Since barometer reading X_i is not necessary for the storm event X_j, this is a counter example. Thus the "necessity" is rejected.

We reject sufficiency by using the definition of conditional probability as follows. Since conditional probability equals joint divided by marginal, we can rewrite (1) as

$$\frac{P(X_i \cap X_j)}{P(X_i)} > P(X_j),$$

or as

$$P(X_i \cap X_j) > P(X_i) * P(X_j).$$

The inequality's sense remains intact if we divide both sides by a positive quantity, $P(X_j) > 0$, to yield the inequality:

$$\frac{P(X_i \cap X_j)}{P(X_j)} > P(X_i).$$

Thus we must always have

$$P(X_i|X_j) > P(X_i) \quad a.e. \tag{2}$$

We have proved that Suppes' test satisfies conditions for $X_j \rightarrow X_i$ as well as $X_i \rightarrow X_j$ at all times. A result finding bidirectional causality $X_i \leftrightarrow X_j$ all the time means that the condition is (1) is not sufficient for $X_i \rightarrow X_j$. □

Some philosophers and economists (e.g., Clive Granger) have suggested that the path $X_i \rightarrow X_j$ should further require that X_i must occur in time before X_j occur, to help achieve asymmetry. However, this is needlessly restrictive and inapplicable for human agents (who read newspapers) acting strategically at time t in anticipation of events at time $t + 1$.

Remark 2. (Asymmetry via flipped models). Logically consistent probabilistic causality theory must retain robust asymmetry even when our causality testing condition(s) are stressed by flipping the cause and effect (X_i and X_j). Since Eqs. (1) and (2) suggesting opposite causal directions are proved to coexist, we need to go beyond the inequality signs and consider the relative magnitudes of the differences: $(P(X_j|X_i) - P(X_j))$, and $(P(X_i|X_j) - P(X_i))$, in order to generalize Suppes' nondeterministic theory.

Remark 3. (Confounders and controls distinguished). The causal path $X_i \rightarrow X_j$ assessment is often affected by two types of often present related events X_k. It is convenient to distinguish between two types of X_k: (i) "confounder" and (ii) "control" variables, even though the two may be synonymous for many readers. First, we define "confounder" X_k as the true underlying cause behind the apparent X_i for the outcome X_j. For example, the true cause of weather events X_j is "atmospheric pressure" X_k and not "barometer reading" as X_i. Second, we define X_k as a "control" event if both (X_i, X_k) may be causing X_j, but we are interested in knowing if X_i causes X_j over and above the effect of X_k. For example, let X_j be health outcome, X_i be some medicine, then X_k, the patient's age, is commonly used as a control. A confounder can be treated as a control, but the converse may not hold true.

Theorem 1 (Stochastic causality).

(a) *Assuming data on all confounders X_k defined in Remark 3 are available, the causal path $X_i \rightarrow X_j$ holds if and only if (iff)*

$$(P(X_j|X_i, X_k) - P(X_j|X_k)) > (P(X_i|X_j, X_k) - P(X_i|X_k)), \quad a.e. \tag{3}$$

(b) *Assuming data on all controls X_k defined in Remark 3 are available, the causal path $X_i \to X_j$ holds iff*

$$(P(X_j|X_i,\ X_k) - P(X_j)) > (P(X_i|X_j,\ X_k) - P(X_i)),\quad a.e. \tag{4}$$

Proof. We remove the obstacles to proving necessity in our proof of Lemma 1 by explicitly including X_k among conditions of Theorem 1. The obstacle to proving sufficiency of Suppes' condition arising from simultaneous existence of Eqs. (1) and (2) is removed here by going deeper into the magnitudes underlying inequalities and focusing on their relative sizes, not just signs. □

A philosopher Salmon (1977, p. 151) suggests replacing the probabilities of *events* appearing in Eq. (1) by causally connected *processes* defined as "spatio-temporally continuous entities" having their own physical status. That is, we need to replace probabilities of events involving $P(X_i, X_j, X_k)$ by densities $f(X_i, X_j, X_k)$ of data generating processes (DGPs). Then, the iff condition from Theorem 1 (a) becomes

$$(f(X_j|X_i,\ X_k) - f(X_j|X_k)) > (f(X_i|X_j,\ X_k) - f(X_i|X_k))\ \ a.e. \tag{5}$$

The slightly simpler iff condition from Theorem 1 (b) becomes

$$(f(X_j|X_i,\ X_k) - f(X_j)) > (f(X_i|X_j,\ X_k) - f(X_i))\ \ a.e. \tag{6}$$

When we accept Salmon's suggestion to consider causality in terms of variable DGPs instead of events, we have an important advantage. We can use widely accepted multiple regression to remove the effect of control variables, not readily available for probabilities of events.

However, conditions (5) and (6) involve difficult to quantify conditional densities. Hansen (2004) suggests a two-step estimator for conditional density estimation from nonparametric regression residuals. However, (5) and (6) also involve quantification of numerical differences between two (conditional) densities. Fortunately, financial economists have developed sophisticated tools for quantification of probabilistic "dominance" of one density over another, called "stochastic dominance" of various orders, discussed later (Definition 3). Hence direct quantification of (5) and (6) is feasible, but left for future work.

Instead of direct estimation of (5) and (6) it is easier to estimate fitted values of conditional expectation functions ($E\hat{f}$) available from nonparametric nonlinear kernel regressions. Our two-step method for implementing the condition in Theorem 1 (a) treats the presumed true cause X_k as a confounder. The first step uses kernel regression: $X_j = f(X_k) + error$, with residual $e_{jk}^{(1)} = X_j - E\hat{f}(X_k)$. Our second step uses kernel regression: $e_{jk}^{(1)} = f(X_i, X_k) + error$, leading to second step residual $e_{j|ik}^{(2)} = e_{jk}^{(1)} - E\hat{f}(X_i, X_k)$. Since it is customary to define residuals as observed minus fitted values, the regression residual value quantifies the negative of the left-hand side (LHS) of (5). The right-hand side of (5) is obtained by flipping i and j.

Theorem 1 (b) requiring only one step is a bit simpler to implement. One computes kernel regression residual: $e_{j|ik} = X_j - E\hat{f}(X_i, X_k)$, while treating all X_k as controls. Remark 3 notes that a confounder can be treated a control, but not vice versa. Since one usually does not know in advance whether X_k is a confounder or control, we should start treating X_k as control. When one suspects that X_k may be a strong cause of X_j overwhelming X_i, we can then implement Theorem 1 (a) requiring two steps.

Note that our one-step method approximates $f(X_j|X_i, X_k) - f(X_j)$ appearing on the LHS of the inequality (6) by the negative of kernel regression residual $e_{j|ik}$ described above. One can avoid working with such negatives on both sides by rewriting (6) as

$$(f(X_j) - f(X_j|X_i, X_k)) < (f(X_i) - f(X_i|X_j, X_k)) \quad a.e.$$

while replacing (>) by (<). Sample realizations from marginal densities appearing on both sides of (6), $f(X_j)$, $f(X_i)$, are simply the available data values: X_{jt}, X_{it}; $t = 1, \ldots, T$.

This chapter develops three criteria (Cr1–Cr3, described later in Section 4) which also rely on kernel regression residuals to help decide the causal path from nonexperimental data, without ruling out bidirectional causality: $X_i \leftrightarrow X_j$, in a subset of cases, consistent with a properly stochastic (nondeterministic) causality concept.

In short, we want to quantify stochastic causality of Theorem 1 after replacing probabilities by densities. We standardize all DGPs (X_i, X_j, X_k) to make sure that regression residual magnitudes on two sides of flipped kernel regressions are comparable. The fitted values ($E\hat{f}$) of kernel regressions mentioned above are sample realizations of conditional densities in the form of conditional expectation functions. Finally we are ready to propose a practical implementation of Theorem 1 (b) by defining the following.

Definition 1 (Stochastic kernel causality). Assuming three conditions: (A1) that conditional expectation functions are consistently estimated, (A2) that all DGPs are standardized, and (A3) that all X_k are control variables, we have the causal path $X_i \to X_j$ iff absolute errors in conditional expectation functions predicting X_j are "smaller" than similar errors in predicting X_i. Replacing true unknown errors by residuals we have:

$$|e_{j|ik}| = |X_j - E[\hat{f}(X_j|X_i, X_k)]| < |X_i - E[\hat{f}(X_i|X_j, X_k)]| = |e_{i|jk}|, \quad a.e., \qquad (7)$$

where the notation X_k refers to any number (0, 1, 2, …) of control variables. The operator E in Eq. (7) refers to *conditional expectation functions* readily available from kernel regressions. If there are T observations in our DGPs, we will need to consider inequalities among two sets of estimates of T errors, compared by using *stochastic* dominance methods described below. Hence the name "stochastic kernel causality" seems appropriate.

An intuitive example may be crime rate as X_i, police deployment rate as X_j and income in the locality as X_k. We conclude that $X_i \to X_j$ if X_i predicts X_j better (smaller absolute residuals) than vice versa, after allowing for X_k. A version without X_k using European data illustrated in a vignette of R package "generalCorr," Vinod (2017), implements (7).

1.1 Computational agenda and decision rules

Stochastic kernel causality of Definition 1 is an empirical not deterministic concept. Our agenda in the sequel is to develop three deeper manifestations of the inequalities in (7) into three empirical criteria, Cr1–Cr3, to further develop a sample unanimity index $ui \in [-100, 100]$, summarizing the three criteria into a single number. Now our proposed decision rules after choosing a threshold, $\tau = 5$, say, are

Ru.1: If $(ui < -\tau)$ the causal path is: $X_i \to X_j$
Ru.2: If $(ui > \tau)$ the causal path is: $X_j \to X_i$
Ru.3: If $(|ui| \leq \tau)$ we obtain bidirectional causality: $X_i \leftrightarrow X_j$, that is, the variables are jointly dependent.

CC-SEM literature refers to bidirectional causality as "endogeneity problem." It is generally solved by inserting another equation in a simultaneous equation model and by using two-stage least squares (2SLS) to replace the endogenous variable on the right-hand side (RHS) of the equation by the fitted values obtained from that additional equation. Even though the fitted values obtained from that additional equation can be viewed as an instrumental variable (IV), our criticism of instrumental variables for *testing* exogeneity in Section 3.1.2 does not apply to *estimations* similar to those involving 2SLS.

An outline of the remaining chapter follows. Section 2 briefly reviews kernel regressions, which can be skipped by readers who are familiar kernel regressions. Section 3 briefly describes the CC-SEM models linking endogenous, exogenous terminology with stochastic kernel causality. Section 4 describes a quantification of stochastic kernel causality by Cr1–Cr3. Section 8 reports a simulation of our decision rules. Section 9 considers statistical inference using the bootstrap. Section 10 considers a topical example of predicting recessions. Section 11 contains a summary and final remarks.

2 Kernel regression review

Linearity of the regression model is often a matter of convenience rather than an evidence-based choice. Back in 1784, the German philosopher Kant said: "Out of the crooked timber of humanity no straight thing was ever made." Since social sciences and medicine deal with human agents, evidence supporting linearity is often missing.

The main reason for using nonparametric nonlinear kernel regression in applied work is to avoid misspecification of the functional form. Best fitting kernel regression line is often jagged which does not have any polynomial or sinusoidal form. However, it provides a superior fit (higher R^2) by not assuming a functional form.

A disadvantage used to be computational difficulty, which has recently disappeared. Remaining disadvantages are that kernel regressions fail to provide partial derivatives and that out-of-sample forecasts can be poor. Determination of causal paths is unaffected by these disadvantages.

Assuming that variables X_k are absent for ease of exposition, without loss of generality (wlog), let us denote by *LjRi* a model having X_j on the LHS and X_i on the RHS to be estimated by a nonlinear nonparametric kernel regression:

$$LjRi: \quad X_{jt} = G_1(X_{it}) + \epsilon_{j|i}, \quad t = 1, \ldots, T, \tag{8}$$

where errors are no longer Normal and independent. Our nonparametric estimate $g_1(X)$ of the population conditional mean function $G_1(X)$ is

$$g_1(X) = \frac{\sum_{t=1}^{T} X_{jt} K\left(\frac{X_{it} - X}{h}\right)}{\sum_{t=1}^{T} K\left(\frac{X_{it} - X}{h}\right)}, \tag{9}$$

where $K(.)$ is the well-known Gaussian kernel function and h is the bandwidth often chosen by leave-one-out cross validation, Li and Racine (2007) and Vinod (2008, Sec. 8.4). It is well known that kernel regression fits are superior to OLS.

The flipped kernel regression *LiRj* is obtained by interchanging X_j and X_i in Eq. (8).

$$LiRj: \quad X_{it} = G_2(X_{jt}) + \epsilon_{i|j}, \quad t = 1, \ldots, T. \tag{10}$$

Proposition 1 (Optimality of g_1). *Assume that g_1 in Eq. (9) belongs to $\mathcal{B}$, a class of Borel measurable functions having finite second moment, then g_1 is an optimal predictor of of X_j given X_i, in the sense that it minimizes the mean squared error (MSE) in the class of Borel measurable functions.*

Proposition 1 is proved as Theorem 2.1 in Li and Racine (2007).

Proposition 2 (Kernel regression is CAN). *Assume that*

- **(i)** *$\{X_{it}, X_{jt}\}$ are iid, and $g_1(x)$, joint density as well as error variance functions are twice differentiable.*
- **(ii)** *K is a bounded second order kernel.*
- **(iii)** *As $T \to \infty$, $Th^3 \to \infty$, and $Th^7 \to 0$.*

Then kernel regression estimate of the conditional expectation function g_1 is consistent and asymptotically normal (CAN).

Proof. See Theorem 2.7 of Li and Racine (2007) for further details and extensions to multivariate and local polynomial generalizations, including a proof of consistency and asymptotic normality. □

2.1 Counterfactuals in kernel regressions

Counterfactuals are defined as "what has not happened but could happen" in available data. Since experimental manipulation is often not an option, especially in social sciences, many authors use virtual manipulation involving counterfactuals, implicit in cross validation described next.

Proposition 3 (Counterfactuals in cross validation). *Considering* $\{X_{it}, X_{jt}\}$ *data, when we pretend that t-th observation is absent, even though it is present, we have a counterfactual. Now leave-one-out cross validation used to determine bandwidth h appearing in* (9) *of kernel regressions minimizes a weighted error sum of squares*

$$min_h \ \frac{1}{T}\Sigma_t[Y_t - \hat{g}_{1,-tL}]^2 W(X_t), \tag{11}$$

where $W(.)$ *is a weight function, subscript* $(-t)$ *denotes omitting t-th observation and where the subscript* (L) *refers to local linear fit. We employ cross validation as a counterfactual in our determination of* (g_1, g_2) *conditional expectation functions, which in turn determine our causal direction and its strength. This is explained in the sequel.*

2.2 Kernel regression and consistency

It is straightforward to also include control (confounding) variables X_k in Eqs. (8) and (10). Let LjRi now suggest left-side X_j right-side (X_i, X_k).

$$LjRi: \ \ X_{jt} = G_1(X_{it}, X_{kt}) + \epsilon_{j|ik}, \ \ t = 1, \ldots, T. \tag{12}$$

The kernel regression estimate g_1 of the conditional expectation function $G_1 = E(X_j|X_i, X_k)$ is consistent, only if the true unknown errors in Eq. (12) are orthogonal to the regressors with probability limit satisfying:

$$\text{plim}_{T\to\infty}(\epsilon_{j|ik}X_{it})/T = 0. \tag{13}$$

If the true relation is nonlinear, a high order polynomial, say, researcher using a linear model is implicitly letting high order terms merge into the regression error. Since the merged error is correlated with the regressor due to misspecification, it will induce endogeneity. Our use of nonparametric kernel regressions prevents endogeneity induced by hidden nonlinearities.

3 Cowles commission SEMs

Koopmans (1950) formulated the consistency requirement of Eq. (13) as exogeneity of X_i and went on to require that each RHS variables should "approximately cause" the LHS variable. He was not interested in determining causal path itself. Engle et al. (1983) (hereafter "EHR") were the first to use flipped models to show that linear regression of X_i on X_j and vice versa have identical R^2 values, implying that exogeneity test based on causality is ambiguous. Our use of nonlinear nonparametric kernel regressions removes identical R^2 values, and the ambiguity. More important, it suggests that Eq. (13) provides an important causality criterion which appears to have been ignored in philosophical causality literature.

All references to causality became out of favor in the CC-SEM literature due to EHR and Holland (1986) (with discussion by many experts including Rubin and Granger). The prevailing view was that causality must be deterministic, not stochastic following Rubin's dictum: "No causation without manipulation." Deterministic causality is often too time-consuming and costly, if not utterly impractical with observational data.

This section further explores the link between stochastic causality and CC-SEM methods for further insights. For example, presence of endogenous variables implying bidirectional causality, $X_i \leftrightarrow X_j$, is accepted as a fact of life, implying that (X_i, X_j) are jointly dependent variables. CC-SEM literature refers to it as the "endogeneity problem" or even disease (Bound et al., 1995).

An example from CC-SEM exhibiting endogeneity problem for X_j is

$$X_i = f_1(X_j, X_k) + u_1 \tag{14}$$

$$X_j = f_2(X_i) + u_2, \tag{15}$$

where two variables (X_i, X_j) are endogenous (also known as jointly dependent) and one variable X_k is exogenous. The specification of the LHS and RHS variables is based on economic theory and researcher judgment that RHS variables are exogenous or approximately causal.

Models with similar equations are called structural equation models (SEM) with additive noise in graph theory literature, briefly reviewed in our Appendix A. A graph $\mathcal{G} = (V, \mathcal{E})$ consists of nodes from the index set V_I and edges $\mathcal{E} \subseteq V_I^2$ connecting them, after ruling out any edge from a node to itself. Graph theory links the p variables in V with edges and directed arrows signifying causal paths. Directed acyclic graphs (DAGs) and Bayesian nets developed by computer scientists and others are reviewed by Peters et al. (2014), Pearl (2009), and Reiss (2016). Our Eq. (14) implies two causal paths: $(X_j \to X_i,\ \ X_k \to X_i)$, whereas Eq. (15) implies the path: $(X_k \to X_j)$. Eqs. (14) and (15) together imply $(X_k \to X_j \to X_i)$ which is both directed and noncyclic. Hence, it is a directed acyclic graph, or DAG.

CC-SEM methods suggest writing a reduced form of equations obtained by replacing X_j in (14) by $f_2(X_k)$, making the RHS a more complicated function $f_3(X_k)$. Thus we have exogenous variables on the RHSs of both equations. These equations are "identified" if one can uniquely determine the structural equation coefficients from reduced form coefficients. If f_1, f_2 are linear regressions the model is not identified. If they are nonlinear nonparametric, they are identified, since f_3 will not be confused with f_2.

3.1 Need for alternative exogeneity tests

Current literature offers two approaches for exogeneity testing: (i) using sequential cuts of a joint density to assess weak exogeneity, and (ii) Hausman–Wu type testing based on instrumental variables. We motivate our approach by discussing the limitations of these approaches in the next two sections.

3.1.1 Weak exogeneity and its limitations

Write a joint density as a product of a conditional and marginal density in two ways upon flipping X_i with X_j:

$$f(X_j, X_i) = f(X_j|X_i) \times f(X_i), \tag{16}$$

$$= f'(X_i|X_j) \times f'(X_j). \tag{17}$$

In the absence of nonparametric tools, EHR rewrite Eq. (16) after conditioning on explicit parameters $\lambda = (\lambda_1, \lambda_2)$ as:

$$f(X_j, X_i|\lambda) = f(X_j|X_i, \lambda_1) \times f(X_i|\lambda_2), \tag{18}$$

related to a factoring of the likelihood function, needed for maximum likelihood (ML) estimation. Now EHR's widely accepted "weak exogeneity" is complicated, because it requires Eq. (18) to implement a "sequential cut" extending Barndorff-Nielsen notion of a cut for the exponential family of distributions.

Definition 2 (EHR weak exogeneity). X_i is weakly exogenous for parameters of interest, ψ, if there exists a reparameterization $\lambda = (\lambda_1, \lambda_2)$ where

(i) ψ is a function of λ_1, and
(ii) $[f(X_j|X_i, \lambda_1) \times f(X_i|\lambda_2)]$ operates a "sequential cut" defined in Eq. (18).

Properties of EHR weak exogeneity.

[WE1] Parameter distinctions: A distinction between parameters of interest, ψ, and other (nuisance) parameters λ_2 is a crucial part of the definition.

[WE2] Granger causality irrelevant: EHR state (p. 290) that "Granger noncausality is neither necessary nor sufficient for weak exogeneity."

[WE3] Invariance: EHR assume that ψ are invariant to policy changes to avoid the famous Lucas critique.

[WE4] Inability to test: EHR flip a two-equation simultaneous equations model (their equations numbered 27 and 28 vs 30 and 31) to argue on page 288 that "the choice of parameters of interest is the sole determinant of weak exogeneity, which is, therefore not directly testable."

3.1.2 Hausman–Wu test and its limitations

Lacking a direct exogeneity test, Wu (1973) had originally provided an indirect exogeneity test, which was later popularized as the Hausman–Wu test. It defines a vector of contrasts, $d = b_{OLS} - b_{IV}$, between ordinary least squares (OLS), an efficient but potentially inconsistent (due to endogeneity) estimator on the one hand, and another inefficient but consistent (by assumption) IV estimator. The covariance matrix of d can be shown to be $V_d = V(b_{IV}) - V(b_{OLS})$, and a quadratic form, $d'(V_d)^{-1}d$, is asymptotically a $\chi^2(p)$, with p degrees of freedom. The Hausman–Wu test amounts to medieval diagnosing of a disease (endogeneity) by showing that a cure (b_{IV}) works.

Actually, the IV remedy has been found to be seriously flawed as shown by Bound et al. (1993) with a provocative title *"the cure can be worse than the disease"* and Bound et al. (1995). Of course, there are several applications including two-stage least squares where IV estimators are extremely useful. One must distinguish between IV-based tests and IV estimators used to remedy endogeneity.

3.1.3 Limitations of IV-based tests

Certain caution is needed in using IV-based tests for exogeneity. For example, authors, including Bound et al. (1993) and Kiviet and Niemczyk (2007), have warned that in finite samples instrumental variable IV estimators "have systematic estimation errors too, and may even have no finite moments." Moreover they can be very inefficient (even in large samples) and unnecessarily change the original specification. We list the following disadvantages:

(IV.1) One must replace each X_i with ad hoc, potentially weak and/or irrelevant instrumental variable from a set $\widetilde{Z}$, before testing for exogeneity of X_i. The set $\widetilde{Z}$ needs to be exhaustive and each element needs to be available to the researcher.
(IV.2) The test needs to be repeated for each potential $\widetilde{Z}_i$ replacing each X_i.
(IV.3) Davidson and MacKinnon (1993, p. 241) show that degrees of freedom p for the $\chi^2(p)$ test are too large when only a subset of p variables in V are exogenous.
(IV.4) The Chi-square sampling distribution is subject to unverified assumptions of linearity and normality, especially unrealistic for human subjects in finite samples.

3.1.4 OLS super-consistency implications

Stock (1987) considers OLS estimate of marginal propensity to consume (MPC) for the Keynesian consumption function when both consumption and income are nonstationary, or measured in levels. Stock proves (p. 1042) that Haavelmo's "simultaneous equations bias" disappears asymptotically, because OLS is super-consistent, reaching the true value at a fast asymptotic rate of T. Since the bias will remain present in finite samples, our Cr1 of (21) chooses the flip with a smaller potential bias. If we are estimating relations involving nonstationary variables, endogeneity will not necessarily induce asymptotic inconsistency.

3.1.5 CC-SEM implications for stochastic kernel causality

The properties WE1–WE4, and the four disadvantages of indirect exogeneity testing listed as IV.1–IV.4 make it clear that a direct test for exogeneity is much needed. Our direct test follows Koopmans in requiring that RHS exogenous variable should approximately cause the LHS variable. The Hausman–Wu test criterion will be incorporated in the first criterion for causality (Cr1) in Eq. (21) explained in the next section. Hence exogeneity of X_i in the LjRi model based on Eq. (12) can be tested by using our decision rule for assessing the causal path $X_i \rightarrow X_j$ described later in Section 7.

4 Stochastic kernel causality by three criteria

Having defined kernel regressions and CC-SEM exogeneity methods we are ready to consider deeper manifestations on the inequalities (7) in Definition 1 of stochastic kernel causality. First define two conditional expectation functions estimated by flipped kernel regressions as:

$$g_{j|ik} = E[\hat{f}(X_j|X_i,X_k)], \quad g_{i|jk} = E[\hat{f}(X_i|X_j,X_k)]. \tag{19}$$

Now our Definition 1 of (7) under assumptions A1–A3 for the causal path $X_i \rightarrow X_j$ requires the kernel regression residuals to satisfy for $t = 1, \ldots, T$

$$|e_{j|ik}| = |X_j - g_{j|ik}| < |X_i - g_{i|jk}| = |e_{i|jk}|, \ a.e. \tag{20}$$

The "a.e." inequality (20) among T numbers can manifest itself in many distinct ways. Our criteria employ sophisticated comparisons of T numbers by comparing their densities, not just summary statistics. Stochastic dominance methods of four orders compare rolling moments (mean, variance, skewness, kurtosis) defined locally over rolling small neighborhoods.

Thus we are ready to fulfill our "computational agenda" mentioned in Section 1.1 requiring definitions of criteria Cr1–Cr3, ultimately leading to a sample unanimity index *ui*.

Our first criterion Cr1 described next evaluates finite sample implications of assumption A1 requiring consistency of conditional expectation functions,

$g_{j|ik}$ and $g_{i|jk}$. We plug observable residuals into the (consistency) exogeneity condition (13), yielding two sets of T multiplications $e_{j|ik}X_t$ and $e_{i|jk}Y_t$. Our Cr1 assumes that closeness to zero of these expressions reveals relative speeds of convergence.

4.1 First criterion Cr1 for $X_i \rightarrow X_j$

Since Kernel regressions are CAN (Proposition 2) the conditional expectation functions ($g_{j|ik}$, $g_{i|jk}$) are consistent. Since speeds of convergence can differ, one should prefer the conditioning with a faster convergence rate. The obvious finite sample indicators of speeds of convergence are available from Eq. (13) when we replace the true unknown errors by residuals. If $g_{j|ik}$ converges faster to its true value than $g_{i|jk}$, the T values implied by the "plim" expression of the LjRi model should be closer to zero than the similar "plim" expression of the flipped LiRj model. Now the condition for the causal path $X_i \rightarrow X_j$ making X_i more exogenous in the LjRi model, than X_j is exogenous in the flipped LiRj model, is the inequality:

$$\text{Cr1}: \ |e_{j|ik}X_t| < |e_{i|jk}Y_t|, \ \ a.e. \tag{21}$$

4.2 Second criterion Cr2 for $X_i \rightarrow X_j$

The validity of the causal path $X \rightarrow Y$ requires that independent changes in X_i lead to (dependent) changes in X_j, leading to LjRi model providing a better fit than LiRj. Note that the fit is measured by the size of residuals which are numerically comparable due to assumption (A2). Hence we require

$$\text{Cr2}: \ |e_{j|ik}| < |e_{i|jk}|, \ \ a.e., \tag{22}$$

which defines Cr2 from Eq. (20).

4.3 Third criterion Cr3 for $X_i \rightarrow X_j$

Following Vinod (2014) an aggregate manifestation of the "a.e." inequality (7) involving residuals: $e_{j|ik}$, $e_{i|jk}$ can be stated in terms of a higher coefficient of determination R^2 for one of the two flipped models. The effect of X_k variable(s), if any, on X_i, X_j is netted out in these computations to yield our third criterion:

$$\text{Cr3}: \ R^2_{j|i,k} = 1 - \frac{\Sigma(e_{j|ik})^2}{(\text{TSS})} > 1 - \frac{\Sigma(e_{i|jk})^2}{(\text{TSS})} = R^2_{i|j,k}, \tag{23}$$

where TSS denotes the total sum of squares, which is $(T-1)$ for standardized data, and where the conditioning in the two models is denoted by subscripts to the R^2.

An equivalent requirement using generalized partial correlation coefficients from Vinod (2017) for $X_i \rightarrow X_j$ is

$$|r^*_{(j|i;\,k)}| > |r^*_{(i|j;\,k)}|. \tag{24}$$

An advantage of Cr3 is that it can be computed without having to standardize the data.

R package "generalCorr" reports the generalized partial correlation coefficients in Eq. (24), if desired. The R function "pacorMany" provides partial correlation coefficients of first column with all others. Recall that Theorem 1 (a) Eq. (5) considers netting out of the confounders X_k from both causal X_i and outcome X_j variables, exactly as it is implemented in computing (24).

5 Numerical evaluation of Cr1 and Cr2

This section provides further remarks on numerical evaluation of inequalities in (21) and (22) which hold *a.e*. That is, a "small" number of inequality reversals are permissible. Fortunately, financial economists have a solution to the quantification of a choice problem characterized by (fuzzy) inequalities which are violated for subsets of points.

Suppose an investor has data on two probability distributions of returns (r_{at}, r_{bt}). If these returns satisfy the inequality ($r_{at} > r_{bt}$), *a.e*., then the clear choice is investment "a." However, real world portfolios 'a' almost always beating "b" are very rare. Moreover investors often want to compare not only the mean, but also the variance, skewness, and kurtosis, if not the entire distribution of returns. Hence, financial economists have developed a concept of stochastic dominance of four orders to, respectively, compare the local mean, variance, skewness, and kurtosis.

This section develops Cu(sd1) to Cu(sd4) as four sets numbers to quantify all four important features of "fuzzy" inequalities between two densities in (21) and (22). This will eventually lead to the unanimity index. Of course, investors seek *higher* returns, while we seek *lower* absolute values of residuals in Eqs. (21) and (22), implying that we must change the sign before using the Finance algorithm. Ours is claimed to be one of the first applications of stochastic dominance in resolving fuzzy inequalities unrelated to Finance.

Definition 3 (Stochastic Dominance). Density $f(X)$ dominates another density $f(Y)$ in the first order if the respective empirical cumulative distribution functions (ecdf) satisfy: $F(x) \leq F(y)$.

It may be unintuitive but true that the dominating density $f(X)$ with larger magnitudes has a smaller cumulative density $F(x)$ in Definition 3. The stochastic dominance is surveyed in Levy (1992) and four orders of stochastic dominance are discussed next.

6 Stochastic dominance of four Orders

The first order stochastic dominance (SD1) is defined in Definition 2. It is well known that SD1 provides a comprehensive picture of the ranking between two probability distributions with a focus on locally defined first moment (mean). This section attempts to discuss quantification of SD1 to SD4 following the theory and software available in Vinod (2008, chap. 4).

The underlying computation requires bringing the two densities on a common "support," requiring ecdf's to have up to $2T$ possible jumps or steps. Hence there are $2T$ estimates of $F(x) - F(y)$ denoted by a $2T \times 1$ vector (sd1). Anderson (1996) shows how a simple premultiplication by a large patterned matrix implements computation of (sd1). Let us use a simple cumulative sum, Cu(sd1), whose sign (+ 1, 0, −1) helps summarize the first order stochastic dominance into only one number.

Second order dominance (SD2) of $f(x)$ over $f(y)$ requires further integrals of ecdf's to satisfy: $\int F(x) \leq \int F(y)$. One computes the numerical integral by using the trapezoidal rule described in terms of a large patterned matrix whose details are given in Vinod (2008, chap. 4) and in Anderson (1996). The $2T$ estimates of SD2 denoted by (sd2) are locally defined variances. Their simple cumulative sum is denoted as Cu(sd2), whose sign (+ 1, 0, −1) summarizes the information regarding second order dominance.

Similarly, SD of order 3 is estimated by a vector (sd3) of $2T$ locally defined skewness values defined from $\int\int F(x) \leq \int\int F(y)$. The sd3 is further summarized by the sign of Cu(sd3).

Analogous SD of order 4 for kurtosis requires $\int\int\int F(x) \leq \int\int\int F(y)$ and measures investor "prudence" according to Vinod (2004). Cumulative sum of point-wise kurtosis estimates of SD4 are Cu(sd4), whose sign (+ 1, 0, −1) summarizes the SD4 dominance information.

Remark 4. Dominance of four orders associated with the first four moments yield four $2T \times 1$ vectors: sd1–sd4. Their cumulative sums, are denoted as Cu(sd1) to Cu(sd4), whose signs are generally the same as the sign of their averages. These signs are indicators of the overall direction of the inequality suggested by T distinct signs of sd1–sd4 values.

6.1 Weighted sum of signs of Cu(sd1) to Cu(sd4)

Now we quantify the three inequality criteria. The inequalities for criteria Cr1 and Cr2 are fuzzy, requiring the use of stochastic dominance methods. It is convenient to rewrite the inequalities (22) and (21) as

$$\text{Cr1}: \; sign(|e_{j|ik}X_t| - |e_{i|jk}Y_t|), \text{ and}$$

$$\text{Cr2}: \; sign(|e_{j|ik}| - |e_{i|jk}|),$$

where the smaller magnitudes suggest a superior specification. Hence the sign (-1) suggests an outcome where the causal path is $X_i \to X_j$.

Depending on the order of stochastic dominance, we have standard tools described in Remark 4 and Vinod (2008, Sec. 4.3) for estimating four numbers, Cu(sd1) to Cu(sd4), quantifying stochastic dominance values. Their signs and magnitudes are determined by the local behavior of the first four moments of the underlying densities of absolute values involved in Cr1 and Cr2 expressions above.

Since it is cumbersome to deal with signs of four numbers, we construct a weighted sum of their signs. What weights do we choose for combining the signs, (−1, 0, +1), not magnitudes of Cu(sd1) to Cu(sd4)? Statistical theory suggests that weights on magnitudes should be *inversely* proportional to the increasing sampling variances of the first four central moments. (σ^2, $2\sigma^4$, $6\sigma^6$, $96\sigma^8$) from a normal parent (applying central limit theory to means of (sdℓ) according to Kendall and Stuart (1977, p. 258). If $\sigma^2 = 0.5$ the declining weights become (2, 2, 1.33, 0.17). The weights when aggregating the signs of cumulative sums (not magnitudes of means) obviously need a mild decline. Based on a small simulation our chosen weights are: (1.2/4, 1.1/4, 1.05/4, 1/4) which sum to 1.085.

Vinod (2017) R package "generalCorr" provides an option to change the weights. Let N_{cr1} denotes a single number weighted sum of four signs of Cu(sd1) to Cu(sd4) associated with Cr1. Similarly, let N_{cr2} denote a single number weighted sum of four signs of Cu(sd1) to Cu(sd4) quantifying Cr2.

Note that the sign from the inequality of two R^2 values (23) involved in our third criterion Cr3 is already only one number in the interval [−1, 0, +1]. We do not need any weighted sum for Cr3. We simply evaluate the sign

$$\text{Cr3}: \ sign(R^2_{j|i,k} - R^2_{i|j,k}), \tag{25}$$

to compute our N_{cr3}. This choice of sign(.) expression for Cr3 makes sure that the sign (-1) represents the situation where the causal path is $X_i \to X_j$. Thus we have defined our sign function evaluations for the three criteria Cr1–Cr3 such that the sign (-1) always means $X_i \to X_j$.

6.2 Unanimity index summarizing signs

The preponderance of evidence regarding the sign is summarized by the grand total of these three numbers N_{cri}, $i = 1, 2, 3$, summarizing the empirical support for a causal path. We compute

$$N_{all} = N_{cr1} + N_{cr2} + N_{cr3}. \tag{26}$$

The interpretation of N_{all} is simply that negative values support the path $X_i \to X_j$ and also support treating X_i as exogenous in a model for joint density $f(X_i, X_j, X_k)$.

Vinod (2017) proves that $N_{all} \in [-3.175, 3.175]$. Since the number 3.175 is unintuitive, we transform it into our sample unanimity index (ui) defined by the relation:

$$ui = 100(N_{all}/3.175), \quad ui \in [-100, 100], \tag{27}$$

for easier interpretation as signed index numbers. The sign of *ui* indicates the estimated direction of the causal path and magnitude represents the extent of unanimity between the quantified criteria Cr1–Cr3. The population index is denoted by upper case letters *UI*.

7 Review of decision rule computations

At this point it is somewhat repetitious but useful to review the above discussion implementing our computational agenda in Section 1.1 leading to our decision rules.

Result 1. Assuming A1–A3, stochastic kernel causality of Definition 1 compares kernel regression: $X_j = G(X_i, X_k) + \epsilon_1$, implying $X_i \to X_j$ with its flipped cousin: $X_i = G(X_j, X_k) + \epsilon_2$, implying $X_j \to X_i$.

The empirically superior causality paths among the flipped cousins are determined by three criteria:

(Cr1): compares consistency and exogeneity condition in (21),
(Cr2): compares smallness of absolute values of residuals in (22), and
(Cr3): compares R^2 values of the two models as in (23).

Four orders of stochastic dominance representing local mean, variance, skewness, and kurtosis in comparing densities involving residuals $e_{j|ik}$, $e_{i|jk}$ yield four numbers: Cu(sd1) to Cu(sd4) for Cr1 using numerical integrations. A weighted average of the signs of these four numbers yields one number N_{cr1} as the representative sign for Cr1. We have similar four numbers of Cr2 and another number N_{cr2} summarizing them. The last criterion Cr3 yields only one number, $N_{cr3} = -sign(R^2_{j|i,k} - R^2_{i|j,k}) \in [-1, 0, 1]$. The number should be -1 for the path $X_i \to X_j$.

Thus we have one number summary of the signs of all three criteria as $N_{all} = N_{cr1} + N_{cr2} + N_{cr3}$. This represents the preponderance of evidence for any particular sign. Finally our sample unanimity index is a simple transformation of N_{all} defined as $ui = 100(N_{all}/3.175)$ which must lie in an intuitive range: $ui \in [-100, 100]$. Choosing a threshold value $\tau = 5$ we can conclude with high probability that $X_i \to X_j$ if and only if $(ui < -\tau)$.

Proof Sketch. We have distilled as much information as possible in the two sets of residuals and two flipped models to assess the causal path. Our resulting decision rules treat the causal paths as empirical questions, without ruling out bidirectional causality associated with jointly dependent variables

in CC-SEM terminology. If additional computational resources are available, a bootstrap sampling distribution of the sample *ui* provides a confidence interval. The "high probability" claimed in Result 1 is further supported by a simulation reported in Section 8, and by examples where cause is known in the vignettes accompanying the R package "generalCorr."

8 Simulation for checking decision rules

Following our Definition 1 and decision rules in Section 1.1 we generate the X_1 variable (assumed to be exogenous) independently and then define X_2 to depend on X_1 after adding a noise term, $\epsilon \sim N(0, 1)$, a the standard normal deviate. Our decision rules are known to perform better in the *absence* of normality and linearity. Hence all our experiments using ϵ are handicapping our decision rules. Nevertheless we want to see if they work reasonably well.

In the following experiments X_1 is an independently generated (exogenous) DGP, and hence the causal path is known to be $X_1 \to X_2$, by construction. We use sample sizes: $T = 50, 100, 300$, to check if our decision rules correctly assess the causal path, despite the handicap of linearity and/or normality.

Let m denote the count for indeterminate signs when we repeat the experiments $N = 1000$ times. Define the success probability (suPr) for each experiment as:

$$(suPr) = \frac{(count\ of\ correct\ signs)}{N - m}. \tag{28}$$

The simulation considers four sets of artificial data where the causal direction is known to be $X_1 \to X_2$.

1. Time regressor: $X_1 = \{1, 2, 3, \ldots, T\}$
   $X_2 = 3 + 4X_1 + \epsilon$
2. Unit root quadratic:
   X_1 has T random walk series from cumulative sum or standard normals.
   $X_2 = 3 + 4X_1 - 3X_1^2 + \epsilon$
3. Two uniforms:
   X_1, Z_1 each have T uniform random numbers
   $X_2 = 3 + 4X_1 + 3Z_1 + \epsilon$
4. Three uniforms:
   X_1, Z_1, Z_2 each have T uniform random numbers
   $X_2 = 3 + 4X_1 + 5Z_1 - 6Z_2 + \epsilon$

The simulation required about 36 h on a Dell Optiplex Windows 10 desktop running Intel core i5-7500, cpu at 3.40 GHz, RAM 8 GB, R version 3.4.2.

The large success proportions (suPr) reported in row 7 (for $T = 50$), row 15 (for $T = 100$) and row 23 (for $T = 300$) of Table 1 assume the threshold

TABLE 1 Summary statistics for results of using the "ui" measure for correct identification of causal path indicated by its positive sign using $N = 1000$ repetitions, $T = 50, 100, 300$ sample sizes along three horizontal panels

Row	Stat.	Expm=1	Expm=2	Expm=3	Expm=4
1	Min. $T = 50$	31.496	−100.000	−100.000	−100.000
2	1st Qu.	63.780	31.496	31.496	−31.496
3	Median	100.000	31.496	31.496	37.008
4	Mean	82.395	33.725	24.386	27.622
5	3rd Qu.	100.000	100.000	37.008	37.008
6	Max.	100.000	100.000	100.000	100.000
7	suPr	1.000	0.793	0.808	0.712
8	Min. $T = 100$	31.496	−100.000	−100.000	−100.000
9	1st Qu.	63.780	31.496	31.496	31.496
10	Median	81.102	31.496	31.496	37.008
11	Mean	74.691	33.106	32.822	35.879
12	3rd Qu.	100.000	100.000	37.008	37.008
13	Max.	100.000	100.000	100.000	100.000
14	suPr	1.000	0.787	0.892	0.803
15	Min. $T = 300$	31.496	−100.000	−31.496	−63.780
16	1st Qu.	81.102	31.496	31.496	37.008
17	Median	81.102	31.496	31.496	37.008
18	Mean	80.357	43.020	42.973	42.117
19	3rd Qu.	100.000	100.000	37.008	37.008
20	Max.	100.000	100.000	100.000	100.000
21	suPr,$\tau = 0$	1.000	0.829	0.987	0.963
22	suPr,$\tau = 15$	1.000	0.833	0.988	0.970
23	suPr,$\tau = 20$	1.000	0.835	0.989	0.971
24	suPr,$\tau = 25$	1.000	0.836	0.989	0.971

Success probabilities (suPr) show convergence as T increases in the three panels.

$\tau = 0$. The results for the four experiments in four columns show that our decision rules using a "ui" from Cr1 to Cr3 work well. The effect on success probabilities of the choice of the threshold is studied for the $T = 300$ case by using $\tau = 0$, 15, 20, 25, respectively, along rows 21–24.

Moreover since the success probabilities "suPr" for $\tau = 0$ along rows 7, 14, and 21 increase as $T = 50$, 100, 300 increases, this suggests desirable asymptotic convergence-type feature. Thus, our decision rules are supported by the simulation.

9 A bootstrap exogeneity test

Statistical inference regarding causal paths and exogeneity uses the rescaled version *ui* from Eq. (27) of N_{all} of (26), for estimating the population parameter *UI*.

Bootstrap percentile confidence interval: We suggest a large number J of bootstrap resamples of (X, Y, Z) data to obtain $(N_{all})_j$ and $(\text{ui})_j$ using any bootstrap algorithm. These $(j = 1, \ldots, J)$ values provide an approximation to the sampling distribution of "sum" or "ui." We can easily sort the J values from the smallest to the largest and obtain the "order statistics" denoted as $(\text{ui})_{(j)}$, with parenthetical subscripts. Now a $(1 - \alpha)100$ percent confidence interval is obtained from the quantiles at $\alpha/2$ and $1 - \alpha/2$. For example, if $\alpha = 0.05$, $J = 999$, 95% confidence interval limits are: $(\text{ui})_{(25)}$ and $(\text{ui})_{(975)}$.

Recalling the decision rules Ru.1–Ru.3 of Section 1.1, if both confidence limits fall inside one of the two half-open intervals, we have a statistically significant conclusion. For example, Ru.1 states that: If $(ui < -\tau)$ the causal path is: $X_i \to X_j$. If instead of a point estimate we have two limits, we want both confidence limits of *ui* lie in the same half-open interval: $[-100, -5)$. Then, we have a statistically significant conclusion that $X_i \to X_j$, or equivalently that X_i is exogenous.

This chapter uses the maximum entropy bootstrap (meboot) R package described in Vinod and López-de-Lacalle (2009) because it is most familiar to me, retains the dependence structure in the data, and is recently supported by simulations in Yalta (2016), Vinod (2015) and elsewhere. An advantage of meboot is that it permits bootstrap inference even if the variables in the model are not stationary. Following Stock (1987) such specification in data levels (without differencing or detrending) allows the estimators to be super-consistent.

9.1 Summarizing sampling distribution of *ui*

The approximate sampling distribution is revealed by J (=999) resampled estimates of *ui*. A simple way of learning the properties of these estimates is in terms of the usual summary statistics. This will be illustrated later in Table 4 for our illustrative example.

Another way involves computing bootstrap proportion of significantly positive or negative values. Let m denote the bootstrap count of indeterminate signs when $(\text{ui}) \in [-\tau, \tau]$, where the threshold $\tau = 5$ can be changed by the

researcher depending on the problem at hand. Now define a bootstrap approximation to the proportion of significantly positive signs as:

$$P^*(+1) = \frac{(count\ of\ ui_j > \tau)}{J-m}. \tag{29}$$

Similarly, a bootstrap approximation to the proportion of significantly negative signs is:

$$P^*(-1) = \frac{(count\ of\ ui_j < -\tau)}{J-m}. \tag{30}$$

10 Application example

An application with some topical interest is briefly discussed in this section. Since the US economy has had a long stretch of growth, tools for predicting a recession are all the more important. Macroeconomists and Federal Reserve researchers being aware of their failure to forecast the last great recession of 2007–2008 have developed new data series. For example, Gilchrist and Zakrajek (2012) excess bond premium (EBP) series has been shown to predict recession risk. The term-spread, defined as the difference between long term yield (10-year) and short-term yield (1-year) on government securities is shown by Bauer and Mertens (2018) to be an excellent predictor of recessions.

Instead of directly predicting discrete events like recessions, we are attempting to study what macroeconomic variables drive EBP and term-spread themselves. Our term-spread is denoted as "Dyld" or difference in yields on 10-year and 6-month government securities and discussed later in Section 10.1. We use Federal Reserve Bank's fairly long quarterly data set from 1973Q1 to 2017Q1.

We study the following potential causes behind EBP by considering the endogeneity of variables in the following nonparametric regression:

$$\text{EBP} = f(\text{Yld10}, \text{eFFR}, \text{CrCrea}, \text{CrDstr}, \text{UnemR}, \text{M2}, \text{MbyP}, \text{YbyHrs}, \text{JD}, \text{JC}), \tag{31}$$

where self-explanatory symbols are: yield on 10-year treasury bonds (Yld10, not seasonally adjusted), effective federal funds rate (eFFR), and credit creation (CrCrea, not seasonally adjusted), credit destruction (CrDstr, not seasonally adjusted), unemployment rate (UnemR), money stock (M2, seasonally adjusted billions of dollars), MbyP (ratio of M to PGDP or real money supply), YbyHrs (ratio of GDP to hours, or productivity), JD (job destruction), JC (job creation). Arguments for using separate variables for CrCrea and CrDstr are found in Contessi and Francis (2013) with additional references.

TABLE 2 Excess bond premium and possible causes

	Cause	Response	Strength	Corr.	*P* value
1	EBP	Yld10	47.244	0.0866	0.25161
1b	EBP	Dyld	31.496	0.0416	0.58258
2	EBP	eFFR	31.496	0.0902	0.23248
3	EBP	CrCrea	31.496	−0.0606	0.42322
4	EBP	CrDstr	31.496	0.2617	0.00043
5	EBP	UnemR	31.496	0.1108	0.14222
6	M2	EBP	31.496	−0.0536	0.47843
7	EBP	MbyP	31.496	0.0195	0.79659
8	YbyHrs	EBP	31.496	−0.0588	0.43693
9	JD	EBP	31.496	0.47	0
10	JC	EBP	31.496	−0.1323	0.07915

Table 2 explicitly reports for each flipped pair the "cause" and "response" such that the left-hand variable EBP in Eq. (31) is present in all pairs. The column entitled "strength" reports the absolute value $|ui|$ value. The names of variables to go in the "cause" and "response" columns are dictated by the sign of ui. For example, line 6 has M2 in the "cause" column and EBP in the "response" column, because $ui < 0$ implies that M2 is exogenous. The column entitled "corr" reports Pearson correlation coefficient with EBP, while the column entitled "*P*-value" reports the "*P*"-value for testing the null of zero correlation. Of course, kernel causality and exogeneity need not agree with traditional correlation inference, since the latter assumes linearity and normal distributions.

Whenever $ui > 0$, we place EBP in the "cause" column. Table 2 line 1b reports that ui is positive and smaller than that for "Yld10" along row 1. We have focused more on EBP than "Dyld" because EBP has greater independent innovations than "Dyld" according to line 1b, where "Dyld" does not "cause" EBP. The simple correlation between EBP and "Dyld" is statistically insignificant and lower than the correlation between EBP and Yld10 reported along row 1.

Note that only M2, YbyHrs, JD, and JC are likely to be self-driven (exogenous) causing the excess bond premium, while all other variables seem to be endogenous, being caused by EBP.

TABLE 3 Term spread between 10-year to 6-month treasury yields and possible causes

	Cause	Response	Strength	Corr.	*P* value
1	Yld10	Dyld	31.496	−0.1862	0.0131
2	Dyld	eFFR	100	−0.5463	0
3	Dyld	CrCrea	37.008	−0.1666	0.02668
4	Dyld	CrDstr	100	0.3107	3e−05
5	Dyld	UnemR	31.496	0.5149	0
6	M2	Dyld	31.496	0.2412	0.00122
7	MbyP	Dyld	31.496	−0.0014	0.98522
8	YbyHrs	Dyld	31.496	0.2718	0.00025
9	Dyld	JD	37.008	−0.0403	0.5939
10	Dyld	JC	100	−0.1359	0.07121

10.1 Variables affecting term spread

Before we turn to statistical inference associated with the results above, we include results for causal paths and their strengths when the dependent variable in (31) is "Dyld," or difference in yields.

$$\text{Dyld} = f(\text{Yld10, eFFR, CrCrea, CrDstr, UnemR, M2, MbyP, YbyHrs, JD, JC}). \tag{32}$$

The results in Table 3 show that independent variation in "Dyld," similar to "EBP," drives that in variables: (dffFFR, CrCrea, CrDtr, UnemR, JD, JC). By contrast, "Dyld" is driven by variables (Yld10, M2, MbyP, YbyHrs). This contrasts with the driver variables (M2, YbyHrs, JD, JC) in Table 2. The common drivers are money supply M2 and productivity "YbyHrs."

10.2 Bootstrap inference on Estimated Causality Paths

What about sampling variability of *ui*? We resample the data 999 times using the "meboot" package to keep the time series properties of the data unchanged. The summary statistics of the 999 estimates of *ui* for 10 variables are split into two Tables 4 and 5.

Table 6 shows that our approximate sampling distribution results provide a distinct piece of information not covered by the results about the strength or *P*

TABLE 4 Summary statistics of 999 bootstrap estimates of causal directions and strengths, Part 1

	Yld10	eFFR	CrCrea	CrDstr	UnemR
Min.	−31.50	−100.00	−31.50	−31.50	−31.50
1st Qu.	31.50	−31.50	−31.50	31.50	−31.50
Median	47.24	−31.50	31.50	31.50	−31.50
Mean	55.18	−4.88	6.36	29.32	−10.84
3rd Qu.	81.10	31.50	31.50	31.50	31.50
Max.	100.00	100.00	100.00	100.00	100.00

TABLE 5 Summary statistics of 999 bootstrap estimates of causal directions and strengths, Part 2

	M2	MbyP	YbyHrs	JD	JC
Min.	−31.50	−31.50	−31.50	−31.50	−100.00
1st Qu.	−31.50	31.50	−31.50	−31.50	−31.50
Median	−31.50	31.50	−31.50	−31.50	−31.50
Mean	−31.50	32.77	−31.50	−1.29	−29.55
3rd Qu.	−31.50	31.50	−31.50	31.50	−31.50
Max.	−31.50	100.00	−31.50	100.00	100.00

TABLE 6 Bootstrap success rates for causal direction using 999 resamples.

	Variable	$P(\pm 1)$
1	Yld10	0.9737
2	eFFR	0.6232
3	CrCrea	0.537
4	CrDstr	0.951
5	UnemR	0.7124
6	M2	1
7	MbyP	0.964
8	YbyHrs	1
9	JD	0.5486
10	JC	0.9758

value in Table 2. The table contains proportion of negative or positive (whichever is most prevalent) in each column described as bootstrap success rates, defined in Eqs. (30) and (29).

We recommend careful analysis of each causal pair with the help of scatterplots. We include only two plots here for brevity: (i) EBP-UnemR pair where UnemR is found to be endogenous, and (ii) EBP-M2 pair where M2 is found to be exogenous. Histograms of the two variables are seen in the diagonal panels of Figs. 1 and 2. The South West panels have a scatter diagram and locally best fitting free hand curve. The number in the North East panels is the ordinary correlation coefficient whose font size suggests its statistical significance.

Fig. 1 depicts a scatterplot having a mildly up-down-up pattern. This may explain why the Pearson correlation coefficient of 0.11 is statistically insignificant, since it does not capture nonlinear relations. Endogeneity of unemployment rate suggests that it is likely an effect of recessions and not

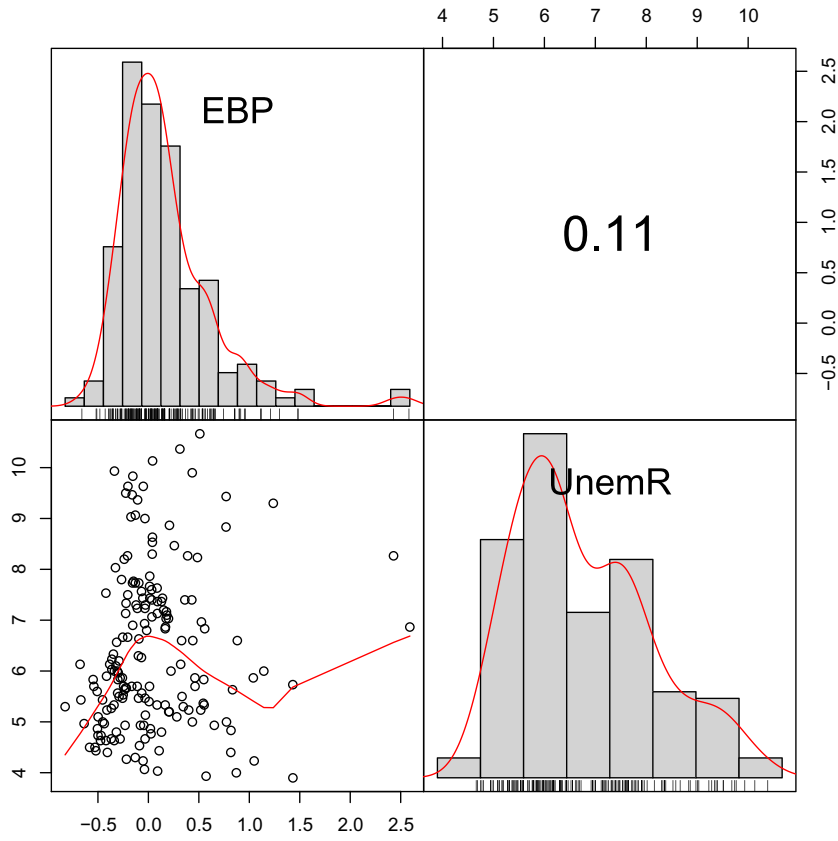

FIG. 1 Scatterplot with nonlinear curve: EBP-UnemR.

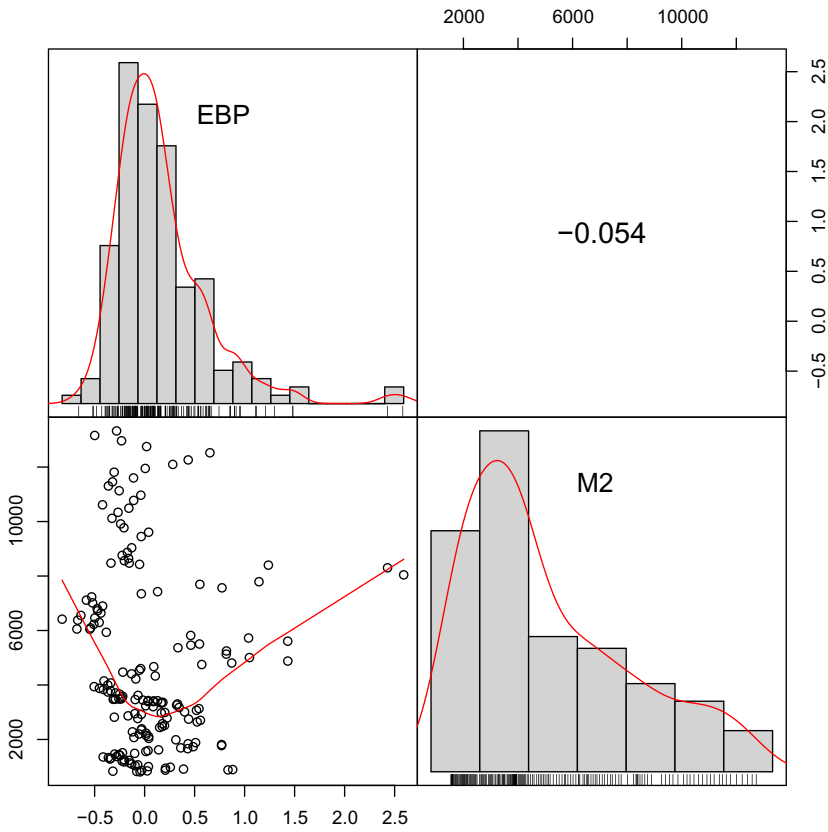

FIG. 2 Scatterplot with nonlinear curve: EBP-M2.

a cause. Note that Fig. 2 suggests that the variation in M2 is weakly exogenous. Its scatterplot is U-shaped and quite noisy confirming highly insignificant and small Pearson correlation coefficient. Thus a decline in M2 may help predict recessions.

11 Summary and final remarks

We update Suppes' "probabilistic causality theory" from philosophical literature and propose a new "stochastic causality theory" of Theorem 1. If one replaces probabilities of events with densities of DGPs, its direct quantification seems difficult. Hence we use kernel regressions to define our "stochastic causality" (Definition 1). A review Section 7 summarizes the derivation of the unanimity index ($ui \in [-100, 100]$) providing our decision rules. This index uses three criteria Cr1–Cr3 and four orders of stochastic

dominance. Our decision rules are simulated in Section 8 with high success rate. Our new bootstrap test for exogeneity in Section 9 provides statistical inference for *ui* using about a thousand estimates.

Descriptive statistics of these 1000 estimates can provide a quick view of the sampling distribution of *ui* to assess the preponderant sign and hence the causal direction. We also provide a confidence interval and identify a bidirectional causal path when ($ui \in [-\tau, \tau]$) or too close to zero. Now ($ui > -\tau$) directly identifies an RHS variable as having an endogeneity problem. It may well need an extra equation in a simultaneous equations model.

Engle et al. (1983, p. 288) admit that their "weak exogeneity" is not "directly testable" as it involves arbitrarily defined distinction between parameters of interest (ψ) and nuisance parameters (λ_2). Hausman–Wu indirect exogeneity tests use IV estimators which can "do more harm than good" (Bound et al., 1995, p. 449), and are criticized as being "very inefficient" by Kiviet and Niemczyk (2007), Dufour, and others. Medicine has long rejected medieval-style diagnoses of diseases by simply showing that a cure works. Hence there is a long-standing need for a direct alternative which avoids having to specify, collect data on and try an exhaustive set of several potential instrumental variables, and concluding that a particular RHS variable is exogenous only if no IV "works." Since it is hard to be sure that one has attempted an exhaustive set of IVs, the endogeneity problem is likely to be over-diagnosed and treated with dubious instruments.

An illustrative example in Section 10 considers a novel macroeconomic model explaining the "excess bond premium" (EBP) known to be a good predictor of US recessions. Alternatively we attempt to explain term spread "Dyld" between long-term and short-term government bond interest rates. Our Table 2 suggests that US investors worried about an impending recession should pay attention to innovations in two key variables: money stock (M2) and productivity (YbyHrs) which are found to be kernel exogenous.

Clearly, practitioners can use our unanimity index implemented with very few lines of code. The ability to incorporate control variables in our analysis is particularly valuable for causality estimation and testing. There are several potential applications in all scientific areas including exploratory hypothesis formulation, big data, and artificial intelligence. One recent paper, Lister and Garcia (2018), uses our decision rules to conclude that global warming causes arthropod deaths. Another paper, Allen and Hooper (2018), uses them to explore causes of volatility in stock prices.

Acknowledgments

I thank Prof. J. Francis for suggesting the "excess bond premium" application and for providing the data and Prof. Philip Shaw for comments. This explains and expands on my keynote address in Jammu, at the 54th annual meeting of the Indian Econometric Society on March 9, 2018.

Appendices

Appendix A. Review of graph theory

Let us begin this admittedly brief and incomplete review by providing a mapping between CC-SEM textbook jargon (Vinod, 2008, chap. 6) and graph theory-SEM jargon from the Stanford Encyclopedia, intended to enhance the communication between the two camps.

1. (left-hand endogenous variable) ≈ (child or descendant variable)
2. (included right-hand endogenous variables) ≈ (parent variables)
3. (excluded right-hand endogenous variable) ≈ (variables with coefficients set to zero a priori)
4. (included right-hand exogenous variable) ≈ (variables included to incorporate Reichenbach's common cause principle)
5. (excluded right-hand exogenous variable) ≈ (various causal identification requirements)

In graph theory the determination of the correct DAG among p variables in V makes it identifiable. Correct DAGs must satisfy, Reiss (2016), Markov condition, minimality, faithfulness, Gaussianity, and incorporate interventions via Pearl's "do" notation.

Assume that a graph $\mathcal{G}$ is DAG. A joint density $f(V)$ satisfies the Markov condition (MC) relative to graph $\mathcal{G}$ if and only if it satisfies three conditions explained in the encyclopedia:

(Screen Off) Let X in V and every set of variables Y also in V excluding the variables which are caused by or "descendants or children of" X the conditional probabilities satisfy:

$$P(X|parent(X),Y) = P(X|parent(X))$$

In other words, given the values of the variables that are parents of X, the values of the variables in Y (which includes no descendants of X) make no further difference to the probability that X will take on any given value.

(Factorization) once we know the conditional probability distribution of each variable given its parents, we can compute the complete joint distribution over all of the variables. $P(V) = \Pi_i P(X_i|parent(X_i))$. This captures Reichenbach's common cause principle.

(d-separation) Pearl's sufficient condition for statistical independence in potentially large graphs. Let $X, Y \in V$, $Z \in V_{-X, -Y}$, where the notation with minus subscript means those variables are excluded from the set, then $P(X, Y|Z) = P(X|Z) \times P(Y|Z)$

Reichenbach's conjunctive fork used for achieving asymmetry is defined by the following formulas:

$$P(X \cap Y|Z) = P(X|Z) \times P(Y|Z) \tag{A.1}$$

$$P(X \cap Y|Z^{-}) = P(X|Z^{-}) \times P(Y|Z^{-}) \tag{A.2}$$

$$P(X|Z) > P(X|Z^{-}) \tag{A.3}$$

$$P(Y|Z) > P(Y|Z^{-}), \tag{A.4}$$

where Z^{-} denotes the situation where Z is absent.

Appendix B. For R code

An R package "generalCorr" has various functions implementing all tools in this chapter. It contains three vignettes showing how to use the R functions with several examples including a discussion of the intuition behind them. The following code is a good way to start an R session.

```
rm(list=ls())
options(prompt = " ", continue = " ", width = 68,
useFancyQuotes = FALSE)
print(date())
```

The above code replaces the R prompt ">" and continuation symbol "+" by blanks to facilitate direct copy and paste of the R code. Most code outputs are suppressed in this appendix for brevity.

Assume the data file is in the form of an excel type workbook or spreadsheet. I place important information about data sources, longer variable names, etc. in the first three lines of excel workbook. The fourth line has variable names as column headings exactly one per column. Variable naming conventions in R are case sensitive, do not allow names with spaces or math symbols, (%, +, /, =), and cannot start with a number. For example, "9x" or "x/y" cannot be valid variable names.

Next, the excel workbook is "saved as" a comma separated "csv" file. It is a good idea to open the file in a notepad to get rid of some extra commas or stuff which might creep in at the end of some lines or at end of the file, if one is editing the excel workbook and removes some text along rows or columns. Finally, we are ready to read the data as:

```
ad="http://www.fordham.edu/economics/vinod/macroebp73to17.csv"
da=read.table(ad, skip=3,sep=",",header=TRUE)
summary(da)
attach(da)
eFFR=effFFR #brief notation effective federal funds rate
```

The "attach" command allows us to access data variables by name.

In the following we access the "generalCorr" package and define a matrix "mtx" where the first variable "EBP" is the left-side variable in Eq. (31) and other 10 variables are listed next.

```
library(generalCorr)
options(np.messages=FALSE)
mtx=cbind(EBP,Yld10,eFFR,CrCrea,CrDstr,
UnemR,M2,MbyP,YbyHrs,JD,JC)
c1=causeSummary(mtx)#fast causality analysis
xtable(c1)#latex table output
parcorMany(mtx) #matrix of partial correlations,SLOW
```

Table 2 is produced by the above code.

The command "bootPairs" below uses 999 data resamples to assess sampling variability of the estimated *ui* values. It created 10 columns of 999 numbers called "b1" here and took about 10 h to implement on a home PC.

```
b1=bootPairs(mtx, n999=999)
a1summ=apply(b1,2,summary)
a1sum2=a1summ*(100/3.175)
xtable(round(a1sum2[,1:5], 3))
xtable(round(a1sum2[,6:10], 3))
```

Tables 4 and 5 are produced by the "xtable" commands in the above code.

Following code produces Table 6.

```
n1=colnames(mtx)[2:11]
bsign=round(bootSign(b1),4)
xtable(cbind(n1,bsign))
```

The code for our Figs. 1 and 2 is next.

```
library(PerformanceAnalytics)
chart.Correlation(cbind(EBP,M2))
chart.Correlation(cbind(EBP,UnemR))
```

Thus anyone with Internet access and rudimentary knowledge of R can estimate causal directions and strengths between a set of variables rather simply.

References

Allen, D.E., Hooper, V., 2018. Generalized correlation measures of causality and forecasts of the VIX using non-linear models. Sustainability 10 (8), 2695. https://doi.org/10.3390/su10082695. https://www.mdpi.com/2071-1050/10/8/2695.

Anderson, G., 1996. Nonparametric tests of stochastic dominance in income distributions. Econometrica 64 (5), 1183–1193.

Bauer, M. D., Mertens, T. M., 2018. Economic forecasts with the yield curve. FRBSF Economic Letter, Federal Reserve Bank of San Francisco, California, https://www.frbsf.org/economic-research/publications/economic-letter/2018/march/economic-forecasts-with-yield-curve.

Bound, J., Jaeger, D.A., Baker, R., 1993. The cure can be worse than the disease: a cautionary tale regarding instrumental variables. NBER Working Paper No. 137, http://ssrn.com/paper=240089.

Bound, J., Jaeger, D.A., Baker, R., 1995. Problems with instrumental variables estimation when the correlation between the instruments and the endogenous explanatory variables is weak. J. Am. Stat. Assoc. 90, 443–450.

Contessi, S., Francis, J., 2013. U.S. commercial bank lending through 2008:q4: new evidence from gross credit flows. Economic Inquiry 51 (1), 428–444.

Davidson, R., MacKinnon, J.G., 1993. Estimation and Inference in Econometrics. Oxford Univ. Press, New York.

Engle, R.F., Hendry, D.F., Richard, J.F., 1983. Exogeneity. Econometrica 51, 277–304.

Gilchrist, S., Zakrajek, E., 2012. Credit spreads and business cycle fluctuations. AmEcon. Rev. 102 (4), 1692–1720.

Hansen, B.E., 2004. Nonparametric Conditional Density Estimation. University of Wisconsin, Department of Economics. https://www.ssc.wisc.edu/bhansen/papers/ncde.pdf.

Holland, P.W., 1986. Statistics and causal inference. J. Am. Stat. Assoc. 81, 945–970 (includes discussion by many authors).

Imbens, G.W., Rubin, D.B., 2015. Causal Inference for Statistics, Social, and Biomedical Sciences: An Introduction. Cambridge University Press. https://doi.org/10.1017/CBO9781139025751.

Kendall, M., Stuart, A., 1977. The Advanced Theory of Statistics, fourth ed. vol. 1. Macmillan Publishing Co., New York.

Kiviet, J.F., Niemczyk, J., 2007. The asymptotic and finite-sample distributions of OLS and simple IV in simultaneous equations. Comput. Stat. Data Anal. 51, 3296–3318.

Koopmans, T.C., 1950. When is an equation system complete for statistical purposes. Yale University. http://cowles.econ.yale.edu/P/cm/m10/m10-17.pdf.

Levy, H., 1992. Stochastic dominance and expected utility: survey and analysis. Manag. Sci. 38 (4), 555–593.

Li, Q., Racine, J.S., 2007. Nonparametric Econometrics. Princeton University Press.

Lister, B.C., Garcia, A., 2018. Climate-driven declines in arthropod abundance restructure a rainforest food web. Proc. Natl. Acad. Sci. 115, 1–10. http://www.pnas.org/content/early/2018/10/09/1722477115.full.pdf.

Pearl, J., 2009. Causality: Models, Reasoning and Inference. Wiley, New York.

Peters, J., Mooij, J., Janzig, D., Zscheischler, J., Scholkopf, B., 2014. Causal discovery with continuous additive noise models. J. Mach. Learn. Res. 15, 2009–2053. http://jmlr.org/papers/volume15/peters14a/peters14a.pdf.

Reiss, J., 2016. Suppes' probabilistic theory of causality and causal inference in economics. J. Econ. Methodol. 23 (3), 289–304. https://doi.org/10.1080/1350178X.2016.1189127.

Salmon, W.C., 1977. An "at-at" theory of causal influence. Philos. Sci. 44 (2), 215–224. http://www.unige.ch/lettres/baumgartner/docs/kausa/protect/salmon.pdf.

Stock, J.H., 1987. asymptotic properties of least squares estimators of cointegating vectors. Econometrica 55 (5), 1035–1056.

Suppes, P., 1970. A Probabilistic Theory of Causality. Amsterdam, North-Holland.

Vinod, H.D., 2004. Ranking mutual funds using unconventional utility theory and stochastic dominance. J. Empir. Financ. 11 (3), 353–377.

Vinod, H.D., 2008. Hands-on Intermediate Econometrics Using R: Templates for Extending Dozens of Practical Examples. World Scientific, Hackensack, NJ. http://www.worldscibooks.com/economics/6895.html.

Vinod, H.D., 2014. Matrix algebra topics in statistics and economics using R. In: Rao, M.B., Rao, C.R. (Eds.), Handbook of Statistics: Computational Statistics with R, vol. 34. Elsevier Science, North Holland, New York, pp. 143–176.

Vinod, H.D., 2015. New bootstrap inference for spurious regression problems. J. Appl. Stat. 1–34. http://www.tandfonline.com/doi/full/10.1080/02664763.2015.1049939.

Vinod, H.D., 2017. Causal Paths and Exogeneity Tests in generalCorr Package for Air Pollution and Monetary Policy, https://cloud.r-project.org/web/packages/generalCorr/vignettes/generalCorr-vignette3.pdf.

Vinod, H.D., López-de-Lacalle, J., 2009. Maximum entropy bootstrap for time series: the meboot R package. J. Stat. Softw. 29 (5), 1–19. http://www.jstatsoft.org/v29/i05/.

Wu, D.-M., 1973. Alternative tests of independence between stochastic regressors and disturbances. Econometrica 77 (5), 733–750.

Yalta, A.T., 2016. Bootstrap inference of level relationships in the presence of serially correlated errors: a large scale simulation study and an application in energy demand. Comput. Econ. 48, 339–366. https://doi.org/10.1007/s10614-015-9530-7.

Zalta, E.N., 2018. The Stanford Encyclopedia of Philosophy. The Metaphysics Research Lab, Stanford University. https://plato.stanford.edu.

Chapter 3

Adjusting for bias in long horizon regressions using R

Kenneth D. West[a,*] and Zifeng Zhao[b]
[a]Department of Economics, University of Wisconsin-Madison, Madison, WI, United States
[b]Department of Information Technology, Analytics and Operations, Mendoza College of Business, University of Notre Dame, Notre Dame, IN, United States
[*]Corresponding author: e-mail: kdwest@wisc.edu

Abstract
Long horizon regressions that rely on linear models are common in many applied fields. Examples from economics include forecasting inflation 12 quarters ahead (Crone et al., 2013) and relating 120 month ahead changes in exchange rates to current period variables (Snaith et al., 2013). We describe R code to implement recently developed procedures that adjust long horizon regressions to lessen bias in parameter estimates (West, 2016).

Keywords: Least squares bias, Small sample bias, Bias reduction, Multistep forecast, Direct forecast, VAR model

1 Introduction

In this chapter we consider small sample bias in long horizon least squares regressions in discrete time linear time series models. A leading application is to "direct" multistep forecasts. We describe R code that adjusts for small sample bias in such regressions. Such adjustments may be important because in some specifications such bias is arbitrarily large for an arbitrarily long horizon (West, 2016).

We begin by reviewing long horizon regressions and the direct method for making a multistep forecast. We then describe R functions to implement recently developed procedures that modify long horizon regression parameters to lessen bias. We close with the code for an empirical application.

Throughout, our topic is solely construction of bias adjusted regression estimates, taking as given a set of regressors or predictors. That is, we do not discuss selection of predictors nor many other topics that are important

Handbook of Statistics, Vol. 41. https://doi.org/10.1016/bs.host.2019.01.006

in applied work such as construction of confidence intervals and forecast evaluation. See West and Zhao (2018) for how bias adjustment relates to forecast evaluation via mean squared prediction error.

The R code can be downloaded from https://www.ssc.wisc.edu/~kwest/appendices/appendices.htm.

2 Long horizon regressions

Let y_t be a scalar time series, with data running up to time T. We suppose that one wishes to model or forecast y_{t+q} for some horizon $q+1>0$. If the data are monthly, $q+1$ is measured in months, and similarly for frequencies other than monthly.

In economics applications, the relevant range for $q+1$ runs from one step ahead ($q+1=1$) to $q+1$ in excess of 100. One step ahead forecasts—business investment next quarter, employment next month, and so on—are ubiquitous. Multistep forecasts are also common. The Survey of Professional Forecasters asks participants to forecast GDP and inflation up to five quarters ahead; Mark (1995) forecasts exchange rates 16 quarters ahead; and Crone et al. (2013) forecast inflation 12 quarters ahead. Policy and academic work sometimes look at even longer horizons. Lunsford and West (2017) forecast interest rates 10 years ahead; Snaith et al. (2013) relate 120 month ahead changes in exchange rates to period t predictors; and Hjalmarsson (2011) relates 10 year ahead stock returns to period t predictors. These examples could be multiplied many times over.

Note that the final two examples used the verb "relates" and not "forecasts." Such research evaluates the connection between a many step ahead variable and a set of regressors using in-sample analysis only. The R procedures we describe here are just as valuable for such in-sample analysis as it is for analysis that involves forecasting. However, to focus the discussion, we often shall describe our R functions in terms of forecasts.

Much though not all the relevant work—both in- and out-of-sample—relies on stationary linear models, which we maintain here. Specifically, we assume that the forecast or in-sample modeling of a stationary variable y_{t+q} relies on the projection of y_{t+q} onto a constant and a $(k\times 1)$ vector X_{t-1}. Write the population least squares projection as

$$y_{t+q}=\alpha+X_{t-1}'\beta+\eta_{t+q}. \tag{1}$$

The disturbance η_{t+q} is unobserved and is defined as the difference between y_{t+q} and the population projection of y_{t+q} onto a constant and X_{t-1}.

In economics, there are two broad classes of applications. In the first class, the elements of X_{t-1} are financial market or survey variables that are hypothesized to be good predictors of an economic variable y. A simple example occurs in the Snaith et al. (2013) paper cited above:

- Let s_t be the log of the end of month nominal exchange rate (say, dollars per British pound), so that

$$\Delta s_t \equiv s_t - s_{t-1}$$

is approximately percentage change in the exchange rate. Observe that with this definition,

$$y_{t+q} \equiv s_{t+q} - s_{t-1} = \Delta s_{t+q} + \Delta s_{t+q-1} + \cdots + \Delta s_t$$

is approximately the percentage change in the exchange rate from month $t-1$ to month $t+q$.

- Let i_t be the nominal return on a nominally safe $q+1$ month US bond. Since we are assuming for the sake of illustration that the data are monthly, if $q+1=120$, then i_t is the interest rate on a 10-year US Treasury bond. ("Nominally safe" means: the borrower [i.e., the US government] will not default, and the return is only guaranteed in nominal rather than real inflation adjusted terms.) Let i_t^* be the corresponding foreign interest rate (the comparable interest rate in the UK, in this example).

Then a certain economic model says that the interest differential on 10-year bonds $i_t - i_t^*$ well explains the cumulative change in the exchange rate over the following 10 years. So the regression run is

$$y_{t+q} \equiv s_{t+q} - s_{t-1} = \alpha + \beta_1 (i_t - i_t^*) + \eta_{t+q};\ X'_{t-1} \equiv i_t - i_t^* \text{ and } k=1. \quad (2)$$

The second broad class of applications are ones where X_{t-1} consists of deterministic terms and lags of y_t and perhaps other variables. An example is Marcellino et al. (2006), who examine prediction of many monthly economic series at horizons up to $q+1=24$ months, using both univariate (X_{t-1} includes lags of y_t) and bivariate (X_{t-1} includes lags of y_t and of a second variable) models. In the simplest possible case, the model is univariate and the forecasting horizon is one step ahead ($q+1=1$). Then (1) is

$$y_t = \alpha + \beta_1 y_{t-1} + \beta_2 y_{t-2} + \ldots + \beta_k y_{t-k} + \eta_t;\ X'_{t-1} = (y_{t-1} \ldots y_{t-k}). \quad (3)$$

A specification such as (3) indicates that the researcher thinks an AR(k) model well approximates y_t. For this same set of predictors, a multistep direct prediction ($q+1>1$) relies on the projection

$$y_{t+q} = \alpha + \beta_1 y_{t-1} + \beta_2 y_{t-2} + \cdots + \beta_k y_{t-k} + \eta_{t+q}. \quad (4)$$

Eq. (4) relies on the fact that a multistep prediction of an AR(k) model is linear in k lags of the variable. Of course, the coefficients $\alpha, \beta_1, \ldots, \beta_k$ in (4) are different from those in (3) (except when $q+1=1$).

A modest permutation of (4) involves predicting the average rather than point in time forecast. For example, we generally are less interested in inflation in the fourth quarter of next year than we are in average inflation over the next four quarters. This sort of application involves a regression of the form

$$\begin{aligned} y_{t+q} &\equiv \left(x_{t+q} + x_{t+q-1} + \cdots + x_t\right)/(q+1) \\ &= \alpha + \beta_1 x_{t-1} + \beta_2 x_{t-2} + \cdots + \beta_k x_{t-k} + \eta_{t+q}. \end{aligned} \tag{5}$$

Again, the coefficients in (5) are, in general, different from those in (4).

Each of (3)–(5) illustrated the direct multistep forecast of y_{t+q}. Note for future use that in (4) and (5), if, indeed, the hypothesized AR(k) model is correct, the disturbance follows a moving average process of order q:

$$\eta_{t+q} \sim \mathrm{MA}(q). \tag{6}$$

In many though not all motivations for regressions in the first class of applications, illustrated by (2), it is also true that $\eta_{t+q} \sim \mathrm{MA}(q)$.

A brief digression on methods for multistep forecasts: for forecasts such as those in (4) and (5), an alternative approach is to recursively generate j period ahead forecasts by using $j-1$ period ahead forecasts. This is the approach of Box and Jenkins (1976), for example. Let "^" denote a least squares estimate. For example, in the model (4), the Box and Jenkins method constructs one and two step ahead forecasts via

$$\text{one step ahead forecast} = \hat{\alpha} + \hat{\beta}_1 y_T + \cdots + \hat{\beta}_k y_{T-k+1}, \tag{7a}$$

$$\begin{aligned} \text{two step ahead forecast} = \hat{\alpha} + \hat{\beta}_1 \times (\text{one step ahead forecast}) + \hat{\beta}_2 y_T + \\ \ldots + \hat{\beta}_k y_{T-k+2}. \end{aligned} \tag{7b}$$

This is sometimes called the *iterated* or *plug-in* method of forecasting. Our procedures for bias adjustment are trivially applicable for one step ahead forecasts such as (7a), when the direct and iterated methods are identical. But they are not directly applicable for multistep iterated forecasts such as (7b). For a theoretical comparison of iterated and direct forecasts, see Ing (2003). We focus on direct forecasts because they are dominant in economics.

3 Bias adjustment for long horizon regressions

3.1 Introduction

Least squares estimators of time series models are biased in finite samples. That is, even if we make (mild) assumptions so that estimates are consistent for underlying population quantities, in finite samples the expectation of the least squares estimator is not, in general, equal to the underlying population quantity. For the simple AR(1) model (in (3), $k=1$ and $\eta_t \sim$ i.i.d.), Kendall (1954) suggested that

$$E\hat{\beta}_1 \approx \beta_1 - \frac{(1+3\beta_1)}{T}. \tag{8}$$

More generally, West (2016) shows that for a $k \times 1$ vector b that depends on own- and cross-moments of X_{t-1} and η_t, the least squares estimator of (8) satisfies

$$E\hat{\beta} = \beta + \frac{b}{T} + O\left(T^{-3/2}\right). \tag{9}$$

In the simple AR(1) model underlying (8), $k=1$, $\beta=\beta_1$ and $b=-(1+3\beta_1)$. The small sample bias we are concerned with in this chapter is the b/T term in (9). We describe R code to construct an estimate $\hat{b}$, yielding a bias adjusted estimate $\hat{\beta}-\hat{b}/T$.

Bias in estimate of the constant term α (defined in (1)) follows from:

$$\left(\text{bias to order } T^{-1} \text{ in} \hat{\alpha}\right) = \left(EX_t'\right)b.$$

Given an estimate $\hat{b}$ supplied by the R code we are about to describe, and a sample average $\overline{X}$, one can adjust for such bias via:

$$(\text{bias adjusted estimate of } \alpha) = \hat{\alpha} - \overline{X}'\hat{b}/T.$$

Because such an adjustment follows directly from adjustment for bias in the slope coefficient vector β, we shall not further discuss bias adjustment of α.

A number of papers have derived b/T when forecasts are one step ahead ($q+1=1$) and η_t is a conditionally homoskedastic martingale difference. See Shaman and Stine (1988) for the univariate AR (Eq. 4) and the summary in Engsted and Pedersen (2014) when the one step ahead forecast comes from an equation from a vector autoregression. West (2016) derives b/T for arbitrary horizons $q+1$ and allowing time varying second moments in η_{t+q}. Our R procedures implement a subset of the results allowed in West (2016). In particular, our code coheres with a special case of the theory in West (2016). This special case requires that (a) a certain cumulant condition holds that rules out time varying second moments and (b) η_{t+q} is uncorrelated not just with X_{t-1} itself but also with all lags of X_{t-1} (i.e., $E\eta_{t+q}X_{t-j}' = 0$ for $j=2, 3, \ldots$). (This last condition is generally assumed under the null of the model but may fail under misspecification. For example, in (4), if $y_t \sim \text{AR}(k)$, then $E\eta_{t+q}X_{t-j}' = 0$ for all $j \geq 1$ and the condition holds. But if $y_t \sim \text{AR}(m)$ for some $m > k$, then in (4) the condition fails and $E\eta_{t+q}X_{t-j}' \neq 0$ for at least one $j > 1$.) In any application in which (a) or (b) fail, the estimate $\hat{b}$ that our code delivers should be taken with a larger than usual grain of salt.

See West (2016) for details. One important theoretical result from West (2016): in regressions such as (2) or (5) where the left hand side variable is a long horizon sum or average of a stationary variable, bias b is arbitrarily big in absolute value for an arbitrarily long horizon q. This is a general result for

such left hand side variables, and is not specific to the examples such as (2) and (5). Hence a bias adjustment is especially appealing for such regressions.

To understand the parameters and/or moments that must be passed to our functions that estimate b, it may be helpful to note that b depends on own- and cross-covariances of the right hand side variables X_{t-1} and the disturbance η_{t+q}. The first pair of functions that we are about to describe **(longhor1, longhor)** construct estimates of the relevant second moments and the user needs only to pass a parameter specifying a certain lag length. The second pair of functions that we describe **(proc_vb_ma0, prov_vb_maq)** rely partially on the user to construct the relevant second moments; prior to invoking the functions, the user is required to have estimated a certain autoregression or vector autoregression, with the results of that estimation passed to our R functions.

The first pair of functions **(longhor1, longhor)** are high-level, but require that X_{t-1} consist solely of lags of a single variable such as in (2)–(5). This single variable may or may not be lags of the left hand side variable; the "may" case is illustrated in (3) and (4), the "may not" in (2) and (5). The second pair of functions **(proc_vb_ma0, prov_vb_maq)** are lower level, requiring more work from the user. But they do not restrict the specification of X_{t-1}.

3.2 R function longhor1

One of our R functions is most easily motivated with reference to any of (3)–(5). The user passes a vector **yseries** with data on the left hand side variable and a second vector **xseries** with data on the right hand side variable. The two vectors are different in the case of (5) but are the same in the cases of (3) or (4). The user also specifies the integer order of the lead of the left hand side relative to the right hand side **nq** ($=q$ in (1)) the integer number of lags **nk** ($=k$ in (1)) on the right hand side, integer pointers **first** and **last** to the sample period and an integer **narlag** that should be set to **nk** if (3)–(5) is of interest and whose presence is explained below. The function returns three **nk** $\times$ 1 vectors:

- **vbias** ($= b$, as defined in (9));
- **betahat** (= least squares $\hat{\beta}$);
- **betahat_adj** (= bias adjusted $\hat{\beta} =$ **betahat** $-$ (**vbias**/T), $T =$ **last-first** $+1$).

See Table 1, which summarizes this information.

To clarify dating and variable definitions: the regression of interest is **yseries**, **nq** periods ahead, on lags 1 to **nk** of **xseries**:

$$\begin{aligned}\mathbf{yseries}(t+\mathbf{nq}) = \alpha + \beta_1 \mathbf{xseries}(t-1) + \cdots + \beta_k \mathbf{xseries}(t-\mathbf{nk}) \\ + \text{disturbance}(t+\mathbf{nq}), \\ t+\mathbf{nq} = \mathbf{first}, \ldots, t+\mathbf{nq} = \mathbf{last}.\end{aligned} \tag{10}$$

Since the regression is run with **yseries** dates running from **first** to **last**, the dates on $\mathbf{xseries}_{t-1}$ go from $t-1 =$ **first_nq** -1 to $t-1 =$ **last_nq** -1.

TABLE 1 Function longhor1

```
result <- longhor1(yseries, xseries, first, last, nq, nk, narlags)
vbias <- result[[1]]; betahat <- result[[2]]; betahat_adj
<- result[[3]]
```

The right hand side variables in the regression of interest consist of a constant and lags of a single variable, such as in (2)–(6). The left hand side may or may not be a lead of the same variable.

Passed by user	
yseries	vector for the left hand side variable
xseries	vector for the right hand side variable
first	integer start date for the left hand side variable in the regression
last	integer end date for the left hand side variable in the regression
nq	q: integer horizon, **nq** ≥ 0
nk	k: integer number of lags k of **xseries** to include on the r.h.s. of the regression (1)
narlag	integer number of lags to include in estimating an AR model for **xseries** (needed to compute the bias). The user should insure that **narlag** is sufficient to produce a white noise residual in this autoregression
Returned to user	
result[[1]]	**nk** $\times 1$ vector: $\hat{b}$ = estimate of $k \times 1$ numerator of bias to order T^{-1} (T = sample size)
result[[2]]	**nk** $\times 1$ vector: $\hat{\beta}$ = ordinary least squares estimate of $k \times 1$ β
result[[3]]	**nk** $\times 1$ vector, bias adjusted $\hat{\beta}$, **result[[3]] = result[[2]] − (result[[1]]/T), T = last-first + 1**
Functions invoked directly or indirectly by **longhor1**: **proc_vb_ma0**, **proc_vb_maq**, and **proc_vbias**.	

To further clarify dating, consider a concrete example. Suppose that **yseries** includes 99 observations. For simplicity of exposition, assume data are annual and run from 1901 to 1999. Thus **yseries**(4) is data from 1904 and **xseries** (11) is data from 1911, for example. Suppose further that **nq** = 7 and **nk** = 2 = **narlag**. Suppose, finally, that one wishes to run the regression with left hand side data running from 1910 to 1990 (thus not using some of the data):

$$\begin{aligned} &\mathbf{yseries}_{t+7} = \text{const.} + \beta_1 \mathbf{xseries}_{t-1} + \beta_2 \mathbf{xseries}_{t-2} + \text{disturbance}, \\ &\quad t+7 = 1910, \ldots, 1990 \end{aligned} \tag{11}$$

Thus the vector of the left hand side variable and the matrix of stochastic right hand side variables are

$$\begin{pmatrix} \mathbf{yseries}_{1910} \\ \mathbf{yseries}_{1911} \\ \vdots \\ \mathbf{yseries}_{1990} \end{pmatrix}, \begin{pmatrix} \mathbf{xseries}_{1902} & \mathbf{xseries}_{1901} \\ \mathbf{xseries}_{1903} & \mathbf{xseries}_{1902} \\ \vdots & \vdots \\ \mathbf{xseries}_{1982} & \mathbf{xseries}_{1981} \end{pmatrix}.$$

Then one invokes **longhor1** via

```
result<-longhor1(yseries, xseries, 10, 90, 7, 2, 2),
vbias<-result[[1]]; betahat<-result[[2]]; betahat_adj<-result[[3]].
```
(12)

In this example, the procedure returns three 2×1 vectors: **betahat** $\equiv \hat{\beta} \equiv (\hat{\beta}_1 \hat{\beta}_2)'$, **vbias** $= \hat{b}$, **betahat_adj** $\equiv \hat{\beta} - \hat{b}/T$, where $T = 81$.

More generally, the procedure is invoked via

```
result<-longhor1(yseries, xseries, first, last, nq, nk, narlag)
vbias<-result[[1]]; betahat<-result[[2]]; betahat_adj<-result[[3]].
```
(13)

Note: The code does not do error checks for missing data. So, suppose in the concrete example just given, where available data run from 1901 to 1999, that the user passes **first** $= 5$ along with **nq** $= 7$. Then the function would assume the first observation on the left hand side variable is 1905 and the first observation on **xseries** is 8 years earlier ($8 =$ **nq** $+ 1$), i.e., 1897—a date that is not in the sample. Results are unpredictable if, as in this illustration, parameters point to data that are not available.

Procedure **longhor1** can also handle applications such as (2), at the additional cost of the user specifying a lag length for an autoregression in the right hand side variable. Let x_t be a generic right hand side variable in a regression, with $x_t = i_t - i_t^*$ in (2) as an example. As noted above, b depends on own- and cross-covariances of the right hand side variables X_{t-1}—in this case, simply x_t—and the disturbance η_{t+q}. In **longhor1**, to compute the necessary second moments, the code relies in part on the presumption that the dynamics of x_t can be approximated by a finite order autoregression. The user must specify the order of this autoregression, i.e., the order of an autoregression in x_t that produces an approximately white noise disturbance. That is the purpose of the parameter **narlag**. In (3)–(5), it will normally be the case that **narlag** $=$ **nk**—one chooses to use k lags in the regression because use of k lags produces an approximately white noise disturbance. But in (2), the theory that leads to the regression does not constrain the order of an approximating autoregression for $x_t (= i_t - i_t^*)$.

To illustrate: suppose that the user decides that an AR(4) produces an approximately white noise disturbance in $i_t - i_t^*$. Then for (2), one invokes **longhor1** with

- **nk** = 1 (because there is only one stochastic right hand side variable), and
- **narlag** = 4 (on the user's conclusion that an AR(4) in $i_t - i_t^*$ produces an approximately white noise residual).

3.3 R function longhor

This is a generalization of **longhor1** in which a vector autoregression rather than an autoregression is used to compute autocovariances of the variables whose lags are in X_{t-1}. In the exchange rate example (2), one might suppose that sharper estimates of the moments of $i_t - i_t^*$ will result from use of the time series of exchange rates in addition to the time series of $i_t - i_t^*$.

Again let x_t be the variable whose lags are in X_{t-1}. Let $w_{1t}, \ldots, w_{nt}$ be n additional variables thought to have useful information about the autocovariances of x_t. Put these n additional variables in a matrix **Wseries**. Then to compute autocovariances from a VAR in $(x_t, w_{1t}, \ldots, w_{nt})$ one invokes **longhor** passing **Wseries** and setting **nWseries** = n. See Table 2.

3.4 R functions proc_vb_ma0 and proc_vb_maq

These are low level functions invoked by **longhor** and **longhor1**. They are flexible enough to allow computation of bias to order T in any least squares regression. An example of a regression covered by these procedures but not allowed by **longhor** or **longhor1** is a direct forecast that relies on a bivariate information set

TABLE 2 Function longhor

```
result <- longhor(yseries, xseries, Wseries, nWseries, first,
                  last, nq, nk, narlag)
vbias <- result[[1]]; betahat <- result[[2]]; betahat_adj
<- result[[3]]
```

The right hand side variables in the regression of interest consist of a constant and lags of a single variable, such as in (2)–(6). The left hand side may or may not be a lead of the same variable.

Parameters are as for **longhor1**, with the two additional parameters defined as

Wseries	matrix containing variables in addition to *xseries* to be used in the VAR that will be used to compute autocovariances of x_t
nWseries	the number of series or columns in **Wseries**. If **nWseries** = 0, the procedure calls **longhor1** to compute b

Functions invoked directly or indirectly by **longhor**: **longhor1**, **proc_vb_ma0**, **proc_vb_maq**, and **proc_vbias**.

$$y_{t+q} = \alpha + \beta_1 y_{t-1} + \beta_2 y_{t-2} + \cdots + \beta_m y_{t-m} + \beta_{m+1} x_{t-1} + \beta_{m+2} x_{t-2} + \cdots + \beta_k x_{t-m} + \eta_{t+q};$$
$$X'_{t-1} = (y_{t-1}, y_{t-2}, \ldots, y_{t-m}, x_{t-1}, x_{t-2}, \ldots, x_{t-m}). \tag{14}$$

In (14), $k = 2m$. Of course our functions also allow the left hand side variable to be not point in time as in (14) but an average or cumulated sum as in (2) or (5). The key difference between (14) and specifications covered by **longhor** and **longhor1** is that the latter require that X_{t-1} consist of lags of a single variable, whereas there are lags of two different variables on the right hand side of (14). The functions **proc_vb_ma0** and **proc_vb_maq** accommodate not only two but any number of different variables on the right hand side of the regression of interest.

In contrast to **longhor** and **longhor1**, **proc_vb_ma0** and **proc_vb_maq** require the user to do preliminary calculations before being invoked. First, the user, and not these functions, is required to estimate $\hat{\beta}$. These functions will compute $\hat{b}$ but not $\hat{\beta}$. Second, the user, and not these functions, must estimate a vector autoregression whose variables include those in X_{t-1}, passing certain results from this vector autoregression to these two functions.

To illustrate how to invoke these functions, let us use (14), setting $m = 2$ for concreteness:

$$y_{t+q} = \alpha + \beta_1 y_{t-1} + \beta_2 y_{t-2} + \beta_3 x_{t-1} + \beta_4 x_{t-2} + \eta_{t+q};\ k = 4;\ X'_{t-1} = (y_{t-1}, y_{t-2}, x_{t-1}, x_{t-2}). \tag{15}$$

The user needs to specify a vector autoregressive model for the right hand side variables in (15) that can be used by our R functions to deliver accurate estimates of the autocovariances of X_{t-1}. The fact that there are two lags on the right hand side of (15) suggests that a vector autoregression of order 2 will suffice. Let $Y_t(2 \times 1) = (y_t,\ x_t)'$. Write the VAR(2) as

$$Y_t = \text{const.} + \Phi_1 Y_{t-1} + \Phi_2 Y_{t-2} + V_t;\ V_t \sim \text{i.i.d};\ \Omega_V \equiv E V_t V_t'. \tag{16}$$

In (16), Φ_1, Φ_2, and Ω_V are 2×2; V_t is 2×1; here and in subsequent equations "const." is an inessential vector of constants whose dimension may be different in different equations.

The user must estimate a VAR such as (16) and pass to our R functions the estimates of the autoregressive coefficients (Φ_1 and Φ_2 in example (16)) and the variance–covariance matrix of the disturbance to the VAR (Ω_V in example (16)) to our R functions. These estimates are passed after rewriting the VAR in the VAR(1) companion form. For the VAR (16), the companion form is

$$\begin{pmatrix} Y_t \\ Y_{t-1} \end{pmatrix} = \text{const.} + \begin{pmatrix} \Phi_1 & \Phi_2 \\ I_2 & 0_{2\times 2} \end{pmatrix} \begin{pmatrix} Y_{t-1} \\ Y_{t-2} \end{pmatrix} + \begin{pmatrix} V_t \\ 0_{2\times 1} \end{pmatrix}, \text{written compactly as}$$

$$\underset{4\times 1}{Z_t} = \text{const.} + \underset{4\times 4}{\Phi}\ \underset{4\times 1}{Z_{t-1}} + \underset{4\times 1}{U_t}\,. \tag{17}$$

The general setup: let Z_t be the $n_Z \times 1$ vector of variables in the VAR used to compute moments related to X_t, with the VAR written in companion form. That is,

$$\underset{n_Z\times 1}{Z_t - EZ_t} = \underset{n_Z\times n_Z}{\Phi}\,(Z_{t-1} - EZ_{t-1}) + \underset{n_Z\times 1}{U_t}\,, \quad \underset{n_Z\times n_Z}{\Omega_U} = EU_tU_t', \quad \underset{k\times 1}{X_t} = \underset{k\times n_Z}{P_X}\ \underset{n_Z\times 1}{Z_t} \tag{18}$$

Note that the elements of X_t are elements of Z_t.

The user must compute and pass to the code: the dimension k of X_t (called **nk** in the code), the dimension of n_Z of Z_t (called **nZtwid** in the code), P_X and estimates $\hat{\Phi}$ and $\hat{\Omega}_U$ (called **PX**, **phitwid**, and **omegaUtwid** in the code). In the example (16) and (17), **nk** = 4, **nZtwid** = 4,

$$\mathbf{PX} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

and, letting "$\hat{}$"denote a least squares estimate or residual,

$$\mathbf{phitwid} = \begin{pmatrix} \hat{\Phi}_1 & \hat{\Phi}_2 \\ I_2 & 0_{2\times 2} \end{pmatrix}, \mathbf{omegaUtwid} = \begin{pmatrix} \hat{\Omega}_V & 0_{2\times 2} \\ 0_{2\times 2} & 0_{2\times 2} \end{pmatrix}, \hat{\Omega}_V = T^{-1}\sum \hat{V}_t\hat{V}_{t-1}'.$$

The user must also pass some moments related to the cross-covariances between Z_t or X_t on the one hand and η_t on the other. We supply separate function calls for (1) η_{t+q} i.i.d., and (2) $\eta_{t+q} \sim \text{MA}(q)$. The first is a special case of the second.

Let $\hat{\eta}_{t+q}$ be the least squares residuals. Let Z_t be the vector of variables in the companion form VAR. The two separate function calls are:

(1) $q = 0$ and $\eta_t \sim$ i.i.d.: The user needs to compute and pass an estimate of $E\eta_t Z_t'$, called **EetaZtwid0**. In example (16) and (17), in which $Z_t' = (y_t, y_{t-1}, x_t, x_{t-1})$,

$$\textbf{EetaZtwid0} = \left(T^{-1}\sum \hat{\eta}_t y_t, T^{-1}\sum \hat{\eta}_t y_{t-1}, T^{-1}\sum \hat{\eta}_t x_t, T^{-1}\sum \hat{\eta}_t x_{t-1}\right).$$

(To prevent misunderstanding: yes, in this example $T^{-1}\sum \hat{\eta}_t y_{t-1} = 0$ and $T^{-1}\sum \hat{\eta}_t x_{t-1} = 0$ by construction, since least square residuals are orthogonal to the regressors.)

(2) $\eta_{t+q} \sim \text{MA}(q)$: The user passes the integer parameter **nq** (the value of q) as well as (i) a matrix **EetaZtwid** and (ii) a vector **EXeta**. (i) and (ii) are defined as follows:

(i) EetaZtwid is a matrix of dimension $(q+1) \times n_Z$. In this matrix, for $i = 0, \ldots, q$, the $(i+1)$st row is a $1 \times$ **nZtwid** estimate of $E\eta_{t+q}Z'_{t+i}$. The estimate can be computed by the user as

$$\text{estimate of} E\eta_{t+q}Z'_{t+i} = T^{-1}\sum \hat{\eta}_{t+q}Z'_{t+i}.$$

(ii) EXeta is an estimate of the **nk** $\times 1$ vector $E(X_t + X_{t+1} + \cdots + X_{t+q-1})\eta_{t+q}$. The estimate can be computed by the user as

$$\textbf{EXeta} = T^{-1}\sum \left(X_t + X_{t+1} + \cdots + X_{t+q-1}\right)\hat{\eta}_{t+q}.$$

Here is the syntax to invoke the functions. The functions return the **nk** $\times 1$ estimate of $\hat{b}$, called **vbias**.

• $\eta_t \sim$ i.i.d : (19a)

```
vbias<-proc_vb_ma0(nZtwid, phitwid, omegaUtwid, nk, PX, EetaZtwid0)
```

• $\eta_t \sim MA(q)$: (19b)

```
vbias<-proc_vb_maq(nZtwid, phitwid, omegaUtwid, nk, PX, EetaZtwid, EXeta, nq).
```

Relative to the procedure used when $\eta_t \sim$ i.i.d., the procedure for $\eta_t \sim \text{MA}(q)$ requires that the user change one parameter (**EetaZtwid0** → **EetaZtwid**) and include two additional parameters (**EXeta** and **nq**). See Tables 3 and 4.

After obtaining **vbias** $\equiv \hat{b}$ from either procedure, the user must divide by sample size T, to obtain bias adjusted estimate $= \hat{\beta} - \frac{b}{T}$.

4 R code for an empirical application

Table 5 has R code for an application using **longhor1**. It estimates (4) with lag length $k = 2$ and horizon $q + 1 = 12$.

TABLE 3 Function proc_vb_ma0

```
vbias <- proc_vb_ma0(nZtwid, phitwid, omegaUtwid, nk, PX,
EetaZtwid0)
```

- In (1), $q=0$ and the regression disturbance $\eta_t \sim$ i. i. d. The right hand side variables in the regression are not restricted to be a constant and lags of a single variable.
- The user has estimated an auxiliary regression, written in companion form as

$$\underset{n_Z\times 1}{Z_t - EZ_t} = \underset{n_Z\times n_Z}{\Phi}(Z_{t-1} - EZ_{t-1}) + \underset{n_Z\times 1}{U_t}, \quad \underset{n_Z\times n_Z}{\Omega_U} = EU_tU_t', \quad \underset{k\times 1}{X_t} = \underset{k\times n_Z}{P_X}\ \underset{n_Z\times 1}{Z_t}.$$

  This regression produces an approximately white noise disturbance U_t.
- The user has also estimated (1), yielding least squares residuals $\{\hat{\eta}_t\}$.

Passed by user	
nZtwid	n_Z: integer number of variables in the auxiliary VAR (13)
phitwid	$\hat{\Phi}$: matrix of autoregressive estimates in (13)
omegaUtwid	$\hat{\Omega}_U$: matrix of estimates of the variance–covariance matrix in (13)
nk	k: integer number of variables in regression (1)
PX	P_X: matrix selecting from Z_t the right hand side variables in the regression of interest in (13) ($P_X = I$ is possible)
EetaZtwid0	vector estimate of $E\eta_t Z_t'$, i.e., $T^{-1}\sum \hat{\eta}_t Z_t'$
Returned to user	
vbias	**nk** $\times 1$ vector: $\hat{b}$ = estimate of $k\times 1$ numerator of bias to order T^{-1}

Functions invoked by **proc_vb_ma0**: **proc_vbias**.

TABLE 4 Function proc_vb_maq

```
vbias <- proc_vb_maq(nZtwid, phitwid, omegaUtwid, nk, PX,
                     EetaZtwid, EXeta, nq)
```

- In (1), q may be any integer and the regression disturbance $\eta_{t+q} \sim$ MA(q). The right hand side variables in the regression are not restricted to be a constant and lags of a single variable.
- The user has estimated an auxiliary regression, written in companion form as

$$\underset{n_Z\times 1}{Z_t - EZ_t} = \underset{n_Z\times n_Z}{\Phi}(Z_{t-1} - EZ_{t-1}) + \underset{n_Z\times 1}{U_t}, \quad \underset{n_Z\times n_Z}{\Omega_U} = EU_tU_t', \quad \underset{k\times 1}{X_t} = \underset{k\times n_Z}{P_X}\ \underset{n_Z\times 1}{Z_t}.$$

  This regression produces an approximately white noise disturbance U_t.
- The user has also estimated (1), yielding least squares residuals $\{\hat{\eta}_{t+q}\}$.

Continued

TABLE 4 Function proc_vb_maq**—Cont'd**

Passed by user	
nZtwid	n_Z: integer number of variables in the auxiliary VAR (13)
phitwid	$\hat{\Phi}$: matrix of autoregressive estimates in (13)
omegaUtwid	$\hat{\Omega}_U$: matrix of estimates of the variance–covariance matrix in (13)
nk	k: integer number of variables in regression (1)
PX	P_X: matrix selecting from Z_t the right hand side variables in the regression of interest in (13) ($P_X = I$ is possible)
EetaZtwid	estimates of $E\eta_{t+q}Z'_{t+i-1}$, $i=1$ to $q+1$. Matrix of dimension (**nq**+1) **nZtwid** (i.e., of dimension $(q+1)\,n_Z$). Row $i+1$ has the estimate of $E\eta_{t+q}Z'_{t+i}$, e.g., the first row of **EetaZtwid** is $T^{-1}\sum\hat{\eta}_{t+q}Z'_t$
EXeta	vector estimate of $E(X_t+X_{t+1}+\cdots+X_{t+q-1})\eta_{t+q}$, i.e., $T^{-1}\sum(X_t+X_{t+1}+\cdots+X_{t+q-1})\hat{\eta}_{t+q}$
nq	integer value of q in (1)
Returned to user	
vbias	**nk** $\times 1$ vector: $\hat{b}$ = estimate of $k\times 1$ numerator of bias to order T^{-1}
Functions invoked by **proc_vb_maq**: **proc_vbias**.	

TABLE 5 R code to illustrating use of longhor1

```
rm(list=ls())
library(MASS)
library(expm)
source("lagmatrix.R")
source("proc_vbias.R")
source("proc_vb_ma0.R")
source("proc_vb_maq.R")
source("UtilFunc_OLS.R")
source("longhor.R")
source("longhor1.R")

# Parameterwe Setting
T <- 240 # sample size
ARp <- 2 # Order of AR
q_horizon <- 11 # Forecast horizon is q+1
```

TABLE 5 R code to illustrating use of longhor1 **—Cont'd**

```
####
#insert code here to read data into "test data" from 1 to N, with
N>=T+q_horizon+Arp
# <code to read in data>
####


# OLS AR coefficients bias correction
result <- longhor1(yseries=test_data, xseries=test_data,
first=ARp+q_horizon+1,
                   last=ARp+q_horizon+T, nq=q_horizon, nk=ARp,
                   narlag=ARp)
vbias <- result[[1]]; betahat <- result[[2]]; betahat_adj
      <- result[[3]]
print(vbias)
print(betahat)
print(betahat_adj)
```

Acknowledgment

We thank Ziqi Chen for assistance in preparing the manuscript.

References

Box, G.E.P., Jenkins, G.M., 1976. Time Series Analysis: Forecasting and Control. Holden Day, San Francisco.

Crone, T.M., Khettry, N.K., Mester, L.J., Novak, J.A., 2013. Core measures of inflation as predictors of total inflation. J. Money Credit Bank. 45 (2), 505–519.

Engsted, T., Pedersen, T.Q., 2014. Bias-correction in vector autoregressive models: a simulation study. Econometrics 2, 45–71.

Hjalmarsson, E., 2011. New methods for inference in long-horizon regressions. J. Financ. Quant. Anal. 46, 815–839.

Ing, C., 2003. Multistep prediction in autoregressive processes. Economet. Theor. 254–279.

Kendall, M.G., 1954. Note on bias in the estimation of autocorrelation. Biometrika 41, 403–404.

Lunsford, K.G., West, K.D., 2017. Some evidence on secular drivers of U.S. safe real rates. In: Federal Reserve Bank of Cleveland Working Paper, pp. 17–23.

Marcellino, M., Stock, J.H., Watson, M.W., 2006. A comparison of direct and iterated multistep AR methods for forecasting macroeconomic time series. J. Econ. 135, 499–526.

Mark, N.C., 1995. Exchange rates and fundamentals: evidence on long-horizon predictability. Am. Econ. Rev. 85 (1), 201–218.

Shaman, P., Stine, R.A., 1988. The bias of autoregressive coefficient estimator. J. Am. Stat. Assoc. 83, 842–848.

Snaith, S., Coakley, J., Kellard, N., 2013. Does the forward premium puzzle disappear over the horizon? J. Bank. Financ. 37, 3681–3693.

West, K.D., 2016. Approximate Bias in time series regression. Manuscript. In: University of Wisconsin.

West, K.D., Zhao, Z., 2018. Improving Forecasts with Bias Adjustment. Manuscript in preparation. University of Wisconsin and University of Notre Dame.

Chapter 4

Hypothesis testing, specification testing, and model selection based on the MCMC output using R☆

Yong Li[a], **Jun Yu**[b] **and Tao Zeng**[c,*]

[a]*Hanqing Advanced Institute of Economics and Finance, Renmin University of China, Beijing, China*

[b]*School of Economics and Lee Kong Chian School of Business, Singapore Management University, Singapore, Singapore*

[c]*School of Economics, Academy of Financial Research, and Institute for Fiscal Big-Data & Policy of Zhejiang University, Zhejiang University, Zhejiang, China*

[*]*Corresponding author: e-mail: ztzt6512@gmail.com*

Abstract

This chapter overviews several MCMC-based test statistics for hypothesis testing and specification testing and MCMC-based model selection criteria developed in recent years. The statistics for hypothesis testing can be viewed as the MCMC version of the "trinity" of test statistics based in maximum likelihood (ML), namely, the likelihood ratio (LR) test, the Lagrange multiplier (LM) test, and the Wald test. The model selection criteria correspond to two predictive distributions. One of them can be viewed as the MCMC version of widely used information criterion, AIC. The asymptotic distributions of the test statistics and model selection criteria are discussed. The test statistics and model selection criteria are applied to several popular models using real data, one of which involves latent variables. The implementation is illustrated in R with the MCMC output obtained by R2WinBUGS.

JEL classification: C11, C12

Keywords: AIC, DIC, Information matrix, LR test, LM test, Markov chain Monte Carlo, Latent variable, Wald test

☆Li gratefully acknowledges the financial support of the Chinese Natural Science Fund (No. 71773130). Yu would like to acknowledge the financial support from Singapore Ministry of Education Academic Research Fund Tier 3 under the grant number MOE2013-T3-1-017. R code that implement our methods can be found at http://www.mysmu.edu/faculty/yujun/Handbook_Rcode.zip.

Handbook of Statistics, Vol. 41. https://doi.org/10.1016/bs.host.2018.12.003

1 Introduction

In economics and finance, statistical models with increasing complexity have been used more and more often. Typically empirical analysis of statistical models involves calculating and maximizing the log-likelihood function, leading to the maximum likelihood (ML) estimator. The ML estimator (MLE) has desirable asymptotic properties of consistency, normality, and efficiency under broad conditions, facilitating hypothesis testing, specification testing, and model selection. The asymptotic normality and efficiency of MLE make the well-known trinity of tests in ML popular in practice, i.e., the likelihood ratio (LR) test, the Wald test, and the Lagrange Multiplier (LM) test. In addition, some specification tests, such as the information matrix based tests, are based on MLE. Furthermore, some widely used information criteria for model selection, such as AIC, BIC, and HQ, are based on MLE.

Unfortunately, many statistical models face with a great deal of difficulties empirically in the sense that they cannot be easily estimated by ML. Examples include but not are restricted to latent variable models, continuous time models, models with complicated parameter restrictions, models in which the log-likelihood is not available in closed-form or is unbounded, models in which parameters are not point identified, high dimensional models for which numerical optimization is difficult to use, models with multiple local optimum in the log-likelihood function.

While for some of these models, alternative estimation methods, such as GMM, can be used. These alternative methods are generally less efficient than ML. With rapidly enhanced power in computing technology, the MCMC method has been used more and more frequently to provide the full likelihood analysis of models. MCMC is typically regarded as a Bayesian approach as it samples from the posterior distribution and the posterior mean is often chosen to be the Bayesian parameter estimate.

After the MCMC output is obtained, a few questions naturally arise. The first question is how to conduct hypothesis testing as one typically does after MLE is used to estimate a model. The second question is how to perform the specification test of the estimated model. The third question is how to compare alternative models that are not necessarily nested by each other. Hypothesis testing, specification testing and model selection are of fundamental importance in empirical studies. Therefore, MCMC-based answers to these questions become critically in practice. The traditional Bayesian answer to these questions is to use the gold standard, the Bayes factors (BFs), or it variants. The BFs basically compare the posterior model probabilities of candidate models, conditional on the data. Despite its appeal in the statistical interpretation, BFs suffer a few serious theoretical and computational difficulties. For example, it is not well-defined under improper priors. It subjects to Jeffreys-Lindley's paradox, that is, it tends to reject the null hypothesis even when the null is correct. For many models, BFs are difficult to compute.

The aim of this chapter is to overview the literature on MCMC-based statistical inference. However, we focus on test statistics and model selection

criteria which can be justified in a frequentist set up, in the same way as how the ML-based methods are justified. Since MCMC was introduced initially as a Bayesian tool, it is not immediately obvious how to make statistical inference based on the MCMC output in the frequentist framework. The essence of the literature is to treat MCMC as a sampling method and resort to the frequentist framework to obtain the asymptotic theory of various statistics based on the MCMC output in repeated sampling.

The statistics for hypothesis testing developed in the literature can be viewed as the MCMC version of the "trinity" of the tests in ML. The statistics for specification testing can be viewed as the MCMC version of the information matrix based test. One of the model selection criteria can be viewed as the MCMC version of AIC. Their asymptotic properties of these statistics are reviewed. The methods are illustrated using some important models widely used in economics and finance in a real data setting. The implementation is illustrated in R with the MCMC output obtained by R2WinBUGS.

MCMC can be used to sample from distributions other than the posterior. In a seminar paper, Chernozhukov and Hong (2003) proposed to use MCMC to sample from quasi-posterior. Moreover, the MCMC output may be used for other types of statistical inference. One example is to construct the confidence sets for identified sets of parameters in econometric models defined through a likelihood or a vector of moments; see Chen et al. (2016). Review of these studies are beyond of the scope of this chapter.

The chapter is organized as follows. Section 2 reviews the MCMC technique and introduces the implementation of MCMC using the R package. We also briefly explain the inferencial approach typically adopted in the Bayesian literature. Section 3 overviews several statistics for hypothesis testing based on the MCMC output. Section 4 overviews the MCMC-based test statistics for specification. Section 5 reviews DIC, an MCMC version of AIC, and other related information criteria. Section 6 gives the empirical illustrations. Section 7 concludes the chapter. R code that implement our methods can be found at http://www.mysmu.edu/faculty/yujun/Handbook_Rcode.zip.

2 MCMC and its implementation in R

Without loss of generality, we take the latent variable models as an example, to explain why ML is difficult to use and to describe how to obtain the MCMC output. Let $\mathbf{y}=(y_1, \ldots, y_n)$ denote the data generated from a probability measure P_0 on the probability space $(\Omega, \mathrm{F}, P_0)$. Let $\mathbf{z}=(\mathbf{z}_1, \mathbf{z}_2, \ldots, \mathbf{z}_n)'$ be the latent variables. The latent variable model is indexed by the some P-dimensional parameter vector, $\boldsymbol{\theta}$. Furthermore, $p(\mathbf{y}|\boldsymbol{\theta})$ is used to denote the observed-data likelihood function, and $p(\mathbf{y}, \mathbf{z}|\boldsymbol{\theta})$ is denoted as the complete-data likelihood function. The relationship between these two likelihood functions is given by

$$p(\mathbf{y}|\boldsymbol{\theta}) = \int p(\mathbf{y}, \mathbf{z}|\boldsymbol{\theta}) d\mathbf{z}. \tag{1}$$

In many latent variable modes, especially dynamic latent variable models, the latent variable $\mathbf{z}$ is often dependent on the sample size and its dimension is the same as or larger than the number of the sample size. When the sample size is large, the integral is high-dimensional. Often the integral does not have a closed-form solution and cannot be reduced into lower dimension integrals. In this case, it will be very difficult to accurately approximate the integral numerically. Consequently, ML is difficult to implement.

Now, we review the basic idea of MCMC. Let $p(\boldsymbol{\theta})$ be prior distribution assigned for parameter $\boldsymbol{\theta}$. Since the observed likelihood $p(\mathbf{y}|\boldsymbol{\theta})$ is intractable, it is very difficult to draw the random observations from the posterior distribution $p(\boldsymbol{\theta}|\mathbf{y})$ directly. To deal with this difficulty, the data-augmentation strategy (Tanner and Wong, 1987) can be applied to augment the parameter space from $\boldsymbol{\theta}$ to $(\boldsymbol{\theta}, \mathbf{z})$. As a result, the likelihood function becomes $p(\mathbf{y}|\boldsymbol{\theta},\mathbf{z})$ which typically is available in closed-form. The MCMC technique, such as Gibbs sampler, draws random samples from the joint posterior distribution $p(\boldsymbol{\theta}, \mathbf{z}|\mathbf{y})$. More concretely, we start with an initial value $[\boldsymbol{\theta}^{(0)}, \mathbf{z}^{(0)}]$, and then at the jth iteration, conditional on the current values $[\boldsymbol{\theta}^{(j)}, \mathbf{z}^{(j)}]$,

(a) generate $\boldsymbol{\theta}^{(j+1)}$ from $p(\boldsymbol{\theta}|\mathbf{z}^{(j)}, \mathbf{y})$;
(b) generate $\mathbf{z}^{(j+1)}$ from $p(\mathbf{z}|\boldsymbol{\theta}^{(j+1)}, \mathbf{z})$.

To get rid of the effect of the initial value, some random observations are discarded as the burn-in observations. After that, the simulated random samples can be regarded as efficient random draws (though correlated in general) from the joint posterior distribution $p(\boldsymbol{\theta}, \mathbf{z}|\mathbf{y})$. These correlated random samples are the MCMC output.

Based on the MCMC output, the parameter estimate can be obtained. For example, Bayesian estimates of $\boldsymbol{\theta}$ can be easily obtained as the sample mean of the generated random samples. Specifically, let $\{\boldsymbol{\theta}^{(j)}, j=1, 2, \ldots, J\}$ be effective random observations generated form the joint posterior distribution $p(\boldsymbol{\theta}, \mathbf{z}|\mathbf{y})$. Then Bayesian estimates of $\boldsymbol{\theta}$ is

$$\overline{\theta} = \frac{1}{J}\sum_{i=1}^{J} \theta^{(j)}.$$

This estimate is justified when the loss function is quadratic.

Under some regularity conditions, it is well documented in the literature (see, for example, Gelman et al., 2013) that the posterior distribution has a limiting normal distribution given by

$$\boldsymbol{\theta} - \hat{\boldsymbol{\theta}}|\mathbf{y} \overset{a}{\sim} N\left(0, \left[-\frac{1}{n}\frac{\partial^2 \ln p\left(\hat{\boldsymbol{\theta}}|\,\mathbf{y}\right)}{\partial\boldsymbol{\theta}\partial\boldsymbol{\theta}'}\right]^{-1}\right), \tag{2}$$

where $\hat{\boldsymbol{\theta}}$ is the posterior mode $\left(\text{i.e., } \hat{\boldsymbol{\theta}} = \arg\max \ \ln p(\boldsymbol{\theta}|\mathbf{y})\right)$ and

$$p(\boldsymbol{\theta}|\mathbf{y}) = \frac{p(\mathbf{y}|\boldsymbol{\theta})p(\boldsymbol{\theta})}{\int p(\mathbf{y}|\boldsymbol{\theta})p(\boldsymbol{\theta})\mathrm{d}\boldsymbol{\theta}}.$$

Furthermore, under extra regularity conditions, when $p(\boldsymbol{\theta}) = O_p(1)$, Li et al. (2017a) showed that the relationship between the posterior mean $\overline{\boldsymbol{\theta}}$ and the posterior mode $\hat{\boldsymbol{\theta}}$ can be expressed as

$$\overline{\boldsymbol{\theta}} = \hat{\boldsymbol{\theta}} + O_p\left(n^{-1}\right), \tag{3}$$

$$\widehat{Var}(\boldsymbol{\theta}|\mathbf{y}) = \left[-\frac{\partial^2 \ln p\left(\mathbf{y}|\hat{\boldsymbol{\theta}}\right)}{\partial\boldsymbol{\theta}\partial\boldsymbol{\theta}'}\right]^{-1} + O_p\left(n^{-2}\right). \tag{4}$$

The large sample properties in (2), (3) and (4) provide the fountainhead from which all the methods reviewed in this chapter springs.

In practice, however, MCMC procedures are not easy to implement using nonconventional software that is not widely available among researchers and practitioners. Therefore, it is practically important to find efficient software packages which can free the researchers from tedious programming and debugging. For this purpose, under the R language environment, Sturtz et al. (2005) introduced a so-called R2WinBUGS package combined with a free software WinBUGS1.4 to obtain the MCMC output. R is an extremely powerful language and environment for statistical computation and graphics which is available free of charge. WinBUGS is a user-friendly software package that implements the Gibbs sampler. It does sampling-based posterior computations for a variety of statistical models such as random effects, generalized linear, proportional hazards, latent variable, and frailty models. The latest version of WinBUGS is Win-BUGS1.4 which was developed by the medical Research Council Biostatistics Unit and the department of Epidemiology and Public Health of the Imperial College School of Medicine at St Mary's Hospital. It is available free of charge at http://www.mrc-bsu.cam.ac.uk/bugs/ An introduction to this software can be found in Spiegelhalter et al. (2003).

In this chapter, using the R language, we implement R2WinBUGS to get the MCMC outputs and then use R to compute the test statistics and the information criteria discussed below. The R code can be downloaded online where the detailed explanation for R commands is provided line by line in the R scripts by us. For more details about R2WinBUGS and WinBUGS1.4, one can refer to Sturtz et al. (2005) and Spiegelhalter et al. (2003). Special tailored R packages to obtain the MCMC output to fit particular statistical models are also available. For example, the R package named MCMC-Pack

was developed by Martin and Quinn (2005). Our R code to compute the test statistics and the information criteria discussed below may be also applied to the MCMC output generated by MCMCPack.

3 Hypothesis testing based on the MCMC output

3.1 Hypothesis testing under decision theory

Assume that a statistical model $M \equiv \{p(\mathbf{y}|\boldsymbol{\theta})\}$ is used to fit the data. The P-dimensional parameter vector $\boldsymbol{\theta}$ can be divided into two parts $\boldsymbol{\theta} = (\vartheta', \boldsymbol{\psi}')'$ where $\vartheta \in \boldsymbol{\Theta}$ denote a vector of p-dimensional parameter of interest and $\boldsymbol{\psi} \in \boldsymbol{\Psi}$ a vector of q-dimensional nuisance parameter. We are interested in knowing whether or not ϑ is equal to some value to verify a particular theory. Hence, the point null hypothesis problem can be written as

$$\begin{cases} H_0 : \vartheta = \vartheta_0 \\ H_1 : \vartheta \neq \vartheta_0 \end{cases}. \tag{5}$$

In this section, we discuss the hypothesis testing problem from a decision viewpoint.

Consider a decision problem whose decision space has two statistical decisions, to accept H_0 (name it d_0) or to reject H_0 (name it d_1). We may specify a loss function denoted by $\{\mathcal{L}[d_i, (\boldsymbol{\theta}, \boldsymbol{\psi})], i = 0, 1\}$ to measure the consequence of the statistical decision d_i. Let $p(\vartheta, \boldsymbol{\psi}|\mathbf{y})$ be the posterior distribution with some given prior $p(\vartheta, \boldsymbol{\psi})$, and $\mathbf{T}(\mathbf{y}, \vartheta_0)$ be a test statistic for hypothesis testing which is a function of the data $\mathbf{y}$. When the expected posterior loss of accepting H_0 is sufficiently larger than the expected posterior loss of rejecting H_0, i.e.,

$$\mathbf{T}(\mathbf{y}\vartheta_0) = \int_{\Theta}\int_{\Psi} \{\mathcal{L}[d_0(\vartheta\boldsymbol{\psi})] - \mathcal{L}[d_1(\vartheta\boldsymbol{\psi})]\} p(\vartheta, \boldsymbol{\psi}|\,\mathbf{y}) \mathrm{d}\vartheta \mathrm{d}\boldsymbol{\psi} > c,$$

we can say that the statistical decision of accepting H_0 might be inappropriate with some confidence so that the statistical decision to reject H_0 can be done naturally. For more details about hypothesis testing under decision theory, one can refer to Bernardo and Rueda (2002) and Bernardo and Smith (2006).

In practice, it is enough to specify the net loss function denoted by $\Delta\mathcal{L}[H_0, (\vartheta, \boldsymbol{\psi})] = \mathcal{L}[d_0, (\vartheta, \boldsymbol{\psi})] - \mathcal{L}[d_1, (\vartheta, \boldsymbol{\psi})]$. Hence, the test statistic can be rewritten as

$$\mathbf{T}(\mathbf{y}, \vartheta_0) = \int_{\Theta}\int_{\Psi} \Delta\mathcal{L}[H_0, (\vartheta, \boldsymbol{\psi})] p(\vartheta, \boldsymbol{\psi}|\,\mathbf{y}) \mathrm{d}\vartheta \mathrm{d}\boldsymbol{\psi} = E_{\boldsymbol{\theta}|\mathbf{y}}(\Delta\mathcal{L}[H_0, (\vartheta, \boldsymbol{\psi})]).$$

3.2 The choice of loss function for hypothesis testing

In the subsection, we review the loss functions for the purpose of constructing hypothesis test statistics. We show that the BFs correspond to the discrete loss

function that takes values of 0 and 1. To overcome the shortcomings of BFs, alterative continuous loss functions have been proposed in the literature to construct new test statistics based on the MCMC output. There is a more fundamental difference between these new test statistics and the BFs. The new test statistics are justified in a frequentist setup, that is, by assuming that $\mathbf{y}$ comes out of the data generating process in a repeated experiment whereas BFs is justified in a Bayesian setup, that is, the decision is made conditional on $\mathbf{y}$.

3.2.1 BFs and 0–1 loss function

If the 0–1 loss function is used, that is,

$$\mathcal{L}[d_0, (\boldsymbol{\vartheta}, \boldsymbol{\psi})] = \begin{cases} 0 & \text{if } \boldsymbol{\vartheta} = \boldsymbol{\vartheta}_0 \\ 1 & \text{if } \boldsymbol{\vartheta} \neq \boldsymbol{\vartheta}_0 \end{cases}, \quad \mathcal{L}[d_1, (\boldsymbol{\vartheta}, \boldsymbol{\psi})] = \begin{cases} 1 & \text{if } \boldsymbol{\vartheta} = \boldsymbol{\vartheta}_0 \\ 0 & \text{if } \boldsymbol{\vartheta} \neq \boldsymbol{\vartheta}_0 \end{cases},$$

the net loss function $\Delta\mathcal{L}[H_0, (\boldsymbol{\vartheta}, \boldsymbol{\psi})]$ is given by

$$\Delta\mathcal{L}[H_0, (\boldsymbol{\vartheta}, \boldsymbol{\psi})] = \begin{cases} -1 & \text{if } \boldsymbol{\vartheta} = \boldsymbol{\vartheta}_0 \\ 1 & \text{if } \boldsymbol{\vartheta} \neq \boldsymbol{\vartheta}_0 \end{cases}.$$

Hence, the test statistic based on this discrete loss function is given by

$$\begin{aligned} \mathbf{T}(\mathbf{y}, \boldsymbol{\vartheta}_0) &= \int_{\Theta}\int_{\Psi} \Delta\mathcal{L}[H_0, (\boldsymbol{\vartheta}, \boldsymbol{\psi})] p(\boldsymbol{\vartheta}, \boldsymbol{\psi} | \, \mathbf{y}) \mathrm{d}\boldsymbol{\vartheta}\mathrm{d}\boldsymbol{\psi} \\ &= \int_{\Theta}\int_{\Psi} \Delta\mathcal{L}[H_0, (\boldsymbol{\vartheta}, \boldsymbol{\psi})] \frac{p(\mathbf{y} | \, \boldsymbol{\vartheta}, \boldsymbol{\psi}) p(\boldsymbol{\vartheta}, \boldsymbol{\psi})}{p(\mathbf{y})} \mathrm{d}\boldsymbol{\vartheta}\mathrm{d}\boldsymbol{\psi}, \end{aligned}$$

where $p(\mathbf{y}) = \int_{\Theta}\int_{\Psi} p(\mathbf{y} \mid \boldsymbol{\vartheta}, \boldsymbol{\psi}) p(\boldsymbol{\vartheta}, \boldsymbol{\psi}) \mathrm{d}\boldsymbol{\vartheta}\mathrm{d}\boldsymbol{\psi}$ is the marginal likelihood.

In general, a positive probability w is assigned to the event $\boldsymbol{\vartheta} = \boldsymbol{\vartheta}_0$, such that a reasonable prior for $\boldsymbol{\vartheta}$ with a discrete support at $\boldsymbol{\vartheta}_0$ can be given by

$$p(\boldsymbol{\vartheta}) = \begin{cases} w & \text{if } \boldsymbol{\vartheta} = \boldsymbol{\vartheta}_0 \\ (1-w)\pi(\boldsymbol{\vartheta}) & \text{if } \boldsymbol{\vartheta} \neq \boldsymbol{\vartheta}_0 \end{cases}.$$

where $\pi(\boldsymbol{\vartheta})$ is a prior distribution. Hence, the test statistic under this discrete prior distribution can be expressed as

$$\begin{aligned} \mathbf{T}(\mathbf{y}, \boldsymbol{\vartheta}_0) &= \int_{\Theta}\int_{\Psi} \Delta\mathcal{L}[H_0, (\boldsymbol{\vartheta}, \boldsymbol{\psi})] \frac{p(\mathbf{y} | \, \boldsymbol{\vartheta}, \boldsymbol{\psi}) p(\boldsymbol{\vartheta}, \boldsymbol{\psi})}{p(\mathbf{y})} \mathrm{d}\boldsymbol{\vartheta}\mathrm{d}\boldsymbol{\psi} \\ &= -\int_{\Psi} \frac{p(\mathbf{y} | \, \boldsymbol{\vartheta}_0, \boldsymbol{\psi}) p(\boldsymbol{\vartheta}_0, \boldsymbol{\psi})}{p(\mathbf{y})} \mathrm{d}\boldsymbol{\vartheta}\mathrm{d}\boldsymbol{\psi} + \int_{\Theta}\int_{\Psi} \frac{p(\mathbf{y} | \, \boldsymbol{\vartheta}, \boldsymbol{\psi}) p(\boldsymbol{\vartheta}, \boldsymbol{\psi})}{p(\mathbf{y})} \mathrm{d}\boldsymbol{\vartheta}\mathrm{d}\boldsymbol{\psi} \\ &= -\int_{\Psi} \frac{p(\mathbf{y} | \, \boldsymbol{\vartheta}_0, \boldsymbol{\psi}) p(\boldsymbol{\psi} | \, \boldsymbol{\vartheta}_0) p(\boldsymbol{\vartheta} = \boldsymbol{\vartheta}_0)}{p(\mathbf{y})} \mathrm{d}\boldsymbol{\vartheta}\mathrm{d}\boldsymbol{\psi} + \int_{\Theta}\int_{\Psi} \frac{p(\mathbf{y} | \, \boldsymbol{\vartheta}, \boldsymbol{\psi}) p(\boldsymbol{\psi} | \, \boldsymbol{\vartheta}) p(\boldsymbol{\vartheta})}{p(\mathbf{y})} \mathrm{d}\boldsymbol{\vartheta}\mathrm{d}\boldsymbol{\psi} \\ &= -\int_{\Psi} \frac{p(\mathbf{y} | \, \boldsymbol{\vartheta}_0, \boldsymbol{\psi}) p(\boldsymbol{\psi} | \, \boldsymbol{\vartheta}_0) w}{p(\mathbf{y})} \mathrm{d}\boldsymbol{\vartheta}\mathrm{d}\boldsymbol{\psi} + \int_{\Theta}\int_{\Psi} \frac{p(\mathbf{y} | \, \boldsymbol{\vartheta}, \boldsymbol{\psi}) p(\boldsymbol{\psi} | \, \boldsymbol{\vartheta}) (1-w)\pi(\boldsymbol{\vartheta})}{p(\mathbf{y})} \mathrm{d}\boldsymbol{\vartheta}\mathrm{d}\boldsymbol{\psi}, \end{aligned}$$

where $p(\boldsymbol{\psi} \mid \boldsymbol{\vartheta})$ is the conditional prior distribution.

From this formula, we can see that the decision criterion can be made as

$$\text{Reject } H_0 \text{ iff } \int_{\Psi} p(\mathbf{y}|\,\boldsymbol{\vartheta}=\boldsymbol{\vartheta}_0,\boldsymbol{\psi})\omega p(\boldsymbol{\psi}|\,\boldsymbol{\vartheta}=\boldsymbol{\vartheta}_0)\mathrm{d}\boldsymbol{\psi} < \int_{\Theta}\int_{\Psi} p(\mathbf{y}|\,\boldsymbol{\vartheta},\boldsymbol{\psi})p(\boldsymbol{\psi}|\,\boldsymbol{\vartheta})(1-w)p(\boldsymbol{\vartheta})\mathrm{d}\boldsymbol{\theta}\mathrm{d}\boldsymbol{\psi}$$

To represent the prior ignorance, in practice, the probability w is set to 1/2 and the criterion becomes:

$$\text{Reject } H_0 \text{ iff } B_{01} = \frac{\int_{\Psi} p(\mathbf{y}|\,\boldsymbol{\vartheta}=\boldsymbol{\vartheta}_0,\boldsymbol{\psi})p(\boldsymbol{\psi}|\,\boldsymbol{\vartheta}=\boldsymbol{\vartheta}_0)\mathrm{d}\boldsymbol{\psi}}{\int_{\Theta}\int_{\Psi} p(\mathbf{y}|\,\boldsymbol{\vartheta},\boldsymbol{\psi})p(\boldsymbol{\psi}|\,\boldsymbol{\vartheta})\pi(\boldsymbol{\vartheta})\mathrm{d}\boldsymbol{\vartheta}\mathrm{d}\boldsymbol{\psi}} = \frac{m_0}{m_1} < 1,$$

where $\{m_k,\ k=0,\ 1\}$ are marginal likelihoods. B_{01} is the well-known BF defined as the ratio of the marginal likelihoods (Kass and Raftery, 1995).

Although BF is intuitively appealing and has a strong probabilistic interpretation, it is known to suffer from some theoretical and computational difficulties. First, when a subjective prior $\pi(\boldsymbol{\vartheta})$ is not available, Jeffreys' prior or reference prior (Bernardo and Smith, 2006; Jeffreys, 1961) are often used to reflect the lack of prior information. Jeffreys' prior and reference prior are generally improper. It follows that $\pi(\boldsymbol{\vartheta}) = C\,f\,(\boldsymbol{\vartheta})$, where $f\,(\boldsymbol{\vartheta})$ is a nonintegrable function, and C is an arbitrary positive constant. In this case, the BF can be expressed as

$$B_{01} = \frac{1}{C}\frac{\int_{\Psi} p(\mathbf{y}|\,\boldsymbol{\psi},\boldsymbol{\vartheta}_0)p(\boldsymbol{\psi}|\,\boldsymbol{\vartheta}_0)\mathrm{d}\boldsymbol{\psi}}{\int_{\Theta}\int_{\Psi} p(\mathbf{y}|\,\boldsymbol{\vartheta},\boldsymbol{\psi})p(\boldsymbol{\psi}|\,\boldsymbol{\vartheta})f(\boldsymbol{\vartheta})\mathrm{d}\boldsymbol{\vartheta}\mathrm{d}\boldsymbol{\psi}}.$$

Clearly, the BF is ill-defined since it depends on the arbitrary constant, C.

Second, to address the ill-defined problem of BF under the improper prior, a proper prior $\pi(\boldsymbol{\vartheta})$ with a large variance (that is a vague prior) has been proposed to represent the prior ignorance. While in this case the BF is well-defined, it has a tendency to favor the null hypothesis even when the null hypothesis is correct, giving rise to the notorious Jeffreys-Lindley's paradox; see Poirier (1995), Robert (1993, 2001). Jeffreys-Lindley's paradox leads to researchers to find variations to the BF. Examples include *partial Bayes factor* (O'Hagan, 1991), the *intrinsic Bayes factor* (Berger and Perrichi, 1996), and the *fractional Bayes factor* (O'Hagan, 1995). These variants basically split the data $\mathbf{y}$ into a training sample and a testing sample. The training sample is used to update an uninformative prior to obtain an informative prior. Unfortunately, they suffer from more or less arbitrary choices of training samples, weights for averaging training samples, and fractions, respectively.

Last but not least, for the latent variable model and many other models, calculation of the marginal likelihood M_k, $k=0$, 1 often involves intractable high-dimensional integrals, and, as a result, BFs are generally very difficult to calculate; see Han and Carlin (2001) for an excellent review of methods for calculating the BFs from the MCMC output.

3.2.2 *Bernardo and Rueda (2002) and the KL loss function*

Bernardo and Rueda (2002, BR hereafter) pointed out that if $\boldsymbol{\vartheta}$ is a continuous parameter, hypothesis testing forces the use of a nonregular (not absolutely continuous) "sharp" prior concentrating a positive probability mass so that the null hypothesis H_0 must have a strictly positive prior probability. This nonregular prior structure leads to the theoretical difficulties of BFs. To overcome these difficulties, Bernardo and Rueda (2002) suggested using a continuous loss function based on the Kullback–Leibler0 (KL) divergence to replace the discrete loss function, i.e.,

$$KL[p(x), q(x)] = \int p(x) \ln \frac{p(x)}{q(x)} dx,$$

where $p(x)$ and $q(x)$ are any two regular probability density functions. Then, the corresponding hypothesis test statistic can be given by:

$$\mathbf{T}_{BR}(\mathbf{y}, \boldsymbol{\vartheta}_0) = E_{\boldsymbol{\theta}|\mathbf{y}}(\min\{KL[p(\mathbf{y}|\boldsymbol{\vartheta}, \boldsymbol{\psi}), p(\mathbf{y}|\boldsymbol{\vartheta}_0, \boldsymbol{\psi})], KL[p(\mathbf{y}|\boldsymbol{\vartheta}_0, \boldsymbol{\psi}), p(\mathbf{y}|\boldsymbol{\vartheta}, \boldsymbol{\psi})]\}).$$

While $\mathbf{T}_{BR}(\mathbf{y}, \boldsymbol{\vartheta}_0)$ is well-defined under improper priors, since the KL divergence function often does not have a closed-form expression, $\mathbf{T}_{BR}(\mathbf{y}, \boldsymbol{\vartheta}_0)$ is difficult to compute for the latent variable model. Moreover, BR suggested choosing threshold values based on the normal distribution to implement the test. The rationale for basing threshold values on the normal distribution conceivably comes from the fact that many test statistics are asymptotically normally distributed. Therefore, BR's approach is not Bayesian as the sampling distribution of the test statistic is used and it is based on the idea of repeated sampling, not conditional on $\mathbf{y}$.

3.2.3 *Li and Yu (2012) and the $\mathcal{Q}$ loss function*

To address the computational problem in $\mathbf{T}_{BR}(\mathbf{y}, \boldsymbol{\theta}_0)$, Li and Yu (2012, LY hereafter) proposed a loss function based on the $\mathcal{Q}$function used in the EM algorithm (Dempster et al., 1977) to replace the KL divergence function. For any two points such as $\boldsymbol{\theta}_1$ and $\boldsymbol{\theta}_2$ defined in the parameter space, the $\mathcal{Q}$ function can be expressed as

$$\mathcal{Q}(\boldsymbol{\theta}_1|\boldsymbol{\theta}_2) = E_{\mathbf{Z}|\mathbf{y},\boldsymbol{\theta}_2}[\ln p(\mathbf{y}, \mathbf{z}|\boldsymbol{\theta}_1)].$$

Compared with the observed data likelihood function $p(\mathbf{y}|\boldsymbol{\theta})$, the $\mathcal{Q}$ function is easier to evaluate for the latent variable model. Let $\boldsymbol{\theta}_0=(\boldsymbol{\vartheta}_0, \boldsymbol{\psi})$, Li and Yu (2012) defined a new continuous net loss function as:

$$\Delta\mathcal{L}(\boldsymbol{\theta},\boldsymbol{\theta}_0)=\{\mathcal{Q}(\boldsymbol{\theta},\boldsymbol{\theta})-\mathcal{Q}(\boldsymbol{\theta}_0,\boldsymbol{\theta})\}+\{\mathcal{Q}(\boldsymbol{\theta}_0,\boldsymbol{\theta}_0)-\mathcal{Q}(\boldsymbol{\theta},\boldsymbol{\theta}_0)\},$$

and proposed a MCMC-based test statistic as:

$$\mathrm{T}_{LY}(\mathbf{y},\boldsymbol{\theta}_0)=E_{\boldsymbol{\theta}|\mathbf{y}}[\Delta\mathcal{L}(\boldsymbol{\theta},\boldsymbol{\vartheta}_0)].$$

While T_{LY} $(\mathbf{y}, \boldsymbol{\theta}_0)$ is well-defined under improper priors and easy to compute for the latent variable model, one still needs to specify some threshold values. Again, threshold values lack of rigorous statistical justifications. Importantly, the need to specify some threshold values suggests that LY's approach is not Bayesian.

3.2.4 *Li et al. (2014) and LR-type loss function*

To address the problem in choosing threshold values, Li et al. (2014, LZY hereafter) introduced another net continuous loss function based on the deviance function (Spiegelhalter et al., 2002) given by

$$\Delta\mathcal{L}[H_0,(\boldsymbol{\vartheta},\boldsymbol{\psi})]=2\ln p(\mathbf{y}|\boldsymbol{\vartheta},\boldsymbol{\psi})-2\ln p(\mathbf{y}|\boldsymbol{\vartheta}_0,\boldsymbol{\psi}).$$

The corresponding test statistic is

$$\mathbf{T}_{LZY}(\mathbf{y},\boldsymbol{\vartheta}_0)=2\int[\ln p(\mathbf{y}|\boldsymbol{\vartheta},\boldsymbol{\psi})-\ln p(\mathbf{y}|\boldsymbol{\vartheta}_0,\boldsymbol{\psi})]p(\boldsymbol{\vartheta},\boldsymbol{\psi}|\mathbf{y})\mathrm{d}\boldsymbol{\vartheta}\mathrm{d}\boldsymbol{\psi}. \quad (6)$$

Since the likelihood function $p(\mathbf{y}|\boldsymbol{\vartheta}, \boldsymbol{\psi})$ is often intractable for the latent variable model, to achieve computational tractability, under some regularity conditions, Li et al. (2014) developed an asymptotically equivalent form for $\mathbf{T}_{LZY}$ $(\mathbf{y}, \boldsymbol{\vartheta}_\mathbf{0})$, i.e.,

$$\begin{aligned}\mathbf{T}^*_{LZY}(\mathbf{y},\boldsymbol{\vartheta}_0)=2D+2\left[\ln p\left(\overline{\boldsymbol{\vartheta}},\overline{\boldsymbol{\psi}}\right)-\ln p(\overline{\boldsymbol{\psi}}|\boldsymbol{\vartheta}_0)\right]-2\int\ln p(\boldsymbol{\vartheta}|\boldsymbol{\psi})p(\boldsymbol{\theta}|\mathbf{y})\mathrm{d}\boldsymbol{\theta}\\-\left[p+q-\mathbf{tr}\left[-L^{(2)}_{0n}(\overline{\boldsymbol{\psi}})V_{22}\left(\overline{\boldsymbol{\theta}}\right)\right]\right],\end{aligned} \quad (7)$$

where $\overline{\boldsymbol{\theta}}=\left(\overline{\boldsymbol{\vartheta}},\overline{\boldsymbol{\psi}}\right)'$ is the posterior mean of $\boldsymbol{\theta}$ under H_1, and

$$D=\int_0^1\left\{\left(\overline{\boldsymbol{\vartheta}}-\boldsymbol{\vartheta}_0\right)'\left[E_{\mathbf{z}1\mathbf{y},\overline{\boldsymbol{\theta}}_b}\left(S_1\left(\mathbf{y},\mathbf{z}|\,\overline{\boldsymbol{\theta}}_b\right)\right)\right]\right\}\mathrm{d}b,$$

with $\overline{\boldsymbol{\theta}}_b=(1-b)\overline{\boldsymbol{\theta}}_*+b\overline{\boldsymbol{\theta}}$, for $b\in[0,\ 1]$, $\overline{\boldsymbol{\theta}}_*=(\boldsymbol{\vartheta}_0,\overline{\boldsymbol{\psi}})'$, $S(\mathbf{y},\ \mathbf{z}|\boldsymbol{\theta})=\partial\ \ln p(\mathbf{y},\ \mathbf{z}|\boldsymbol{\theta})/\partial\boldsymbol{\theta}$, $S_1(\cdot)$ being the subvector of $S(\mathbf{y},\ \mathbf{z}|\boldsymbol{\theta})$ corresponding to $\boldsymbol{\vartheta}$, $V_{22}\left(\overline{\boldsymbol{\theta}}\right)=E\left[(\boldsymbol{\psi}-\overline{\boldsymbol{\psi}})(\boldsymbol{\psi}-\overline{\boldsymbol{\psi}})'|\,\mathbf{y},H_1\right]$, the submatrix of $V\left(\overline{\boldsymbol{\theta}}\right)$ corresponding to $\boldsymbol{\psi}$, and $L^{(2)}_{0n}(\boldsymbol{\psi})=\partial^2\ \ln\ p(\mathbf{y},\ \boldsymbol{\psi}\,|\,\boldsymbol{\vartheta}_0)/\partial\boldsymbol{\psi}\,\partial\boldsymbol{\psi}'$.

To compute $\mathbf{T}^*_{LZY}(\mathbf{y}, \boldsymbol{\vartheta}_0)$, one mainly needs to evaluate the second derivative of $\ln p(\mathbf{y}|\boldsymbol{\theta})$. The well-known Louis formula by Louis (1982) suggests

$$\begin{aligned}\frac{\partial^2 \ln p(\mathbf{y}|\boldsymbol{\theta})}{\partial\boldsymbol{\theta}\partial\boldsymbol{\theta}'} &= E_{\mathbf{z}|\mathbf{y},\boldsymbol{\theta}}\left\{\frac{\partial^2 \ln(\mathbf{y},\mathbf{z}|\boldsymbol{\theta})}{\partial\boldsymbol{\theta}\partial\boldsymbol{\theta}'}\right\} + Var_{\mathbf{z}|\mathbf{y},\boldsymbol{\theta}}\{S(\mathbf{y},\mathbf{z}|\boldsymbol{\theta})\} \\ &= E_{\mathbf{z}|\mathbf{y},\boldsymbol{\theta}}\left\{\frac{\partial^2 \ln(\mathbf{y},\mathbf{z}|\boldsymbol{\theta})}{\partial\boldsymbol{\theta}\partial\boldsymbol{\theta}'} + S(\mathbf{y},\mathbf{z}|\boldsymbol{\theta})S(\mathbf{y},\mathbf{z}|\boldsymbol{\theta})'\right\} \\ &\quad - E_{\mathbf{z}|\mathbf{y},\boldsymbol{\theta}}\{S(\mathbf{y},\mathbf{z}|\boldsymbol{\theta})\}E_{\mathbf{z}|\mathbf{y},\boldsymbol{\theta}}\{S(\mathbf{y},\mathbf{z}|\boldsymbol{\theta})\}',\end{aligned}$$

where all the expectations are taken with respect to the conditional distribution of $\mathbf{z}$ given $\mathbf{y}$ and $\boldsymbol{\theta}$. Hence, we can use the following formula to calculate the second derivative of the observed-data likelihood function,

$$\begin{aligned}&E_{\mathbf{z}|\mathbf{y},\boldsymbol{\theta}}\left\{\frac{\partial^2 \ln(\mathbf{y},\mathbf{z}|\boldsymbol{\theta})}{\partial\boldsymbol{\theta}\partial\boldsymbol{\theta}'} + S(\mathbf{y},\mathbf{z}|\boldsymbol{\theta})S(\mathbf{y},\mathbf{z}|\boldsymbol{\theta})'\right\} \\ &\approx \frac{1}{J}\sum_{i=1}^{J}\left\{\frac{\partial^2 \ln(\mathbf{y},\mathbf{z}^{(j)}|\boldsymbol{\theta})}{\partial\boldsymbol{\theta}\partial\boldsymbol{\theta}'} + S\left(\mathbf{y},\mathbf{z}^{(j)}|\boldsymbol{\theta}\right)S\left(\mathbf{y},\mathbf{z}^{(j)}|\boldsymbol{\theta}\right)'\right\}, \\ &E_{\mathbf{z}|\mathbf{y},\boldsymbol{\theta}}\{S(\mathbf{y},\mathbf{z}|\boldsymbol{\theta})\} \approx \frac{1}{J}\sum_{i=1}^{J} S\left(\mathbf{y},\mathbf{z}^{(m)}|\boldsymbol{\theta}\right) = \frac{1}{J}\sum_{i=1}^{J}\frac{\partial \ln p(\mathbf{y},\mathbf{z}^{(j)}|\boldsymbol{\theta})}{\partial\boldsymbol{\theta}},\end{aligned}$$

where $\{\mathbf{z}^{(j)}, j=1, 2, \ldots, J\}$ are the MCMC samples of $\mathbf{z}$.

Since $\mathbf{T}_{LZY}$ is the posterior mean of the difference in deviance, $\mathbf{T}_{LZY}$ and $\mathbf{T}^*_{LZY}$ can be understood as the MCMC version of LR test. Li et al. (2014) pointed out that the proposed test statistic appeals in four aspects. First, they are well-defined under improper priors. Second, they do not suffer from Jeffreys-Lindley's paradox and, hence, can be used under non-informative vague priors. Third, at least, $\mathbf{T}^*_{LZY}$ is not difficult to compute. For the latent variable model, $\mathbf{T}^*_{LZY}(\mathbf{y}, \boldsymbol{\vartheta}_0)$ only involves the second derivative which is not very difficult to evaluate from the MCMC output.

Finally, under some mild regularity conditions, when the likelihood information dominates the prior information, Li et al. (2014) proved that under the null hypothesis

$$\begin{aligned}\mathbf{T}_{LZY}(\mathbf{y}, \boldsymbol{\vartheta}_0) &\overset{a}{\sim} \boldsymbol{\epsilon}'\left[\mathbf{I}\mathbf{J}_{11}^{1/2}(\boldsymbol{\theta}_0)\mathbf{J}_{11}(\boldsymbol{\theta}_0)\mathbf{I}\mathbf{J}_{11}^{1/2}(\boldsymbol{\theta}_0)\right]\boldsymbol{\epsilon} \\ &\quad - \left[p+q-\mathbf{tr}[-L_{0n}^{(2)}(\overline{\boldsymbol{\theta}})V_{22}(\overline{\boldsymbol{\theta}})]\right],\end{aligned} \tag{8}$$

$$\begin{aligned}\mathbf{T}^*_{LZY}(\mathbf{y}, \boldsymbol{\vartheta}_0) &\overset{a}{\sim} \boldsymbol{\epsilon}'\left[\mathbf{I}\mathbf{J}_{11}^{1/2}(\boldsymbol{\theta}_0)\mathbf{J}_{11}(\boldsymbol{\theta}_0)\mathbf{I}\mathbf{J}_{11}^{1/2}(\boldsymbol{\theta}_0)\right]\boldsymbol{\epsilon} \\ &\quad - \left[p+q-\mathbf{tr}[-L_{0n}^{(2)}(\overline{\boldsymbol{\theta}})V_{22}(\overline{\boldsymbol{\theta}})]\right],\end{aligned} \tag{9}$$

where $\boldsymbol{\epsilon}$ is a standard multivariate normal variate, $\boldsymbol{\theta_0} = (\boldsymbol{\vartheta_0}, \boldsymbol{\psi}0)$ the true value of $\boldsymbol{\theta}$, $\mathbf{J}(\boldsymbol{\theta_0})$ the Fisher information matrix given by

$$\mathbf{J}(\boldsymbol{\theta}_0) = \frac{1}{n}\int -L_n^{(2)}(\boldsymbol{\theta}_0)p(\mathbf{y}|\,\boldsymbol{\theta}_0)\mathrm{d}\mathbf{y},$$

$\mathbf{IJ}(\boldsymbol{\theta_0})$ the inverse of $\mathbf{J}(\boldsymbol{\theta_0})$, $\mathbf{J}_{11}(\boldsymbol{\theta_0})$, and $\mathbf{IJ}_{11}(\boldsymbol{\theta_0})$ the submatrices of $\mathbf{J}(\boldsymbol{\theta_0})$ and $\mathbf{IJ}(\boldsymbol{\theta_0})$, respectively, corresponding to $\boldsymbol{\vartheta}$. The asymptotic distributions given in (8) and (9) are obtained under the assumptions of repeated sampling and the diverged sample size. Clearly, the set up is also in the frequentist domain. A drawback of the test is that it is not asymptotically pivotal because the asymptotic distribution depends on some unknown population parameters.

3.2.5 *Li et al. (2015) and LM-type loss function*

To address the nonpivotal problem in the test statistic of Li et al. (2014, 2015) proposed to use a quadratic loss function given by

$$\Delta\mathcal{L}[H_0, \boldsymbol{\theta}] = \left(\boldsymbol{\vartheta} - \overline{\boldsymbol{\vartheta}}\right)' C_{\vartheta\vartheta}\left(\overline{\boldsymbol{\theta}}_0\right)\left(\boldsymbol{\vartheta} - \overline{\boldsymbol{\vartheta}}\right), \tag{10}$$

where

$$C(\boldsymbol{\theta}) = s(\boldsymbol{\theta})s(\boldsymbol{\theta})', \mathbf{s}(\boldsymbol{\theta}) = \frac{\partial \ln p(\mathbf{y}|\,\boldsymbol{\theta})}{\partial \boldsymbol{\theta}},$$

and $s(\boldsymbol{\theta})$ the score function of $\boldsymbol{\theta}$, $C_{\vartheta\vartheta}(\boldsymbol{\theta})$ is the submatrix of $C(\boldsymbol{\theta})$ corresponding to $\boldsymbol{\vartheta}$ and is semipositive definite, $\overline{\boldsymbol{\theta}}_0 = (\boldsymbol{\vartheta}_0, \overline{\boldsymbol{\psi}}_0)$ is the posterior mean of $\boldsymbol{\vartheta}$ under H_0, $\overline{\boldsymbol{\theta}}$ is the posterior mean of $\boldsymbol{\theta}$ under H_1. Based on this quadratic loss, naturally, the test statistic is given by

$$\mathbf{T}_{LLY}(\mathbf{y}, \boldsymbol{\vartheta}_0) = \int \Delta\mathcal{L}[H_0, \boldsymbol{\theta}]p(\boldsymbol{\theta}|\,\mathbf{y})\mathrm{d}\boldsymbol{\theta} = \int \left(\boldsymbol{\vartheta} - \overline{\boldsymbol{\vartheta}}\right)' C_{\vartheta\vartheta}\left(\overline{\boldsymbol{\theta}}_0\right)\left(\boldsymbol{\vartheta} - \overline{\boldsymbol{\vartheta}}\right)p(\boldsymbol{\theta}|\,\mathbf{y})\mathrm{d}\boldsymbol{\theta}, \tag{11}$$

where $p(\boldsymbol{\theta}|\mathbf{y})$ is the posterior distribution of $\boldsymbol{\theta}$ under H_1.

To compute $\mathbf{T}_{LLY}(\mathbf{y}, \boldsymbol{\vartheta}_0)$, one mainly needs to evaluate the first derivative of $\ln p(\mathbf{y}|\boldsymbol{\theta})$. For the latent variable model, $\ln p(\mathbf{y}|\boldsymbol{\theta})$ is often intractable. Under the EM algorithm (Dempster et al., 1977), it can be shown that

$$\frac{\partial \ln p(\mathbf{y}|\,\boldsymbol{\theta})}{\partial \boldsymbol{\theta}} = E_{\mathbf{z}|\mathbf{y},\boldsymbol{\theta}}\{S(\mathbf{y}, \mathbf{z}|\,\boldsymbol{\theta})\} \approx \frac{1}{J}\sum_{i=1}^{J} S\left(\mathbf{y}, \mathbf{z}^{(j)}|\,\boldsymbol{\theta}\right) = \frac{1}{J}\sum_{i=1}^{J}\frac{\partial \ln p\left(\mathbf{y}, \mathbf{z}^{(j)}|\,\boldsymbol{\theta}\right)}{\partial \boldsymbol{\theta}}$$

where $\{\mathbf{z}^{(j)}, j = 1, 2, \ldots, J\}$ are the MCMC samples of $\mathbf{z}$.

The proposed test can be viewed as the MCMC version of LM test. To see the link, let the LM statistic (Breusch and Pagan, 1980) be

$$\mathbf{LM} = s_{\vartheta}\left(\hat{\boldsymbol{\theta}}_0\right)\left[-\mathbf{IL}_{\vartheta\vartheta}^{(2)}\left(\hat{\boldsymbol{\theta}}\right)\right]s_{\vartheta}\left(\hat{\boldsymbol{\theta}}_0\right),$$

where $\hat{\boldsymbol{\theta}}_0 = (\boldsymbol{\vartheta}_0, \hat{\boldsymbol{\psi}}_0)$ is the MLE of $\boldsymbol{\theta}$ under the null hypothesis, $s_{\vartheta}(\boldsymbol{\theta})$ is subvector of $s(\boldsymbol{\theta})$ corresponding to $\boldsymbol{\vartheta}$, $\mathbf{IL}_{\vartheta\vartheta}(\boldsymbol{\theta})$ is the submatrix of $\mathbf{IL}(\boldsymbol{\theta})$ corresponding to $\boldsymbol{\vartheta}$, $\mathbf{IL}^{(2)}(\boldsymbol{\theta})$ is the inverse matrix of $\mathbf{L}^{(2)}(\boldsymbol{\theta}) := \partial^2 \ln p(\mathbf{y}|\boldsymbol{\theta})/\partial\boldsymbol{\theta}\partial\boldsymbol{\theta}'$. Under some regularity assumptions, when the null hypothesis is true and the likelihood dominates the prior, Li et al. (2015) showed that

$$\mathbf{T}_{LLY}(\mathbf{y}, \boldsymbol{\vartheta}_0) = \mathbf{LM} + o_p(1) \xrightarrow{d} \chi^2(p).$$

The test statistic $\mathbf{T}_{LLY}(\mathbf{y}, \boldsymbol{\vartheta}_0)$ has a few nice properties. For example, it is well-defined under an improper prior and immune to Jeffreys-Lindley's paradox. In addition, for the latent variable model it is not difficult to compute with the EM algorithm. Finally, it follows a pivotal χ^2_p asymptotically, and hence, it is easy to obtain threshold values.

3.2.6 *Li et al. (2019) and Wald-type loss function*

Although the test statistic proposed by Li et al. (2015) is convenient to calculate and has some good properties, it requires the MCMC output to be obtained twice, one under H_0 and the other under H_1. Based on another quadratic loss function, Li et al. (2019) proposed a test statistic which is only by-product of the MCMC output under H_1, and hence, is easier to compute.

Let the posterior covariance matrix under the alterative hypothesis be

$$\mathbf{V}(\overline{\boldsymbol{\theta}}) = E\left[(\boldsymbol{\theta} - \overline{\boldsymbol{\theta}})(\boldsymbol{\theta} - \overline{\boldsymbol{\theta}})' | \mathbf{y}, H_1\right] = \int (\boldsymbol{\theta} - \overline{\boldsymbol{\theta}})(\boldsymbol{\theta} - \overline{\boldsymbol{\theta}})' p(\boldsymbol{\theta}|\mathbf{y}) \mathrm{d}\boldsymbol{\theta},$$

where $\overline{\boldsymbol{\theta}}$ is the posterior mean of $\boldsymbol{\theta}$ under the alternative hypothesis H_1. Li et al. (2019) proposed the following net loss function for hypothesis testing

$$\Delta\mathcal{L}[H_0, \boldsymbol{\theta}] = (\boldsymbol{\vartheta} - \boldsymbol{\vartheta}_0)' \left[\mathbf{V}_{\vartheta\vartheta}(\overline{\boldsymbol{\theta}})\right]^{-1} (\boldsymbol{\vartheta} - \boldsymbol{\vartheta}_0),$$

where $\mathbf{V}_{\vartheta\vartheta}(\overline{\boldsymbol{\theta}})$ is the submatrix of $\mathbf{V}(\overline{\boldsymbol{\theta}})$ corresponding to $\boldsymbol{\vartheta}$, $\left[\mathbf{V}_{\vartheta\vartheta}(\overline{\boldsymbol{\theta}})\right]^{-1}$ is the inverse matrix of $\mathbf{V}_{\vartheta\vartheta}(\overline{\boldsymbol{\theta}})$. Then, the test statistic can be established as follows:

$$\mathbf{T}_{LLYZ}(\mathbf{y}, \boldsymbol{\vartheta}_0) = \int (\boldsymbol{\vartheta} - \boldsymbol{\vartheta}_0)' \left[\mathbf{V}_{\vartheta\vartheta}(\overline{\boldsymbol{\theta}})\right]^{-1} (\boldsymbol{\vartheta} - \boldsymbol{\vartheta}_0) p(\boldsymbol{\theta}|\mathbf{y}) \mathrm{d}\boldsymbol{\theta}. \tag{12}$$

To see the link between $\mathbf{T}_{LLYZ}(\mathbf{y}, \boldsymbol{\vartheta}_0)$ and the Wald statistic, define the Wald statistic by (Engle, 1984)

$$\mathbf{Wald} = \left(\hat{\boldsymbol{\vartheta}}_{ML} - \boldsymbol{\vartheta}_0\right)' \left[-\mathbf{IL}^{(2)}_{\vartheta\vartheta}\left(\hat{\boldsymbol{\theta}}_{ML}\right)\right]^{-1} \left(\hat{\boldsymbol{\vartheta}}_{ML} - \boldsymbol{\vartheta}_0\right)',$$

where $\hat{\boldsymbol{\theta}}_{ML} := \left(\hat{\boldsymbol{\vartheta}}_{ML}, \hat{\boldsymbol{\psi}}_{ML}\right)$ is the ML estimate of $\boldsymbol{\theta}$. Under some regularity assumptions, when the null hypothesis is true and the likelihood dominates the prior, Li et al. (2019) showed that

$$\mathbf{T}_{LLYZ}(\mathbf{y}, \boldsymbol{\vartheta}_0) = \mathbf{Wald} + o_p(1) \xrightarrow{d} \chi^2(p).$$

This is why $\mathbf{T}_{LLYZ}(\mathbf{y}, \boldsymbol{\vartheta}_0)$ may be viewed as a MCMC version of the Wald test.

It can be seen that $\mathbf{T}_{LLYZ}(\mathbf{y}, \vartheta_0)$ shared some nice properties with the test of Li et al. (2015). First, it is well-defined under improper prior distributions and avoids Jeffreys-Lindley's paradox. Second, the asymptotic distribution is pivotal so that the threshold values can be easily obtained from the $\chi^2(p)$ distribution. Most importantly, it is only by-product of the posterior output under H_1, and hence, is easier to compute.

Table 1 summarize the MCMC-based trinity of the tests and their key properties. It is important to emphasize that although they are constructed from the MCMC output which contains random draw from the Bayesian posterior distribution, the statistical inference made by the three tests is not conditional on the data. Instead, the justification of the three tests is done in a frequentist framework, requiring repeated sampling from the DGP and an asymptotic argument.

4 Specification testing based on the MCMC output

Detection of specification problems in economics has been a major concern. After ML is applied to estimate the model, several specification tests may be used, including the information matrix test of White (1982), the IOS and IOS_A tests of Presnell and Boos (2004). Recently, Li et al. (2018) proposed a specification test based on the MCMC output which can assess the validity of the model specification and can tell the source of model misspecification if the null model is rejected.

Let model P be a collection of candidate models indexed by parameters $\boldsymbol{\theta}$ whose dimension is q. Let $P_{\boldsymbol{\theta}}$ denote P indexed by $\boldsymbol{\theta}$. We say the model P is correctly specified if there exists $\boldsymbol{\theta}$, such that $P_0 \in P_{\boldsymbol{\theta}}$.

Arguably the best known specification test is based on the information matrix proposed by White (1982). For *i.i.d.* case, let $p(\mathbf{y}|\boldsymbol{\theta})$ denote the likelihood function of Model $P\boldsymbol{\theta}$ and

$$\begin{aligned}
\mathbf{s}(\mathbf{y},\boldsymbol{\theta}) &:= \partial \ln p(\mathbf{y}|\boldsymbol{\theta})/\partial\boldsymbol{\theta}, \mathrm{h}(\mathbf{y},\boldsymbol{\theta}) &&:= \partial^2 \ln p(\mathbf{y}|\boldsymbol{\theta})/\partial\boldsymbol{\theta}\partial\boldsymbol{\theta}' \\
\mathbf{H}(\boldsymbol{\theta}) &:= \int \mathbf{h}(\mathbf{y},\boldsymbol{\theta}) p(\mathbf{y}|\boldsymbol{\theta}) \mathrm{d}\mathbf{y}, \mathrm{J}(\boldsymbol{\theta}) &&:= \int \mathbf{s}(\mathbf{y},\boldsymbol{\theta})\mathbf{s}'(\mathbf{y},\boldsymbol{\theta}) p(\mathbf{y}|\boldsymbol{\theta}) \mathrm{d}\mathbf{y}
\end{aligned}$$

Let $d(\mathbf{y},\theta) := vech[\mathbf{h}(\mathbf{y},\theta) + \mathbf{s}(\mathbf{y},\theta)\mathbf{s}'(\mathbf{y},\theta)]$, where *vech* is the columnwise vectorization with the upper portion excluded. Let the ML-based sample counterparts of $\mathbf{H}(\boldsymbol{\theta})$ and $\mathbf{J}(\boldsymbol{\theta})$ be

$$\hat{\mathbf{H}}_n\left(\hat{\boldsymbol{\theta}}_{ML}\right) := \frac{1}{n}\sum_{t=1}^{n} \mathbf{h}\left(y_t, \hat{\boldsymbol{\theta}}_{ML}\right), \hat{\mathbf{J}}_n\left(\hat{\boldsymbol{\theta}}_{ML}\right) := \frac{1}{n}\sum_{t=1}^{n} \mathbf{s}\left(y_t \hat{\boldsymbol{\theta}}_{ML}\right)\mathbf{s}'\left(y_t, \hat{\boldsymbol{\theta}}_{ML}\right).$$

Let $D_n\left(\hat{\boldsymbol{\theta}}_{ML}\right) = \frac{1}{n}\sum_{t=n}^{n} d\left(y_t, \hat{\boldsymbol{\theta}}_{ML}\right)$ and $\dot{D}_n\left(\hat{\boldsymbol{\theta}}_{ML}\right) = \partial D_n\left(\hat{\boldsymbol{\theta}}_{ML}\right)/\partial\boldsymbol{\theta}$. If the model is correctly specified, then $\mathbf{H}(\boldsymbol{\theta}) + \mathbf{J}(\boldsymbol{\theta}) = 0$. White (1982) proposed the following information matrix test

$$\mathrm{IMT} = nD_n\left(\hat{\boldsymbol{\theta}}_{ML}\right) V_n^{-1}\left(\hat{\boldsymbol{\theta}}_{ML}\right) D_n\left(\hat{\boldsymbol{\theta}}_{ML}\right), \tag{13}$$

TABLE 1 Summary of MCMC-based trinity of tests

	T_{LZY}	T_{LLY}	T_{LLYZ}
Expression	$2\left[\ln p(\overline{\boldsymbol{\vartheta}}, \overline{\boldsymbol{\psi}}) - \ln p(\overline{\boldsymbol{\psi}} \mid \boldsymbol{\vartheta}_0)\right]$ $-2\int \ln p(\boldsymbol{\vartheta} \mid \boldsymbol{\psi}) p(\boldsymbol{\theta} \mid \mathbf{y}) \mathrm{d}\boldsymbol{\theta} + 2D$ $-\left[p + 1 - \mathbf{tr}[-L_{0n}^{(2)}(\overline{\boldsymbol{\psi}}) V_{22}(\overline{\boldsymbol{\theta}})\right]$	$\int (\boldsymbol{\vartheta} - \overline{\boldsymbol{\vartheta}})' \mathrm{C}_{\vartheta\vartheta}(\overline{\boldsymbol{\vartheta}}_0)$ $(\boldsymbol{\vartheta} - \overline{\boldsymbol{\vartheta}}) p(\boldsymbol{\vartheta} \mid \mathbf{y}) \mathrm{d}\boldsymbol{\vartheta}$	$\int (\boldsymbol{\vartheta} - \overline{\boldsymbol{\vartheta}}_0)' \left[\mathbf{V}_{\vartheta\vartheta}\overline{\boldsymbol{\theta}}\right]^{-1}$ $(\boldsymbol{\vartheta} - \boldsymbol{\vartheta}_0) p(\boldsymbol{\theta} \mid \mathbf{y}) \mathrm{d}\boldsymbol{\theta}$
Prior	Improper or proper	Improper or proper	Improper or proper
Jeffreys-Lindley's Paradox	No	No	No
Asymptotic theory	$\boldsymbol{\epsilon}' \left[\mathbf{I}\mathbf{J}_{11}^{1/2}(\boldsymbol{\vartheta}_0) \mathbf{J}_{11}(\boldsymbol{\vartheta}_0) \mathbf{I}\mathbf{J}_{11}^{1/2}(\boldsymbol{\vartheta}_0)\right] \boldsymbol{\epsilon}$ $-\left[p + q - \mathbf{tr}\left[-L_{0n}^{(2)}(\overline{\boldsymbol{\theta}}) V_{22}(\overline{\boldsymbol{\theta}})\right]\right]$	$\chi^2(p)$	$\chi^2(p)$
Asymptotic pivotal	No	Yes	Yes

where

$$V_n\left(\hat{\boldsymbol{\theta}}_{ML}\right)=\frac{1}{n}\sum_{t=1}^{n}v_t\left(\hat{\boldsymbol{\theta}}_{ML}\right)v_t\left(\hat{\boldsymbol{\theta}}_{ML}\right)',$$

$$v_t\left(\hat{\boldsymbol{\theta}}_{ML}\right)=d\left(y_t,\hat{\boldsymbol{\theta}}_{ML}\right)-\dot{D}_n\left(\hat{\boldsymbol{\theta}}_{ML}\right)\hat{\mathbf{H}}_n^{-1}\left(\hat{\boldsymbol{\theta}}_{ML}\right)\mathbf{s}\left(y_t,\hat{\boldsymbol{\theta}}_{ML}\right).$$

He then showed that $\text{IMT}\xrightarrow{d}\chi^2$ as $n\to\infty$ under the null hypothesis.

Presnell and Boos (2004) proposed an alternative test—the "in-and-out" likelihood ratio (IOS) test for models with *i.i.d.* observations,

$$\text{IOS}=\ln\frac{\prod_{t=1}^{n}p\left(y_t,\hat{\boldsymbol{\theta}}_{ML}\right)}{\prod_{t=1}^{n}p\left(y_t,\hat{\boldsymbol{\theta}}_{ML}^{(t)}\right)}=\sum_{t=1}^{n}\left[\ln p\left(y_t|\,\hat{\boldsymbol{\theta}}_{ML}\right)-\ln p\left(y_t,\hat{\boldsymbol{\theta}}_{ML}^{(t)}\right)\right],$$

where $\hat{\boldsymbol{\theta}}_{ML}^{(t)}$ be the MLE of $\boldsymbol{\theta}$ when the t-th observation, y_t, is deleted from the whole sample. They showed that the asymptotic form of IOS is

$$\text{IOS}_A=\mathbf{tr}\left[-\hat{\mathbf{H}}_n^{-1}\left(\hat{\boldsymbol{\theta}}_{ML}\right)\hat{\mathbf{J}}_n\left(\hat{\boldsymbol{\theta}}_{ML}\right)\right],\tag{14}$$

and $\text{IOS}-\text{IOS}_A=o_p\ (n^{-1/2})$. Like IMT, IOS_A also compares $\hat{H}_n\left(\hat{\boldsymbol{\theta}}_{ML}\right)$ with $\hat{J}_n\left(\hat{\boldsymbol{\theta}}_{ML}\right)$, but in a ratio form instead of an additive form. Under the null hypothesis, $\text{IOS}_A\xrightarrow{p}q$ and $n^{1/2}\ (\text{IOS}_A-q)$ converges to a normal distribution with zero mean and finite variance. It is well documented in the literature that the asymptotic distributions poorly approximate their finite sample counterparts for IMT, IOS, and IOS_A. As a result, they all suffer from serious bias distortions if the critical values for testing are based on the asymptotic distributions. The poor finite sample performance of these tests is not surprising as the asymptotic theory is derived based on the convergence of the sample high order moments, whose speed is slow. To reduce the size distortion of these tests, bootstrap methods have been proposed to obtain the critical values. Unfortunately, bootstrap methods are computationally demanding.

For weakly dependent data, let $\mathbf{y}^t:=(y_1,\ldots,y_t)$ and

$$\begin{aligned}
\mathbf{s}(\mathbf{y}^t,\boldsymbol{\theta}) &:=\frac{\partial\ln p(\mathbf{y}^t|\,\boldsymbol{\theta})}{\partial\boldsymbol{\theta}}, & \mathbf{h}(\mathbf{y}^t,\boldsymbol{\theta}) &:=\frac{\partial^2\ln p(\mathbf{y}^t|\,\boldsymbol{\theta})}{\partial\boldsymbol{\theta}\partial\boldsymbol{\theta}'},\\
\mathbf{s}_t(\boldsymbol{\theta}) &:=\mathbf{s}(\mathbf{y}^t,\boldsymbol{\theta})-\mathbf{s}(\mathbf{y}^{t-1},\boldsymbol{\theta}), & \mathbf{h}_t(\boldsymbol{\theta}) &:=\mathbf{h}(\mathbf{y}^t,\boldsymbol{\theta})-\mathbf{h}(\mathbf{y}^{t-1},\boldsymbol{\theta}),\\
\hat{\mathbf{J}}_n(\boldsymbol{\theta}) &:=\frac{1}{n}\sum_{t=1}^{n}\mathbf{s}_t(\boldsymbol{\theta})\mathbf{s}_t'(\boldsymbol{\theta}), & \hat{\mathbf{H}}_n(\boldsymbol{\theta}) &:=\frac{1}{n}\sum_{t=1}^{n}\mathbf{h}_t(\boldsymbol{\theta}).
\end{aligned}$$

and $V\left(\overline{\boldsymbol{\theta}}\right)=\int\left(\boldsymbol{\theta}-\overline{\boldsymbol{\theta}}\right)\left(\boldsymbol{\theta}-\overline{\boldsymbol{\theta}}\right)'p(\boldsymbol{\theta}|\,\mathbf{y})\mathrm{d}\boldsymbol{\theta}$, a natural MCMC-based informative matrix test statistic can be defined as:

$$\text{BIMT}=\mathbf{tr}\left[nV\left(\overline{\boldsymbol{\theta}}\right)\hat{\mathbf{J}}_n\left(\overline{\boldsymbol{\theta}}\right)\right]=n\int\left(\boldsymbol{\theta}-\overline{\boldsymbol{\theta}}\right)'\hat{\mathbf{J}}_n\left(\overline{\boldsymbol{\theta}}\right)\left(\boldsymbol{\theta}-\overline{\boldsymbol{\theta}}\right)p(\boldsymbol{\theta}|\,\mathbf{y})\mathrm{d}\boldsymbol{\theta},\tag{15}$$

Under some mild regularity conditions, Li et al. (2018) showed that under the null hypothesis, $n^{1/2}$ (BIMT/$q-1$) has the same asymptotic distribution as $n^{1/2}$ (IOS$_A$/$q-1$). Hence, BIMT may be regarded as the MCMC-based version of IOS$_A$. Unfortunately but not surprisingly, BIMT inherits the size distortion problem of IOS$_A$ and bootstrap methods must be used.

Due to this size distortion problem, Li et al. used a technique of Fan et al. (2015) to construct a new specification test statistic. In particular, they propose to expand $p(\mathbf{y}|\boldsymbol{\theta})$, the model in concern, to a larger model denoted by $p(\mathbf{y}|\boldsymbol{\theta}_L)$ where $\boldsymbol{\theta}_L=(\boldsymbol{\theta}',\boldsymbol{\theta}_E')'$ with $\boldsymbol{\theta}_E$ being a q_E-dimensional vector. So the expanded model $p(\mathbf{y}|\boldsymbol{\theta}_L)$ nests the original model $p(\mathbf{y}|\boldsymbol{\theta})$.

It is assumed that if the specification $p(\mathbf{y}|\boldsymbol{\theta})$ is correct, then the true value of $\boldsymbol{\theta}_E$ is zero. The final specification test statistic of Li et al. (2018) has the form of

$$\text{BMT}=\mathbf{tr}\left\{C_E\left(\mathbf{y},\left(\overline{\boldsymbol{\theta}},\boldsymbol{\theta}_E=0\right)\right)V_E\left(\overline{\boldsymbol{\theta}}_L\right)\right\}+\sqrt{n}(\text{BIMT}/q-1)^2, \tag{16}$$

where $C_E\left(\mathbf{y},\left(\overline{\boldsymbol{\theta}},\boldsymbol{\theta}_E=0\right)\right)$ is the submatrix of C $(\mathbf{y},\boldsymbol{\theta}_L)$ corresponding to $\boldsymbol{\theta}_E$ evaluated at $\left(\overline{\boldsymbol{\theta}},\boldsymbol{\theta}_E=0\right)$ and $V_E\left(\overline{\boldsymbol{\theta}}_L\right)$ is the submatrix of $V_E(\boldsymbol{\theta}_L)$ corresponding to $\boldsymbol{\theta}_E$ evaluated at, $\overline{\boldsymbol{\theta}}_L$ and

$$s(\mathbf{y},\boldsymbol{\theta}_L)=\frac{\partial \ln p(\mathbf{y}|\boldsymbol{\theta}_L)}{\partial\boldsymbol{\theta}_L},C(\mathbf{y},\boldsymbol{\theta}_L)=s(\mathbf{y},\boldsymbol{\theta}_L)s(\mathbf{y},\boldsymbol{\theta}_L)',$$

$$V\left(\overline{\boldsymbol{\theta}}_L\right)=E\left[\left(\boldsymbol{\theta}_L-\overline{\boldsymbol{\theta}}_L\right)\left(\boldsymbol{\theta}_L-\overline{\boldsymbol{\theta}}_L\right)'|\,\mathbf{y}\right]=\int\left(\boldsymbol{\theta}_L-\overline{\boldsymbol{\theta}}_L\right)\left(\boldsymbol{\theta}_L-\overline{\boldsymbol{\theta}}_L\right)'p(\boldsymbol{\theta}_L|\mathbf{y})\mathrm{d}\boldsymbol{\theta}_L,$$

with $\overline{\boldsymbol{\theta}}_L$ being the posterior mean of $\boldsymbol{\theta}_L$ in the expanded model. It can be seen that BIMT is used as the power enhancement function.

Under a set of regularity conditions, Li et al. showed that if the model is correctly specified, $BMT\xrightarrow{d}\chi^2(q_E)$; but if the model is misspecified with $q^*\neq q$, then

$$\mathbf{tr}\left\{C_E\left(\mathbf{y},\left(\overline{\boldsymbol{\theta}},\boldsymbol{\theta}_E=0\right)\right)V_E\left(\overline{\boldsymbol{\theta}}_L\right)\right\}=\sqrt{n}(q^*/q-1)^2+O_p\left(\sqrt{n}\right),\text{BMT}\sim O_p\left(\sqrt{n}\right),$$

where $q^*=\mathbf{tr}[-\mathbf{H}\,(\boldsymbol{\theta}^*)^{-1}\,\mathbf{J}\,(\boldsymbol{\theta}^*)]$ with $\boldsymbol{\theta}^*$ being the pseudo true value of $\boldsymbol{\theta}$, where

$$\mathbf{H}(\boldsymbol{\theta}^*) \coloneqq \lim_{n\to\infty}\mathbf{H}_n(\boldsymbol{\theta}^*)\,\text{and}\,\mathbf{J}(\boldsymbol{\theta}^*)\coloneqq\lim_{n\to\infty}\mathbf{J}_n(\boldsymbol{\theta}^*),$$

$$\mathbf{J}_n(\boldsymbol{\theta}) \coloneqq \int\hat{\mathbf{J}}_n(\boldsymbol{\theta})p(\mathbf{y})d\mathbf{y},\mathbf{H}_n(\boldsymbol{\theta})\coloneqq\int\hat{\mathbf{H}}_n(\boldsymbol{\theta})p(\mathbf{y})d\mathbf{y},$$

BMT has several nice properties. First, compared with IM, IOS, and IOS$_A$, BMT is based on the MCMC output. When the likelihood function is difficult to optimize but the MCMC draws from the posterior distribution are available, BMT is easier to compute than IM, IOS, and IOS$_A$. Second, when $\sqrt{n}(\text{BIMT}/q-1)^2$ does not have the size distortion problem, it is most likely that BMT will not suffer from size distortion. As a result, no bootstrap method is needed and intensive computational effort is avoided.

5 Model selection based on the MCMC output

Model selection is a very important statistical decision in practice. Many important and widely used information criteria have been proposed to select from candidate models in the literature. Examples include AIC, BIC, and HQ. Most of them require that MLE is available. The most well-known model selection criterion based on the MCMC output is DIC of Spiegelhalter et al. (2002). DIC is constructed based on the posterior distribution of the log-likelihood or the deviance, and has several desirable features. First, DIC is simple to calculate from the MCMC output when the likelihood function is available in closed-form. Second, DIC is applicable to a wide range of statistical models. Third, unlike BFs, DIC is not subject to Jeffreys-Lindley's paradox and can be defined under improper priors. In this section, we first review the DIC for models when the asymptotic theory for ML is applicable, paying particular attention to the asymptotic justification of DIC. We also discuss how to obtain DICs when there are latent variables. In both cases, the loss function is the plug-in predictive loss. We also discuss the information criteria when the loss function is the Bayesian predictive loss.

5.1 DIC for regular models

We first review DIC for regular models, that is, when the asymptotic theory given by (2), (3) and (4) holds true. Spiegelhalter et al. (2002) proposed the DIC for Bayesian model comparison. The criterion is based on the deviance

$$D(\boldsymbol{\theta}) = -2\ln p(\mathbf{y}|\boldsymbol{\theta}),$$

and takes the form of

$$\mathrm{DIC} = D\left(\overline{\boldsymbol{\theta}}\right) + 2P_D, \tag{17}$$

where P_D, used to measure the model complexity and also known as "effective number of parameters," is defined as the difference between the posterior mean of the deviance and the deviance evaluated at the posterior mean of the parameters:

$$P_D = \overline{D(\boldsymbol{\theta})} - D\left(\overline{\boldsymbol{\theta}}\right) = -2\int [\ln p(\mathbf{y}|\boldsymbol{\theta}) - \ln p(\mathbf{y}|\boldsymbol{\theta})]p(\mathbf{y}|\boldsymbol{\theta})\mathrm{d}\boldsymbol{\theta}, \tag{18}$$

with $\overline{\boldsymbol{\theta}}$ being the posterior mean of $\boldsymbol{\theta}$.

Under some regularity conditions, Li et al. (2017a) gives a rigorous decision-theoretic justification. Let $g(\mathbf{y})$ be the data generating process of $\mathbf{y}$, $\mathbf{y}_{rep} = (y_{1,rep}, \ldots, y_{n,rep})'$ denote the future replicate data with $\mathbf{y}$. Hence, the plug-in predictive distribution based on replicate data is $-2\ln p\left(\mathbf{y}_{rep}|\overline{\boldsymbol{\theta}}(\mathbf{y})\right)$

where $\overline{\boldsymbol{\theta}}(\mathbf{y})$ is the posterior mean under the data $\mathbf{y}$. Consider the plug-in predictive distribution $p\left(\mathbf{y}_{rep}|\overline{\boldsymbol{\theta}}(\mathbf{y})\right.$ in the following KL divergence

$$KL\left[g\left(\mathbf{y}_{rep}\right), p\left(\mathbf{y}_{rep}|\,\overline{\boldsymbol{\theta}}_n(\mathbf{y})\right)\right] = E_{\mathbf{y}_{rep}}\left[\ln\frac{g\left(\mathbf{y}_{rep}\right)}{p\left(\mathbf{y}_{rep}|\,\overline{\boldsymbol{\theta}}_n(\mathbf{y})\right)}\right]$$
$$= E_{\mathbf{y}_{rep}}\left[\ln g\left(\mathbf{y}_{rep}\right)\right] + E_{\mathbf{y}_{rep}}\left[-\ln p\left(\mathbf{y}_{rep}|\,\overline{\boldsymbol{\theta}}_n(\mathbf{y})\right)\right].$$

The smaller this KL divergence, the better the candidate model in predicting $g(\mathbf{y}_{rep})$. Since $g(\mathbf{y}_{rep})$ is the true DGP and $E_{\mathbf{y}_{rep}} \ln g(\mathbf{y}_{rep})$ is independent with candidate models, it is dropped from the above equation. Li et al. (2017a) showed that DIC is an unbiased estimator of $E_{\mathbf{y}}E_{\mathbf{y}_{rep}}\left[-2\ln p\left(\mathbf{y}_{rep}|\,\overline{\boldsymbol{\theta}}(\mathbf{y})\right)\right]$ asymptotically, i.e., $E_{\mathbf{y}}E_{\mathbf{y}_{rep}}\left[-2\ln p\left(\mathbf{y}_{rep}|\,\overline{\boldsymbol{\theta}}\right)\right] = E_{\mathbf{y}}(\mathrm{DIC}) + o(1)$. The key assumptions to obtain the asymptotic unbiasedness include that the candidate models are good approximation to the true DGP, the consistency and asymptotic normality of MLE, and the expression for the asymptotic variance of MLE. For details, see Li et al. (2017a).

The above decision-theoretic justification to DIC is that DIC selects a model that asymptotically minimizes the risk, which is the expected KL divergence between the DGP and the plug-in predictive distribution $p\left(\mathbf{y}_{rep}|\,\overline{\boldsymbol{\theta}}(\mathbf{y})\right)$ where the expectation is taken with respect to the DGP. A key difference between AIC and DIC is that the plug-in predictive distribution is based on different estimators. In AIC, the ML estimate, $\hat{\boldsymbol{\theta}}_{ML}(\mathbf{y})$, is used while in DIC the Bayesian posterior mean, $\overline{\boldsymbol{\theta}}(\mathbf{y})$, is used.

When $\ln p(\mathbf{y}|\boldsymbol{\theta})$ has a closed-form expression, it can be seen that DIC is trivial to compute from the MCMC output. DIC has been incorporated into a Bayesian software, WinBUGS. This explains why DIC has been widely used in practice for model selection.

5.2 Bayesian predictive distribution as the loss function

Unfortunately, the plug-in predictive distribution is not invariant to parameterization. As a result, DIC is sensitive to parameterization. Alternatively, we may use the Bayesian predictive distribution as a loss function. The Bayesian predictive distribution is not only a full proper predictive distribution, but also invariant to reparameterization.

Let $p(\mathbf{y}_{rep}|\mathbf{y})$ be the Bayesian predictive distribution, that is,

$$p\left(\mathbf{y}_{rep}|\,\mathbf{y}\right) = \int p\left(\mathbf{y}_{rep}|\,\boldsymbol{\theta}\right)p(\boldsymbol{\theta}|\,\mathbf{y})d\boldsymbol{\theta}.$$

The KL divergence based on the Bayesian predictive distribution is given by

$$KL\left[g\left(\mathbf{y}_{rep}\right), p\left(\mathbf{y}_{rep}|\,\mathbf{y}\right)\right] = E_{\mathbf{y}_{rep}}\left(\ln g\left(\mathbf{y}_{rep}\right)\right) - E_{\mathbf{y}_{rep}}\left(\ln p\left(\mathbf{y}_{rep}|\,\mathbf{y}\right)\right). \tag{19}$$

Li et al. (2017a) obtained the information criterion based on the Bayesian predictive distribution as

$$\mathrm{DIC}^{BP} = D\left(\overline{\boldsymbol{\theta}}\right) + (1 + \ln 2)P_D. \tag{20}$$

Under some regularity assumptions, Li et al. showed that DIC^{BP} is an unbiased estimator of $E_{\mathbf{y}}E_{\mathbf{y}_{rep}}[-2\ln p(\mathbf{y}_{rep}|\,\mathbf{y})]$ asymptotically, i.e., $E_{\mathbf{y}}E_{\mathbf{y}_{rep}}[-2\ln p(\mathbf{y}_{rep}|\,\mathbf{y})] = E_y(DIC^{BP}) + o(1)$. Clearly, DIC^{BP} is as easy to compute as DIC. Since DIC is monitored in WinBUGS, no additional effort is needed for calculating DIC^{BP}.

5.3 Integrated DIC for latent variable models

Unfortunately, not all models are regular. A well-known nonregular model in economics is a class of models with incidental parameters which leads to the incidental parameter problem. In this class of models, the information about the incidental parameters stops accumulating after a finite number of observations have been taken; see Neyman and Scott (1948) and Lancaster (2000) for details about the incidental parameter problem.

As shown in Gelman et al. (2013), the incidental parameter problem can lead that the ML estimator is inconsistent and Bayesian large sample theory becomes invalid. When this is the case, the asymptotic justification of DIC does not hold because of the failure of these standard asymptotic theory.

In general, the latent variable model given in (1) does not have incidental parameters and hence the incidental parameter problem is not applicable. As explained earlier, for many latent variable models, the likelihood function is very difficult to be accurately approximate, rendering ML difficult to implement. To facilitate the posterior analysis, the data-augmentation strategy of Tanner and Wong (1987) is often used to augment the parameter space to $(\boldsymbol{\theta}, \mathbf{z})$, changing the likelihood function to $p(\mathbf{y}|\,\boldsymbol{\theta}, \mathbf{z})$ which typically has a closed-form expression. Denote the sample mean of $\mathbf{z}$, $\boldsymbol{\theta}$ by $\overline{\mathbf{z}}$, $\overline{\boldsymbol{\theta}}$, obtained from the MCMC output. Applying DIC developed earlier to the data-augmented MCMC output leads to

$$\mathrm{DIC}^{DA} = D\left(\overline{\mathbf{z}}, \overline{\boldsymbol{\theta}}\right) + 2P_D^{DA}, \tag{21}$$

$$\begin{aligned} P_D^{DA} &= \overline{D(\mathbf{z}, \boldsymbol{\theta})} - D\left(\overline{\mathbf{z}}, \overline{\boldsymbol{\theta}}\right) \\ &= -2\int\left[\ln p(\mathbf{y}|\,\mathbf{z}, \boldsymbol{\theta}) - \ln p\left(\mathbf{y}|\,\overline{\mathbf{z}}, \overline{\boldsymbol{\theta}}\right)\right] p(\mathbf{z}, \boldsymbol{\theta}|\,\mathbf{y})\mathrm{d}\mathbf{z}\mathrm{d}\boldsymbol{\theta}, \end{aligned} \tag{22}$$

where $D(\mathbf{z}, \boldsymbol{\theta}) = -2\ln p(\mathbf{y}|\,\mathbf{z}, \theta)$ which is typically available in closed-form. This way of calculating DIC is monitored and implemented in Win-BUGS,

following the suggestion of Spiegelhalter et al. (2002). Clearly the use of data augmentation not only facilitates MCMC sampling, but also makes DIC easier to calculate from the MCMC output.

Unfortunately, the data augmentation technique introduces incidental parameters to the model which lead to the incidental parameter problem. This is because, as discussed before, in many latent variable models, the latent variable $\mathbf{z}$ is often dependent on the sample size and its dimension is the same as or larger than the number of the sample size. As a result, the model becomes non-regular after the parameter space is expanded to $(\boldsymbol{\theta}, \mathbf{z})$. In particular, the ML estimator of $\mathbf{z}$ is typically inconsistent and the Bayesian large sample theory is invalid for $\mathbf{z}$. Although data augmentation makes DIC easy to calculate, it invalidates the asymptotic justification of DIC. DIC based on the data augmentation technique, as calculated in (21) and (22), is no longer asymptotically unbiased estimator of $E_{\mathbf{y}}E_{\mathbf{y}_{rep}}\left[-2\ln p\left(\mathbf{y}_{rep}|\overline{\boldsymbol{\theta}}(\mathbf{y})\right)\right]$. As a result, for the latent variable model, DIC, as how it is currently monitored and implemented in Win-BUGS, should not be used.

To address this problem, Li et al. (2017b) introduced an integrated DIC (IDIC) which integrates the latent variable out of the deviance and the penalty term. IDIC is given by

$$\text{IDIC} = D\left(\overline{\boldsymbol{\theta}}\right) + 2P_D^I, \tag{23}$$

where

$$P_D^I = \mathbf{tr}\left\{\mathbf{I}\left(\overline{\boldsymbol{\theta}}\right)V\left(\overline{\boldsymbol{\theta}}\right)\right\}, \tag{24}$$

and

$$\mathbf{I}(\boldsymbol{\theta}) = -\frac{\partial^2 \ln p(\mathbf{y}|\boldsymbol{\theta})}{\partial\boldsymbol{\theta}\partial\boldsymbol{\theta}'}, V\left(\overline{\boldsymbol{\theta}}\right) = E\left[\left(\boldsymbol{\theta}-\overline{\boldsymbol{\theta}}\right)\left(\boldsymbol{\theta}-\overline{\boldsymbol{\theta}}\right)'|\,\mathbf{y}\right].$$

Li et al. (2017b) showed that under regularity conditions, IDIC is an asymptotically unbiased estimator of $E_{\mathbf{y}}E_{\mathbf{y}_{rep}}\left[-2\ln p\left(\mathbf{y}_{rep}|\overline{\boldsymbol{\theta}}(\mathbf{y})\right)\right]$.

Similarly, if the loss function the Bayesian predictive distribution, one may obtain an alternative information criterion, which is $IDIC^{BP}$ by Li et al. (2017b) and is defined as

$$\text{IDIC}^{BP} = D\left(\overline{\boldsymbol{\theta}}\right) + (1 + \ln 2)P_D^I, \tag{25}$$

As shown in Li et al. (2017a), $E_{\mathbf{y}}E_{\mathbf{y}_{rep}}(-2\ln p(\mathbf{y}_{rep}|\mathbf{y})) = E_{\mathbf{y}}[\text{IDIC}^{BP}] + o(1)$.

5.4 Computing IDIC for latent variable models

For the latent variable model, $\ln p(\mathbf{y}|\boldsymbol{\theta})$ generally does not have an analytical expression. As a result, computing $\ln p\left(\mathbf{y}|\overline{\boldsymbol{\theta}}\right)$ and P_D^I is not trivial, in sharp

contrast to the quantities in (21) and (22). Li et al. (2017b) introduced a very general approach to computing IDIC.

Let

$$p(\mathbf{y}, \mathbf{z}|\overline{\boldsymbol{\theta}}, b) = p(\mathbf{y}|\mathbf{z}, \overline{\boldsymbol{\theta}})^b p(\mathbf{z}|\overline{\boldsymbol{\theta}})$$

$$p(\mathbf{y}|\overline{\boldsymbol{\theta}}, b) = \int p(\mathbf{y}, \mathbf{z}|\overline{\boldsymbol{\theta}}, b)\mathrm{d}\mathbf{z} = \int p(\mathbf{y}|\mathbf{z}, \overline{\boldsymbol{\theta}})^b p(\mathbf{z}|\overline{\boldsymbol{\theta}})\mathrm{d}\mathbf{z},$$

$$p(\mathbf{z}|\mathbf{y}, \overline{\boldsymbol{\theta}}, b) = \frac{p(\mathbf{y}, \mathbf{z}|\overline{\boldsymbol{\theta}}, b)}{p(\mathbf{y}|\overline{\boldsymbol{\theta}}, b)} = \frac{p(\mathbf{y}|\mathbf{z}, \overline{\boldsymbol{\theta}})^b p(\mathbf{z}|\overline{\boldsymbol{\theta}})}{p(\mathbf{y}|\overline{\boldsymbol{\theta}}, b)},$$

so that

$$p(\mathbf{y}|\overline{\boldsymbol{\theta}}, 1) = \int p(\mathbf{y}|\mathbf{z}, \overline{\boldsymbol{\theta}}) p(\mathbf{z}|\overline{\boldsymbol{\theta}})\mathrm{d}\mathbf{z} = \int p(\mathbf{y}, \mathbf{z}|\overline{\boldsymbol{\theta}})\mathrm{d}\mathbf{z} = p(\mathbf{y}|\overline{\boldsymbol{\theta}}),$$

$$p(\mathbf{y}|\overline{\boldsymbol{\theta}}, 0) = \int p(\mathbf{y}|\mathbf{z}, \overline{\boldsymbol{\theta}})^0 p(\mathbf{z}|\overline{\boldsymbol{\theta}})\mathrm{d}\mathbf{z} = \int p(\mathbf{z}|\overline{\boldsymbol{\theta}})\mathrm{d}\mathbf{z} = 1$$

$$p(\mathbf{z}|\mathbf{y}, \overline{\boldsymbol{\theta}}, 1) = \frac{p(\mathbf{z}, \mathbf{y}|\overline{\boldsymbol{\theta}}, 1)}{p(\mathbf{y}|\overline{\boldsymbol{\theta}}, 1)} = \frac{p(\mathbf{y}|\mathbf{z}, \overline{\boldsymbol{\theta}}) p(\mathbf{z}|\overline{\boldsymbol{\theta}})}{p(\mathbf{y}|\overline{\boldsymbol{\theta}}, 1)} = \frac{p(\mathbf{y}|\mathbf{z}, \overline{\boldsymbol{\theta}}) p(\mathbf{z}|\overline{\boldsymbol{\theta}})}{p(\mathbf{y}|\overline{\boldsymbol{\theta}})} = p(\mathbf{z}|\mathbf{y}, \overline{\boldsymbol{\theta}}),$$

$$p(\mathbf{z}|\mathbf{y}, \overline{\boldsymbol{\theta}}, 0) = \frac{p(\mathbf{z}, \mathbf{y}|\overline{\boldsymbol{\theta}}, 0)}{p(\mathbf{y}|\overline{\boldsymbol{\theta}}, 0)} = \frac{p(\mathbf{y}|\mathbf{z}, \overline{\boldsymbol{\theta}})^0 p(\mathbf{z}|\overline{\boldsymbol{\theta}})}{p(\mathbf{y}|\overline{\boldsymbol{\theta}}, 0)} = \frac{p(\mathbf{z}|\overline{\boldsymbol{\theta}})}{1} = p(\mathbf{z}|\overline{\boldsymbol{\theta}}).$$

Using the path sampling technique of Gelman and Meng (1998), Li et al. showed that

$$\begin{aligned} \ln p(\mathbf{y}|\overline{\boldsymbol{\theta}}) - \ln 1 &= \ln \frac{f(1)}{f(0)} = \int_0^1 \frac{\partial \ln f(b)}{\partial b}\mathrm{d}b \\ &= \int_0^1 E_{\mathbf{z}|\mathbf{y}, \overline{\boldsymbol{\theta}}, b}\left[\ln p(\mathbf{y}|\mathbf{z}, \overline{\boldsymbol{\theta}})\right]\mathrm{d}b := \int_0^1 u(b)\mathrm{d}b, \end{aligned} \tag{26}$$

where $f(b) = p(\mathbf{y}|\overline{\boldsymbol{\theta}}, b)$ such that $f(1) = p(\mathbf{y}|\overline{\boldsymbol{\theta}})$ and $f(0) = 1$.

In many cases, $\int_0^1 u(b)\mathrm{d}b$ in (26) does not have an analytical solution. Following Gelman and Meng (1998), we can numerically approximate it using the trapezoidal rule. In particular, we can choose a set of fixed grids $\left\{b_{(s)} = \frac{s}{S}\right\}_{s=0}^{S}$ such that $b_{(0)} = 0 < b_{(1)} < b_{(2)} < \ldots < b_{(S)} = 1$, and then approximate the integral by

$$\ln p(\mathbf{y}|\overline{\boldsymbol{\theta}}) \approx \frac{1}{S}\left(\frac{u(0)}{2} + \sum_{s=1}^{s-1} u(b_s) + \frac{u(1)}{2}\right).$$

Since $\ln p(\mathbf{y}|\mathbf{z}, \overline{\boldsymbol{\theta}})$ often has an analytical expression, $\ln p(\mathbf{y}|\overline{\boldsymbol{\theta}})$ can be conveniently obtained using the above formula.

To compute P_D^I, it mainly needs to evaluate the second derivative of ln $p(\mathbf{y}|\boldsymbol{\theta})$. Again, the well-known Louis formula suggests that

$$\begin{aligned}\frac{\partial^2 \ln p(\mathbf{y}|\boldsymbol{\theta})}{\partial\boldsymbol{\theta}\partial\boldsymbol{\theta}'} &= E_{\mathbf{z}|\mathbf{y},\boldsymbol{\theta}}\left\{\frac{\partial^2 \ln(\mathbf{y},\mathbf{z}|\boldsymbol{\theta})}{\partial\boldsymbol{\theta}\partial\boldsymbol{\theta}'}\right\} + Var_{\mathbf{z}|\mathbf{y},\boldsymbol{\theta}}\{S(\mathbf{y},\mathbf{z}|\boldsymbol{\theta})\} \\ &= E_{\mathbf{z}|\mathbf{y},\boldsymbol{\theta}}\left\{\frac{\partial^2 \ln(\mathbf{y},\mathbf{z}|\boldsymbol{\theta})}{\partial\boldsymbol{\theta}\partial\boldsymbol{\theta}'} + S(\mathbf{y},\mathbf{z}|\boldsymbol{\theta})S(\mathbf{y},\mathbf{z}|\boldsymbol{\theta})'\right\} \\ &\quad - E_{\mathbf{z}|\mathbf{y},\boldsymbol{\theta}}\{S(\mathbf{y},\mathbf{z}|\boldsymbol{\theta})\}E_{\mathbf{z}|\mathbf{y},\boldsymbol{\theta}}\{S(\mathbf{y},\mathbf{z}|\boldsymbol{\theta})\}'.\end{aligned}$$

Hence, we can use the following formula to calculate the second derivative of the observed-data likelihood function,

$$\begin{aligned}&E_{\mathbf{z}|\mathbf{y},\boldsymbol{\theta}}\left\{\frac{\partial^2 \ln(\mathbf{y},\mathbf{z}|\boldsymbol{\theta})}{\partial\boldsymbol{\theta}\partial\boldsymbol{\theta}'} + S(\mathbf{y},\mathbf{z}|\boldsymbol{\theta})S(\mathbf{y},\mathbf{z}|\boldsymbol{\theta})'\right\} \\ &\approx \frac{1}{J}\sum_{j=1}^{J}\left\{\frac{\partial^2 \ln(\mathbf{y},\mathbf{z}^{(j)}|\boldsymbol{\theta})}{\partial\boldsymbol{\theta}\partial\boldsymbol{\theta}'} + S\left(\mathbf{y},\mathbf{z}^{(j)}|\boldsymbol{\theta}\right)S\left(\mathbf{y},\mathbf{z}^{(m)}|\boldsymbol{\theta}\right)'\right\}, \\ &E_{\mathbf{z}|\mathbf{y},\boldsymbol{\theta}}\{S(\mathbf{y},\mathbf{z}|\boldsymbol{\theta})\} \approx \frac{1}{J}\sum_{j=1}^{J} S\left(\mathbf{y},\mathbf{z}^{(j)}|\boldsymbol{\theta}\right),\end{aligned}$$

where $\{\mathbf{z}^{(j)}, j=1, 2, \ldots, J\}$ are the MCMC samples.

The main difference between DIC, given in (17) and (18), and IDIC, given in (23) and (24), lies in P_D and P_D^I. To compute P_D, we need to evaluate $E_{\boldsymbol{\theta}|\mathbf{y}}[\ln p(\mathbf{y}|\boldsymbol{\theta}))] \approx \frac{1}{J}\sum_{j=1}^{J} \ln p\left(\mathbf{y}|\boldsymbol{\theta}^{(j)}\right)$. For the latent variable models, without knowing the analytical form of ln $p(\mathbf{y}|\boldsymbol{\theta})$, computing $\frac{1}{J}\sum_{j=1}^{J} \ln p\left(\mathbf{y}|\boldsymbol{\theta}^{(j)}\right)$ is very expensive since one has to evaluate ln p $(\mathbf{y}|\boldsymbol{\theta}^{(j)})$ for J times with J being large. To compute P_D^I in IDIC, one only needs to compute the second derivative once.

Two well-known classes of latent variable models are the linear Gaussian state space model and the nonlinear non-Gaussian state space model. For these two classes of models, some recursive algorithms, such as the Kalman filter and particle filter algorithms, can be used to facilitate the computation of IDIC. There are existing R packages to implement the Kalman filter and particle filter algorithms; see Tusell (2011). Hence, the proposed method here can be combined with these R packages.

6 Empirical illustrations

In this section, we illustrate the proposed test statistics and model selection criteria using three popular examples in economics and finance. The first example contains asset pricing models with a t error distributions. The likelihood functions of these models not only have the analytical form, but also can

be rewritten as in the latent variable form. These two alternative ways of rewritting the models allow us to check the problem in DIC with data augmentation. The second example contains stochastic volatility models, where the volatility is latent. In the second example, the analytical expression of the observed data likelihood does not exist.

6.1 Statistical inference in asset pricing models

Asset pricing models are one of important models in modern finance. There models generally assume that the return distribution is normal. Unfortunately, there has been overwhelming empirical evidence against normality for asset returns, which have led researchers to investigate asset pricing models with heavy-tailed distributions. Zhou (1993) and Kan and Zhou (2017) suggested to use the multivariate t distribution to replace the multivariate normal distribution. Moreover, on the basis of the efficient market theory, the asset excess premium should not be statistically different from zero. At last, the multivariate t distribution can be rewritten as scale-mixture framework to become a latent variable model. Hence, we consider the following six asset pricing models:

$$
\begin{aligned}
&Model\,1: R_t = \boldsymbol{\beta}'\boldsymbol{F}_t + \epsilon_t, \epsilon_t \sim N[\mathbf{0}, \boldsymbol{\Sigma}];\\
&Model\,2: R_t = \alpha + \boldsymbol{\beta}'\boldsymbol{F}_t + \epsilon_t, \epsilon_t \sim N[\mathbf{0}, \boldsymbol{\Sigma}];\\
&Model\,3: R_t = \boldsymbol{\beta}'\boldsymbol{F}_t + \epsilon_t, \epsilon_t \sim t[\mathbf{0}, \boldsymbol{\Sigma}, \nu];\\
&Model\,4: R_t = \boldsymbol{\beta}'\boldsymbol{F}_t + \epsilon_t, \epsilon_t \sim N(\mathbf{0}, \boldsymbol{\Sigma}/\omega_t), \omega_t \sim \Gamma\left(\frac{\nu}{2}, \frac{\nu}{2}\right);\\
&Model\,5: R_t = \boldsymbol{\alpha} + \boldsymbol{\beta}'\boldsymbol{F}_t + \epsilon_t, \epsilon_t \sim t[\mathbf{0}, \boldsymbol{\Sigma}, \nu];\\
&Model\,6: R_t = \boldsymbol{\alpha} + \boldsymbol{\beta}'\boldsymbol{F}_t + \epsilon_t, \epsilon_t \sim N(\mathbf{0}, \boldsymbol{\Sigma}/\omega_t), \omega_t \sim \Gamma\left(\frac{\nu}{2}, \frac{\nu}{2}\right),
\end{aligned}
$$

where R_t is the excess return of portfolio at period t with $N \times 1$ dimension, $\boldsymbol{F}_t$ a $K \times 1$ vector of factor portfolio excess returns, $\boldsymbol{\alpha}$ a $N \times 1$ vector of intercepts, $\boldsymbol{\beta}$ a $N \times K$ vector of scaled covariances, ϵ_t the random error, $t = 1$, 2, …, n. For convenience, we restrict $\boldsymbol{\Sigma}$ to be a diagonal matrix and ν to be a known constant as $\nu = 3$. It is noted that Model 4 is the scale-mixture distributional representation of Model 3, and Model 5 is the scale mixture distributional representation of Model 6.

Monthly returns of 25 portfolios, constructed at the end of each June, are the intersections of 5 portfolios formed on size (market equity, ME) and 5 portfolios formed on the ratio of book equity to market equity (BE/ME). The Fama/French's three factors, market excess return, SMB (Small Minus Big), HML (High Minus Low) are used as the explanatory factors (Fama and French, 1993). The sample period is from July 1926 to July 2011, so that $N = 25$, $n = 1021$. The data are freely available from the data library of Kenneth French.[a]

[a] http:/mba.tuck.dartmouth.edu/pages/faculty/ken.french/data_library.html.

Making inference for the asset pricing models has attracted a considerable amount of attentions in the empirical asset pricing literature. Avramov and Zhou (2010) provided an excellent review of the literature on Bayesian portfolio analysis. As to Bayesian inference, we need specify the prior distributions for parameters. Here, to represent the prior ignorance, we assign some vague conjugate prior distributions, that is,

$$\alpha_i \sim N[0, 100], \beta_{ij} \sim N[0, 100], \phi_{ii}^{-1} \sim \Gamma[0.01, 0.01].$$

Based on the R language, we use R2WinBUGS to get the MCMC outputs, and draw 100,000 random observations from the posterior distributions in each model where the first 40,000 is used as the burn-in sample, and the next 60,000 iterations is collected with every 3th observation as effective observations. Hence, these are 20,000 effective observations.

6.1.1 Hypothesis testing for asset pricing models

In asset pricing theory, the efficient market theory suggests that the excess premium $\boldsymbol{\alpha}$ should be zero. Hence, we can write this problem as a hypothesis to be tested as:

$$H_0 : \boldsymbol{\alpha} = 0 \times \mathbf{1}_N, H_1 : \boldsymbol{\alpha} \neq 0 \times \mathbf{1}_N,$$

where $\mathbf{1}_N$ is an N-dimensional vector with unit elements. Model 6 is the most general model which can nest other models, hence, based on this model, we discuss the asset pricing testing problem above.

In Section 4, among of those approaches, we have shown that the threshold values by Bernardo and Rueda (2002) and Li and Yu (2012) are difficult to calibrate. Hence, here, we only consider the statistics respectively developed by (Li et al., 2014, 2015, 2019). Based on 20,000 MCMC samples, we calculate the three test statistics, $\mathbf{T}_{LZY}(\mathbf{y}, \boldsymbol{\vartheta}_0)$, $\mathbf{T}_{LLY}(\mathbf{y}, \boldsymbol{\vartheta}_0)$ and $\mathbf{T}_{LLYZ}(\mathbf{y}, \boldsymbol{\vartheta}_0)$. We report the results in Table 2.

Obviously, from these results, according to the critical values from $\chi^2(25)$, under 5% significant level, all the test statistics reject the null hypothesis. Hence, we can conclude that the mean–variance efficiency does not held in practice. As to these test statistics, more details, one can refer to Li et al. (2014, 2015, 2019). At last, according to the Savage-Dickey Density Ratio approach by Verdinelli and Wasserman (1995), it can be shown that.

$\hat{BF} = 1.069$ which provide mild evidence to support H_0 which is contractive to the results from the hypothesis testing statistics. This reason lies that in this section, we use the vague prior to do the hypothesis testing so that BFs suffer from the Jeffreys-Lindley's paradox. It should be very suggested to use BFs for doing hypothesis testing when the prior information is not available. More details about the Jeffreys-Lindley's paradox, see the discussion by Li et al. (2015).

TABLE 2 Asset pricing testing in M_6

Hypothesis	$\alpha = 0$
$\mathbf{T}_{LZY}(\mathbf{y}, \vartheta_0)$	140.5191
$\mathbf{T}_{LLY}(\mathbf{y}, \vartheta_0)$	153.5680
$\mathbf{T}_{LLZY}(\mathbf{y}, \vartheta_0)$	184.4315

6.1.2 Specification testing for asset pricing models

In this subsection, we take the standard Fama–French three-factor asset pricing model (Fama and French, 1993) that is, model 2 as an example for illustrating the proposed approach. The standard asset pricing model is given by

$$Model\,2: R_t = \alpha + \beta_1 R_{mt} + \beta_2 SMB_t + \beta_3 HML_t + \epsilon_t, \epsilon_t \sim N[\mathbf{0}, \mathbf{\Sigma}]$$

where R_m is the excess market return, *SMB* stands for "Small [market capitalization] Minus Big" and *H ML* for "High [book-to-market ratio] Minus Low"; they measure the historic excess returns of small caps over big caps and of value stocks over growth stocks.

Here, for checking the model misspecification, the expanded model can be specified as

$$Model\,2E: R_t = \boldsymbol{\alpha} + \boldsymbol{\beta}_1 R_{mt} + \boldsymbol{\beta}_{1E} R_{mt}^2 + \boldsymbol{\beta}_2 SMB_t + \boldsymbol{\beta}_3 HML_t + \epsilon_t, \epsilon_t \sim N[\mathbf{0}, \mathbf{\Sigma}]$$

Hence, according to Section 4, we can write this model misspecification problem as a hypothesis to be tested as:

$$H_0: \beta_{1E} = 0,\ H_1: \beta_{1E} \neq 0$$

Following Section 4, the proposed test statistic can be given by

$$\mathrm{BMT} = \mathbf{tr}\left\{C_E[\mathbf{y}, ((\boldsymbol{\alpha}, \boldsymbol{\beta}_1, \boldsymbol{\beta}_2, \boldsymbol{\beta}_3, \mathbf{\Sigma}), \boldsymbol{\beta}_{1E} = 0)] V_E\left(\overline{\boldsymbol{\theta}}_L\right)\right\} + \sqrt{1021}(\mathrm{BIMT}/125 - 1)^2$$

Hence, based on 20,000 effective observation drawn from the posterior distribution, we can compute the corresponding statistics which are reported in Table 3. It is noted that if the model is correctly specified, *BMT* converges to $\chi^2(25)$ distribution. Given this χ^2 distribution, under 0.05 significant level, the critical value is 37.65. Hence, according to the table, we can conclude that *BMT* strongly reject the null hypothesis which means that the asset price model is misspecified (Table 3).

6.1.3 Model comparison for asset pricing models

We make a model comparison of these asset pricing models. Based on 20,000 effective observations, we calculate DICs, and BFs. Table 4 reports P_D, P_D^{DA},

TABLE 3 Results of specification test for model 2

Item	Value
BIMT	610
$\mathbf{tr}\{C_E[\mathrm{y}, ((\alpha, \beta_1, \beta_2, \beta_3, \Sigma), \beta_{1E}=0)]V_E(\bar{\boldsymbol{\theta}}_{\mathrm{L}})$	444
$\sqrt{1021}(\mathrm{BIMT}/125-1)^2$	481
BMT	925

TABLE 4 Model selection results for Fama–French three factor models

Model	M_1	M_2	M_3	M_4	M_5	M_6
# of Parameters	100	125	100	100	125	125
P_D	100	125	100	100	125	125
DIC	−119,842	−119,880	−133,088	−133,088	−133,202	−133,202
DIC^{BP}	−119,872	−119,918	−133,118	−133,118	−133,240	−133,240
P_D^{DA}	—	—	—	1021	—	1046
DIC^{DA}	—	—	—	−134,777	—	−134,897
P_D^I	100	125	100	100	126	126
IDIC	−119,842	−119,880	−133,087	−133,087	−133,201	−133,201
IDIC^{BP}	−119,873	−119,918	−133,118	−133,118	−133,240	−133,240

P_D^I, DIC, DIC^{BP}, DIC^{DA}, IDIC, and IDIC^{BP} for all six models. Note that only M_4 and M_6 has the latent variable so that P_D^{DA} and DIC^{DA} are only reported for these two models. Furthermore, M_3 and M_4 are the same model with different distribution expression, M_5 and M_6 are the same model with different distribution expression. Hence, as to the same model with different distribution expression, P_D, P_D^I, DIC, DIC^{BP}, IDIC, and IDIC^{BP} are equal for the same model.

From Table 4, we can get some interesting finding. First, as expected, DIC^{DA} in Model 3 is quite different from that in Model 4 although these two models are the same, but only have different distribution expression. The main reason is that in Model 4, the scale-mixture specification is used and, hence, a sequence of latent variables, $\{\omega_t\}$ are treated as parameters. For the same reason, DIC^{DA} in Model 5 is quite different from that in Model 6. As argued earlier, this conceptual

difficulty is due to lack of the theoretical foundation. Second, DIC, DIC^{BP}, IDIC, and IDIC^{BP} do not suffer from the same difficulty as DIC^{DA}. For Model 3 (and Model 5), they are identical to those for Model 4 (and Model 6). Third, the theoretical results show that P_D and P_D^I should be close to the actual number of the parameters, P, if the posterior distribution is well approximated by the normal distribution and the use of uninformative priors is used. The results can be confirmed from this table. Most importantly, we see that P_D is almost identical to P_D^I in all models. Not surprisingly, DIC and IDIC are almost the same in all models and DIC^{BP} and IDIC^{BP} are almost the same. This confirm the theoretical result that P_D and P_D^I can be well approximated. In addition, all DICs provide the evidence to support M_6 is the best model for prediction among these six models.

In addition, as to P_D and P_D^I, we need point out that in terms of the computational cost, for Models 3 and 5, P_D^I can require less efforts than P_D. The reason is that P_D involves $\int \ln p(\mathbf{y}|\boldsymbol{\theta})p(\boldsymbol{\theta}|\mathbf{y})\mathrm{d}\boldsymbol{\theta}$, which is approximated by $\frac{1}{J}\sum_{j=1}^{J} \ln p\left(\mathbf{y}|\boldsymbol{\theta}^{(j)}\right)$. This quantity is much more expensive to compute because it requires numerical evaluation of $\ln p(\mathbf{y}|\boldsymbol{\theta}^{(j)})$ for J times. For example, here, based on the 20,000 posterior random observations, one has to evaluate $\ln p(\mathbf{y}|\boldsymbol{\theta}^{(j)})$ 20,000 times. Fortunately, as to asset pricing models, $\ln p(\mathbf{y}|\boldsymbol{\theta}^{(j)})$ has closed-form. However, as to other models such that $\ln p(\mathbf{y}|\boldsymbol{\theta})$ does not have analytical form, obviously, IDIC is more advantageous than DIC.

At last, in order to check the reliability of the general computation approach by Section 5.4, we take model 6 as an example. Since the likelihood function $\ln p(\mathbf{y}|\boldsymbol{\theta})$ has analytical form, we can easily get that $D(\overline{\boldsymbol{\theta}}) = -133452$. Using the approximation approach in Section 5.4, we give the approximated value of $D(\overline{\boldsymbol{\theta}})$, that is, $\hat{D}(\overline{\boldsymbol{\theta}})$ under different grids and report the results in the Table 5. From this table, it can be observed that with the increasing grid S, the proposed approach can approximate $D(\overline{\boldsymbol{\theta}})$ very well.

TABLE 5 The approximated value of $D(\overline{\boldsymbol{\theta}})$ based on Section 5.4

Hypothesis	$\hat{D}(\overline{\boldsymbol{\theta}})$
$S=200$	−133,436
$S=400$	−133,437
$S=800$	−133,451
$S=900$	−133,452

6.2 Statistical inference in stochastic volatility models

Stochastic volatility (SV) models are one of the important models to model the time-varying volatility in financial econometrics. The basic SV model is composed of two equations, one is measurement equation, the other is state equation where the logarithmic volatility is the state variable which is often assumed to follow an AR(1) model. The basic form can be written as

$$y_t = \alpha + \exp(h_t/2)u_t, u_t \sim N(0,1),$$
$$h_t = \mu + \phi(h_{t-1} - \mu) + v_t, v_t \sim N(0, \tau^2),$$

where $t = 1, 2, \ldots, n$, y_t is the continuously compounded return, h_t the unobserved log-volatility, $h_0 = \mu$, u_t, and v_t are independent for all t. In this chapter, we denote this model by M_1.

An important and well documented empirical feature in many financial time series is the leverage effect (Black, 1976). Hence, following Yu (2005), a fundamental extension of the basic SV model is to incorporate the leverage effect. The leverage effect SV model can be defined as:

$$y_t = \alpha + \exp(h_t/2)u_t, u_t \sim N(0,1)$$
$$h_{t+1} = \mu + \phi(h_t - \mu) + v_{t+1}, v_{t+1} \sim N(0, \tau^2)$$

with

$$\begin{pmatrix} u_t \\ v_{t+1} \end{pmatrix} \overset{i.i.d.}{\sim} N\left\{ \begin{pmatrix} 0 \\ 0 \end{pmatrix}, \begin{pmatrix} 1 & \rho \\ \rho & 1 \end{pmatrix} \right\}$$

and $h_0 = \mu$. In this model, ρ captures the leverage effect if $\rho < 0$. In the empirical literature, there is a negative relationship between the expected future volatility and the current return. We denote this model as M_2.

To carry out Bayesian analysis, following Meyer and Yu (2000), the prior distributions are specified as follows:

$$\alpha \sim N(0, 100), \mu \sim N(0, 100),$$
$$\phi \sim Beta(1,1), 1/\tau^2 \sim \Gamma(0.001), \rho \sim \text{Unit}(-1, 1)$$

This type prior can be regarded as a noninformative prior to represent the prior ignorance.

The dataset consists of 945 daily mean-corrected returns on Pound/−Dollar exchange rates, covering the period between 01/10/81 and 28/06/85. Here, using R language, we use R2WinBUGS to run MCMC to get the outputs. After a burn-in period of 10,000 iterations, we save every 20th value for the next 100,000 iterations to get 5000 effective draws. The same dataset was used in Kim et al. (1998) and Meyer and Yu (2000). The posterior mean and standard error of parameters in the two competing model are reported in Table 6. Note that the in M_2, the posterior mean of ρ is very close to zero, relative to its posterior standard error.

TABLE 6 Posterior mean and standard error of parameters in M_1 and M_2

Parameter	M_1		M_2	
	Mean	*SE*	*Mean*	*SE*
μ	−0.6733	0.3282	−0.6485	0.3377
φ	0.9733	0.0127	0.9802	0.0138
ρ	—	—	−0.0575	0.1570
τ	0.1698	0.0378	0.1661	0.0391

6.2.1 Hypothesis testing for stochastic volatility models

In this chapter, the hypothesis that we are concerned can be expressed as:

$$H_0 : \rho = 0,\ H_1 : \rho \neq 0$$

Here, ρ is the interest parameter, the nuisance parameter is denoted by $\boldsymbol{\psi} = (\mu, \varphi, \tau^{-2})$, $\boldsymbol{\theta} = (\rho, \boldsymbol{\psi}) = \rho, (\mu, \varphi, \tau^2)$. Again, based on 20,000 effective observation, we calculate the three test statistic, that is, $\mathbf{T}_{LZY}(\mathbf{y}, \boldsymbol{\vartheta}_0)$, $\mathbf{T}_{LLY}(\mathbf{y}, \boldsymbol{\vartheta}_0)$, and $\mathbf{T}_{LLYZ}(\mathbf{y}, \boldsymbol{\vartheta}_0)$. We report all the results in Table 7.

From this table, according to the critical values calibrated from their asymptotic distribution, under 5% significant level, all three test statistics fail to reject the null hypothesis. The result is correspond with estimation result, that is, $\rho = -0.0575$. Furthermore, this provide enough evidence to support that leverage effect in this exchange data is not obvious.

6.2.2 Specification testing for SV models

The dataset used here contains the daily returns on AUD/USD exchange rates from January 2005 to December 2012. Following a suggestion of a referee,

TABLE 7 Hypothesis hypothesis results for the leverage effect

Hypothesis	$\rho = 0$
$\mathbf{T}^*_{LZY}(\mathbf{y}, \boldsymbol{\vartheta}_0)$	−0.6870
$\mathbf{T}_{LLY}(\mathbf{y}, \boldsymbol{\vartheta}_0)$	0.1659
$\mathbf{T}_{LLZY}(\mathbf{y}, \boldsymbol{\vartheta}_0)$	1.7050

before we apply BMT to the SV model, we first test the *i.i.d.* normal model with constant mean and constant variance given by

$$y_t = \alpha + \varepsilon_t, \varepsilon_t \overset{i.i.d.}{\sim} N(0, \sigma^2) \tag{27}$$

An AR(1) model is used as the expanded model

$$y_t = \alpha + \beta y_{t-1} + \varepsilon_t, \varepsilon_t \overset{i.i.d.}{\sim} N(0, \sigma^2). \tag{28}$$

The Bayesian MCMC method is implemented to estimate the parameters with the following vague prior

$$\alpha \sim N(0, 100\sigma^2), \beta \sim N(0, 100\sigma^2), \sigma^{-2} \sim \Gamma(0.001, 0.001).$$

For the above two models, we draw 20,000 MCMC samples from the posterior distribution and compute BMT.

The critical value of $\chi^2(1)$ is 6.63 at the 1% significance level. BMT is 251.52, rejecting the *i.i.d.* normal model. This conclusion is not surprising as the volatility of stock returns is stochastic. However, J_1 is 0.2858 (i.e., $J_0 = 251.23$) which is less than the critical value of $\chi^2(1)$. Using J_1 alone only suggests that we cannot reject $\beta = 0$ in Model (28). This conclusion is also not surprising as the weekly returns have very weak serial correlations.

Next, we change the null model to the following basic SV model,

$$\begin{aligned} y_t &= \alpha + \exp(h_t/2)u_t, u_t \overset{i.i.d.}{\sim} N(0, 1), \\ h_t &= \mu + \phi(h_{t-1} - \mu) + \tau v_t, v_t \overset{i.i.d.}{\sim} N(0, 1). \end{aligned} \tag{29}$$

The expanded model is as follows:

$$\begin{aligned} y_t &= \alpha + \beta_1 y_{t-1} + \exp(h_t/2)u_t, u_t \overset{i.i.d.}{\sim} N(0, 1). \\ h_t &= \mu + \phi(h_{t-1} - \mu) + \tau v_t, v_t \overset{i.i.d.}{\sim} N(0, 1). \end{aligned} \tag{30}$$

The following vague priors are used

$$\begin{aligned} &\alpha \sim N(0, 100), \ \phi \sim Beta(1, 1), \\ &\tau^{-2} \sim \Gamma(0.001, 0.001), \ \beta_1 \sim N(0.5, 100). \end{aligned}$$

To obtain BMT, we draw 110,000 MCMC samples from the posterior distribution and discard the first 10,000 as burning-in observations, and store the remaining samples as effective observations in both models. In this case, BMT = 0.4279 which is less than the critical value of $\chi^2(1)$, suggesting that the basic SV model is not misspecified.

6.2.3 Model comparison of SV models

Hence, we consider the model comparison of these two models. Since the models are of a nonlinear non-Gaussian form and both $p(\mathbf{y}|\boldsymbol{\theta})$ are not available in closed-form, the approach provided in Section 5 is implemented to compute

TABLE 8 Model selection results for M_1 and M_2

Model	M_1	M_2
P_D^{DA}	53.60	31.33
$D(\mathbf{z}, \overline{\boldsymbol{\theta}})$	1695.40	1693.36
DIC^{DA}	1802.52	1756.21
P_D^I	2.32	3.24
$D(\overline{\boldsymbol{\theta}})$	1837.81	1837.78
IDIC	1842.50	1844.30
IDIC^{BP}	1841.80	1843.30
BF_{21}	0.2174	

DICs, and the Savage Dickey density ratio (Verdinelli and Wasserman, 1995) is implemented to calculate BFs. Hence, DIC requires tedious computational efforts. Here, we only report the results of DIC^{DA}, IDIC, P_D^{DA}, P_D^I, and BFs in Table 8.

From this table, we can get the following findings. First, DIC^{DA} and IDIC suggest different rankings of the competing models where DIC^D suggests that M_2 is better that M_1, IDIC and IDIC^{BP} both suggest M_1. According to DIC^{DA}, it can be observed that M_1 and M_2 perform nearly the same judged by the model fit term, $D(\mathbf{z}, \overline{\boldsymbol{\theta}})$. However, M_2 reduces P_D^7 by 22.3 over M_1. This reduction of the model complexity is the reason why DIC^{DA} prefers M_2. This result is surprising as the posterior mean of the leverage effect is nearly zero as reported in Table 8 and not accord with the hypothesis testing results. Obviously, as to SV models, when the latent variable is regarded as parameters, the number of parameters exceeds the number of observations, say $n+3$ in M_1 and $n+4$ in M_1. Hence, an important season to lead the surprising results lie that DIC^{DA} is lack of rigorously theoretical foundation and should be cautious to be used in practice although its computation is simple.

Second, IDIC and IDIC^{BP} both suggest that M_1 is slightly better that M_2 although the difference is not large. In IDIC, P_D^I is 2.32 in M_1 and 3.24 in M_2. These values are very close to the actual numbers of parameters in the two models. It is noted that M_2 has one extra parameter so that this difference is reasonable. Moreover, M_1 and M_2 perform nearly the same judged by $D(\overline{\boldsymbol{\theta}})$. These findings give the reason why M_1 is slightly better that M_2. Third, BFs suggest that M_1 is the better model, consistent with the ranking of IDIC. This empirical example clearly demonstrates that IDIC is a more reliable model selection criterion that DIC^{DA}. In addition, although IDIC and IDIC^{BP} both

select the basic SV model, they imply that different predictive distribution should be used. From the theoretical analysis, as to predictive problem, the model selection results suggest that the basic SV model with Bayesian predictive distribution should be used because this decision can yield smallest risk asymptotically when M_1, M_2, plug-in predictive distribution and Bayesian predictive distribution are candidate use.

7 Concluding remarks

In this chapter, instead of making refinements for BFs, we overviews some alterative approaches developed in the recent literature for hypothesis testing and model selection methods. The approaches are established after the MCMC output is available. We show that these approaches not only have good theoretical properties, but also, do not require tedious additional computational efforts. Hence, with the advance of MCMC techniques and expanding computing facility, these approaches can be applied into a variety of complex models, especially latent variable models.

As to the hypothesis testing, we overviews several statistics for hypothesis testing which can be regarded as the MCMC version of the "trinity" of test statistics widely used in the frequentist domain, namely, LR test, LM test, and Wald test. Their asymptotic distributions are discussed based on a set of regular conditions. Furthermore, we overview the well-known DIC and its extensions. The asymptotic property of DICs are also discussed compared with AIC. At last, we illustrate the methods using econometric models with real data, some of which involve latent variables. The implementation is illustrated by R code with the MCMC output obtained by R2WinBUGS.

References

Avramov, D., Zhou, G., 2010. Bayesian portfolio analysis. Annu. Rev. Financ. Econ. 2, 25–47.

Berger, J.O., Perrichi, L.R., 1996. The intrinsic Bayes factor for model selection and prediction. J. Am. Stat. Assoc. 91, 109–122.

Bernardo, J.M., Rueda, R., 2002. Bayesian hypothesis testing: a reference approach. Int. Stat. Rev. 70, 351–372.

Bernardo, J.M., Smith, A.F.M., 2006. Bayesian Theory, second ed. John Wiley & Sons Canada, Limited, Chichester.

Black, F., 1976. Studies of stock market volatility changes. Proc. Am. Stat. Assoc. Bus. Econ. Stat. Sec. 177–181.

Breusch, T.S., Pagan, A.R., 1980. The Lagrange multiplier test and its applications to model specification in econometrics. Rev. Econ. Stud. 47, 239–253.

Chen, X., Christensen, T.M., O'Hara, K., Tamer, E., 2016. MCMC Confidence Sets for Identified Sets. Working Paper, Yale University.

Chernozhukov, V., Hong, H., 2003. An MCMC approach to classical estimation. J. Econ. 115 (2), 293–346.

Dempster, A.P., Laird, N.M., Rubin, D.B., 1977. Maximum likelihood from incomplete data via the EM algorithm. J. R. Stat. Soc. Ser. B 39 (1), 1–38.

Engle, R.F., 1984. Wald, likelihood ratio, and Lagrange multiplier tests in econometrics. Handb. Econ. 2, 775–826.

Fama, E.F., French, K.R., 1993. Common risk factors in the returns on stocks and bonds. J. Financ. Econ. 33, 3–56.

Fan, J., Liao, Y., Yao, J., 2015. Power enhancement in high-dimensional cross-sectional tests. Econometrica 83 (4), 1497–1541.

Gelman, A., Meng, X.-L., 1998. Simulating normalizing constants: From importance sampling to bridge sampling to path sampling. Stat. Sci. 13, 163–185.

Gelman, A., Carlin, J., Stern, H., Dunson, D., Vehtari, A., Rubin, D., 2013. Bayesian Data Analysis, third ed. Chapman and Hall/CRC.

Han, C., Carlin, B.P., 2001. Markov chain Monte Carlo methods for computing Bayes factor: a comparative review. J. Am. Stat. Assoc. 96, 1122–1132.

Jeffreys, H., 1961. Theory of Probability, third ed. Oxford University Press, Oxford.

Kan, R., Zhou, G., 2017. Modeling non-normality using multivariate t: implications for asset pricing. China Financ. Rev. Inter. 7, 2–32.

Kass, R.E., Raftery, A.E., 1995. Bayes factor. J. Am. Stat. Assoc. 90, 773–795.

Kim, S., Shephard, N., Chib, S., 1998. Stochastic volatility: Likelihood inferenceand comparison with ARCH models. Rev. Econ. Stud. 65, 361–393.

Lancaster, T., 2000. The incidental parameter problem since 1948. J. Econ. 95 (2), 391–413.

Li, Y., Yu, J., 2012. Bayesian hypothesis testing in latent variable models. J. Econ. 166, 237–246.

Li, Y., Zeng, T., Yu, J., 2014. A new approach to Bayesian hypothesis testing. J. Econ. 178, 602–612.

Li, Y., Liu, X.B., Yu, J., 2015. A Bayesian chi-squared test for hypothesis testing. J. Econ. 189, 54–69.

Li, Y., Yu, J., Zeng, T., 2017a. Deviation Information Criterion: Justification and Variation. Working paper. Singapore Management University.

Li, Y., Yu, J., Zeng, T., 2017b. Integrated Deviation Information Criterion for Latent Variable Models. Working paper. Singapore Management University.

Li, Y., Yu, J., Zeng, T., 2018. Specification tests based on MCMC output. J. Econometrics 207, 237–260.

Li, Y., Liu, X.B., Yu, J., Zeng, T., 2019. A posterior-based Wald-type statistic for hypothesis testing. Working paper. School of Economics, Singapore Management University.

Louis, T.A., 1982. Finding the observed information matrix when using the EM algorithm. J. R. Stat. Soc. Ser. B 44, 226–233.

O'Hagan, A., 1991. Discussion of posterior Bayes factors by M. Aitkin. J. R. Stat. Soc. Ser. B 53, 136.

O'Hagan, A., 1995. Fractional Bayes factors for model comparison, (with discussion). J. R. Stat. Soc. Ser. B 57, 99–138.

Martin, A.D., Quinn, K.M., 2005. MCMCpack 0.6-6. http://mcmcpack.wustl.edu/.

Meyer, R., Yu, J., 2000. BUGS for a Bayesian analysis of stochastic volatility models. Econ. J. 3, 198–215.

Neyman, J., Scott, E.L., 1948. Consistent estimates based on partially consistent observations. Econometrica 16, 1–32.

Poirier, D.J., 1995. Intermediate Statistics and Econometrics: A Comparative Approach. The MIT Press.

Presnell, B., Boos, D.D., 2004. The IOS test for model misspecification. J. Am. Stat. Assoc. 99 (465), 216–227.

Robert, C., 1993. A note on Jeffreys-Lindley paradox. Stat. Sin. 3, 601–608.

Robert, C., 2001. The Bayesian Choice: From Decision-Theoretic Foundations to Computational Implementation, second ed. Springer Texts in Statistics.
Spiegelhalter, D., Best, N.G., Carlin, B., van der Linde, A., 2002. Bayesian measures of model complexity and fit. J. R. Stat. Soc. Ser. B 64, 583–639.
Spiegelhalter, D., Thomas, A., Best, N.G., Lunn, D., 2003. WinBUGS User Manual. Version 1.4. MRC Biostatistics Unit, Cambridge, England.
Sturtz, S., Ligges, U., Gelman, A., 2005. R2WinBUGS: a package for running WinBUGS from R. J. Stat. Softw. 39 (3), 1–16.
Tanner, T.A., Wong, W.H., 1987. The calculation of posterior distributions by data augmentation. J. Am. Stat. Assoc. 82, 528–540.
Tusell, F., 2011. Kalman filtering in R. J. Stat. Softw. 12 (2), 1–27.
Verdinelli, I., Wasserman, L., 1995. Computing Bayes factors using a generalization of the Savage-Dickey density. J. Am. Stat. Assoc. 90 (430), 614–618.
White, H., 1982. Maximum likelihood estimation of misspecified models. Econometrica 50, 1–25.
Yu, J., 2005. On leverage in a stochastic volatility models. J. Econ. 127, 165–178.
Zhou, G., 1993. Asset-pricing tests under alternative distributions. J. Financ. 48, 1927–1942.

Further reading

Akaike, H., 1973. Information theory and an extension of the maximum likelihood principle. In: Second International Symposium on Information Theory, vol. 1. Springer Verlag, pp. 267–281.
Chen, C.F., 1985. On asymptotic normality of limiting density function with Bayesian implications. J. R. Stat. Soc. Ser. B 47, 540–546.
Chib, S., 1995. Marginal likelihood from the Gibbs output. J. Am. Stat. Assoc. 90, 1313–1321.
Chib, S., Jeliazkov, I., 2001. Marginal likelihood from the Metropolis-Hastings output. J. Am. Stat. Assoc. 96, 270–281.
Creal, D., 2012. A survey of sequential Monte Carlo methods for economics and finance. Econ. Rev. 31, 245–296.
Doucet, A., Johansen, A.M., 2011. A tutorial on particle filtering and smoothing: fifteen years later. In: Crisan, D., Boris, R. (Eds.), The Oxford Handbook of Nonlinear Filtering. Oxford University Press.
Doucet, A., Shephard, N., 2012. Robust Inference on Parameters via Particle Filters and Sandwich Covariance Matrices. Working Paper, Harvard University.
Geweke, J., 2007. Bayesian model comparison and validation. Am. Econ. Rev. 97, 60–64.
Kadane, J.B., Lazar, N.A., 2004. Methods and criteria for model selection. J. Am. Stat. Assoc. 99 (465), 279–290.
Shephard, N., 2005. Stochastic Volatility: Selective Readings. Oxford University Press.
Spiegelhalter, D., Best, N., Carlin, B., van der Linde, A., 2014. The deviance information criterion: 12 years on. J. R. Stat. Soc. Ser. B 76, 485–493.
Vehtari, A., Ojanen, J., 2012. A survey of Bayesian predictive methods for model assessment, selection and comparison. Stat. Surv. 6, 142–228.

This page intentionally left blank

Part II

Multivariate Models

This page intentionally left blank

Chapter 5

Dynamic panel GMM using R☆

Peter C.B. Phillips[a,*] **and Chirok Han**[b]

[a]*Sterling Professor of Economics and Professor of Statistics at Yale University, New Haven, CT, United States*

[b]*Professor of Economics at Korea University, Seoul, Republic of Korea*

[*]*Corresponding author: e-mail: peter.phillips@yale.edu*

Abstract

GMM methods for estimating dynamic panel regression models are heavily used in applied work in many areas of economics and more widely in the social and business sciences. Software packages in STATA and GAUSS are commonly used in these applications. We provide a new R program for difference GMM, system GMM, and within-group estimation for simulation with the model we consider that is based on a standard first-order dynamic panel regression with individual- and time-specific effects. The program lacks the generality of a full package but provides a foundation for further development and is optimized for speed, making it particularly useful for large panels and simulation purposes. The program is illustrated in simulations that include both stationary and nonstationary cases. Particular attention in the simulations is given to analyzing the impact of fixed effect heterogeneity on bias in system GMM estimation compared with the other methods.

Keywords: Bias, Difference GMM, Dynamic panel, System GMM, Within-group estimation

JEL Classification: C32, C33

1 Introduction

Longitudinal studies record changing information about the same cross section units over time. The sample may relate to individuals, households, firms, or collections of individuals in the form of cohorts or industries. The data may involve economic characteristics such as income, expenditure, and employment or indicators of health, well-being, and socioeconomic status. Longitudinal data can be arranged in a matrix where each row tracks a different individual or unit

☆Phillips acknowledges support from a Kelly Fellowship at the University of Auckland. We thank Dr. Hyoungjong Kim for comments on the paper and assistance with the GMM R code and verification of final results.

Handbook of Statistics, Vol. 41. https://doi.org/10.1016/bs.host.2019.01.002

(i) at different points in time (t), thereby constituting a panel $\{y_{it}: i=1, \ldots, N; t=1, \ldots, T\}$ of observations of N individuals at T time periods leading to NT total observations of a particular variable (or vector of variables) y_{it}. Such data are commonly known as panel data and models that represent their process of generation are known as panel data models.

Panel data offer many more opportunities for learning about real world phenomena than cross section data and time series data. Primary among these is the capacity to measure changes in behavior or outcomes over time, to study the duration of certain characteristics, and to record the timing and impact of events. These appealing features have led to the creation of many longitudinal studies at city, regional, national, and international levels. Among the earliest studies of this type are the Panel Study of Income Dynamics (PSID)[a] which commenced in 1968 as a survey of some 70,000 households in the United States, the Dunedin Multidisciplinary Health and Development Study[b] which commenced in 1972 following a group of around 1000 individuals born in New Zealand during 1972–73, and the Wisconsin Longitudinal Study,[c] following 10,317 individual graduates from Wisconsin High Schools since 1957.

Recent longitudinal studies have broadened the fields of enquiry to include topics that are becoming of growing importance to modern society. Among many such examples, we mention here only two. One is the effect of aging demographics in many countries of the world. Aging impacts individuals in terms of income, retirement decisions, housing, health, general functionality, day-to-day life, and well-being. Longitudinal survey information on these aspects of aging population assists policy makers in designing programs to address changing societal needs as demographics evolve. Two such studies are the Australian Longitudinal Study on Aging[d] and the Singapore Life Panel.[e]

A second area where longitudinal data now plays a vital role is in assessing the environmental impact of climate change. Rising temperatures and sea levels associated with anthropogenic sources in the modern industrial age have major global implications for human society and more generally for all life on Earth. Methods by which such changes are being assessed have relied in the past on the use of global climate models that simulate the evolution of atmospheric and oceanic conditions in response to incoming radiation, the filtering effects of aerosols, and the heat retention capacity of greenhouse gases at various station locations around the globe. Observational data recorded at some of the land-based stations may also be employed to assess the impact of the various driving forces behind climate. Panel models have recently been designed and estimated with both these data sources to determine Earth's climate sensitivity

[a] https://psidonline.isr.umich.edu/.
[b] https://dunedinstudy.otago.ac.nz/.
[c] https://www.ssc.wisc.edu/wlsresearch/.
[d] http://www.flinders.edu.au/sabs/fcas/alsa/alsa_home.cfm.
[e] https://slp.smu.edu.sg/sms/.

to increases in greenhouse gas emissions (Magnus et al., 2011; Storelvmo et al., 2016, 2018; Phillips et al., 2018).

Two important features of these climate econometric models are worth noting: (i) panel dynamics are incorporated to capture internal dynamic workings of the climate system and (ii) time-specific effects in each period are included to capture the explicit influence on station level temperature of aggregate variables that reflect prevailing global climate conditions. The second specification builds simultaneity into the system that provides feedback from global to station level data. This type of macro to micro feedback is to be expected in complex interdependent systems such as global climate. But it also manifests in many other settings where there is macroeconomic or community-wide social influence on individual behavior. The dynamic panel model that we use for illustration in the present study embodies this feedback feature.

Since the early 1980s and, in particular since the study by Nickell (1981) on dynamic panel bias, econometric methods have played a major role in the development of suitable methodology for estimation and inference in dynamic panel models. Prominent among these methods have been moment-based methods such as generalized method of moments (GMM) which work from clearly defined moment equations and carefully constructed instrumental variables based on both differences and levels of past observations.[f] GMM methods for estimating dynamic panel regression models are used in empirical research throughout the social, business, and medical sciences. Several different versions of these methods are available, including options for the inclusion of certain instruments and the use of additional estimating equations or moment conditions.

The present contribution provides a new suite of programs written in R. The programs provide for estimation and inference based on so-called difference GMM (hereafter, diff-GMM), system GMM (hereafter, sys-GMM), and within-group (WG) methods. These R programs complement software in STATA,[g] GAUSS,[h] and the R `plm` package (Croissant and Millo, 2018) that are presently available for applications. The new programs use fast computational algorithms that are particularly useful in large panels and simulation exercises. The programs are written for a standard first-order dynamic panel regression with individual- and time-specific effects. They lack the generality of a software package. But the code provided can be varied and extended to deal with models of greater complexity. The code has been extensively tested against existing software packages in STATA.

[f]Readers are referred to the works of Arellano (2003), Baltagi (2013), Hsiao (2014), Pesaran (2015), and Wooldridge (2010) for textbook discussions of these methods.
[g]STATA is a registered trademark of StataCorp LLC.
[h]GAUSS is a registered trademark of Aptech Systems, Inc.

Some illustrative simulations of the new programs are provided for a dynamic panel autoregression that allows for stationary and nonstationary cases, as well as time-specific effects that are determined by global driver variables. The findings reveal some notably large bias effect differences between sys-GMM estimation and the other methods. These biases are sourced in high levels of fixed effect heterogeneity in relation to equation error variance, corroborating analytic evidence by Hayakawa (2007, 2015).

2 A dynamic panel model with macro drivers

To illustrate the main methods of estimating dynamic panel regressions, we use the following model with both individual-specific and time-specific components. The model differs from the usual formulation in that the time-specific effects depend on global averages, thereby building into the framework a distinctive feedback and simultaneity. As discussed in Section 1, this type of feedback is likely to be present in many applications where macro level influences affect micro observations via individual and firm decisions that are made while cognizant of prevailing macro conditions.

Prominent examples occur in real estate and climate studies. For instance, individual real estate sales in a specific region may depend on local dynamics as well as regional determinants (such as immigration, and state, county, or city policy decisions) and national level determinants (such as prevailing interest rates and inflation). Likewise models of Earth's climate may involve station level dynamics with station level individual fixed effects as well as time-specific effects that involve global influences including variations in solar radiation that reach the Earth's surface and growing levels of atmospheric carbon dioxide and other greenhouse gases. In both cases, attention also needs to be given to potential nonstationary elements and trending behavior in the component variables, such as secularly rising real estate prices, growth in greenhouse gas concentrations, global temperature changes, shrinking glaciers, and rising sea levels.

The model we will use in our R simulation follows the design of the global climate econometric model in Storelvmo et al. (2016) and the simulation model used in Phillips (2018). It is designed to embody some of the characteristic dependencies and interactions described earlier and is given by the following two equations:

$$y_{it} = \alpha_i + \beta_1 y_{it-1} + \beta_2 x_{it-1} + \lambda_{t-1} + u_{it} \tag{1}$$

$$\lambda_{t-1} = \gamma_0 + \gamma_1 \overline{y}_{t-1} + \gamma_2 \overline{x}_{t-1} + \gamma_3 z_{t-1} \tag{2}$$

where $(\overline{y}_{t-1}, \overline{x}_{t-1}) = \left(N^{-1}\sum_{i=1}^{N} y_{it-1}, N^{-1}\sum_{i=1}^{N} x_{it-1}\right)$ are cross section aggregates of y_{it-1} and x_{it-1}, z_{t-1} is exogenous and u_{it} is a disturbance.

Eq. (1) has the usual dynamic panel form with both individual fixed effects α_i, time-specific effects λ_t, and predetermined inputs (y_{it-1}, x_{it-1}). Eq. (2) characterizes the aggregate effects that influence individual members of the panel $y_{it.}$ This equation presently includes only the aggregate predetermined variables $(\overline{y}_{t-1}, \overline{x}_{t-1}, z_{t-1})$ and may be augmented by including a disturbance term $u_{\lambda t}$.

In this chapter, we will treat x_{it-1} and z_{it-1} as exogenous variables with respective generating mechanisms

$$\begin{aligned} x_{it} &= \rho_x x_{it-1} + v_{xit} \\ z_t &= \mu_0 + \mu_1 t + z_t^0, \quad \text{and} \quad z_t^0 = \rho_z z_{t-1}^0 + v_{zt}. \end{aligned}$$

For the purposes of this illustration of GMM methods, we will assume that the equation errors $(u_{it}, v_{xit}, v_{zt}) \sim_d iid\, \mathcal{N}(0, \Sigma_v)$ over both indices i and t, with $\sum_v = \text{diag}(\sigma_u^2, \sigma_x^2, \sigma_z^2)$. The fixed effects are assumed to be drawn from $\alpha_i \sim_d iid\, \mathcal{N}\left(0, \sigma_\alpha^2\right)$ with zero mean as a standardization (given the presence of γ_0 in (2)).

Cross section aggregation of (1) and combination with (2) gives the aggregate dynamic equation

$$\overline{y}_t = (\overline{\alpha} + \gamma_0) + (\beta_1 + \gamma_1)\overline{y}_{t-1} + (\beta_2 + \gamma_2)\overline{x}_{t-1} + \gamma_3 z_{t-1} + \overline{u}_t \tag{3}$$

The stability condition for the aggregate dynamics in (3) is $|\beta_1 + \gamma_1| < 1$ and the condition for the individual level dynamics in (1) is $|\beta_1| < 1$. Under these conditions y_{it} is generated by a stable dynamic system about a stochastic trend x_{it} and a stochastic trend with drift driven by the time-specific effects λ_t that are in turn driven by the two nonstationary global variables $\overline{x}_t$ and $\overline{z}_t$ via (2). Similarly, at the aggregate level, the global $\overline{y}_t$ satisfies the stable dynamic Eq. (3) about the aggregate stochastic trend $\overline{x}_t$ and global stochastic trend with drift z_t.

This type of behavior is to be anticipated in many socioeconomic and geophysical contexts where stable dynamic forces interact with exogenously driven trend mechanisms. For instance, in studying global climate systems, energy balance considerations involve the interaction of incoming solar radiation, internal heat generation within the Earth itself and from anthropogenic surface sources with greenhouse gas, atmospheric pollutants, and aerosol production. This interaction between stable solar radiation and trending atmospheric conditions means that the both local and global climate systems may be modeled as evolving dynamically around stochastic and deterministic trends.

We use within-group (WG), diff-GMM, and sys-GMM estimation of (1) and estimate time-specific effects λ_t by global averaging of the fitted Eq. (1) using the normalization $\overline{\alpha} = 0$ giving

$$\hat{\lambda}_t = \overline{y}_t - \hat{\beta}_1 \overline{y}_{t-1} - \hat{\beta}_2 \overline{x}_{t-1}.$$

The resulting residual time series $\hat{\lambda}_t$ is then used to estimate the parameters of (2). Ordinary least squares (OLS) may be used in this regression. Recognizing the nonstationarity of the series at the aggregate level, an efficient

cointegration regression method may be employed such as dynamic least squares (DOLS; Saikkonen, 1991; Phillips and Loretan, 1991; Stock and Watson, 1993), fully modified least squares (FM-OLS; Phillips and Hansen, 1990), or a more recent efficient methods like trend IV regression (Phillips, 2014). The latter methods require further R programs and are therefore not included here. In the present treatment, OLS is used.

3 R code for dynamic panel estimation

Unlike C and Gauss, R implements matrices in column-major storage order. As we consider samples in which $N > T$, it is more efficient to store $(y_{1t}, \ldots, y_{Nt})'$ as a column. With this feature taken into consideration, we explain the R code developed here and applied in our simulation exercise.

3.1 Data generation

We first set N (`nsize`), T (`tsize`), and the number of initial observations (`burn`) to discard:

```
nsize <- 1000
tsize <- 40
burn <- 20
t.all <- tsize + burn
```

The parameters to set for our simulation exercise are β_1, β_2, γ_0, γ_1, γ_2, γ_3, σ_α, σ_u, σ_x, σ_z, μ_0, μ_1, ρ_x, and ρ_z. These are specified as follows:

```
beta1 <- .25
beta2 <- .1
gamma0 <- 5
gamma1 <- -.1
gamma2 <- .1
gamma3 <- 3
sigma <- list(alpha=5, u=1, x=1, z=1)
mu0 <- 1
mu1 <- .005
rho <- list(z=1, x=1)
```

As will become apparent in our discussion of the findings, the primary influences on performance in parameter estimation are the signal-to-noise ratios $SNR_x = \sigma_x/\sigma_u$, $SNR_z = \sigma_z/\sigma_u$, the relative sample size ratio $\frac{N}{T}$, and the fixed effect heterogeneity-to-noise ratio $FNR_\alpha = \sigma_\alpha/\sigma_u$. In particular, when $FNR_\alpha \geq 10$ sys-GMM estimation suffers from substantial bias. For the settings above, we have $|\beta_1 + \gamma_1| = 0.15$ and $|\beta_1| = 0.25 < 1$. So both local and global stability conditions hold. The settings $\rho_x = \rho_z = 1$ and $\mu_1 = 0.005$ imply that the x_{it} are a collection of independent stochastic trends and z_t is a stochastic trend with drift. The dynamic system then evolves locally and globally in a stable fashion around stochastic and deterministic trends driven by the exogenous inputs x_{it} and z_t.

Given these parameter values, we generate the panel data $x_{it}=\rho_x x_{it-1}+v_{xit}$ with $v_{xit}\sim\mathcal{N}\left(0,\sigma_x^2\right)$ as follows:

```
set.seed(1)
x <- sigma$x * matrix(rnorm(nsize*t.all), nsize, t.all)
for (j in 2:ncol(x)) x[,j] <- rho$x * x[,j-1] + x[,j]
if (rho$x==1) x <- x-x[,burn-1]
```

where we subtract $x_{i,-1}$ from x_{it} in case $\rho_x=1$ in order to prevent x_{i0} from having too large a cross-sectional variance due to large burn. The time series $z_t=\mu_0+\mu_1 t+z_t^0$ with $z_t^0=\rho_z z_{t-1}^0+v_{zt}$ is generated as follows:

```
trend <- mu0 + mu1 * seq(-burn+1, tsize)
z0 <- filter(rnorm(t.all), rho$z, method="recursive")
if (rho$z==1) z0 <- z0-z0[burn-1]
z <- trend + z0
```

where again z_{-1} is subtracted in case $\rho_z=1$. The idiosyncratic errors $u_{it}\sim\mathcal{N}\left(0,\sigma_u^2\right)$ and the individual effects $\alpha_i\sim\mathcal{N}\left(0,\sigma_\alpha^2\right)$ are drawn by the following code:

```
alpha <- sigma$alpha * rnorm(nsize)
u <- sigma$u * matrix(rnorm(nsize*t.all), nsize, t.all)
```

With these components in hand, we recursively generate y_{it} as follows:

```
xbar <- colMeans(x)
y <- matrix(NA, nsize, t.all)
y[,1] <- alpha + u[,1]
for (j in 2:ncol(y)) {
  lambda <- gamma0 + gamma1 * mean(y[,j-1]) +
       gamma2 * xbar[j-1] + gamma3 * z[j-1]
y[,j] <- alpha + beta1*y[,j-1] + beta2*x[,j-1] + lambda + u[,j]
}
```

(Note the line break in the algebra between lines 5 and 6 occurs after the "+," not before, because otherwise R regards the first line as a complete sentence and ignores the subsequent line.) We finally trim the initial observations using

```
y <- y[, burn:t.all]
x <- x[, burn:t.all]
z <- z[ burn:t.all]
```

After this, `y` is the $N\times(T+1)$ matrix of y_{it}, and `x` that of x_{it}, for $t=0, 1, \ldots, T$. The `z` object stores the $(T+1)$-vector of z_t for $t=0, 1, \ldots, T$. x_{iT} and z_T are not used for estimation.

For estimation, it is convenient to prepare the $N\times T$ matrices of y_{it}, y_{it-1}, and x_{it-1} and the vectors $\overline{y}_t,\overline{y}_{t-1},\overline{x}_{t-1}$ and z_{t-1} for $t=1, 2, \ldots, T$. Below, they are called `y2`, `y1`, `x1`, `y2bar`, `y1bar`, `x1bar`, and `z1`, where the "`2`" suffix is for no lag and the "`1`" for one lag.

```
ybar <- colMeans(y)
xbar <- colMeans(x)

idx.lag0 <- 2:ncol(y)
idx.lag1 <- idx.lag0 - 1

y2 <- y[,idx.lag0]
y1 <- y[,idx.lag1]
x1 <- x[,idx.lag1]
z1 <- z[ idx.lag1]
y2bar <- ybar[idx.lag0]
y1bar <- ybar[idx.lag1]
x1bar <- xbar[idx.lag1]
```

3.2 Within-group estimation

We first estimate the model $y_{it} = \alpha_i + \beta_1 y_{it-1} + \beta_2 x_{it-1} + \theta_t + u_{it}$, where θ_t denotes (unobserved) common time effects. The fastest method is to regress the "two-way" within deviations (within-group and within-period deviations) of y_{it} on those of y_{it-1} and x_{it-1}. The two-way within deviations are obtained by the following user-written `Within2` function:

```
Within2 <- function(x) {
   z <- x - rep(colMeans(x), rep.int(nrow(x), ncol(x)))
   z - rowMeans(z)
}
```

The first line inside the function removes the cross-sectional averages,[i] and the second line the within-group averages. The within-group estimates of β_1 and β_2 are obtained by the following:

```
y2d <- Within2(y2)
y1d <- Within2(y1)
x1d <- Within2(x1)

wg <- .lm.fit(cbind(as.vector(y1d), as.vector(x1d)),
              as.vector(y2d))$coef
```

The `as.vector()` function converts a matrix into a vector very fast.[j] The dotted function `.lm.fit()` is considerably faster than the standard `lm()` function (see the help document and try benchmarking) and is only slightly slower than manual calculation using `solve()`. The resulting `wg` object contains the WG estimates $\hat{\beta}_1$ and $\hat{\beta}_2$. We have verified the results from this procedure against Stata's `xtreg, fe`.

[i] This part of the code was inspired by http://www.gastonsanchez.com/visually-enforced/how-to/2014/01/15/Center-data-in-R/.

[j] See the benchmark reported by David Bellot in https://stackoverflow.com/questions/3823211/convert-a-matrix-to-a-1-dimensional-array.

The γ_j parameters are estimated by regressing $\hat{\theta}_t$ (which corresponds to λ_{t-1}) on $\overline{y}_{t-1}$, $\overline{x}_{t-1}$, and z_{t-1}. For this, $\hat{\theta}_t$ (plus a constant) is obtained as $\overline{y}_t - \hat{\beta}_1 \overline{y}_{t-1} - \hat{\beta}_2 \overline{x}_{t-1}$. Note that the residual vector corresponds to (λ_0, …, λ_{T-1}) so these residuals may now be regressed on $\overline{y}_{t-1}$, $\overline{x}_{t-1}$, and z_{it-1}:

```
EstimateGamma <- function(b) {
   resid <- y2bar - b[1]*y1bar - b[2]*x1bar
   .lm.fit(cbind(1, y1bar, x1bar, z1), resid)$coef
}
ghat <- EstimateGamma(wg)
```

The `EstimateGamma()` function accepts `b` and returns $(\hat{\gamma}_0, \hat{\gamma}_1, \hat{\gamma}_2, \hat{\gamma}_3)$. This procedure uses the random objects `y2bar`, `y1bar`, `x1bar`, and `z1`, but can be placed outside the replication loop for slight time saving because an object inside a function is evaluated not when the function is defined but when the object is actually evaluated. That is, `y2bar`, `y1bar`, and others inside the `EstimateGamma()` function are updated every time the function is called. This procedure is written as a function because the same procedure will be executed for GMM as well as for WG.

3.3 Difference GMM

Croissant and Millo's plm package is fully fledged but slow. To be suitable for simulations, we manually implement the GMM procedures[k] for our model $y_{it} = \alpha_i + \beta_1 y_{it-1} + \beta_2 x_{it-1} + d_t\delta + u_{it}$, $t = 1, \ldots, T$, where α_i are fixed effects, x_{it-1} is strictly exogenous, $d_t = (d_{t2}, \ldots, d_{tT})$ is the vector of time dummies with $d_{ts} = 1$ if $t = s$ and 0 otherwise, and u_{it} are the serially uncorrelated idiosyncratic errors. For notational brevity, let $T_j = T - j$ and $\mathbf{q}_{it} = (x_{it-1}, d_t)$ so the model is $y_{it} = \alpha_i + \beta_1 y_{it-1} + \mathbf{q}_{it}\beta_\mathbf{q} + u_{it}$, where $\beta_\mathbf{q} = (\beta_2, \delta')'$.

For diff-GMM, the dependent variable vector for unit i is $\Delta y_i = (\Delta y_{i2}, \ldots, \Delta y_{iT})'$, the corresponding regressor matrix is

$$\mathbf{X}_i = \begin{pmatrix} \Delta y_{i1} & \Delta \mathbf{q}_{i2} \\ \Delta y_{i2} & \Delta \mathbf{q}_{i3} \\ \vdots & \vdots \\ \Delta y_{iT-1} & \Delta \mathbf{q}_{iT} \end{pmatrix}$$

and the matrix of instruments is

[k]Our understanding of STATA's implementation of difference and system GMM was assisted by reading StataCorp (2015) and by studying output from the STATA `xtabond2` package (Roodman, 2009).

$$\mathbf{W}_i = \begin{pmatrix} y_{i0} & 0 & 0 & \dots & 0 & \dots & 0 & \Delta\mathbf{q}_{i2} \\ 0 & y_{i0} & y_{i1} & \dots & 0 & \dots & 0 & \Delta\mathbf{q}_{i3} \\ \vdots & \vdots & \vdots & & \vdots & & \vdots & \vdots \\ 0 & 0 & 0 & \dots & y_{i0} & y_{iT-2} & & \Delta\mathbf{q}_{iT} \end{pmatrix}. \tag{4}$$

Difference GMM makes use of the orthogonality condition $E(\mathbf{W}_i'\Delta u_i)=0$, where $\Delta u_i=(\Delta u_{i2}, \dots, \Delta u_{iT})'$. With the $\mathbf{W}_i$ disintegrated over t, these orthogonality conditions are (i) $E(y_{is}\Delta u_{it})=0$ for $s\le t-2$ and $t=2, \dots, T$, (ii) $E(\sum_{t=2}^{T}\Delta x_{it-1}\Delta u_{it})=0$, and (iii) $E(\Delta d_t'\Delta u_{it})=0$ for $t=2, \dots, T$. Because x_{it} is strictly exogenous to u_{it}, the single moment restriction in (ii) may be replaced with (ii′) $E(x_{is}\Delta u_{it})=0$ for all s and t, but it is a common practice to use (ii) instead of (ii′) in order to avoid using too many moment conditions. Another way of understanding diff-GMM is as three-stage least squares regression applied to the T_1 differenced equations using $\mathbf{W}_i$ as instruments, with the restriction that the slope parameters are the same for all t. According to this interpretation, the "reduced-form" equations may be written as

$$\Delta y_{it-1} = \pi_{t0} + \sum_{s=0}^{t-2}\pi_{ts}y_{is} + \phi\Delta x_{it-1} + \text{error}_{it}, \quad t=2,\dots,T,$$

where the intercept π_{t0} is time specific due to the time dummies included in the instrument set, but the coefficient ϕ of Δx_{it-1} is common for all t due to the particular way in which $\mathbf{W}_i$ is constructed. In order to make ϕ different across t, the $\mathbf{W}_i$ matrix should also include the interaction of Δx_{it-1} and the time dummies (excluding the dummy for $t=2$), which leads to T_2 more instruments. Interesting as it is to examine the effect of this modification for our model and in general, in the remainder of this study we follow the convention of industry practice and use (4) as the instrument matrix characterization for diff-GMM.

The one-step efficient GMM estimator is computed as

$$\left(S_{wx}'A_1^{-1}S_{wx}\right)^{-1}S_{wx}'A_1^{-1}S_{wy},$$

where $S_{wx}=\sum_{i=1}^{n}\mathbf{W}_i'\mathbf{X}_i$, $S_{wy}=\sum_i\mathbf{W}_i'\Delta y_i$, $A_1=\sum_i W_i'H_1W_i$ with $H_1=D'D$, and D' is the $T_2\times T_1$ difference matrix such that $D'(a_1, \dots, a_t)'=(\Delta a_2, \dots, \Delta a_t)'$. The two-step efficient GMM estimator is obtained by replacing A_1 with $A_2=\sum_i\mathbf{W}_i'\mathbf{e}_i\mathbf{e}_i'\mathbf{W}_i$, where $\mathbf{e}_i$ is the $T_2\times 1$ vector of the residuals from the one-step GMM.

Next we explain how to implement these estimators in R. As mentioned at the outset, R stores matrices in column-major order. We accordingly store variables (such as y_{it}, y_{it-1}, and x_{it-1}) as $N\times T$ matrices `y2`, `y1`, and `x1` in order to speed up calculation. It is not computationally efficient to loop over i and t for calculation. Instead, we want to block-copy the columns of `y2`, `y1`, and `x1` to construct the required matrices. For this purpose, we choose to stack variables across i first and then over t. Also, as R is particularly good at

handling lists, we will sometimes store objects (especially for the instruments) as list objects.

We first create time dummies. For this purpose, let $d_t=(d_{t2}, \ldots, d_{tT})$ be the $1\times T_1$ vector of time dummies such that $d_{ts}=I(s=t)$. To deal with time dummies, we create a "list" of T objects containing the $N\times T_1$ matrices of $\mathbf{1}_N\, d_t$ for $t=1, 2, \ldots, T$.

```
TD <- list()
TD[[1]] <- matrix(0,nsize,tsize-1)
for (j in 2:tsize) {
  TD[[j]] <- matrix(0,nsize,tsize-1)
  TD[[j]][,j-1] <- 1
}
```

The tth object in the `TD` list is the $N\times T_1$ matrix of $\mathbf{1}_N\, d_t$. For example, its second object `TD[[2]]` is the $N\times T_1$ matrix with 1 in the first column and 0 everywhere else. We do not pursue computational efficiency because `TD` is to be generated only once for each simulation setting.

To construct the regressor matrix and the instruments for diff-GMM, we need the differenced time dummies. We store it as another list, called `DTD`, of T_1 objects for $t=2, \ldots, T$.

```
DTD <- list()
for (j in 1:(length(TD)-1)) DTD[[j]] <- TD[[j+1]]-TD[[j]]
```

The resulting `DTD` list contains the $N\times T_1$ matrices of differenced time dummies $\mathbf{1}_N\Delta\, d_2, \ldots, \mathbf{1}_N\Delta\, d_t$. The tth object in the list is $\mathbf{1}_N\Delta\, d_{t+1}$.

The construction of the stacked dependent variable $\mathbf{y}$ is straightforward. We only need to make sure that it is stacked over i first and then over t and that the dimension is $NT_1\times 1$ (for $\Delta y_{i2}, \ldots, \Delta y_{iT}$). To get Δy_{it}, we difference y_{it} by

```
DiffMat <- function(x) x[,-1] - x[,-ncol(x)]
dy2 <- DiffMat(y2)
```

so that `dy2` is the $N\times T_1$ matrix of Δy_{it} for $t=2, \ldots, T$, and the dependent variable vector Δy_i is obtained by `as.vector(dy2)`. For the regressor matrix, the $N\times T_1$ matrix of Δy_{it-1} is obtained by

```
dy1 <- DiffMat(y1)
```

which is then stacked into a vector by `as.vector(dy1)`, and the Δx_{it-1} part is obtained similarly by

```
dx1 <- DiffMat(x1)
```

and then `as.vector(dx1)`. Next, we need to stack the components of `DTD` vertically. A short command for this is "`do.call(rbind, DTD)`."

```
library(Matrix)
DTD.mat <- Matrix(do.call(rbind, DTD), sparse = TRUE)
```

We do not pay particular attention to pursuing time saving for the differenced time dummies because this matrix is not repeatedly created over replications. But it is notable that storing the time dummy matrix as an R sparse matrix greatly helps to save computation time.[1] For the full regressor matrix, we horizontally attach them by:

```
XD <- cbind(as.vector(dy1), as.vector(dx1), DTD.mat)
```

As `DTD.mat` is a sparse matrix, so is `XD`, which is crucial for speed gain. The resulting `XD` matrix is an $NT_1 \times (T+1)$ sparse matrix of the regressors $(\Delta y_{it-1}, \Delta x_{it-1}, \Delta d_t)$ stacked over $i=1, \ldots, N$ first and then over $t=2, \ldots, T$.

Next, for the instruments, we form a list object `WD` containing the $N \times L$ instrument matrices ($L=$the number of columns of $\mathbf{W}_i$) for $t=2, \ldots, T$. The following function will do it.

```
MakeDgmmIV <- function(y1,dx1) {
  n <- nrow(y1)
  p <- ncol(dx1)
  qsize <- p*(p+1)/2
  z <- list()
  m <- 1
  mat0 <- matrix(0, n, qsize)
  for (j in 1:p) {
    z1 <- mat0
    for (k in 1:j) {
      z1[,m] <- y1[,k]
      m <- m+1
    }
    z[[j]] <- cbind(z1, dx1[,j], DTD[[j]])
  }
  z
}
WD <- MakeDgmmIV(y1,dx1)
```

The above body of code creates the list of T_1 instrument matrices, the jth element of which contains the $N \times L$ matrix of instruments for $t=j+1$.

We want to stack the components of `WD` for later use. One method is the short hand command `do.call(rbind,WD)`, but this command is slightly bloated

[1]The authors learned the importance of using R sparse matrices for time saving from information on https://stackoverflow.com/questions/53744906/how-to-make-crossprod-faster/53745063#53745063.

and we can save time by the following iteration which is tailored for the present study.

```
StackList <- function(aList, sparse = FALSE) {
  p <- length(aList)
  n <- nrow(aList[[1]])
  z <- matrix(0, p*n, ncol(aList[[1]]))
  i2 <- 0
  for (j in 1:p) {
    i1 <- i2+1
    i2 <- i2+n
    z[i1:i2,] <- aList[[j]]
  }
  if (sparse) Matrix(z, sparse = TRUE) else z
}
```

We will use `StackList()` whenever the components of a list object (containing matrices with the same dimension) needs to be vertically attached. If the `sparse` option is set, `StackList()` returns a sparse matrix. With this function at hand, the stacked instrument matrix is obtained by the following:

```
WDmat <- StackList(WD, sparse = TRUE)
```

For the one-step diff-GMM, it remains to evaluate $A_1 = \sum_i \mathbf{W}_i' H_1 \mathbf{W}_i$, where $H_1 = D'D$. For this, it is fast and convenient to first obtain the stacked matrix for $D\mathbf{W}_i$ and then compute its cross product matrix. Noting that

$$D\mathbf{W}_i = \begin{pmatrix} -W_{i2} \\ W_{i2} - W_{i3} \\ \vdots \\ W_{iT-1} - W_{iT} \\ W_{iT} \end{pmatrix},$$

we construct the stacked $D\mathbf{W}_i$ matrix as follows:

```
GetDW <- function(W) {
  x <- list()
  x[[1]] <- -W[[1]]
  for (j in 2:length(W)) x[[j]] <- W[[j-1]]-W[[j]]
  append(x, list(W[[j]]))
}
```

Using this function, A_1 is formed by the following

```
MakeA1 <- function(W) {
  a <- StackList(GetDW(W), sparse = TRUE)
  crossprod(a)
}
A1 <- MakeA1(WD)
```

Unless T is small, it is extremely important to store the stacked $D\mathbf{W}_i$ as a sparse matrix, which is done inside the `MakeA1` function above (see the "`sparse = TRUE`" option).

Now that all the constituents are ready, we obtain the one-step efficient GMM estimator as follows:

```
library(MASS)  # for ginv() if necessary
EstimateGMM <- function(Swx, Swy, Omega, ginv. = FALSE) {
  a <- if (ginv.) ginv(as.matrix(Omega))%*%Swx else
       solve(Omega,Swx)
  as.vector(solve(crossprod(a,Swx), crossprod(a,Swy)))
}

Swx <- crossprod(WDmat, XD)
Swy <- crossprod(WDmat, as.vector(dy2))
dg1 <- EstimateGMM(Swx, Swy, A1)
```

This code for the one-step efficient diff-GMM has been verified by comparison with results from STATA's "`xtabond`" command.

The two-step efficient diff-GMM estimator is obtained by replacing A_1 with $A_2 = \sum_i \mathbf{W}_i' \mathbf{e}_i \mathbf{e}_i' \mathbf{W}_i$, where $\mathbf{e}_i$ are the residuals from one-step diff-GMM (corresponding to Δu_{it}). For this, we first get the residuals by

```
du <- dy2-as.vector(XD%*%dg1)
```

which is an $N \times T_1$ matrix of $\Delta \hat{u}_{it}$ for $i = 1, \ldots, n$ (rows) and $t = 2, \ldots, T$ (columns). Then A_2 is obtained by first multiplying $\mathbf{e}_i$ to each column of $\mathbf{W}_i$ element-wise, adding up the results over t, and then getting the cross product of the resulting $N \times L$ matrix, as follows:

```
ListMatCrossprod <- function(W,e) {
  ans <- W[[1]]*e[,1]
  for (j in 2:length(W)) ans <- ans + W[[j]]*e[,j]
  ans
}

MakeClustCov <- function(W,du) {
  crossprod(ListMatCrossprod(W,du))
}
A2 <- MakeClustCov(WD,du)
```

The result of `ListMatCrossprod(WD,du)` is an $N \times L$ matrix, which hardly contains any zeros. It is thus unnecessary to convert it to a sparse matrix before applying `crossprod()`. The two-step efficient diff-GMM estimator is finally obtained by

```
dg2 <- EstimateGMM(Swx, Swy, A2)
```

where the "`EstimateGMM`" function has been defined before. If $N < L$, the `A2` matrix is singular, but R's `solve()` works in that case as well. The entire

body of code for diff-GMM has been verified by comparing results with the outputs from the `xtabond` command of STATA 14.

Given the two-step efficient difference GMM estimates in `dg2`, the γ_j estimates can be computed using `EstimateGamma(dg2)` as before.

3.4 System GMM

System GMM additionally employs the moment restrictions

$$E(y_{it} - \alpha - \beta_1 y_{it-1} - \beta_2 x_{it-1} - \theta_t) = 0, \quad t = 1, \ldots, T,$$
$$E[\Delta y_{it-1}(y_{it} - \alpha - \beta_1 y_{it-1} - \beta_2 x_{it-1} - \theta_t)] = 0, \quad t = 2, \ldots, T.$$

Note the presence of the global intercept and the first moments considered for $t = 1, \ldots, T$.

The full system of equations is then written in stacked form as

$$\begin{pmatrix} \Delta y_{it} \\ y_{it} \end{pmatrix} = \begin{pmatrix} \Delta y_{it-1} & \Delta x_{it-1} & \Delta d_t & 0 \\ y_{it-1} & x_{it-1} & dt & 1 \end{pmatrix} \begin{pmatrix} \beta_1 \\ \beta_2 \\ \delta \\ \beta_0 \end{pmatrix} + \begin{pmatrix} \Delta u_{it} \\ \alpha_i + u_{it} \end{pmatrix}$$

for $t = 1, 2, \ldots, T$, where $\beta_0 = \lambda_1$ is the intercept of the levels equation, and Δy_{i0}, Δx_{i0}, d_1, and d_0 are defined as zero. The instruments for these two equations are $W_{it}^S = \text{diag}(W_{it}^D, W_{it}^L)$, where W_{it}^D is row $t-1$ of $\mathbf{W}_i$ in (4) with W_{i1}^D defined as 0, and $W_{it}^L = [(\Delta y_{it-1})d_t, 1]$.

For the implementation, let us first create data matrices for sys-GMM. We combine `dy2` and `y2`, `dy1` and `y1`, and `dx1` and `x1` in a suitable way for sys-GMM.

```
y2s <- rbind(cbind(0,dy2), y2)
y1s <- rbind(cbind(0,dy1), y1)
x1s <- rbind(cbind(0,dx1), x1)
```

The zeros above correspond to $t = 1$ in the differenced equation. Given them, the dependent variable vector will be `as.vector(y2s)`, and the regressor matrix will be constructed by horizontally attaching `as.vector(y1s)`, `as.vector(x1s)`, the matrix for time effects, and the vector for the constant term in the levels equation.

For the time effects, we first prepend a zero matrix to the DTD list from the diff-GMM part in order to handle $t = 1$ as follows:

```
ZeroMatOf <- function(x) matrix(0, nrow(x), ncol(x))
DTDS <- append(DTD, list(ZeroMatOf(DTD[[1]])), after = 0)
```

Then the full time effects matrix for sys-GMM is constructed as follows:

```
TDS <- mapply(rbind, DTDS, TD, SIMPLIFY = FALSE)
TDS.mat <- Matrix(do.call(rbind, TDS), sparse = TRUE)
```

where we do not pursue computational efficiency because this part of the regressor matrix is generated outside the replication loop. Next, the vector for the constant term in the levels equation is $\mathbf{1}_t \otimes [(0, 1)' \otimes \mathbf{1}_N]$, which is constructed by

```
CONS <- Matrix(rep(rep(c(0,1), each=nsize), tsize),
               sparse = TRUE)
```

The regressor matrix for sys-GMM is now created as follows:

```
XS <- Matrix(cbind(as.vector(y1s), as.vector(x1s),
                   TDS.mat, CONS), sparse = T)
```

The next target is the instrument matrix. We will reuse the matrix `WD` already created for diff-GMM, but we prepend a zero matrix to `WD` to handle $t=1$, as follows:

```
WD <- append(WD, list(ZeroMatOf(WD[[1]])), after=0)
```

The length of `WD` is now T (it was previously T_1), with the first matrix (corresponding to $t=1$) being the $N \times L$ zero matrix. The instrument matrices for the levels equations are constructed by the following code:

```
MakeLgmmIV <- function(dy1) {
  p <- ncol(dy1)
  W <- list()
  w0 <- cbind(matrix(0, nrow(dy1), p), 1)
  W[[1]] <- w0
  for (j in 1:p) {
    w1 <- w0
    w1[,j] <- dy1[,j]
    W[[j+1]] <- w1
  }
  W
}
WL <- MakeLgmmIV(dy1)
```

The resulting `WL` ("`L`" for levels) is a length-T list of $N \times T$ matrices, the tth of which is the matrix of $(\Delta y_{it-1} d_t, 1)$, where the last 1 is for the universal constant instrument for the levels GMM part. Given `WD` and `WL`, the full instrument matrices for the system GMM is obtained by diagonally combining `WD` and `WL`. The `Matrix` library provides the following convenient `bdiag` function for this purpose.

```
WS <- lapply(mapply(bdiag,WD,WL,SIMPLIFY=FALSE), as.matrix)
```

We convert the sparse matrix into a usual matrix by `as.matrix` because our `StackList()` function operates on only usual matrices. Now, the stacked full instrument matrix is obtained by

```
WSmat <- StackList(WS, sparse = TRUE)
```

where `WSmat` is a sparse matrix.

It remains to construct the covariance matrix for the one-step sys-GMM. The one-step (inefficient) sys-GMM by STATA's `xtdpdsys` uses $D'D$ for the diff-GMM part, I for the levels GMM part, and zero for the covariance part. The A_1 variance matrix $\sum_i \mathbf{W}_i' D'D\mathbf{W}_i$ is identical to that for diff-GMM although the dimension of $\mathbf{W}_i$ changes due to the 0 part for $t=1$. The levels GMM part B_1 is the cross product of the levels instrument matrix because the identity transformation is used.

```
B1 <- crossprod(StackList(WL, sparse = TRUE))
```

Then Stata's one-step covariance matrix for sys-GMM is diag(A_1, B_1)

```
AS1 <- bdiag(A1, B1)
```

and the corresponding one-step sys-GMM estimator is obtained by:

```
Swx <- crossprod(WSmat, XS)
Swy <- crossprod(WSmat, as.vector(y2s))
sg1 <- EstimateGMM(Swx, Swy, AS1)
```

For the two-step efficient sys-GMM, the one-step residuals $\mathbf{e}_D$ and $\mathbf{e}_L$ are first obtained for Δu_{it} and $\alpha_i + u_{it}$, respectively. Then A_2^D and A_2^L are calculated by combining the instruments and residuals for the differenced equations and the levels equations, respectively, as before.

```
res <- as.vector(y2s)-as.vector(XS%*%sg1)
res <- matrix(res, nrow=2*nsize)
de <- res[1:nsize,]
ue <- res[seq(nsize+1,2*nsize),]

A2 <- MakeClustCov(WD,de)
B2 <- MakeClustCov(WL,ue)
```

The covariance part (of the differenced part and the levels part) is estimated by $\sum_i \left(\mathbf{W}_{D,i}'\hat{e}_{D,i}\right)\left(\mathbf{W}_{D,i}'\hat{e}_{L,i}\right)'$, where $\mathbf{W}_{D,i}$ and $\mathbf{W}_{L,i}$ are the instrument matrices for the difference GMM and the levels GMM, respectively, and $\hat{e}_{D,i}$ and $\hat{e}_{L,i}$ are the corresponding residuals. This matrix is obtained by the following code.

```
MakeClustCov2 <- function(W1,u1,W2,u2) {
  wu1 <- ListMatCrossprod(W1,u1)
  wu2 <- ListMatCrossprod(W2,u2)
  crossprod(wu1,wu2)
}
AB <- MakeClustCov2(WD,de,WL,ue)
```

Given A2, B2, and AB, the full two-step covariance matrix for sys-GMM is constructed by attaching them into a single matrix:

```
AS2 <- rbind(cbind(A2,AB), cbind(t(AB),B2))
```

Then the two-step sys-GMM estimator is computed as follows:

```
sg2 <- EstimateGMM(Swx, Swy, AS2)
```

Given the estimates of β_1 and β_2, the γ_j parameters are estimated by invoking EstimateGamma(sg2) as before. The code and algorithmic procedure have been verified against xtdpdsys of STATA 14, as now described.

3.5 Code verification and comparison

We have generated data for $N=1000$ and $T=40$ (with burn=20). The dimension of y and x is $N\times(T+1)$ as verified by the following:

```
> dim(y)
[1] 1000 41
> dim(x)
[1] 1000 41
```

For this dataset, the WG, diff-GMM, and sys-GMM estimates are reported as follows:

```
> cbind(wg, dg1[1:2], dg2[1:2], sg1[1:2], sg2[1:2]) wg
[1,] 0.2162380 0.2485650 0.2497906 0.2559667 0.2556852
[2,] 0.1061886 0.0995333 0.1007674 0.1008795 0.1009978
```

The two rows correspond to β_1 and β_2, respectively, and the five columns are for WG, one-step diff-GMM, two-step diff-GMM, one-step sys-GMM, and two-step sys-GMM, respectively.

Now we compare these results with the STATA outputs. For this, we first create a long-format dataset, which is saved in STATA's old .dta format:

```
w <- data.frame(id = as.vector(row(y)),
                year = as.vector(col(y))-1,
                y = as.vector(y), x = as.vector(x))
library(foreign)
write.dta(w, 'sample.dta')
```

Then we load the data from STATA 14 and estimate the coefficients by the following STATA commands:

```
use sample, clear
xtset id year
local tmax = r(tmax)
xtreg y l.(y x) i.year, fe
```

```
qui gen x1 = l.x
qui tab year, gen(yr_)
drop yr_1 yr_2

xtreg y l.(y x) yr_*, fe
xtabond y x1 yr_*
xtabond y x1 yr_*, two
xtdpdsys y x1 yr_*
xtdpdsys y x1 yr_*, two
```

The outputs from the last five STATA commands are tabulated in Table 1. The estimates are identical to those obtained by our R code.

On a computer system with 2.9 GHz Intel Core i7 CPU and 16 GB 2133 MHz LPDDR3 RAM, one pass of the entire body of R estimation procedures took approximately 4.2 s to complete. To compare performance on the same system, STATA 14 reported 4.1 s for the two-step diff-GMM alone and 4.4 s for the two-step sys-GMM. Our R code is about twice as time efficient, though this direct comparison is a little unfair because our R code reuses the diff-GMM code in sys-GMM and STATA's commands produce many ancillary test statistics as well. The R `plm` package seems not to be optimized. The two-step diff-GMM gave the same estimates as STATA in over 60 s, and the two-step sys-GMM used twice as long to produce estimates that are different from those delivered by STATA and our R programs. Computation time fluctuates randomly, but many trials in the course of our simulation exercise suggest that our R code is more time efficient than the existing STATA and R packages for GMM estimation of our dynamic panel data model.

4 Simulation results

We now report simulation results for various parameter settings. We start by exploring the case where $\sigma_\alpha = 1$, $\sigma_x = 1$, and $\sigma_z = 1$ (with $\sigma_u = 1$ throughout) for $N = 100$ and 800 and $T = 10$ and 40. We compare the performance of WG,

TABLE 1 STATA output for the test data

Variable	WG	$DGMM_1$	$DGMM_2$	$SGMM_1$	$SGMM_2$
y_{it-1}	0.2162380	0.2485650	0.2497906	0.2559667	0.2556852
x_{it-1}	0.1061886	0.0995333	0.1007674	0.1008795	0.1009978
Time dummies	Included, unreported				

Note: DGMM and SGMM denote diff-GMM and sys-GMM, respectively. The "1" and "2" suffixes denote the one-step and two-step procedures, respectively.

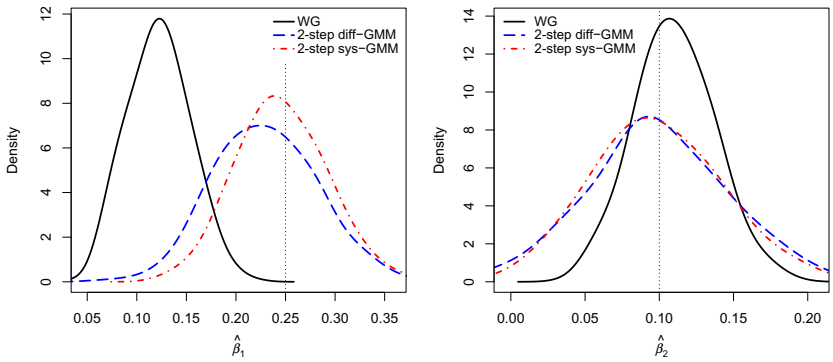

FIG. 1 Estimated densities for $N = 100$ and $T = 10$ for $\sigma_\alpha = 1$.

diff-GMM, and sys-GMM estimators of β_1 and β_2 in the usual setting where the moment restrictions for diff-GMM and those for sys-GMM are all strong. For this case, we expect that WG produces biased estimates and its bias depends intimately on T (Nickell, 1981), that the GMM estimates are consistent, and that sys-GMM is more efficient than diff-GMM. Fig. 1 presents the estimated densities of the estimators for this case.[m] For the AR coefficient (β_1), WG is certainly biased and diff-GMM is less biased. In this case, it is apparent that sys-GMM is the least biased estimator and is evidently more efficient than diff-GMM. For the β_2 parameter, bias is unclear even for WG, although the bias of WG is still larger than that of both GMM estimators.[n]

When N increases to 800 in Fig. 2, the same pattern is observed in a more exaggerated fashion. All the estimators are less scattered, and the bias of WG remains about the same. Comparison with Fig. 1 reveals growing concentration, corroborating the consistency of the GMM estimators of β_1 and β_2, with sys-GMM clearly more efficient than diff-GMM for β_1 estimation. The bias of WG is also evident for β_2 with N this large, although substantially smaller than for β_1.

The bias of the WG estimator decreases as T increases (Nickell, 1981). For example, when $T = 40$ and $N = 800$, the estimated densities are presented in Fig. 3. Bias reduction is noticeable. Also, sys-GMM outperforms diff-GMM only marginally in this case.

---

[m]Unlike WG and sys-GMM, diff-GMM sporadically produces very wild estimates. For $N = 200$ and $T = 20$, there were two instances (out of 1000) of the diff-GMM β_1 estimates being smaller than -1 with a minimum of -3.8 and three instances larger than 1 with a maximum 5.7. This behavior is not observed in the WG estimates or in the sys-GMM estimates. Oddly, the aberrant behavior of diff-GMM occurs only for $N = 200$ and $T = 20$. Possible reasons include proximity to singularity for some datasets. Identifying the reasons for this behavior of diff-GMM requires further investigation and is not pursued here.

[n]As shown in Phillips and Sul (2007, proposition 2), the bias of WG estimates of coefficients of exogenous variables in dynamic panel models is typically smaller than that of the autoregressive coefficient, contrary to the claim in Nickell (1981).

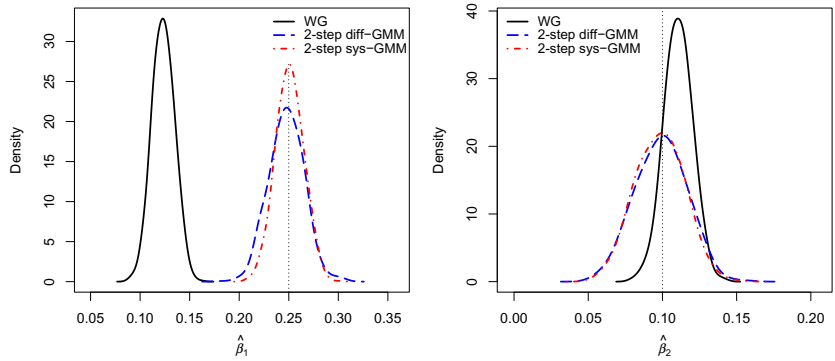

FIG. 2 Estimated densities for $N=800$ and $T=10$ for $\sigma_\alpha=1$.

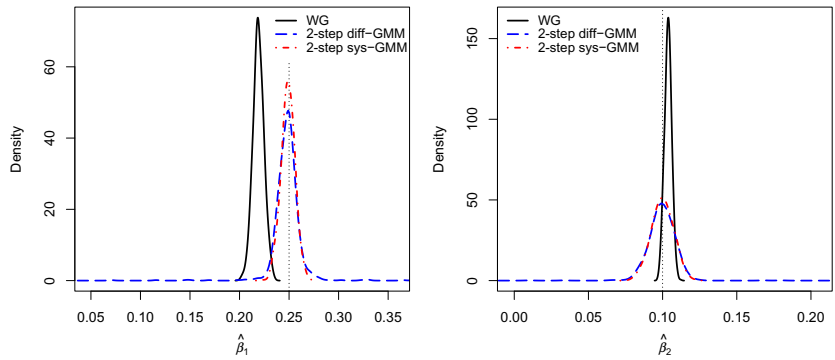

FIG. 3 Estimated densities for $N=800$ and $T=40$ for $\sigma_\alpha=1$.

The weak-instrument problem is manifest when the FNR_α ratio $\sigma_\alpha/\sigma_u=5$ or 10. Fig. 4A shows the estimated densities for $N=100$ and $T=10$ for $\sigma_\alpha=5$. The performance of sys-GMM is remarkably poor for the autoregressive parameter (β_1) in terms of both bias and efficiency, even though sys-GMM uses only nine poor moment restrictions in addition to the many strong instruments employed by diff-GMM. This phenomenon is intriguing because GMM procedures typically attach small weights to noisy moment functions. It may be related to inefficient one-step weighting matrix by STATA's sys-GMM, whose adverse effect has not died out in the second step estimation. That is, STATA's one-step sys-GMM ignores the presence of the fixed effects in the formation of covariance matrix for the levels equations and assumes that the error term in the levels equation has variance σ_u^2 and no serial correlation. As a result, STATA's one-step weight matrix is far from optimal especially if σ_u^2 is large. STATA's procedure also assumes that the differenced equations and the levels equations are mutually uncorrelated, which is not true. When the initial one-step estimator is largely biased due to extreme noise in a subset (the "levels" part) of the

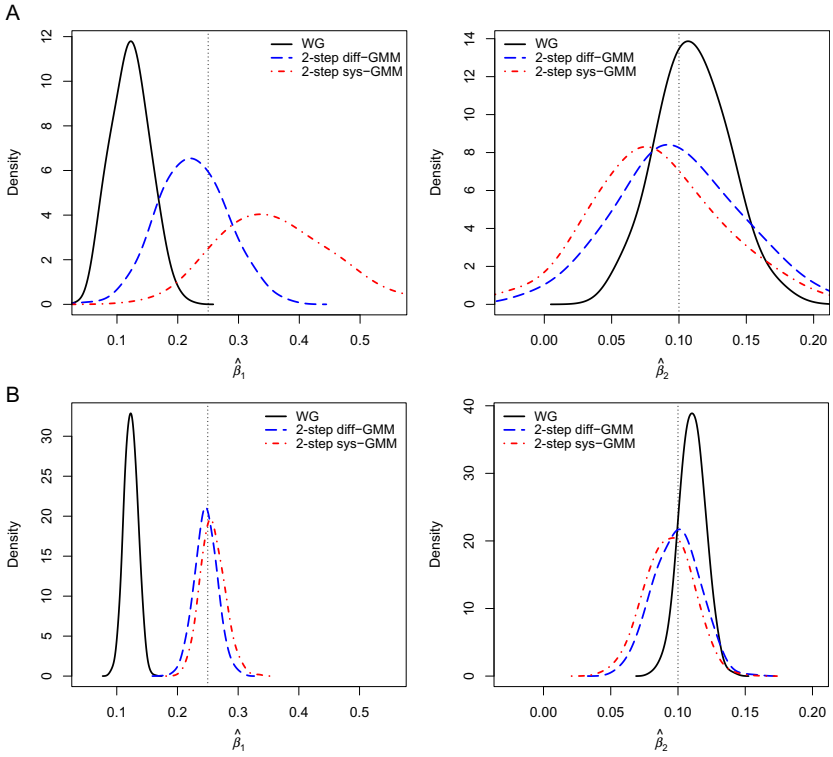

FIG. 4 Estimated densities for $\sigma_\alpha = 5$ and $T = 10$.

moment restrictions, the performance of the corresponding two-step estimator can be compromised if N is not very large. An interesting modification of sys-GMM to achieve improved performance would be the use of diff-GMM as the initial estimator, or a three-step estimation in which $\sigma_\alpha^2/\sigma_u^2$ is estimated in the second step to form a feasible optimal weight and nonparametrically estimated optimal weight is used in the third step. These topics are left for future research.

With N increasing, the weak-instrument problem fades away. For $N = 800$, Fig. 4B exhibits that sys-GMM is close to diff-GMM. We expect that sys-GMM eventually surpasses diff-GMM when N further increases, as the cases with $\sigma_\alpha = 1$ above show.

When T increases to 40, the estimated densities change to Fig. 5 for $N = 100$ in (A) and for $N = 800$ in (B). But increasing T does not remedy the poor performance of sys-GMM.

The poor performance of sys-GMM for large σ_α/σ_u is drastically emphasized when $\sigma_\alpha = 10$ (with $\sigma_u = 1$ as before). Fig. 6 shows the estimated

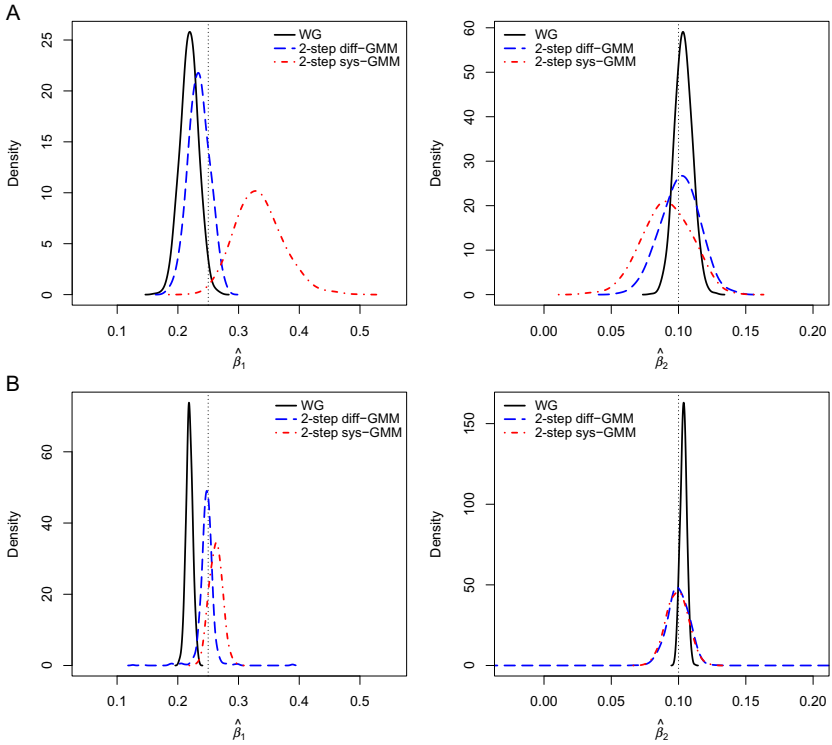

FIG. 5 Estimated densities for $\sigma_\alpha = 5$ and $T = 40$.

densities for $(N, T) = (100, 10)$ in (A) and $(N, T) = (800, 10)$ in (B). Large N is clearly required for sys-GMM to be useful in this case. In contrast, diff-GMM performs well. This analysis is relevant for applied work. Using empirical data on Earth's temperature, downwelling radiation, and CO_2 equivalent greenhouse gas levels over 1964–2005 with $N = 968$ and $T = 42$, Phillips (2018) found a σ_α/σ_u ratio of 15.043 and a sys-GMM estimate of β_1 of 0.8665, more than six times larger than the diff-GMM estimate of 0.1346.

We have also examined different σ_x and σ_z values and results remain qualitatively the same in terms of relative performance: When $\sigma_\alpha = 1$, sys-GMM is better than diff-GMM; when $\sigma_\alpha = 5$ or $\sigma_\alpha = 10$, the performance of sys-GMM is poorer than diff-GMM and unacceptably biased and inefficient for $N = 100$.

The distributions of the $\hat{\gamma}_j$ estimators can be compared using the mathematical fact that $\hat{\beta}_j + \hat{\gamma}_j$ is the same for all estimators (see Phillips, 2018, for this invariance), though differences in variances of $\hat{\beta}_j$ and $\hat{\gamma}_j$ may make the $\hat{\gamma}_j$ estimators appear to be distributed much more closely. For example, for $N = 800$ and

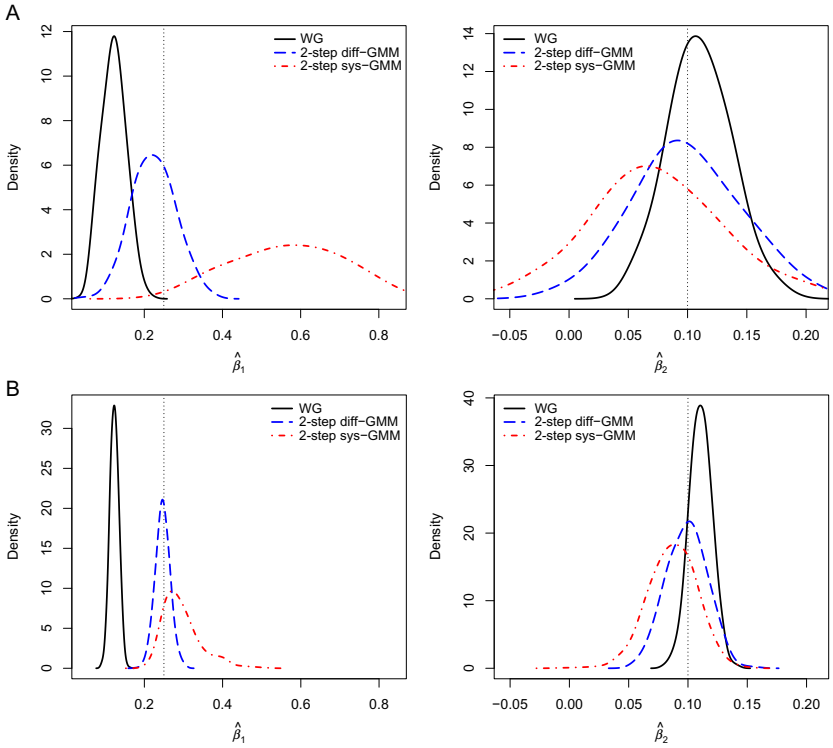

FIG. 6 Estimated densities for $\sigma_\alpha = 10$ and $T = 10$.

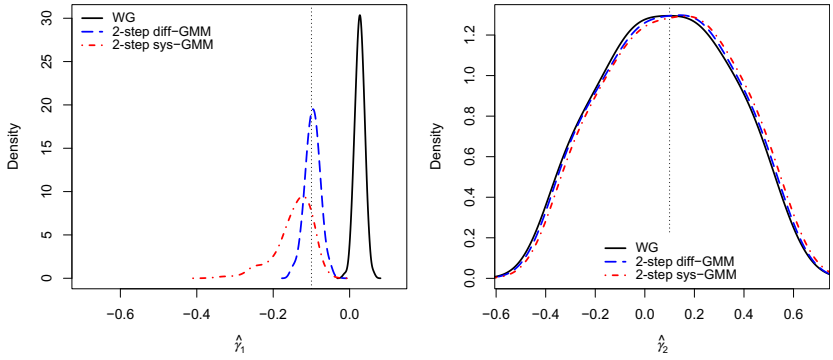

FIG. 7 Estimated densities of $\hat{\gamma}_1$ and $\hat{\gamma}_2$ for $\sigma_\alpha = 10$, $N = 800$, and $T = 10$.

$T = 10$ with $\sigma_\alpha = 10$, the estimated densities of $\hat{\gamma}_1$ and $\hat{\gamma}_2$ are given in Fig. 7. In comparison to Fig. 6B, the panel for $\hat{\gamma}_1$ in Fig. 7 is a clear mirror image of the $\hat{\beta}_1$ panel in Fig. 6B, which matches the fact that the estimates of $\hat{\beta}_j + \hat{\gamma}_j$ are identical. The distributions of $\hat{\gamma}_2$ look, on the other hand, similar to one another, unlike

Fig. 6B. This apparent similarity happens because the variances of $\hat{\gamma}_2$ are substantially larger than those of $\hat{\beta}_2$—so the densities (of $\hat{\gamma}_2$) appear similar on the given scale in Fig. 7, whereas the densities (of $\hat{\beta}_2$) are clearly differentiated in Fig. 6B.

5 Conclusion

Longitudinal data are now an integral part of experimental and empirical studies across a range of disciplines from the medical to the social and business sciences. As the number of such studies continues to grow, use of dynamic panel regression methods that have been developed in econometrics can be expected to become even more widespread than they are at present. Correspondingly, this growth will stimulate demand for open source programs in R that have been validated against existing proprietary software packages such as STATA. The programs presented here provide such an alternative and have the advantage of speed and code efficiency over existing software that makes them particularly useful in simulation exercises, especially for large panels.

References

Arellano, M., 2003. Panel Data Econometrics. Oxford University Press.

Baltagi, B.H., 2013. Econometric Analysis of Panel Data, fifth ed. John Wiley & Sons Ltd.

Croissant, Y., Millo, G., 2018. Panel Data Econometrics With R. John Wiley & Sons Ltd.

Hayakawa, K., 2007. Small sample bias properties of the system GMM estimator in dynamic panel data models. Econ. Lett. 95, 32–38.

Hayakawa, K., 2015. The asymptotic properties of the system GMM estimator in dynamic panel data models when both N and T are large. Econ. Theory 31, 647–667.

Hsiao, C., 2014. Analysis of Panel Data. Cambridge University Press.

Magnus, J.R., Melenberg, B., Muris, C., 2011. Global warming and local dimming: the statistical evidence. J. Am. Stat. Assoc. 106, 452–464.

Nickell, S., 1981. Biases in dynamic models with fixed effects. Econometrica 49, 1417–1426.

Pesaran, M.H., 2015. Time Series and Panel Data Econometrics. Oxford University Press.

Phillips, P.C.B., 2014. Optimal estimation of cointegrated systems with irrelevant instruments. J. Economet. 178, 2100–2224.

Phillips, P. C. B., 2018. Dynamic panel modeling of climate change. Cowles Foundation discussion paper #2150, Yale University.

Phillips, P.C.B., Hansen, B.E., 1990. Statistical inference in instrumental variables regression with I(1) processes. Rev. Econ. Stud. 57, 99–125.

Phillips, P.C.B., Loretan, M., 1991. Estimating long-run economic equilibria. Rev. Econ. Stud. 59, 407–436.

Phillips, P.C.B., Sul, D., 2007. Bias in dynamic panel estimation with fixed effects, incidental trends and cross section dependence. J. Economet. 137, 162–188.

Phillips, P. C. B., T. Leirvik, and T. Storelvmo, 2018. Econometric estimation of Earth's transient climate sensitivity. Working paper, Yale University.

Roodman, D., 2009. How to do xtabond2: an introduction to difference and system GMM in Stata. Stata J. 9 (1), 86–136.

Saikkonen, P., 1991. Asymptotically efficient estimation of cointegration regressions. Economet. Theory 7, 1–21.

StataCorp, 2015. Stata Longitudinal-Data/Panel-Data Reference Manual Release 14. Stata Press, College Station, TX.

Stock, J.H., Watson, M.W., 1993. A simple estimator of cointegrating vectors in higher order integrated systems. Econometrica 61, 783–821.

Storelvmo, T., Leirvik, T., Lohmann, U., Phillips, P.C.B., Wild, M., 2016. Disentangling greenhouse warming and aerosol cooling to reveal Earth's climate sensitivity. Nat. Geosci. 9, 286–289.

Storelvmo, T., Heede, U.K., Leirvik, T., Phillips, P.C.B., Arndt, P., Wild, M., 2018. Lethargic response to aerosol emissions in current climate models. Geophys. Res. Lett. 45, 9814–9823.

Wooldridge, J., 2010. Econometric Analysis of Cross Section and Panel Data, second ed. MIT Press.

Further reading

Kruiniger, H., 2009. GMM estimation of dynamic panel data models with persistent data. Economet. Theory 25, 1348–1391.

Chapter 6

Vector autoregressive moving average models

Wolfgang Scherrer* **and Manfred Deistler***
TU Wien, Vienna, Austria
**Corresponding authors: e-mail: wolfgang.scherrer@tuwien.ac.at; manfred.deistler@tuwien.ac.at*

Abstract
Vector autoregressive moving average (VARMA) processes constitute a flexible class of linearly regular processes with a wide range of applications. In many cases VARMA models allow for a more parsimonious parametrization than vector autoregressive (VAR) models. However, compared to VAR processes the relation between internal parameters and external characteristics (e.g., the autocovariance function) is more involved and estimation is harder since in general the maximum likelihood method here needs numerical optimization. In this contribution we want to give a broad overview of VARMA modeling with an emphasis on structure theory, estimation and practical implementation with the free software environment R and specialized R packages. First we present basic definitions and the interrelation between VAR, VARMA models and state space models. We will show how to compute important characteristics like autocovariance function, spectral density and impulse response functions and how to compute predictions. Then we discuss parametrization issues, including the question how to implement structural information. As mentioned above, estimation of VARMA models is quite involved. Consequently a substantial part of the paper will deal with maximum likelihood estimation and with alternative estimators which are cheaper to compute, but in general not asymptotically efficient. In addition to parameter estimation, model selection, in particular, choosing the best model order is treated.

Keywords: Vector autoregressive moving average process, State space model, Identifiability, Parameter estimation, Model selection

1 Introduction

This contribution aims at giving an introductory survey to structure, estimation, and computational aspects of vector autoregressive moving average (VARMA) and linear state space models. VARMA, respectively, state space models constitute a flexible class of linear dynamic systems for modeling of linearly regular,

Handbook of Statistics, Vol. 41. https://doi.org/10.1016/bs.host.2019.01.004

stationary processes. This contribution is neither self-contained nor does it treat all relevant aspects in VARMA modeling. For the theory of stationary processes we refer, e.g., to Rozanov (1967), Hannan and Deistler (2012), and Brockwell and Davis (1991).

Structure theory is concerned with the analysis of the relation between "external" characteristics of the observed process, such as the population second moments, and the "internal" model parameters. One aspect is identifiability, i.e., the question whether the external characteristics uniquely determine the model parameters. In addition here we consider the so-called realization problem, i.e., the problem of constructing the underlying VARMA (or state space) system from the population second moments of the observed process as well as the continuity of this mapping. This can be seen as an idealized "estimation" problem which gives insight for actual estimation. For these problems see, e.g., Hannan and Deistler (2012), Caines (1988), Söderström and Stoica (1989), Ljung (1999), Lütkepohl (2005), Kalman (1974), and the references therein.

Estimation of VARMA or state space models is quite intricate in particular for the multivariate case, for two reasons. First the parameter spaces are much more complicated compared to the VAR case and second, in general, there does not exist an explicit formula for the maximum likelihood estimate. We discuss (Gaussian) ML estimation and the asymptotic properties of these estimates. In addition we discuss estimation procedures, such as the Hannan–Rissanen–Kavalieris procedure and the CCA subspace procedure, which are, e.g., used as initial estimates for ML estimation.

In general an important part of an overall estimation procedure is model selection, e.g., determining the maximum AR/MA orders (p,q) of a VARMA model. Here we discuss this for the case of information criteria like AIC and BIC. Important references for the estimation of VARMA and state space models are Caines (1988), Hannan and Deistler (2012), Reinsel (1997), Ljung (1999), Lütkepohl (2005), and Tsay (2014).

This contribution also tries to demonstrate the usage of the statistical computer program R (R Core Team, 2015) for the modeling of multivariate time series with VARMA or state space models. We mainly use the MTS package (Tsay, 2015) and the dse package (Gilbert, 2015).

In econometrics the dominant approach for the modeling of multivariate time series is the usage of VAR models. This differs, e.g., from system identification in (control) engineering, where mostly state space models are used. The frequent use of VAR models in econometrics is explained by the fact that the Yule–Walker estimates are easy to calculate and asymptotically efficient. However, due to the vast increase of computation power nowadays also the estimation of VARMA (state space) models does not cause substantial computational problems. We hope that this contribution serves as a motivation for econometricians to try to fit VARMA (state space) models more often.

2 Vector autoregressive moving average models

We assume that the reader is familiar with basic concepts and results from the theory of stationary processes, which may, e.g., be found in Rozanov (1967), Hannan and Deistler (2012), and Brockwell and Davis (1991).

We consider VARMA models of the form

$$y_t = a_1 y_{t-1} + \cdots + a_p y_{t-p} + \epsilon_t + b_1 \epsilon_{t-1} + \cdots + b_q \epsilon_{t-q} \tag{1}$$

where $(\epsilon_t \mid t \in \mathbb{Z})$ is n-dimensional white noise[a] with a positive definite covariance matrix

$$\mathbb{E}\epsilon_t \epsilon_t' = \Sigma > 0 \tag{2}$$

and where $a_j \in \mathbb{R}^{n\times n}$, $j = 1, \ldots, p$ and $b_j \in \mathbb{R}^{n\times n}$, $j = 1, \ldots, q$ are real square matrices. We will always assume *stability*, i.e.,

$$\det(a(z)) \neq 0 \, \forall |z| \leq 1 \tag{3}$$

where $a(z) = I_n - a_1 z - \cdots - a_p z^p$ is the associated AR polynomial. The MA polynomial is defined as $b(z) = I_n + b_1 z + \cdots + b_q z^q$. In addition we assume that the so-called *inverse stability* or *strict miniphase* condition

$$\det(b(z)) \neq 0 \ \forall |z| \leq 1 \tag{4}$$

is satisfied. In Sections 3 and 7.1 we will also consider slightly more general models of the form

$$a_0 y_t = a_1 y_{t-1} + \cdots + a_p y_{t-p} + b_0 \epsilon_t + b_1 \epsilon_{t-1} + \cdots + b_q \epsilon_{t-q} \tag{5}$$

where $a_0 = b_0$ is a nonsingular matrix. Clearly by premultiplication both sides of the above equation by a_0^{-1} we get a model of the form (1). Note that due to the assumption $a_0 = b_0$ the noise (ϵ_t) is the same in (1) and (5). However, the coefficient matrices are different in general, but (for simplicity of notation) we will use the same symbols in (1) and (5). Given these assumptions, the VARMA system (1) has a unique stationary solution given by

$$y_t = \sum_{j \geq 0} k_j \epsilon_{t-j}. \tag{6}$$

The coefficients k_j are obtained by the power series expansion of the so-called *transfer function*

$$k(z) = a^{-1}(z) b(z) = \sum_{j \geq 0} k_j z^j \quad |z| \leq 1. \tag{7}$$

In the following z is used as a complex variable as well as for the backward shift on the integers $\mathbb{Z}$. Therefore, we can also write the process (y_t) in the form

[a]A process $(\epsilon_t \mid t \in \mathbb{Z})$ is called white noise if $\mathbb{E}\epsilon_t = 0$, $\mathbb{E}\epsilon_t \epsilon_s' = 0$ for $t \neq s$ and $\mathbb{E}\epsilon_t \epsilon_t'$ is finite and independent of t.

$$y_t = a^{-1}(z)b(z)\epsilon_t = k(z)\epsilon_t.$$

Due to stability and inverse stability, it can be shown that the above representation is the *Wold representation* of the VARMA process $(y_t \mid t \in \mathbb{Z})$ and $(\epsilon_t \mid t \in \mathbb{Z})$ are the *innovations* of (y_t), i.e., ϵ_{t+1} is the prediction error for the linear, least squares prediction $\hat{y}_{t+1}$ of y_{t+1} given the infinite past $(y_t, y_{t-1}, \ldots)$. See below for a more detailed discussion and Wold (1954) and Rozanov (1967).

Note that VARMA processes have mean zero. In practice therefore in a first step the data has to be demeaned before the VARMA modeling.

The transfer function $k(z) = a^{-1}(z)b(z)$ is a *rational* matrix (i.e., a matrix whose entries are rational functions of z). Vice versa any regular[b] process (y_t) where the transfer function corresponding to the Wold decomposition is rational has a VARMA representation (1). However, in general only the so-called *miniphase* condition

$$\det(b(z)) \neq 0 \; \forall |z| < 1 \tag{8}$$

holds rather than the strict miniphase condition (4). For more details see Section 3.

The sequence $(k_j \mid j \in \mathbb{N}_0)$ is called the *impulse response function* since it represents the impact of a unit pulse on the future values of the process. To say it in another way k_j reflects the influence of the "shock" ϵ_t on the future value y_{t+j}. The impulse response function and the transfer function are in a one-to-one relation given by (7). Note that by our assumptions we have $a(0) = a_0 = b_0 = b(0) = I_n$ and thus $k(0) = k_0 = I_n$.

For interpretation it is often convenient to normalize the error covariance as $\mathbb{E}\bar{\epsilon}_t\bar{\epsilon}_t' = I_n$ where $\bar{\epsilon}_t = H^{-1}\epsilon_t$ and $H^{-1}\Sigma(H^{-1})' = I_n$. The transformed errors $\bar{\epsilon}_t$ are called orthogonalized shocks and the correspondingly transformed impulse response function

$$\left(\bar{k}_j = k_j H \mid j \in \mathbb{N}_0\right) \tag{9}$$

is the so-called orthogonalized impulse response function. Clearly we have

$$y_t = \sum_{j \geq 0} k_j \epsilon_{t-j} = \sum_{j \geq 0} k_j H H^{-1} \epsilon_{t-j} = \sum_{j \geq 0} \bar{k}_j \bar{\epsilon}_{t-j} \tag{10}$$

The transformation matrix H is only unique up to postmultiplication by orthogonal matrices. From the point of view of interpretation the orthogonalized impulse response function has the advantage that the orthogonalized shocks are *instantaneously* and *serially* uncorrelated, i.e., $\mathbb{E}\bar{\epsilon}_{it}\bar{\epsilon}_{js} = 0$ for $i \neq j$ or $t \neq s$.

---

[b] A stationary process is called (linearly) regular if the h-step ahead prediction $\hat{y}_{t+h}$ for y_{t+h} from the infinite past $(y_s \mid s \leq t)$ converges to zero for h going to infinity. A process (y_t) is regular if and only if (y_t) has a causal MA(∞) representation $y_t = \sum_{j \geq 0} k_j \epsilon_{t-j}$ and if ϵ_t is obtained by a linear transformation of y_s, $s \leq t$.

R Demonstration 1 We assume that the reader has some basic knowledge in R (programming), see R Core Team (2015).[c,d]

The MTS package is an "all-purpose toolkit" for analyzing multivariate time series. It handles a wide range of models, e.g., VAR, ARMA models, multivariate volatility models, factor models and error-correction VAR models for co-integrated time series. However, here we will use this package only for modeling by VARMA models and, in particular, by VARMA models in echelon canonical form.

In addition we will use some own utility functions (in particular tools to switch between MTS models and dse objects). These functions are not thoroughly tested, so they have to be used with some care. In particular there is almost no check on the input parameters. The readers of this contribution are free to use this code and to adapt it according to their own preferences and needs. These data, including some data and an Rmd file which contains all the examples and demonstrations of this chapter, are available at https://github.com/WolfgangScherrer/VARMA-modeling.

We will often use a syntax like MTS::function (respectively, dse::function) to clearly indicate which package is used.

The package MTS, which is kind of companion toolbox for the text book (Tsay, 2014), uses a different notation for VARMA models

$$\phi_0 y_t = \phi_1 y_{t-1} + \cdots + \phi_p y_{t-p} + \phi_0 \epsilon_t - \theta_1 \epsilon_{t-1} - \cdots - \theta_q \epsilon_{t-q}$$

The AR and MA coefficients are collected in two matrices $\phi = (\phi_1, \ldots, \phi_p) \in \mathbb{R}^{n \times np}$ and $\theta = (\theta_1, \ldots, \theta_q) \in \mathbb{R}^{n \times nq}$.

The following R code shows how to construct a VARMA model and how to compute and plot (see Fig. 1) the (orthogonalized) impulse response function.

```
> library(MTS)        # load package
> source('tools.R')   # load utility functions
>
> phi0 = matrix(c( 1.0,    0,  0,
+                 -1.199,  1,  0,
+                 -0.638,  0,  1), byrow = TRUE, nrow= 3)
> phi = matrix(c(0.762,  0,      0,     -0.074,  0.137,  -0.313,
+               -0.142, -0.470,  0.543,  0,      0,       0,
+                0.920, -0.775,  0.064,  0,      0,       0),
+              byrow = TRUE, nrow= 3, ncol=6)
> theta = matrix(c( 0.694,-0.116,-0.150,-0.216, 0.269,-0.231,
+                  -0.540,-0.253, 0.708,  0,     0,     0,
```

[c]The computations were carried out with: R version 3.4.4 (2018-03-15), MTS: 1.0, dse: 2015.12.1, QZ: 0.1.6 and kableExtra 0.9.0.

[d]We advise the reader to start experimenting with ARMA models for relatively small dimensions $n \leq 3$ in order to obtain compact outputs and "nice looking" plots.

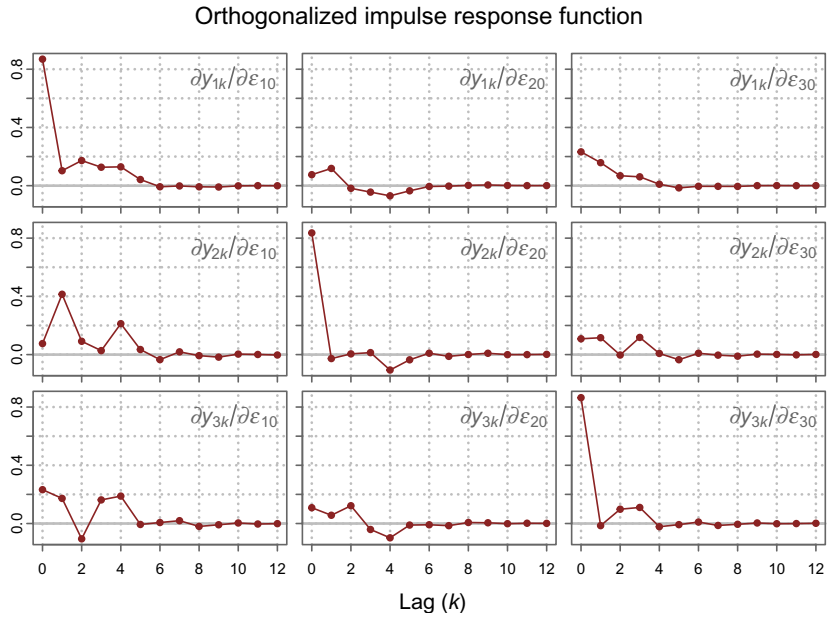

FIG. 1 This picture shows the orthogonalized impulse response function of the example VARMA model of R Demonstration (1).

```
+                  0.748,-0.760, 0.242,  0,     0,     0),
+                byrow = TRUE, nrow= 3, ncol=6)
> sigma = matrix(c(0.815,  0.154, 0.411,
+                  0.154,  0.716, 0.202,
+                  0.411,  0.202, 0.813),  nrow=3)
```

The (orthogonalized) impulse response function may be computed with the function `VARMAirf`. Note that `VARMAirf` assumes $\phi_0 = I$ ($a_0 = I$) and hence we reparametrize the model by $\phi \to \phi_0^{-1}\phi$ and $\theta \to \phi_0^{-1}\theta$.

The function `VARMAirf` returns a list with components `psi` and `irf`. The component `psi` is the matrix $(k_0, k_1, \ldots, k_l) \in \mathbb{R}^{n \times n(l+1)}$. The component `irf` is the matrix[e] $\left(\mathrm{vec}\left(\overline{k}_0\right), \mathrm{vec}\left(\overline{k}_1\right), \ldots, \mathrm{vec}\left(\overline{k}_l\right)\right) \in \mathbb{R}^{n^2 \times (l+1)}$, i.e., `irf` contains the desired orthogonalized impulse response coefficients. Note that the `MTS` package here uses two different methods to represent a 3-dimensional array by a (2-dimensional) matrix!

`VARMAirf` uses the symmetric square root of Σ (computed via an eigenvalue decomposition of Σ) and hence the lag zero coefficient $\overline{k}_0 = k_0 H = H = H'$ is symmetric.

[e] *vec* denotes the vectorization operator, i.e. $vec(X)$ is the mn-dimensional column vector obtained by stacking the columns of the (m,n)-dimensional matrix X.

`VARMAirf` always produces two plots (one for the (orthogonalized) impulse response function and one for the cumulative (orthogonalized) impulse response function). In order to be somewhat more flexible we have implemented a simple function (`plot3d`) for plotting 3-dimensional arrays like the impulse response function or the autocovariance function. The (i, j)-th panel plots the respective (i, j)-component of $\overline{k}_l$ as a function of the lag $l = 0,1,2\ldots$ and hence shows influence of the j-th "orthogonalized shock" $\overline{\epsilon}_{jt}$ on the i-th component $y_{i,t+l}$.

```
> k = MTS::VARMAirf(Phi = solve(phi0,phi),
+                   Theta = solve(phi0, theta),
+                   Sigma = sigma, lag = 12, orth = TRUE)
>
> plot3d(k$irf, dim = c(3,3,13),
+    main='orthogonalized impulse response function',
+    labels.ij="partialdiff*y[i_*k]/partialdiff*epsilon[j_*0]",
+    type='o', lty='solid', col='brown4', pch=19, cex=0.5)
```

For a VARMA model (1), the best linear least squares one-step ahead prediction for y_{t+1} given the infinite past y_t, y_{t-1}, … is given by

$$\hat{y}_{t+1} = a_1 y_t + \cdots + a_p y_{t+1-p} + b_1 \epsilon_t + \cdots + b_q \epsilon_{t+1-q} \tag{11}$$

This is a consequence of the projection theorem. This theorem says that the best approximation of an element in a Hilbert space by an element in a subspace is characterized by the fact that the approximation error is orthogonal to all elements of the subspace, see, e.g., Bachman and Narici (2000). In our context the underlying Hilbert space is the Hilbert space of square integrable (scalar) random variables. Due to the strict miniphase assumption (4) we can express ϵ_t as

$$\epsilon_t = b^{-1}(z)a(z)y_t = \sum_{j \geq 0} l_j y_{t-j}. \tag{12}$$

This representation is called the AR(∞) representation of the process. Due to (12) the prediction $\hat{y}_{t+1}$ is in fact a linear combination of the past and present values y_t, y_{t-1}, …. The prediction errors

$$\hat{u}_{t+1} = (y_{t+1} - \hat{y}_{t+1}) = \epsilon_{t+1} \tag{13}$$

are orthogonal to y_t, y_{t-1}, …. This shows that $\hat{y}_{t+1}$ is indeed the best linear prediction for y_{t+1} and that the ϵ_t's are the innovations of (y_t).

The more general h-step ahead prediction $\hat{y}_{t+h}$ for y_{t+h} given the infinite past and the corresponding prediction errors $\hat{u}_{t+h} = y_{t+h} - \hat{y}_{t+h}$ are

$$\hat{y}_{t+h} = \sum_{j=h}^{\infty} k_j \epsilon_{t+h-j} \tag{14}$$

$$\hat{u}_{t+h} = \sum_{j=0}^{h-1} k_j \epsilon_{t+h-j} \tag{15}$$

This again is a consequence of the projection theorem. In order to express the forecast $\hat{y}_{t+h}$ as a function of the past and present observations $(y_{t-j} \,|\, j \geq 0)$ we use the AR(∞) representation (12). Clearly for $h = 1$ this corresponds to (11). According to (15) the variance of the h-step ahead prediction error is given by

$$\Sigma_h = \mathbb{E}\hat{u}_{t+h}\hat{u}'_{t+h} = \sum_{j=0}^{h-1} k_j \Sigma k'_j \tag{16}$$

If we express this variance in terms of the orthogonalized impulse response coefficients (9) we obtain $\Sigma_h = \sum_{j=0}^{h-1} \overline{k}_j \overline{k}'_j$ and thus the variance of the i-th component of $\hat{u}_{t+h}$ is given by

$$\mathbb{E}\hat{u}^2_{i,t+h} = \sum_{j=0}^{h-1}\sum_{l=1}^{n} \overline{k}^2_{j,il} = \sum_{l=1}^{n} \underbrace{\left(\sum_{j=0}^{h-1} \overline{k}^2_{j,il}\right)}_{=:\sigma^h_{il}}$$

where $\overline{k}_{j,il}$ is the (i,l)-th entry of $\overline{k}_j$. The ratio

$$c^h_{il} = \frac{\sigma^h_{il}}{\sum_{m=1}^{n} \sigma^h_{im}}$$

is the fraction of the prediction error variance of the i-th component due to the l-th component of the (orthogonalized) shocks. Thus we have derived the so-called *forecast error variance decomposition* (FEVD).

R Demonstration 2 The forecast error variance decomposition[f] may be computed by the utility function `fevd`. This function takes as a main argument an arbitrary orthogonal impulse response function (e.g., computed by `ARMAirf`) and returns a list with two components. The first element `vd` is an $(n, n, h_{\max})$-dimensional array where the (i, j, h)-th entry is equal to c^h_{ij} and the second element v is an $(n, h_{\max})$-dimensional matrix where the (i, h)-element is the variance of the i-th component of the h-step ahead forecast error, i.e., $\sum_{m=1}^{n} \sigma^h_{im}$. The maximum forecast horizon $h_{\max}$ is determined by the length of the input.

A plot of this decomposition[g] may be obtained by `plotfevd`, see Fig. 2.

[f] It seems that the function `MTS::FEVdec` has a bug. Furthermore the orthogonalization scheme is "hardwired" (a Cholesky decomposition of Σ). Therefore, we have implemented our own version.

[g] The choice for the `series.names` will become clear later on.

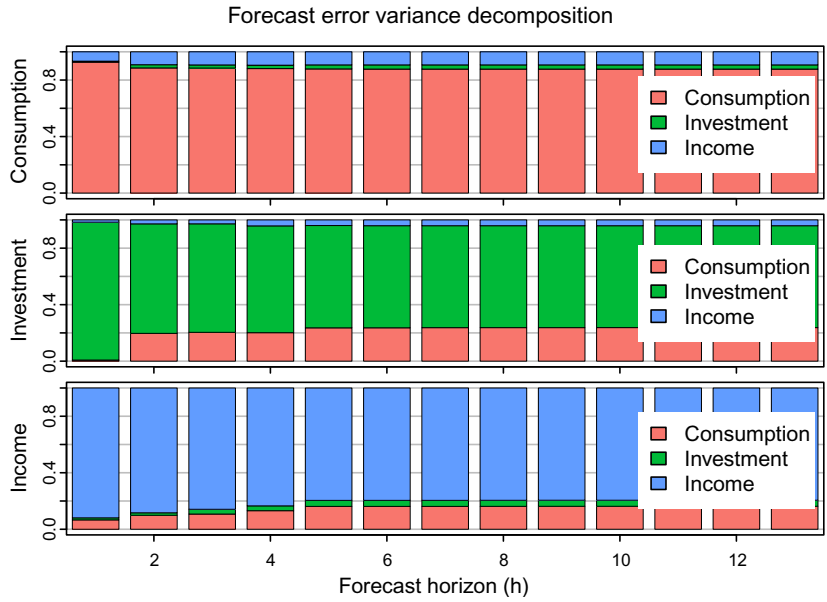

FIG. 2 This picture shows the forecast error variance decomposition computed in R Demonstration (2).

```
> out = fevd(k$irf, dim = c(3,3,13))
> plotfevd(out$vd,
+     series.names = c('consumption', 'investment', 'income'))
```

Note that $\mathbb{E}y_t = 0$ holds. The second moments of y_t (autocovariances), $\gamma_j = \mathbb{E}y_{t+j}y_t'$ may be computed by the so-called *generalized Yule–Walker* equations. First note that

$$\mathbb{E}\epsilon_{t-i}y_{t-j}' = \sum_{l\geq 0}\mathbb{E}\left[\epsilon_{t-i}\epsilon_{t-j-l}'\right]k_l' = \begin{cases} 0 & \text{for } i<j \\ \Sigma k_{i-j}' & \text{for } i\geq j \end{cases}.$$

Postmultiplying the VARMA Eq. (1) on both sides by y_{t-j}' and taking expectation gives

$$\gamma_j = a_1\gamma_{j-1} + \cdots + a_p\gamma_{j-p} + b_j\Sigma k_0' + \cdots + b_q\Sigma k_{q-j}' \quad \text{for } 0\leq j\leq q \tag{17}$$

$$\gamma_j = a_1\gamma_{j-1} + \cdots + a_p\gamma_{j-p} \quad \text{for } j>q \tag{18}$$

where we set $a_0 = b_0 = I_n$. For given parameters $(a_1,\ldots,a_p,b_1,\ldots,b_q,\Sigma)$ the above equation system (together with the symmetry condition $\gamma_j = \gamma_{-j}'$) has a unique solution $(\gamma_j \mid j\in\mathbb{Z})$.

The autocovariance sequence $(\gamma_j \mid j\in\mathbb{Z})$ defines the so-called *spectral density* of the process (y_t) via

$$f(z) = \frac{1}{2\pi} \sum_{j=-\infty}^{\infty} \gamma_j z^j, z \in \mathbb{C} \tag{19}$$

Conversely the autocovariance sequence is obtained from the spectral density through the formula

$$\gamma_j = \int_{-\pi}^{\pi} f\left(e^{-i\lambda}\right) e^{i\lambda j} d\lambda \tag{20}$$

In the literature the spectral density is often defined as a function of the frequencies $\lambda \in [-\pi, \pi]$, i.e., one considers the function $\overline{f}(\lambda) = f\left(e^{-i\lambda}\right)$. In particular the covariance γ_0 is equal to $\gamma_0 = \int_{-\pi}^{\pi} f(e^{-i\lambda}) d\lambda$. From this it can be shown that $f(e^{-i\lambda})\Delta$ is a measure for the size of the contributions of the oscillations with frequencies in the (small) frequency band $[\lambda, \lambda + \Delta]$ to the process (y_t). For a more detailed presentation, see Rozanov (1967).

The spectral density of a VARMA process may be directly expressed in terms of the VARMA parameters $(a_1, \ldots, a_p, b_1, \ldots, b_q, \Sigma)$. We have

$$f(z) = \frac{1}{2\pi} k(z) \Sigma k^*(z) = \frac{1}{2\pi} a^{-1}(z) b(z) \Sigma b^*(z) a^{-*}(z) \tag{21}$$

where

$$k^*(z) = \left[k\left(z^{-1}\right)\right]', \; a^*(z) = \left[a\left(z^{-1}\right)\right]', \; b^*(z) = \left[b\left(z^{-1}\right)\right]'$$
$$\text{and } a^{-*}(z) = \left[a^{-1}\left(z^{-1}\right)\right]'.$$

Hence, there is an amazingly simple relation between the VARMA parameters and the second moments of the process. Formula (21) also implies that $f(\cdot)$ is a *rational* matrix. As can be shown, conversely for any rational spectral density there is an underlying VARMA system, see Rozanov (1967) and Hannan and Deistler (2012).

An important special case is (vector) autoregressive (VAR) models

$$y_t = a_1 y_{t-1} + \cdots + a_p y_{t-p} + \epsilon_t$$

Of course the above statements apply also to this special case, but some things are much simpler.

The one-step ahead prediction is

$$\hat{y}_{t+1} = a_1 y_t + \cdots + a_p y_{t+1-p}$$

and the h-step ahead prediction may be recursively (in h) computed by

$$\begin{aligned} \hat{y}_{t+2} &= a_1 \hat{y}_{t+1} + a_2 y_t + \cdots + a_p y_{t+2-p} \\ \hat{y}_{t+3} &= a_1 \hat{y}_{t+2} + a_2 \hat{y}_{t+1} + a_3 y_t + \cdots + a_p y_{t+3-p} \\ &\vdots \end{aligned}$$

As a special case of (17) and (18) we obtain the *Yule–Walker equations*

$$\gamma_0 = a_1\gamma_{-1} + \cdots + a_p\gamma_{-p} + \Sigma \tag{22}$$

$$\gamma_j = a_1\gamma_{j-1} + \cdots + a_p\gamma_{j-p} \qquad \text{for } j > 0 \tag{23}$$

Hence, we may compute the autocovariance function without computing the impulse response function first.

R Demonstration 3 The autocovariance function of a VARMA process (y_t) may be computed with the function `ARMAcov`. This function returns a list with components `autocov` and `ccm`, where `autocov` stores the autocovariances, i.e., the matrix $(\gamma_0, \gamma_1, \ldots, \gamma_l) \in \mathbb{R}^{n \times n(l+1)}$ and the $(n \times n(l+1))$ matrix `ccm` contains the autocorrelations[h] $\rho_j = \text{diag}\,(\gamma_0)^{-1/2}\gamma_j\,\text{diag}\,(\gamma_0)^{-1/2}$, $j = 0, 1, \ldots, l$.

To be precise `VARMAcov` computes an approximation of the autocovariance function by the finite sum $\sum_{l=0}^{m} k_{j+l}\Sigma k_j'$, where the number m of lags used corresponds to the optional parameter `trun`.

Note that `VARMAcov` always prints the computed autocovariances and autocorrelations (called cross correlation matrices) and hence here we suppress the output of the next `R` block.

The plot of the autocorrelation function (see Fig. 3) is produced with the utility function `plot3d`.

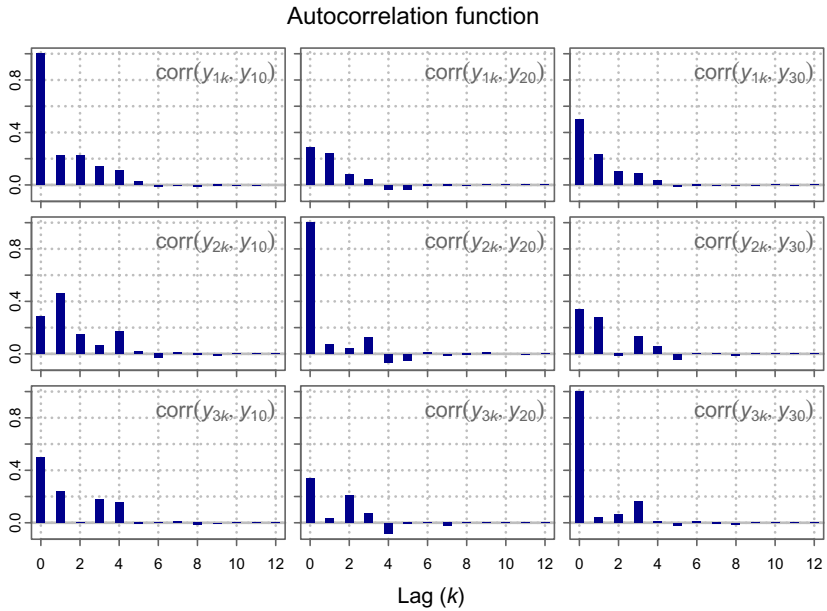

FIG. 3 This picture shows the autocorrelation function computed in R Demonstration (3).

[h]Here $\text{diag}\,(\gamma_0)^{-1/2}$ denotes the diagonal matrix with diagonal elements $(\gamma_{0,\,ii})^{-1/2}$, i.e., the reciprocals of the standard deviations of the components y_{it}.

```
> g = MTS::VARMAcov(Phi = solve(phi0,phi),
+                   Theta = solve(phi0,theta),
+                   Sigma = sigma, lag = 12)
> plot3d(g$ccm, dim = c(3,3,13),
+        main = 'auto correlation function',
+        labels.ij = "corr(list(y[i_*k],y[j_*0]))",
+        type = 'h', col = 'blue4', lwd = 5, lend = 1)
```

3 Identifiability of VARMA systems

One of the main problems in the statistical analysis of VARMA models is that the (population) second moments of the observed process (y_t) (i.e., its spectral density or equivalently its autocovariance function) do not uniquely determine the underlying VARMA system, unless additional assumptions have been imposed. This is the so-called problem of (non-) identifiability. Note also that in the Gaussian case the second moments completely describe the finite dimensional marginal distributions of the observed process (y_t). In this section we first describe a procedure to construct a unique VARMA system for a given spectral density and then a parametrization of VARMA systems which is based on this construction. The discussion follows Hannan and Deistler (2012) and the references given there.

Theorem 1 shows that there is a one-to-one relation between the spectral density $f(z)$ and the pair $(\Sigma, k(z))$ where Σ is the innovation variance and $k(z)$ is the transfer function corresponding to the Wold decomposition of the process.

Theorem 1 (*see, e.g., Rozanov, 1967; Deistler and Scherrer, 2018*). *For a given rational, nonsingular spectral density, f say, there exists a unique factorization as*

$$f(z) = \frac{1}{2\pi} k(z) \Sigma k^*(z)$$

where $k(z)$ is a rational function which has no poles for $|z| \leq 1$ and no zeros for $|z| < 1$ and where $k(0) = I_n$ holds. This transfer function corresponds to the Wold decomposition of the process.

Thus under our assumptions $k(z)$ and Σ are unique for a given spectral density and the question of identifiability reduces to the question under which assumptions $k(z) = a^{-1}(z)b(z)$ uniquely determines $a(z)$ and $b(z)$. Such a pair $(a(z), b(z))$ is called left matrix fraction description (LMFD) of the transfer function $k(z)$. In the following we will consider VARMA systems of the form (5).

Let us first repeat some basic concepts and results for polynomial matrices. A square polynomial matrix $u(z)$ is called common left factor of a pair of polynomial matrices $(a(z), b(z))$ if there exist polynomial matrices $\left(\overline{a}(z), \overline{b}(z)\right)$ such that $a(z) = u(z)\overline{a}(z)$ and $b(z) = u(z)\overline{b}(z)$. A square polynomial matrix

$u(z)$ is called unimodular if $\det u(z)$ is a nonzero constant. This condition is equivalent to the requirement that the inverse matrix $u^{-1}(z)$ is polynomial too. A pair $(a(z), b(z))$ is called left coprime if any common left factor is unimodular.

The degree of a polynomial, $c(z)$ say, is denoted by $\deg(c(z))$ and for a polynomial matrix $a(z)$ the degree, $\deg(a(z))$, is the maximum of the degrees of its entries. Correspondingly the row degrees of a polynomial matrix are the maximal degrees of the entries of the respective rows. A polynomial matrix $a(z)$ with row degrees $(d_i, i = 1, \ldots n)$ (where n is the number of rows of $a(z)$) is called row reduced if the so-called row end matrix

$$a_{[r]} := \lim_{z \to 0} \operatorname{diag}\left(z^{d_1}, \ldots, z^{d_n}\right) a\left(z^{-1}\right)$$

has full rank n.

Suppose we have given a rational transfer function $k(z)$, i.e., a (square) matrix where all entries are rational. We can easily construct a LMFD for $k(z)$, e.g., by setting $a(z) = c(z)I_n$ and $b(z) = c(z)k(z)$ where $c(z)$ is the least common multiple of the denominators of the entries $k_{ij}(z)$ of the matrix $k(z)$. However, this simple approach in general leads to VARMA models of very high orders (p, q). If we start from an arbitrary LMFD $k(z) = \overline{a}^{-1}(z)\overline{b}(z)$ then premultiplying $\overline{a}(z)$ and $\overline{b}(z)$ by a nonsingular polynomial matrix $u(z)$ leads to an equivalent LMFD for $k(z)$ since

$$(u(z)\overline{a}(z))^{-1}\left(u(z)\overline{b}(z)\right) = \overline{a}^{-1}(z)\overline{b}(z) = k(z).$$

Conversely if $u(z)$ is a common left factor of $\left(\overline{a}(z), \overline{b}(z)\right)$, i.e., $\left(\overline{a}(z), \overline{b}(z)\right) = u(z)(a(z), b(z))$ for some polynomial matrices $a(z)$ and $b(z)$, then $(a(z), b(z))$ is another LMFD of $k(z)$, in other words we may "cancel" common left factors. By canceling all nonunimodular common left factors one can construct a left coprime LMFD, $(a(z), b(z))$ say, of k. It can be shown that any other LMFD $\left(\overline{a}(z), \overline{b}(z)\right)$ then is of the form $\left(\overline{a}(z), \overline{b}(z)\right) = u(z)(a(z), b(z))$ where $u(z)$ is polynomial and that $\left(\overline{a}(z), \overline{b}(z)\right)$ is left coprime if and only if $u(z)$ is unimodular.

In the scalar case $(n = 1)$ coprimeness means that the polynomials $a(z)$ and $b(z)$ have no common roots. Together with the condition $a(0) = b(0) = 1$ then the LMFD $(a(z), b(z))$ is unique. Note also that the coprimeness condition implies that the degrees of $a(z)$ and $b(z)$ are minimal among all LMFDs of $k(z)$.

In the multivariate case $(n > 1)$, however, we still have the freedom to premultiply $(a(z), b(z))$ by a unimodular matrix and thus we need additional restrictions to get a unique LMFD.

In the following we describe a procedure to construct a VARMA system from a given rational transfer function $k(z) = \sum_{j \geq 0} k_j z^j$. Such procedures are called *realization* algorithms. The basic equation is

$$a(z)k(z) = b(z)$$

Using the normalization $k(0)=k_0=I_n$ this equation is rewritten as

$$a(z)(k(z)-I_n)=b(z)-a(z).$$

Note that $(b(z)-a(z))$ is a polynomial matrix of degree less than or equal to $r=\max(p,q)$. Therefore

$$a_0k_m-a_1k_{m-1}-\cdots-a_pk_{m-p}=0\ \ \forall m>r=\max(p,q) \tag{24}$$

Let us define the infinite dimensional (block) Hankel matrix of the transfer function as

$$H^{(k)}=\begin{pmatrix} k_1 & k_2 & k_3 & \cdots \\ k_2 & k_3 & k_4 & \cdots \\ k_3 & k_4 & k_5 & \cdots \\ \vdots & \vdots & \vdots & \end{pmatrix} \tag{25}$$

Then (24) can be written as

$$(a_r,a_{r-1},\ldots,a_1,-a_0,0,0,\ldots)H^{(k)}=0 \tag{26}$$

where we set $a_j=0$ for $p<j\le r$. Due to the Hankel structure this relation implies that $H^{(k)}$ has rank at most nr. Furthermore, it follows that the AR polynomial $a(z)$ is closely related to the left kernel of the Hankel matrix $H^{(k)}$.

Clearly $a(z)$ is not uniquely determined from (26). A unique solution is obtained as follows. We select the first linearly independent rows of $H^{(k)}$ which form a basis for its row space. Let $h(i,j)$ denote the j-th row in the i-th block row of the Hankel matrix $H^{(k)}$, i.e., $h(i,j)$ is the $((i-1)n+j)$-th row of $H^{(k)}$. Note that, if $h(i,j)$ is linearly dependent from the preceding rows, the same holds for $h(i+1,j)$. Therefore this basis can be described by a multi index $\nu=(\nu_1,\ldots,\nu_n)$ where ν_j is the number of block rows where the j-th row is selected as basis row. The ν_j's are called the Kronecker indices of the Hankel matrix $H^{(k)}$ or of the underlying transfer function $k(z)$. The sum $\nu_1+\cdots+\nu_n$ is equal to the rank of the Hankel matrix.

Now we express the rows $h(\nu_i+1,i)$, $i=1,\ldots,n$ as a linear combination of the *preceding* basis rows:

$$h(\nu_i+1,i)=-\sum_{j=1}^{i-1}a_{0,ij}h(\nu_i+1,j)+\sum_{j=1}^{n}\sum_{k=1}^{\nu_i}a_{k,ij}h(\nu_i+1-k,j) \tag{27}$$

Since we only use basis rows we set $a_{k,ij}=0$ if $h(\nu_i+1-k,j)$ is not an element of the basis, i.e., if $\nu_i+1-k>\nu_j$. Furthermore let $a_{0,jj}=1$, $a_{0,ij}=0$ for $j>i$ and $a_{k,ij}=0$ for $\nu_i<k\le p$ where $p=\max_{j=1,\ldots,n}\nu_j$ and define the AR polynomial $a(z)=a_0-a_1z-\cdots-a_pz^p$ with coefficient matrices $a_k=(a_{k,ij})_{i,j}$. By construction $b(z)=a(z)k(z)$ is polynomial and hence $(a(z),b(z))$ is a LMFD of the transfer function $k(z)$ as desired. This LMFD has the following properties (see, e.g., Hannan and Deistler, 2012)

1. the row degrees of $(a(z), b(z))$ are $\nu_1, \ldots, \nu_n$.
2. $a_{ij}(z)$ is divisible by $z^{n_{ij}}$ where $n_{ij} = \max(\nu_i + 1 - \nu_j, 1)$ for $j > i$ and $n_{ij} = \max(\nu_i + 1 - \nu_j, 0)$ for $j < i$. The matrix $a_0 = a(0)$ is a lower left triangular matrix with diagonal entries equal to 1.
3. $b(0) = a(0)$.
4. The pair $(a(z), b(z))$ is left coprime and row reduced.

A pair $(a(z), b(z))$ which satisfies these restrictions is said to be in echelon canonical form. Vice versa, as can be shown, if a pair $(a(z), b(z))$ is in echelon canonical form with row degrees $\nu_1, \ldots, \nu_n$ then the Kronecker indices of the corresponding transfer function $k(z) = a^{-1}(z)b(z)$ are equal to $\nu_1, \ldots, \nu_n$.

The set of all pairs $(a(z), b(z))$ corresponding to Kronecker indices $\nu = (\nu_1, \ldots, \nu_n)$ can be easily parametrized by the set of "free" coefficients $a_{k,ij}$ and $b_{k,ij}$ which are not restricted to be equal to one or zero. These free parameters are stacked to a vector $\theta \in \mathbb{R}^d$. Therefore we consider a set $\Theta_\nu \subset \mathbb{R}^d$ of system parameters, where

$$d = \sum_i \left(\sum_{j<i} \min\left(\nu_i + 1, \nu_j\right) + \sum_{j \geq i} \min\left(\nu_i, \nu_j\right) + n\nu_i \right) \tag{28}$$

The set Θ_ν is equal to the set of parameters $\theta \in \mathbb{R}^d$ which in addition satisfy the conditions:

1. $\det(a(z)) \neq 0$ for all z, $|z| \leq 1$ and $\det(b(z)) \neq 0$ for all z, $|z| \leq 1$.
2. the row degrees of $(a(z), b(z))$ are equal to $\nu_1, \ldots, \nu_n$ and $(a(z), b(z))$ is left coprime and row reduced.

As can be shown, Θ_ν is an open subset of $\mathbb{R}^d$. We now consider the mapping $\pi : \Theta_\nu \to U_\nu$ which attaches the transfer function $k(z) = a^{-1}(z)b(z)$ to a parameter vector $\theta \in \Theta_\nu$. The set U_ν is the set of all rational transfer functions with Kronecker indices $\nu = (\nu_1, \ldots, \nu_n)$ which have no poles and no zeros for $|z| \leq 1$. In order to introduce a topology for U_ν, we identify the transfer functions $k(z)$ with the impulse response $(k_j \mid j \in \mathbb{N})$ and endow the set $(\mathbb{R}^{n \times n})^{\mathbb{N}}$ with the product topology of the spaces $\mathbb{R}^{n \times n}$. The set U_ν is considered as a subset of $(\mathbb{R}^{n \times n})^{\mathbb{N}}$ and endowed with the corresponding relative topology. This topology is metrizable and convergence is equivalent to the convergence of the impulse response coefficients, i.e., a sequence of transfer functions $k^{(j)}(z) \in U_\nu$ converges to $k^{(0)}(z) \in U_\nu$ if and only if

$$k_s^{(j)} \to k_s^{(0)} \ \forall s \in \mathbb{N}.$$

As easily can be seen the function π is bijective and continuous. The continuity of the inverse map, which attaches the "free" parameters to a transfer function in U_ν according to Eq. (27), is a straightforward consequence of the fact that we only use basis rows to construct the parameters.

It is impossible to describe the set of all rational transfer functions with a finite dimensional parametrization. Here we have decomposed the set of rational transfer functions into disjoint pieces U_ν each of which can be parametrized continuously with a finite dimensional parameter space. There exist several other possibilities to break down the set of rational transfer functions into pieces and to parametrize these pieces.

In actual applications one has to estimate the Kronecker indices from data. In a multivariate setting this leads to a search over a substantial number of Kronecker indices. Therefore, often one considers a somewhat simplified parametrization where one only specifies the orders p, q and sets $p=q$. If $a_0=b_0=I_n$, $\mathrm{rk}([a_p,b_p])=n$ and $(a(z),b(z))$ are left coprime, then the parameters $(a_1,\ldots,a_p,b_1,\ldots,b_p)$ are identifiable. The corresponding set of transfer functions is the set U_ν with Kronecker indices $\nu=(p,\ldots,p)$. The corresponding parameter space Θ_ν has dimension $2n^2p$. Even for moderate dimensions n the number of free parameters to be estimated may be too high for a given sample size. Hence, it is also common practice to consider subsets where the degrees p and q are separately described, which gives a parameter space of dimension $n^2(p+q)$. Here the conditions for identifiability are: $(a(z),b(z))$ are left coprime, $a_0=b_0=I_n$ and $\mathrm{rk}(a_p)=n$ for $p>q$, $\mathrm{rk}(b_q)=n$ for $q>p$ and $\mathrm{rk}([a_p,b_q])=n$ for $p=q$. The price for these simplified parametrizations is that certain transfer functions cannot be described by this approach.

The Kronecker indices ν_i of the transfer function can also be computed from the block Hankel matrix of the autocovariance function (γ_k). This is an immediate consequence of the formula[i]

$$H^{(\gamma)}=\begin{pmatrix}\gamma_1&\gamma_2&\gamma_3&\cdots\\ \gamma_2&\gamma_3&\gamma_4&\cdots\\ \gamma_3&\gamma_4&\gamma_5&\cdots\\ \vdots&\vdots&\vdots&\end{pmatrix}=\begin{pmatrix}k_1&k_2&k_3&\cdots\\ k_2&k_3&k_4&\cdots\\ k_3&k_4&k_5&\cdots\\ \vdots&\vdots&\vdots&\end{pmatrix}\begin{pmatrix}\Sigma k_0'&0&0&\cdots\\ \Sigma k_1'&\Sigma k_0'&0&\cdots\\ \Sigma k_2'&\Sigma k_1'&\Sigma k_0'&\cdots\\ \vdots&\vdots&\vdots&\ddots\end{pmatrix} \tag{29}$$

Finally we want to remark that the Kronecker indices and the echelon canonical form can be computed from a finite number of impulse response coefficients. If s is an upper bound for the rank of the infinite Hankel matrix then it suffices to consider the finite Hankel matrix

$$H_{s+1}^{(k)}=\begin{pmatrix}k_1&k_2&\cdots&k_{s+1}\\ k_2&k_3&\cdots&k_{s+2}\\ \vdots&\vdots&\ddots&\vdots\\ k_{s+1}&k_{s+2}&\cdots&k_{2s+1}\end{pmatrix} \tag{30}$$

[i]Compare also (18) and (24).

R Demonstration 4 The utility function `impresp2PhiTheta` computes for a given impulse response the Kronecker indices and the corresponding VARMA model in echelon canonical form. The computations are based on a finite submatrix $H_{f,p}$ of the infinite dimensional Hankel matrix with f block rows and p block columns. The numbers f, p are determined from the length of the input sequence. The core computation is to determine a basis for the row space. This is done via a QR decomposition of the transpose $H'_{f,p}$ with the R function `qr`. The output of `impresp2PhiTheta` is a list with components `Phi, Theta, Phi0` (these matrices contain the AR/MA parameters), `kidx` (the vector of Kronecker indices), `Hrank` (the (computed) rank of the Hankel matrix $H'_{f,p}$) and `Hpivot` (as returned by `qr()`). Note that the first `Hrank` elements of the vector `Hpivot` contain the indices of the basis rows of $H_{f,p}$.

The function `MTS::Kronspec` determines the zero/one restrictions imposed by the echelon canonical for given Kronecker indices (`kdx`) and prints a nice representation of these restrictions (for `output=TRUE`).

```
> out = impresp2PhiTheta(k$psi)
> out$kidx # Kronecker indices
[1] 2 1 1
> # display the corresponding AR/MA restrictions
> junk = MTS::Kronspec(out$kidx)
Kronecker indices: 2 1 1
Dimension: 3
Notation:
 0: fixed to 0
 1: fixed to 1
 2: estimation
AR coefficient matrices:
     [,1] [,2] [,3] [,4] [,5] [,6] [,7] [,8] [,9]
[1,]    1    0    0    2    0    0    2    2    2
[2,]    2    1    0    2    2    2    0    0    0
[3,]    2    0    1    2    2    2    0    0    0
MA coefficient matrices:
     [,1] [,2] [,3] [,4] [,5] [,6] [,7] [,8] [,9]
[1,]    1    0    0    2    2    2    2    2    2
[2,]    2    1    0    2    2    2    0    0    0
[3,]    2    0    1    2    2    2    0    0    0
>
> all.equal(cbind(out$Phi0, out$Phi, out$Theta),
+           cbind(phi0,  phi, theta))
[1] TRUE
```

The Kronecker indices are (2,1,1) and hence the rank of the Hankel matrix is $2+1+1=4$ and the rows 1,2,3,4 (i.e., the first 4 rows) of H form a basis. The number of "free parameters" is 24. The last statement of the above `R`

code shows that the VARMA model we have started with is in echelon canonical form.

R Demonstration 5 The package `dse` centers on linear, time invariant ARMA models and state space models. One of the nice features of this package is the unified approach to handle these two model classes. The `dse` package uses an object oriented approach (with the S3 class system) and implements object classes for models, data sets and estimated models. We start with discussing VARMA models. Note that `dse` uses yet another convention for the sign of the AR/MA parameters:

$$a_0 y_t + a_1 y_{t-1} + \cdots + a_p y_{t-p} = b_0 \epsilon_t + b_1 \epsilon_{t-1} + \cdots + b_q \epsilon_{t-q}$$

(V)ARMA models are represented by `ARMA` objects (which are special `TSmodel` objects). Note that the `ARMA` model class may represent more general models, in particular models with exogenous inputs (i.e., VARMAX models) and models with a trend component. In addition note that a_0 and b_0 may be different. However, here we will stick to the simple model above and assume that $a_0 = b_0$.

The AR parameters $a_j \in \mathbb{R}^{n \times n}$ are stored in the $(p+1, n, n)$-dimensional array `A`, where the i-th slot `A[i, , ]` corresponds to the matrix $a_{i-1} \in \mathbb{R}^{n \times n}$. Analogously the MA parameters b_j are stored in the $(q+1, n, n)$-dimensional array `B`.

In order to be able to easily switch between `MTS` and `dse` models and tools we have implemented two utility functions `PhiTheta2ARMA` and `ARMA2PhiTheta`.

The following code chunk loads the `dse` package library and converts the above VARMA model to a `dse::ARMA` object. In addition we check that we can reconstruct the `Theta, Phi` parameters:

```
> library(dse)
Loading required package: tfplot
Loading required package: tframe
Attaching package: 'dse'
Die folgenden Objekte sind maskiert von 'package:stats':

  acf, simulate
>
> arma = PhiTheta2ARMA(Phi = phi, Theta = theta, Phi0 = phi0,
+        output.names = c('consumption','investment','income'))
> arma

A(L) =
1-0.762L1+0.074L2  0-0.137L2  0+0.313L2
-1.199+0.142L1   1+0.47L1   0-0.543L1
-0.638-0.92L1   0+0.775L1   1-0.064L1
```

```
B(L) =
1-0.694L1+0.216L2 0+0.116L1-0.269L2 0+0.15L1+0.231L2
-1.199+0.54L1  1+0.253L1  0-0.708L1
-0.638-0.748L1  0+0.76L1  1-0.242L1
>
> junk = ARMA2PhiTheta(arma, normalizePhi0 = FALSE)
> all.equal(cbind(phi, theta, phi0),
+           cbind(junk$Phi, junk$Theta, junk$Phi0))
[1] TRUE
```

The stability and the miniphase assumption now may be checked with the function `dse::polyrootDet(a))` which computes the roots of the determinant of a polynomial matrix $a(z)=a_0+a_1z+\cdots+a_pz^p$ with coefficients which are stored in the 3-dimensional array `a`.

```
> # check the stability assumption
> min(abs(polyrootDet(arma$A)))>1
[1] TRUE
> # check the (strict) miniphase assumption
> min(abs(polyrootDet(arma$B)))>1
[1] TRUE
```

The "dse" package contains a number of useful utilities for polynomial matrices (e.g., `characteristicPoly, companionMatrix, polydet, ...`).

4 State space models

State space models are an alternative way to describe processes with a rational spectral density. We consider models of the form

$$x_{t+1} = Ax_t + B\epsilon_t$$
$$y_t = Cx_t + \epsilon_t$$

where (ϵ_t) is white noise with a positive definite variance $\Sigma = \mathbb{E}\epsilon_t\epsilon_t'$, x_t is an unobserved random vector called state and $A \in \mathbb{R}^{s\times s}$, $B \in \mathbb{R}^{s\times n}$, $C \in \mathbb{R}^{n\times s}$ are parameter matrices. We always assume *stability*, i.e.,

$$\lambda_{\max}(A) < 1 \tag{31}$$

and *inverse stability* (also called *strict miniphase* assumption)

$$\lambda_{\max}(A - BC) < 1. \tag{32}$$

Here $\lambda_{\max}(X)$ denotes the maximum of the moduli of the eigenvalues of a square matrix X. Given these assumptions there exists a unique stationary solution of this state space system

$$x_t = (I_s - Az)^{-1}zB\epsilon_t = \sum_{j\geq 0} A^jB\epsilon_{t-1-j} \tag{33}$$

$$y_t = Cx_t + \epsilon_t = \Big(C(I_s - Az)^{-1}zB + I_n\Big)\epsilon_t = \epsilon_t + \sum_{j>0} CA^{j-1}B\epsilon_{t-j} \tag{34}$$

Given the strict miniphase assumption it is easy to see that the "inverse" system is

$$x_{t+1} = (A - BC)x_t + By_t = \sum_{j\geq 0}(A - BC)^j By_{t-j} \tag{35}$$

$$\epsilon_t = -Cx_t + y_t = y_t - \sum_{j>0} C(A - BC)^{j-1}By_{t-j} \tag{36}$$

Therefore (34) is the Wold decomposition of the process (y_t) and the ϵ_t's are the innovations for (y_t). It is immediate to see that the h-step ahead prediction for y_{t+h} given the infinite past $(y_s \mid s \leq t)$ is

$$\hat{y}_{t+h} = CA^{h-1}x_t \tag{37}$$

Clearly the transfer function

$$k(z) = C(I_s - Az)^{-1}zB + I_n \tag{38}$$

is rational and correspondingly the spectral density of (y_t) is rational. As can be shown every rational (nonsingular) spectral density can be represented (realized) by a state space system of the above form.

The impulse response coefficients, i.e., the coefficients of the MA(∞) representation $y_t = \sum_{j\geq 0} k_j \epsilon_{t-j}$, respectively, the coefficients of the power series expansion of the transfer function $k(z) = \sum_{j\geq 0} k_j z^j$, immediately follow from (34)

$$k_j = \begin{cases} I_n & \text{for } j = 0 \\ CA^{j-1}B & \text{for } j > 0 \end{cases}$$

The state sequence $(x_t \mid t \in \mathbb{Z})$ is an AR(1) process and the variance $P = \mathbb{E}x_t x_t'$ is the (unique) solution of a so-called Lyapunov equation

$$P = \mathbb{E}x_{t+1}x_{t+1}' = \mathbb{E}(Ax_t + B\epsilon_t)(Ax_t + B\epsilon_t)' = APA' + B\Sigma B'$$

The autocovariance function of (y_t) is given by

$$\gamma_j = \begin{cases} CPC' + \Sigma & \text{for } j = 0 \\ CA^{j-1}M & \text{for } j > 0 \end{cases}$$

where $M = \mathbb{E}x_{t+1}y_t' = APC' + B\Sigma$.

R Demonstration 6 State space models are implemented as `SS` objects in `dse`. However, `dse` uses a different naming convention, i.e., $A \to F$, $B \to K$, and $C \to H$. To convert the above VARMA model to a state space model (in innovation form), we may use the function[j] `toSS`. The function `is.innovSS` checks whether the input is an "innovation form state space" object.

[j]Note that the function `toSS` does not work for ARMA(p, q) models with $q > p$!

```
> ss = dse::toSS(arma)
> ss

 F =
     [,1] [,2] [,3]      [,4]      [,5]      [,6]
[1,]    0    0    0 -0.074000  0.137000 -0.313000
[2,]    0    0    0 -0.088726  0.164263 -0.375287
[3,]    0    0    0 -0.047212  0.087406 -0.199694
[4,]    1    0    0  0.762000  0.000000  0.000000
[5,]    0    1    0  0.771638 -0.470000  0.543000
[6,]    0    0    1  1.406156 -0.775000  0.064000

H =
     [,1] [,2] [,3] [,4] [,5] [,6]
[1,]    0    0    0    1    0    0
[2,]    0    0    0    0    1    0
[3,]    0    0    0    0    0    1

 K =
         [,1]      [,2]      [,3]
[1,] 0.142000 -0.132000 -0.082000
[2,] 0.170258 -0.158268 -0.098318
[3,] 0.090596 -0.084216 -0.052316
[4,] 0.068000  0.116000  0.150000
[5,] 0.479532 -0.077916  0.014850
[6,] 0.215384  0.059008 -0.082300
>
> dse::is.innov.SS(ss)
[1] TRUE
```

We here get a model with a state space dimension $s=6$.

The utility function `SSirf` and `SScov` compute the impulse response and autocovariance function. The outputs returned by these function have the same structure as the output of the corresponding `MTS` function, e.g., `SScov` returns a list with components autocov and `ccm` where both of them are matrices of dimension $(n, n(l+1))$. To compute the state variance P here the function[k] `lyap` is used.

The following code computes the impulse response function and the ACF of the state space model. Of course, since this state space model and the VARMA model above describe the same process the output must be identical to the output we have computed above.

[k]Alternatively one may use the function `dse::Riccati`. However, for the model we use here for testing purposes the (noniterative version) of `Riccati` stops with an error message. The function `lyap` first computes a Schur decomposition of the state transition matrix A (respectively, F). To this end the `QZ` package has to be installed.

```
> k.ss = SSirf(ss, Sigma = sigma, lag.max = 12, orth = TRUE)
> all.equal(k, k.ss, check.attributes = FALSE)
[1] TRUE
>
> g.ss = SScov(ss, Sigma = sigma, lag.max = 12)
> all.equal(g, g.ss)
[1] TRUE
```

5 Identifiability of state space models

Similar to the VARMA case, the relation between the parameters (A,B,C,Σ) and the second moments of the process (y_t) is not one-to-one. Due to the spectral factorization Theorem 1 again it suffices to consider the relation between the state space parameters (A,B,C) and the corresponding transfer function (38).

A state space system (33), (34) is called *controllable* if the controllability matrix

$$\mathcal{C}_s = \left(B, AB, \ldots, A^{s-1}B\right) \in \mathbb{R}^{s\times ns}$$

has rank s, it is called *observable* if the observability matrix

$$\mathcal{O}_s = \left(C', A'C', \ldots, \left(A^{s-1}\right)'C'\right)' \in \mathbb{R}^{ns\times s}$$

has rank s. A state space system is called *minimal* if the state dimension s is minimal among all state space systems describing the same transfer function $k(z) = C(I_s - Az)^{-1}zB + I_n$. It can be shown that a system is minimal if and only it is controllable and observable, see Kalman (1963). This minimality condition is analogous to the left coprime condition for VARMA systems, since it rules out redundant parameters. Even under minimality the state space parameters (A,B,C) are not uniquely determined from the transfer function. As can be shown (Kalman, 1963), two minimal state space systems (A,B,C) and $\left(\tilde{A},\tilde{B},\tilde{C}\right)$ describe the same transfer function if and only if there exists a nonsingular matrix $T \in \mathbb{R}^{s\times s}$ such that

$$\tilde{A} = TAT^{-1}, \tilde{B} = TB, \tilde{C} = CT^{-1} \tag{39}$$

holds.

R Demonstration 7 The `dse` tools `observability` and `reachability` compute the singular values of the observability, respectively, of the reachability matrices.

```
> sv0 = dse::observability(ss)
> svR = dse::reachability(ss)
Singular values of reachability matrix for noise: 0.7443343
0.4589284 0.3712107 0.2409465 5.985088e-17 2.107293e-17
```

```
> signif(rbind(svO,svR),4)
      [,1]   [,2]   [,3]   [,4]      [,5]      [,6]
svO 2.9990 2.6280 1.7220 0.9879 9.206e-01 4.450e-01
svR 0.7443 0.4589 0.3712 0.2409 5.985e-17 2.107e-17
```

Inspecting these singular values shows that the model is not reachable and hence is *not minimal*.

One possibility to achieve a minimal model is to use a "`balancing and truncation`" scheme. The `dse` package offers the function `balanceMittnik(model,n)` to this end. The (optional) parameter n is the desired state dimension. Here we use `n=4` based on the fact that only 4 of the reachability singular values are significantly greater than zero. The following code computes a (minimal) state space model with a state space dimension 4 and checks that this model really is an equivalent description of the process (y_t):

```
> ssb = dse::balanceMittnik(ss, n=4)
> ssb

 F =
           [,1]       [,2]       [,3]       [,4]
[1,] -0.1327857  0.1813460  0.3120305  0.3311117
[2,] -0.7805031  0.1090193  0.2904374 -0.1461123
[3,]  0.0715883 -0.6736422  0.6819328  0.2285732
[4,] -0.2674132 -0.4954136 -0.3930425 -0.2987692

 H =
          [,1]         [,2]       [,3]         [,4]
[1,]  0.02484203  0.07963300 -0.4383435  0.184895023
[2,] -0.56526277  0.02998341 -0.2627269  0.183926049
[3,] -0.19050836 -0.45709445 -0.3611779 -0.001188339

K =
            [,1]       [,2]       [,3]
[1,] -0.76035926  0.2878963  0.1622077
[2,]  0.01509112 -0.1126601  0.2372431
[3,] -0.21419099 -0.1738510 -0.1596715
[4,] -0.04079621  0.2276468  0.3100379

> k.ss = SSirf(ss, Sigma = sigma, lag.max = 12, orth = TRUE)
> all.equal(k, k.ss, check.attributes = FALSE)
[1] TRUE
```

To check the stability assumption we may use the function `stability`. For the miniphase assumption there is no corresponding tool. However, it is not difficult to check this assumption "manually" by computing the eigenvalues of $(A-BC)$ (respectively, using the `dse` notation of $(F-KH)$):

```
> dse::stability(ssb)
The system is stable.
[1] TRUE
attr(,"roots")
          Eigenvalues of F        moduli
[1,] -0.2859941+0.5749215i 0.6421272+0i
[2,] -0.2859941-0.5749215i 0.6421272+0i
[3,]  0.4656927+0.3115639i 0.5603051+0i
[4,]  0.4656927-0.3115639i 0.5603051+0i
>
> lambda = eigen(ssb$F - ssb$K %*% ssb$H)$values
> cat(ifelse((max(abs(lambda))<1),
+            'The system is strictly miniphase\n',
+            'The system is not strictly miniphase\n'))
The system is strictly miniphase
```

The set of all minimal systems (A,B,C) satisfying the stability and miniphase assumption can be embedded in $\mathbb{R}^{s^2+2ns}$, where the equivalence classes corresponding to the nonsingular matrices T in (39) are manifolds of dimension s^2.

In the following we present a procedure (see Ho and Kalman, 1966) to construct a unique state space system from a given rational square transfer function $k(z)$ with $k(0)=I_n$. As we have seen in Section 3, a transfer function, which has a power series expansion $k(z)=\sum_{j=0}^{\infty}k_j z^j$ which converges in a disc around $z=0$, is rational if and only if the corresponding Hankel matrix $H^{(k)}$ (as defined in (25)) has finite rank. Now let $S\in\mathbb{R}^{s\times\infty}$ be a matrix such that the rows of $SH^{(k)}$ form a basis for the row space of $H^{(k)}$ and determine the parameters (A,B,C) by solving the following equations

$$S\begin{pmatrix} k_2 & k_3 & k_4 & \cdots \\ k_3 & k_4 & k_5 & \cdots \\ \vdots & \vdots & \vdots & \end{pmatrix} = ASH^{(k)},\quad B = S\begin{pmatrix} k_1 \\ k_2 \\ k_3 \\ \vdots \end{pmatrix},$$

$$(k_1\ k_2\ k_3\ \cdots) = CSH^{(k)}$$

It is immediate to see that these parameters (A,B,C) are unique for given S and that $k_j=CA^{j-1}B$, $j>0$ holds as desired. Furthermore the system (A,B,C) can be shown to be minimal. The matrix S is unique up to premultiplication with nonsingular matrices $T\in\mathbb{R}^{s\times s}$. If we replace S by $\tilde{S}=TS$ then we obtain $\left(\tilde{A}=TAT^{-1},\tilde{B}=TB,\tilde{C}=CT^{-1}\right)$ by this procedure in accordance to the above discussions. If we want to construct a unique state space system we thus have to make a unique choice for a basis of the row space of the Hankel matrix. One possibility is to chose the first linearly independent rows as basis. This leads to state space systems in echelon canonical form. Similar to the VARMA case one can also construct a parametrization of the set U_ν of

rational transfer functions with given Kronecker indices by state space systems, see, e.g., (Hannan and Deistler, 2012, chapter 2). Here we have only presented one approach for parametrization of state space systems. Alternative parameterizations are, e.g., overlapping parameterizations of the set of transfer functions with a given rank of the Hankel matrix (see Hannan and Deistler, 2012) or data driven local coordinates introduced by McKelvey et al. (2004) and Ribarits et al. (2004).

In the special case where the Hankel matrix has rank $s \geq n$ and where the first s rows are linearly independent we may set $S = (I_s, 0_{s\times\infty})$. Then we obtain a system of the form

$$A = \begin{pmatrix} 0_{s-n\times n} & I_{s-n} \\ A_{21} & A_{22} \end{pmatrix}, B = \begin{pmatrix} B_1 \\ B_2 \end{pmatrix}, C = (I_n \;\; 0_{n\times s-n})$$

with $B_1 \in \mathbb{R}^{s-n\times n}$, $B_2 \in \mathbb{R}^{n\times n}$, $A_{21} \in \mathbb{R}^{n\times n}$, and $A_{22} \in \mathbb{R}^{n\times s-n}$. It can be shown that this case is "generic" within the set of transfer functions where the Hankel matrix has rank s. The corresponding Kronecker indices are given by $\nu = (p, \ldots, p)$ if $s = pn$ and $\nu = (p, \ldots, p, p-1, \ldots, p-1)$ if $s = (p-1)n+k$, $0 < k < n$.

Similar to the VARMA case it suffices to consider a finite dimensional part (30) of the Hankel matrix for the above construction.

R Demonstration 8 The utility function `impresp2SS` implements this Ho-Kalman procedure, i.e., computes a state space model in echelon form for a given impulse response function. By means of the functions `impresp2SS` and `impresp2PhiTheta` we can easily switch forth and back between VARMA and state space models.

```
> sse = impresp2SS(k$psi, type = 'echelon')$ss
> sse

 F =
       [,1]   [,2]   [,3]  [,4]
[1,]  0.000  0.000  0.000 1.000
[2,] -0.142 -0.470  0.543 1.199
[3,]  0.920 -0.775  0.064 0.638
[4,] -0.074  0.137 -0.313 0.762

 H =
     [,1] [,2] [,3] [,4]
[1,]    1    0    0    0
[2,]    0    1    0    0
[3,]    0    0    1    0
 K =
  [,1]     [,2]  [,3]
[1,]  0.068000  0.116000 0.15000
[2,]  0.479532 -0.077916 0.01485
```

```
[3,]  0.215384  0.059008 -0.08230
[4,]  0.193816 -0.043608 0.03230
> k.ss = SSirf(sse, Sigma = sigma, lag.max = 12, orth = TRUE)
> all.equal(k, k.ss, check.attributes = FALSE)
[1] TRUE
```

6 Maximum likelihood estimation

In this section we consider the Gaussian maximum likelihood estimation of VARMA models or state space models. This is in our context the benchmark procedure, which gives consistent and asymptotically efficient estimates. If the innovations are Gaussian, then the same is true for the observations $\mathbf{y}_T = (y_1', \ldots, y_T')'$ and thus $-2T^{-1}$ times the log likelihood (omitting a constant term) is given by

$$L_T(\theta, \Sigma) = T^{-1} \log \det \Gamma_T(\theta, \Sigma) + T^{-1} \mathbf{y}_T' \Gamma_T^{-1}(\theta, \Sigma) \mathbf{y}_T \tag{40}$$

where

$$\Gamma_T(\theta, \Sigma) = \begin{pmatrix} \gamma_0(\theta, \Sigma) & \gamma_{-1}(\theta, \Sigma) & \cdots & \gamma_{-T+1}(\theta, \Sigma) \\ \gamma_1(\theta, \Sigma) & \gamma_0(\theta, \Sigma) & \cdots & \gamma_{-T+2}(\theta, \Sigma) \\ \vdots & \vdots & \ddots & \vdots \\ \gamma_{T-1}(\theta, \Sigma) & \gamma_{T-2}(\theta, \Sigma) & \cdots & \gamma_0(\theta, \Sigma) \end{pmatrix}$$

and $\gamma_k(\theta, \Sigma)$ denote the population covariances if θ is the parameter describing the AR and MA polynomials $a(z)$ and $b(z)$ or the corresponding state space parameters A, B, C and where Σ is the innovation variance. Note that the likelihood function depends on θ only via the transfer function $k(z)$ and this transfer function may described either by a VARMA model or a state space model.

The ML estimate $(\hat{\theta}, \hat{\Sigma})$ is the minimizer of the negative log likelihood function (40) over a suitably defined parameter space $(\Theta \times S) \subset (\mathbb{R}^d \times \mathbb{R}^{n \times n})$, where S denotes the set of all symmetric and positive definite $n \times n$ matrices. The required regularity conditions on the parameter space are as follows:

- The parameter space Θ is an open subset of $\mathbb{R}^d$.
- The systems corresponding to $\theta \in \Theta$ satisfy the stability and inverse stability assumption.
- The mapping attaching the transfer function to the parameter θ is injective (identifiability). This mapping as well as its inverse is continuous.

Such parameterizations are, e.g., the echelon form parametrization and the overlapping parametrization as described in Sections 3 and 5. Of course in most applications the Kronecker indices are not known a priori and have to be determined by a model selection step, see Section 8.

If there is a $\theta_0 \in \Theta$ which corresponds to the true VARMA/state space system generating the data then the ML estimate can be shown to be consistent,

asymptotically normal and asymptotically efficient under general conditions, see Hannan and Deistler (2012).

A main problem in VARMA estimation is that in general there is no explicit formula describing this minimizer as a function of the data and thus numerical optimization procedures have to be used. The above formula (40) is not suited for numerical implementation, since it would need the inverse (and determinant) of a high dimensional ($nT \times nT$) matrix. There are several numerically efficient algorithms to compute the log likelihood function which exploit the block Toeplitz structure of Γ_T. In particular we mention the Kalman filter (see, e.g., Deistler and Scherrer, 2018) which gives the factorization of the joint density function $g(y_1, \ldots, y_T)$ as a product of the conditional probability density functions $g(y_t | y_{t-1}, \ldots, y_1)$. The negative log likelihood function (up to additive and multiplicative constants) is equal to

$$L_T(\theta, \Sigma) = \frac{1}{T} \sum_{t=1}^{T} \left(\log \det \left(\Sigma_{t|t-1} \right) + \mathrm{tr} \left(\Sigma_{t|t-1} e_{t|t-1} e'_{t|t-1} \right) \right)$$

where $e_{t|t-1}$ is the prediction error of the least squares prediction for y_t given the observations $y_1, \ldots, y_{t-1}$ and $\Sigma_{t|t-1}$ is the variance of $e_{t|t-1}$. As noted above these quantities may be efficiently computed by the Kalman filter.

For t going to infinity we have

$$\left(\Sigma_{t|t-1} - \Sigma \right) \to 0, \left(e_{t|t-1} - \epsilon_t \right) \to 0 \text{ and } \left(\epsilon_t - e_t \right) \to 0$$

where (compare (12))

$$e_t = \sum_{j=0}^{t-1} l_j y_{t-j}$$

For the VARMA case the residuals e_t may be computed by the recursion

$$e_t = y_t - a_1 y_{t-1} - \cdots - a_p y_{t-p} - b_1 e_{t-1} - \cdots - b_q e_{t-q} \quad \text{for } t = 1, 2, \ldots$$

where we set $e_t = y_t = 0 \in \mathbb{R}^n$ for $t \leq 0$. An analogous recursion is used for state space models.

This leads to an approximation of the negative log likelihood function

$$\log \det (\Sigma) + \mathrm{tr} \left(\Sigma^{-1} \frac{1}{T} \sum_{t=1}^{T} e_t e'_t \right)$$

For given $k(z)$, i.e., given θ, the minimizer for Σ is the sample variance of the residuals e_t

$$\hat{\Sigma} = \frac{1}{T} \sum_{t=1}^{T} e_t e'_t$$

Plugging this estimate into the (approximate) log likelihood function leads to

$$\log\det\left(\hat{\Sigma}\right)+n \tag{41}$$

This approximation of the likelihood function is often used for the estimation of VARMA or state space systems, see, e.g., the `dse` package by Paul Gilbert, the `MTS` package by Tsay and the MATLAB® `SYSID` toolbox by Ljung. Ljung (1999) called the corresponding method *prediction error* method (PEM).

As can be shown estimates obtained from this approximation are asymptotically equivalent to the exact ML estimates in the sense that $\sqrt{T}$ times the difference of the estimates converges to zero. In finite sample there is some evidence, that the exact ML estimates perform (slightly) better.

In a number of packages neither the stability nor the inverse stability condition are built in the parametrization so that in finite sample the estimation may lead to nonstable or nonminiphase systems.

The negative log likelihood function may have several local minima and therefore we have to start the numerical optimization with a "good" initial estimate in order to mitigate this problem. Nevertheless, also using good initial estimates the optimization may get stuck in a local minimum of the likelihood function.

The optimization of the likelihood function is far from trivial, in particular, if the dimension of the parameter space is large. The numerical burden may be reduced if one performs only one Gauss–Newton step commencing from an initial estimate. This is justified by the fact, that this simplified estimate is asymptotically equivalent to a full ML estimate, if we start with a consistent initial estimate. This result does not mean that in finite samples the estimate could not be improved by iterating the optimization steps (such as Gauss–Newton steps).

An alternative is to use the EM (expectation–maximization) algorithm for optimization of the likelihood function as proposed by Shumway and Stoffer (1982) and Gibson and Ninness (2005).

7 Initial estimates

As discussed in the previous section, ML estimation requires consistent initial estimates. Here we describe two popular procedures, one for the VARMA case and one for the state space case, to obtain such initial estimates.

7.1 Estimation of VARMA models—The Hannan, Rissanen, Kavalieris procedure

We describe an algorithm for estimating the VARMA parameters for given p, q which was first proposed for the scalar case by Åström and Mayne (1982) and analyzed by Hannan and Rissanen (1982) and further investigated by Hannan et al. (1986).

A general problem in VARMA estimation is, that the past ϵ_t's are not directly observed, in contrary to the past y_t's. The basic idea is to replace the past ϵ_t's by suitable estimates and then to obtain estimates for the parameters a_j, b_j by a least squares type formula. In order to obtain the estimates for the innovations ϵ_t we use a "long" autoregression, i.e., an AR($\tilde{p}$) model with a large $\tilde{p}$. In other words we approximate the AR(∞) representation (12) by an AR($\tilde{p}$) model with a large $\tilde{p}$. For given $\tilde{p}$, the AR parameters and the innovation variance are estimated by solving the Yule–Walker equations (23), (22), where we replace the population autocovariances γ_k by their sample counterparts

$$\begin{aligned}
\hat{\gamma}_k &= \frac{1}{T}\sum_{t=1}^{T-k}(y_{t+k}-\overline{y})(y_t-\overline{y})', && \text{for } T>k\geq 0\\
\hat{\gamma}_k &= 0\in\mathbb{R}^{n\times n}, && \text{for } k\geq T\\
\hat{\gamma}_k' &= \hat{\gamma}_{-k}', && \text{for } k\leq 0
\end{aligned}$$

Here T denotes the sample size and $\overline{y}=\frac{1}{T}\sum_{t=1}^{T}y_t$ is the sample mean. Let $\tilde{\Sigma}_{\tilde{p}}$ denote the estimate for Σ for a given order $\tilde{p}$. For estimation of $\tilde{p}$ we use the AIC information criterion (see, e.g., Akaike (1974))

$$AIC(\tilde{p}) = \log\det\tilde{\Sigma}_{\tilde{p}} + \frac{2}{T}\left(\tilde{p}n^2\right). \tag{42}$$

The estimate for $\tilde{p}$ is the minimizer of $AIC(\tilde{p})$ within a given range $0\leq\tilde{p}\leq\tilde{p}_{\max}$. The idea of information criteria like (42) is to formulate a trade off between the fit of the model as described by $\log\det\tilde{\Sigma}_{\tilde{p}}$ and the dimension of the space of free system parameters required to obtain this fit, which here is $d(\tilde{p})=\tilde{p}n^2$. The AIC criterion used in this step may be shown to be optimal for estimating AR(∞) systems, see Shibata (1980).

From the estimates $\tilde{p}$ and the AR($\tilde{p}$) parameters $\tilde{a}_j$, $j=1,\ldots,\tilde{p}$ we get estimates of the innovations by

$$\tilde{\epsilon}_t = y_t - \tilde{a}_1 y_{t-1} - \cdots - \tilde{a}_{\tilde{p}} y_{t-\tilde{p}}, \quad t>\tilde{p}$$

In the next step we consider the "regression"

$$y_t = a_1 y_{t-1} + \cdots + a_p y_{t-p} + b_1\tilde{\epsilon}_{t-1} + \cdots + b_q\tilde{\epsilon}_{t-q} + u_t \tag{43}$$

and estimate the parameters a_j, b_j by ordinary least squares. Strictly speaking (43) is not a regression, since the error u_t is not orthogonal to the regressors $y_{t-1},\ldots,y_{t-p},\tilde{\epsilon}_{t-1},\ldots,\tilde{\epsilon}_{t-q}$. In this step a priori restrictions such as zero restrictions on the entries of the parameter matrices may easily be incorporated. In particular, it is also straightforward to estimate ARMA systems in echelon canonical form.

This regression step could also be iterated, i.e., given estimates $\hat{a}_i$, $\hat{b}_i$ we can reestimate the innovations by

$$\tilde{\epsilon}_t^{(2)} = y_t - \hat{a}_1 y_{t-1} - \cdots - \hat{a}_p y_{t-p} - \hat{b}_1 \tilde{\epsilon}_{t-1}^{(2)} - \cdots - \hat{b}_q \tilde{\epsilon}_{t-q}^{(2)} \tag{44}$$

and then use these "new" estimates $\tilde{\epsilon}_t^{(2)}$ for an additional regression step.

As an estimate for the innovation variance Σ we either use the sample variance of the residuals of the regression step (43) or the sample variance of the estimates of the innovations computed by (44).

As can be shown, see, e.g., Hannan et al. (1986), this Hannan–Rissanen–Kavalieris procedure gives consistent estimates, provided that (p, q) have been chosen appropriately. However, it does not give asymptotically efficient estimates.

7.2 Estimation of state space models—The CCA subspace method

First note that if we had observed the state in (33) and (34), then the estimation of the parameters A, B, C would be rather easy. There exist several procedures for estimating the state, but here we only consider one. In order to motivate the procedure called CCA (see, e.g., Larimore, 1983) consider the following equations. By Eqs. (33) and (34) it follows that for $h \geq 1$

$$y_{t+h} = CA^{h-1}x_{t+1} + \epsilon_{t+h} + CB\epsilon_{t+h-1} + \cdots + CA^{h-2}B\epsilon_{t+1}$$

From the inverse system (35) and (36), we can represent the state x_{t+1} as a function of the past $y_t's$

$$x_{t+1} = By_t + (A-BC)By_{t-1} + \cdots + (A-BC)^{h-1}By_{t+1-h} + (A-BC)^h x_{t+1-h}$$

Combining these two equations we get

$$\begin{pmatrix} y_{t+1} \\ y_{t+2} \\ \vdots \\ y_{t+h} \end{pmatrix} = \begin{pmatrix} C \\ CA \\ \vdots \\ CA^{h-1} \end{pmatrix} \left(B, (A-BC)B, \ldots, (A-BC)^{h-1}B \right) \begin{pmatrix} y_t \\ y_{t-1} \\ \vdots \\ y_{t+1-h} \end{pmatrix} +$$

$$+ \begin{pmatrix} \epsilon_{t+1} \\ CB\epsilon_{t+1} + \epsilon_{t+2} \\ \vdots \\ CA^{h-2}B\epsilon_{t+1} + \cdots + \epsilon_{t+h} \end{pmatrix} + \begin{pmatrix} C \\ CA \\ \vdots \\ CA^{h-1} \end{pmatrix} (A-BC)^h x_{t+1-h}$$

Denoting the sum of the last two terms on the right-hand side by u_t leads to the "regression"

$$y_t^+ = \underbrace{\mathcal{O}_h \mathcal{K}_h}_{=:\beta} y_t^- + u_t = \beta y_t^- + u_t$$

where

$$y_t^+ = \left(y'_{t+1}, y'_{t+2}, \ldots, y'_{t+h}\right)'$$
$$y_t^- = \left(y'_{t-1}, \ldots, y'_{t+1-h}\right)'$$
$$\mathcal{O}_h = \left(C', A'C', \ldots, \left(A^{h-1}\right)'\right)'$$
$$\mathcal{K}_h = \left(B, (A-BC)B, \ldots, (A-BC)^{h-1}B\right).$$

Strictly speaking this is not a regression since the error u_t is not orthogonal to the regressors $(y'_t, \ldots, y_{t+1-h})'$. However, for large h this can be ignored.[1] The coefficient matrix β is of rank s (for $h \geq s$) and thus we estimate $\beta = \mathcal{O}_h\mathcal{K}_h$ by a reduced rank regression technique (see Anderson, 1951) as follows: Let

$$\hat{\Gamma}^+ = \frac{1}{T}\sum_{t=1}^{T} y_t^+ \left(y_t^+\right)', \hat{\Gamma}^- = \frac{1}{T}\sum_{t=1}^{T} y_t^- \left(y_t^-\right)' \text{ and } \hat{H} = \frac{1}{T}\sum_{t=1}^{T} y_t^+ \left(y_t^-\right)'$$

The OLS estimate

$$\hat{\beta} = \hat{H}\left(\hat{\Gamma}^-\right)^{-1}$$

typically has full rank (nh). Therefore, we use a weighted singular value decomposition to obtain an estimate of rank s. Let

$$\left(\hat{\Gamma}^+\right)^{-1/2} \hat{\beta} \left(\hat{\Gamma}^-\right)^{1/2} = U\Sigma V' = U_1\Sigma_1 V_1' + U_2\Sigma_2 V_2'$$

where Σ_1 is the diagonal $(s \times s)$ matrix consisting of the s largest singular values and Σ_2 is the diagonal matrix composed of the remaining $(nh - s)$ singular values. Here $X^{1/2}$ denotes a symmetric square root of a positive definite matrix X and $X^{-1/2}$ denotes the inverse, if it exists. The final estimate for β then is

$$\tilde{\beta} = \left(\hat{\Gamma}^+\right)^{1/2} \left(U_1\Sigma_1 V_1'\right) \left(\hat{\Gamma}^-\right)^{-1/2}$$

This leads to a state estimation of the form

$$\tilde{x}_{t+1} = S\tilde{\beta} y_t^-$$

where $S \in \mathbb{R}^{s \times nh}$ is chosen such that $S\mathbb{E}y_t^+ y_t^-$ has full row rank s. Common choices for S are, e.g., $S = U_1'$ or $S = (I_s, 0)$ in the case that the first s rows of the Hankel matrix $\mathbb{E}y_t^+ y_t^-$ are linearly independent. Given $\tilde{x}_t$ we estimate C by a regression of y_t onto $\tilde{x}_t$ and the matrices A and B by regressing $\tilde{x}_{t+1}$ onto $\tilde{x}_t$ and the residuals of the former regression.

It can be shown that this procedure gives consistent estimates (see, e.g., Deistler et al., 1995) and asymptotically efficient estimates (see Bauer, 2005) under suitable assumptions, in particular requiring that h tends to infinity with

[1] Note that ϵ_{t+j} is orthogonal to y_t^- for $j > 0$ and that $(A-BC)^h x_{t+1-h}$ converges to zero for $h \to \infty$ due to the strict miniphase assumption.

the sample size going to infinity. Typically h is determined by estimating the order of an AR approximation of the system as described in (42). Despite the fact that CCA is asymptotically efficient, in many cases it is only used as an initial estimate for likelihood estimation. For example, the MATLAB® system identification toolbox (ML Sysid TB, n.d. (R2017b)) proposes such an approach.

There are several other subspace methods, see, e.g., for early references Aoki and Havenner (1991), Larimore (1983), Van Overschee and De Moor (1994), and Verhaegen (1994).

8 Model selection

In practice, in most cases specification parameters such as (p, q) for the VARMA case, or the state space dimension s for the state space case, or the Kronecker indices ν are not known a priori and have to be estimated from data.

The most used estimators are derived from so-called information criteria. For simplicity of presentation we discuss this procedure for the case of Kronecker indices only, see Sections 3 and 5. The so-called BIC (Bayesian information criterion), see Schwarz (1978), is defined as

$$BIC(\nu) := \min_{\theta \in \Theta_\nu, \Sigma > 0} (L_T(\theta, \Sigma)) + \frac{\log(T)}{T} d(\nu) \tag{45}$$

where $\nu = (\nu_1, \ldots, \nu_n)$ is the vector of Kronecker indices, $\Theta_\nu \subset \mathbb{R}^{d(\nu)}$ is the corresponding parameter space as described in Sections 3 and 5, and $d(\nu)$ is the dimension of the parameter space Θ_ν, see (28). The estimate for ν is obtained by minimizing $BIC(\nu)$ over the set of all Kronecker indices up to a certain (user defined) upper bound, $\nu_i \leq \nu_{\max}$. Again the idea of BIC is to formulate a trade off between fit and complexity. As has been shown in Hannan (1980), (Hannan and Deistler, 2012, chapter 5) under suitable assumptions this procedure gives a consistent estimator for the Kronecker indices, whereas the AIC criterion $AIC(\nu) = \min(L_T(\theta, \Sigma)) + \frac{2}{T} d(\nu)$ typically over-estimates the Kronecker indices. This is a consequence of the smaller weight of the penalty term $d(\nu)$.

It should be noted, however, that the estimation of the order does not come without a price in the sense that "naive" confidence bounds, obtained from the asymptotic distribution of the ML-estimates for known ν are not reliable in the case where ν has to estimated, see Leeb and Pötscher (2005). Consistent estimation of Kronecker indices also does not necessarily lead to optimal prediction, see, e.g., Shibata (1980).

Due to the high numerical burden of the ML estimation and the fact that the information criterion often has to be computed for a substantial number of parameter spaces, the minimum of the negative log likelihood in (45) is often approximated by the negative log likelihood evaluated at a suitable (initial) estimate of the system leading to

$$\log\det\left(\hat{\Sigma}\right) + \frac{\log T}{T} d(\nu).$$

An alternative method for estimation of the Kronecker indices is based on the fact that the linear dependency relations of the Hankel matrix of the impulse response are the same as for the Hankel matrix of the autocovariances, see (29). Tsay (2014) has proposed a recursive test procedure, using canonical correlations, to estimate the Kronecker indices from data.

Once the Kronecker indices (or the orders (p,q)) have been estimated it may still be advisable to further reduce the dimension of the parameter space, e.g., by "hard-thresholding" where insignificant coefficients are set to zero. This approach is implemented in the MTS package in the procedures `refVARMA` and `refKronfit`.

R Demonstration 9 In order to illustrate the estimation of VARMA and state space models we use the following three quarterly time series from the "FRED (Federal Reserve Economic Data)" database (https://fred.stlouisfed.org):

- `DPIC96` Real Disposable Personal Income
- `GPDIC1` Real Gross Private Domestic Investment
- `PCECC96` Real Personal Consumption Expenditures

The variables are measured in Billions of Chained 2009 Dollars. We consider the quarterly growth rates (i.e., the differences of the log values) and demean and scale the growth rates such that the sample variance is equal to one. The whole date set is split into two parts: The data from 1958 to 1991 (136 observations) are used for the estimation of the models and the data from 1992 to the end of 2017 (104 observations) are used for the comparison of the models (in terms of their predictive power).

We use a matrix y to store the data (for estimation with the `MTS` tools) and a `TSdata` object which is used for the estimation by `dse` tools. For a plot of these data, see Fig. 4.

```
> # read the data into an R data.frame object
> data = read.delim(file = 'prCoIn_Quarterly.txt',header=TRUE)
> data$DATE = as.Date(data$DATE)
>
> # compute the quarterly growth rates
> data$consumption = c(NA,diff(log(data$PCECC96)))
> data$investment = c(NA,diff(log(data$GPDIC1)))
> data$income = c(NA,diff(log(data$DPIC96)))
>
> # skip the data before 1958
> d = as.POSIXlt(data$DATE)
> data = data[d >= as.POSIXlt('1958-01-01'), ]
> d = d[d >= as.POSIXlt('1958-01-01')]
>
```

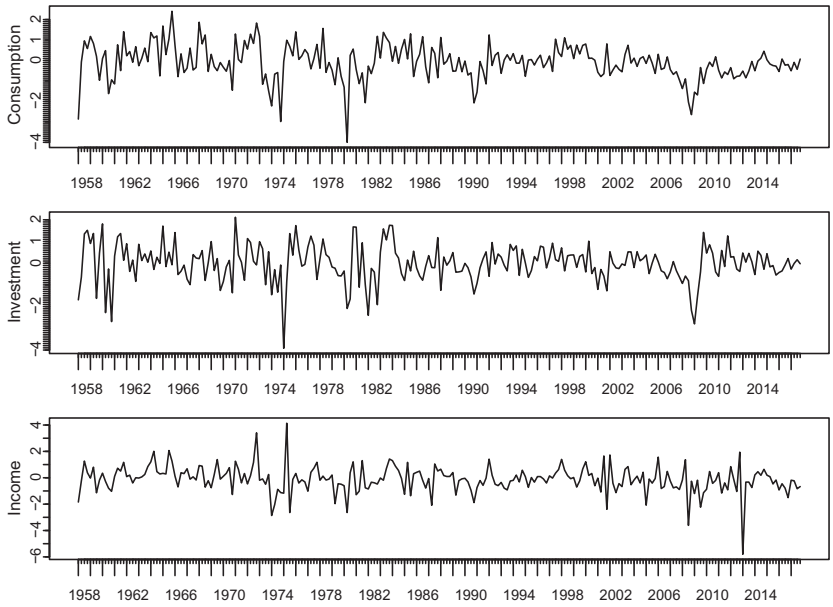

FIG. 4 Plot of the "consumption" data set, see R Demonstration (9).

```
> # collect the time series to be analyzed in the matrix "y"
> y = cbind(data$consumption,data$investment,data$income)
> colnames(y) = c('consumption','investment','income')
> rownames(y) = paste(format(data$DATE,'%Y'),
+                     quarters(data$DATE),sep=' ')
>
> n = ncol(y)  # number of variables
> T.obs = nrow(y) # total sample size
> T.est = sum(d < as.POSIXlt('1992-01-01')) # estimation sample
>
> # demean and scale the data
> y = scale(y, center = colMeans(y[1:T.est,]),
+              scale = sqrt((T.est-1)/T.est)*apply(y[1:T.est,],
+                                        MARGIN = 2, FUN = sd))
>
> data.start = c(1900 + d[1]$year, d[1]$mon %/% 3 +1)
> data.end = c(1900 + d[T.obs]$year, d[T.obs]$mon %/% 3 +1)
> est.end = c(1900 + d[T.est]$year, d[T.est]$mon %/% 3 +1)
>
> # construct "dse" TSdata objects
> sample = dse::TSdata(output = y)
> sample = tframed(sample,list(start=data.start, frequency=4))
```

```
> estimation.sample = tfwindow(sample, end = est.end)
>
> # plot the data
> par(oma = c(0,0,0,0), mar = c(2,2,1,0)+0.1, tcl = -0.2,
+     mgp = c(1.25, 0.15, 0), cex.main = 1, cex.axis = 0.75)
> tfplot(sample)
```

R Demonstration 10 In order to evaluate the "fit" of a given model on a data set we may use the `dse::l()` function which in particular computes the (log) likelihood of the model. The output of this command is a `TSestModel` which is an object which stores the model, the data and the estimation results.

```
> arma = l(arma, estimation.sample)
> summary(arma)
neg. log likelihood = 506.1354 sample length = 136
     consumption investment income
RMSE 0.9304949 1.232342 0.7825009
ARMA:
inputs:
outputs: consumption investment income
    input dimension = 0 output dimension = 3
    order A = 2 order B = 2 order C =
     26 actual parameters  6 non-zero constants
    trend not estimated.
```

Forecasts may be computed by the dse functions forecast, `horizon-Forecasts` and `featherForecasts`. The function `dse::forecast` computes the out-of-sample forecasts for a given model and sample. In the example below we compute the forecasts for 2018–2020, i.e., for forecast horizons $h = 1, 2, \ldots, 12$. The corresponding `MTS` function (for estimated VARMA models) is `MTS::VARMApred`. We also show how to produce a plot of the forecast. However, to save space the plot is not included here.

```
> # compute the "out-of-sample" forecast for 3 years
> z = suppressWarnings(forecast(arma,data=sample,horizon=4*3))
>
> tfplot(z,start=c(2010,1)) # plot the end of the sample
> # extract the "out-of-sample" forecast for 2018
> window(z$forecast[[1]], end=c(2018,4))
   Series 1  Series 2  Series 3
2018 Q1 -0.28900612 -0.04738126 -0.15876220
2018 Q2 -0.14087985 -0.19181475 -0.32920728
2018 Q3 -0.04276265 -0.11987410 -0.02930487
2018 Q4 0.05460322 0.11196985 0.08652213
```

The function `dse::horizonForecasts` computes the h-step ahead forecasts for all time points within a sample. The forecasts are "aligned" with the original data, such that it is easy to compute the forecast errors. In the code below we use the state space model `sse` (which is equivalent to the VARMA model `arma`) in order to show that the syntax is independent of the object class. The optional parameter `discard.before` means that predictions based on data up to this time point should be discarded (i.e., are set to zero).

```
> z = suppressWarnings(horizonForecasts(sse, sample,
+             horizons = c(1,4), discard.before = T.est))
> # plot a subsample
> tfplot(z, start = c(1991,1), end = c(1995,4))

> # extract the forecasts/errors for "income"
> junk = cbind(sample$output[,3], t(z$horizonForecasts[,3]),
+        sample$output[,c(3,3)] - t(z$horizonForecasts[,3]))
> colnames(junk) = c(colnames(z$data$output)[3],
+        t(outer(c('pred h=','err h='),c(1,4),paste,sep="")))
> rownames(junk) = rownames(y)
> round(junk[(T.est-3):(T.est+8),],4)
        income pred h=1 pred h=4  err h=1 err h=4
1991 Q1 -0.7447   0.0000   0.0000 -0.7447 -0.7447
1991 Q2 -0.2067   0.0000   0.0000 -0.2067 -0.2067
1991 Q3 -0.5754   0.0000   0.0000 -0.5754 -0.5754
1991 Q4 -0.0239   0.0000   0.0000 -0.0239 -0.0239
1992  Q1 1.4128  -0.2206  0.0000  1.6334  1.4128
1992  Q2 0.1890   0.4206   0.0000 -0.2316  0.1890
1992 Q3 -0.4929 -0.2781   0.0000 -0.2148 -0.4929
1992 Q4 -0.5981   0.1570  -0.3521 -0.7550 -0.2460
1993 Q1 -0.3536   0.2433   0.2014 -0.5970 -0.5550
1993 Q2 -0.7867 -0.3533  -0.1527 -0.4334 -0.6340
1993 Q3 -0.9186   0.0992   0.0316 -1.0178 -0.9502
1993 Q4 -0.2453   0.0967   0.1324 -0.3421 -0.3778
```

R Demonstration 11 Of course it is easy to estimate autoregressive models with both packages. Here we use `dse::estVARar` which is based on the Yule–Walker equations. The order is estimated by the AIC information criterion. (The estimated order is $p = 2$.) This VAR model will serve as a kind of benchmark model.

```
> model.VAR = dse::estVARXar(estimation.sample, aic = TRUE)
> summary(model.VAR)
neg. log likelihood = 510.5801 sample length = 136
```

```
     consumption investment income
RMSE 0.9402565 0.8669979 0.930077
ARMA: model estimated by estVARXar
inputs:
outputs: consumption investment income
     input dimension = 0  output dimension = 3
     order A = 2   order B = 0  order C =
      18 actual parameters  6 non-zero constants
     trend not estimated.
```

The `dse` package offers a number of functions to estimate state space models. The package author suggest to use `bft` (brute force technique), so we follow this advice. The main idea of this technique (which is a particular *subspace algorithm*) is as follows. First estimate a "long" AR model, i.e., with a high order p. Next convert this model to a state space model and then use a model reduction technique to construct a state space model of the desired order s. This procedure is repeated for a number of AR orders p and state space dimensions s and finally the model which is the best in terms of an information criterion is returned.

The following `R` code estimates two state space models, the first one uses "BIC" as selection criterion (and returns a state model of order $s=1$) and the second one uses "AIC" (which results in $s=3$).

```
> model.BFTbic = dse::bft(estimation.sample,
+                     criterion = 'tbic', verbose = FALSE)
> summary(model.BFTbic)
neg. log likelihood = 531.3362 sample length = 136
     consumption investment income
RMSE 0.9710178 0.8961374 0.9726227
innovations form state space: nested model a la Mittnik
inputs:
outputs: consumption investment income
   input dimension = 0 state dimension = 1 output dimension = 3
   theoretical parameter space dimension = 6
   7 actual parameters 0 non-zero constants
   Initial values not specified.
>
> model.BFTaic = dse::bft(estimation.sample,
+                     criterion = 'taic', verbose = FALSE)
> summary(model.BFTaic)
neg. log likelihood = 513.1865 sample length = 136
     consumption investment income
RMSE 0.9312625 0.8933566 0.9239323
innovations form state space: nested model a la Mittnik
```

```
inputs:
outputs: consumption investment income
   input dimension = 0 state dimension = 3 output dimension = 3
   theoretical parameter space dimension = 18
   27 actual parameters 0 non-zero constants
   Initial values not specified.
```

Next we try to enhance these models with maximum likelihood. The corresponding function is `dse::estMaxLik` which takes an "initial" model as main argument. Note that both `model.BFTbic` and `model.BFTaic` are `TSestModel` objects and thus contain the (estimation) data.

```
> model.BFTbicML = estMaxLik(model.BFTbic)
> summary(model.BFTbicML)
neg. log likelihood = 528.7699  sample length = 136
     consumption investment income
RMSE 0.9584305 0.8920208 0.9712616
innovations form state space: Estimated with max.like/optim
( converged ) from initial model: nested model a la Mittnik
inputs:
outputs: consumption investment income
   input dimension = 0 state dimension = 1 output dimension = 3
   theoretical parameter space dimension = 6
   7 actual parameters 0 non-zero constants
   Initial values not specified.
>
> model.BFTaicML = estMaxLik(model.BFTaic)
> summary(model.BFTaicML)
neg. log likelihood = 509.5183  sample length = 136
     consumption investment  income
RMSE 0.9301279 0.8753634 0.9262648
innovations form state space: Estimated with max.like/optim
( converged ) from initial model: nested model a la Mittnik
inputs:
outputs: consumption investment income
   input dimension = 0 state dimension = 3 output dimension = 3
   theoretical parameter space dimension = 18
   27 actual parameters 0 non-zero constants
   Initial values not specified.
```

The function `dse::estMaxLik` can also deal with VARMA models. However, one first has to find an appropriate initial estimate and we could not a find a suitable `dse` function for this purpose. (Of course one could first estimate a state space model and then convert this to a VARMA model.) `estMaxLik` uses a simple (and flexible) scheme to deal with constraints. It simply treats coefficients (of the initial model) which are zero or one as fixed. One may also impose more complicated constraints with the function `dse::fixConstants`.

However, it is not possible to impose a constraint like $a_0 = b_0$ and thus it is not clear how to estimate VARMA model in echelon canonical form.

For these reasons, we use the `MTS` package for estimation of VARMA models. A VARMA(p,q) model (without further structure) may be estimated with `MTS::VARMA` (or `MTS::VARMAcpp` where the computation of the likelihood is implemented in C++). The initial estimate is computed by the HRK algorithm. The output of this function is a list which in particular contains the estimated AR (`$Phi`) and MA (`$Theta`) parameters. In the following R code an ARMA(1,1) model is estimated. In order to be able to easily compare this ARMA model with the above estimated state space models we convert the result of `MTS::VARMA` to a `dse::TSestModel` object.

```
> out = MTS::VARMACpp(y[1:T.est,], p=1, q=1,
+                     include.mean = FALSE, details = FALSE)
Number of parameters: 18
initial estimates: 0.3866 0.2347 -0.1595 0.4197 -0.0479 0.2851
                   0.7382 -0.2265 -0.1591 -0.3344 -0.2207 0.4608
                   0.0312 -0.0298 -0.31 -0.5184 0.1628 0.1548
Par. lower-bounds: -0.0731 -0.105 -0.5698 -0.0226 -0.3748 -0.1097
                   0.2556 -0.5831 -0.5899 -0.8385 -0.619 -0.0013
                   -0.4539 -0.4132 -0.7546 -1.0476 -0.2554 -0.3303
Par. upper-bounds: 0.8462 0.5743 0.2508 0.862 0.279 0.6799
                   1.2207 0.1301 0.2717 0.1696 0.1777 0.9228
                   0.5162 0.3535 0.1346 0.0108 0.581 0.6398
Final Estimates:  0.2272581 0.3050181 0.2507603 0.2214819
                 0.08512909 -0.1097409 0.255627 0.1300933
                 -0.3924509 -0.158361 0.1776507 0.02029747
                 -0.2315791 -0.1815723 -0.02620168 -0.03041847
                 -0.1641312 0.3666428
Warning in sqrt(diag(solve(Hessian))): NaNs wurden erzeugt

Coefficient(s):
             Estimate  Std. Error  t value  Pr(>|t|)
consumption   0.22726     0.21633    1.051   0.2935
investment    0.30502     0.16703    1.826   0.0678 .
income        0.25076     0.17576    1.427   0.1537
consumption   0.22148          NA       NA       NA
investment    0.08513          NA       NA       NA
income       -0.10974     0.11572   -0.948   0.3430
consumption   0.25563     0.26996    0.947   0.3437
investment    0.13009          NA       NA       NA
income       -0.39245     0.22166   -1.771   0.0766 .
             -0.15836     0.25571   -0.619   0.5357
              0.17765     0.18415    0.965   0.3347
              0.02030     0.16113    0.126   0.8998
             -0.23158     0.10343   -2.239   0.0252 *
```

```
              -0.18157          NA       NA       NA
              -0.02620     0.11180   -0.234   0.8147
              -0.03042     0.20751   -0.147   0.8835
              -0.16413          NA       NA       NA
               0.36664     0.19471    1.883   0.0597 .
---
Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
---
Estimates in matrix form:
AR coefficient matrix
AR( 1 )-matrix
      [,1]   [,2]   [,3]
[1,] 0.227  0.3050  0.251
[2,] 0.221  0.0851 -0.110
[3,] 0.256  0.1301 -0.392
MA coefficient matrix
MA( 1 )-matrix
       [,1]   [,2]    [,3]
[1,] 0.1584 -0.178 -0.0203
[2,] 0.2316  0.182  0.0262
[3,] 0.0304  0.164 -0.3666

Residuals cov-matrix:
           [,1]       [,2]      [,3]
[1,] 1.02816104 0.06541693 0.4179515
[2,] 0.06541693 1.11927510 0.2405560
[3,] 0.41795146 0.24055595 0.9005352
----
aic= 0.02976334
bic= 0.4152618
>
>
> model.ARMA11 = PhiTheta2ARMA(out$Phi, out$Theta,
+                   output.names = seriesNames(sample)$output)
> model.ARMA11 = dse::l(model.ARMA11, estimation.sample)
> summary(model.ARMA11)
neg. log likelihood = 566.4137 sample length = 136
     consumption investment income
RMSE   1.039684   1.063949  0.9585752
ARMA:
inputs:
outputs: consumption investment income
     input dimension = 0 output dimension = 3
     order A = 1 order B = 1  order C =
      18 actual parameters 6 non-zero constants
     trend not estimated.
```

Next we estimate VARMA models in echelon canonical form (with `MTS::Kronfit`) for Kronecker indices $\nu=(1,0,0)$, $\nu=(1,1,0)$, $\nu=(1,1,1)$, and $\nu=(2,1,1)$. The output of these computations is not shown to save space.

```
> out = suppressWarnings(MTS::Kronfit(da = y[1:T.est,],
+               kidx = c(1,0,0), include.mean = FALSE))
> model.ARMA100 = PhiTheta2ARMA(out$Phi, out$Theta, out$Ph0,
+       fix = T, output.names = seriesNames(sample)$output)
> model.ARMA100 = dse::l(model.ARMA100, estimation.sample)
>
> out = suppressWarnings(MTS::Kronfit(da = y[1:T.est,],
+               kidx = c(1,1,0), include.mean = FALSE))
> model.ARMA110 = PhiTheta2ARMA(out$Phi, out$Theta, out$Ph0,
+       fix = T, output.names = seriesNames(sample)$output)
> model.ARMA110 = dse::l(model.ARMA110, estimation.sample)
>
> out = suppressWarnings(MTS::Kronfit(da = y[1:T.est,],
+               kidx = c(1,1,1), include.mean = FALSE))
> model.ARMA111 = PhiTheta2ARMA(out$Phi, out$Theta, out$Ph0,
+       fix = T, output.names = seriesNames(sample)$output)
> model.ARMA111 = dse::l(model.ARMA111, estimation.sample)
>
> out = suppressWarnings(MTS::Kronfit(da = y[1:T.est,],
+               kidx = c(2,1,1), include.mean = FALSE))
> model.ARMA211 = PhiTheta2ARMA(out$Phi, out$Theta, out$Ph0,
+       fix = T, output.names = seriesNames(sample)$output)
> model.ARMA211 = dse::l(model.ARMA211, estimation.sample)
```

Note that the VARMA model which has been used throughout the first sections of this contribution is a "rounded" version of the last model estimated, i.e., the model for $\nu=(2,1,1)$.

```
> all.equal(list(Ph0=phi0,Phi=phi,Theta=theta),
+          lapply(out[c('Ph0','Phi','Theta')],round,3))
[1] TRUE
```

R Demonstration 12 We have 10 candidate models, the VAR(2) model (labeled `VAR` in the following), the two state space models estimated with `dse::bft` (`BFTbic` and `BFTaic`), the two state space models estimated with `dse:estMaxLik` (`BFTbicML` and `BFTaicML`), the VARMA(1,1) model (`ARMA11`) obtained by `MTS:VARMAcpp` and the four echelon form VARMA models (`ARMA100`, `ARMA110`, `ARMA111`, and `ARMA211`) which have been estimated by `MTS:Kronfit`. The dse package provides some nice tools to compare and evaluate a set of estimated models. We start with evaluation the "in-sample" performance of the models with `dse::informationTests`.

```
> info = dse::informationTests(model.VAR, model.BFTbic,
+             model.BFTbicML, model.BFTaic, model.BFTaicML,
+             model.ARMA11, model.ARMA100, model.ARMA110,
+             model.ARMA111, model.ARMA211)
```

We do not show the output of this procedure but collect the (relevant) results in Table 1. The information criteria of course heavily depend on the number of "free" parameters of the respective models. As noted above the default strategy of dse is to consider the coefficients of the parameter matrices which are not equal to one or zero as "free" and to consider the zero/one coefficients as "fixed."

- For a general state space model this gives $2ns+s^2$ parameters. However, this does not account for the fact that the parameter matrices are only unique up to state space transformations, see (39). Therefore `dse::informationTests` also uses the so-called "theoretical number of parameters": $2ns$. In Table 1 we report the values of the information criteria based on this "theoretical number of parameters."
- For VARMA models in echelon form the "actual number of parameters" is not correct, since it does not account for the constraint $a_0=b_0$. The function `dse::fixConstants` allows to set any coefficient as "fixed" or as "free." In order to force `dse::informationTests` to use the correct number of free parameters $2n(\nu_1+\cdots+\nu_n)$ for a model in echelon form, we use `dse::fixConstants` and "fix" all zero/one coefficients **and** all entries of b_0. This is accomplished in the code above by calling `PhiTheta2ARMA` with the optional argument `fix=TRUE`.

The "out-of-sample" performance of these model is evaluated by considering the 1-step ahead prediction errors. The code below computes sample covariance of the $h=1,2,3,4$-step ahead prediction errors on the "validation" sample 1992 to the end of 2017. The results of these computations (for the 1-step ahead prediction error) are summarized in Table 2.

```
> z = dse::forecastCov(model.VAR, model.BFTbic,
+             model.BFTbicML, model.BFTaic, model.BFTaicML,
+             model.ARMA11, model.ARMA100, model.ARMA110,
+             model.ARMA111, model.ARMA211, data = sample,
+             horizons = 1:4, discard.before = T.est)
>
> # extract MSE for each series and the total MSE
> mse = array(0, dim = c(ncol(y)+1,4,n.estimates),
+             dimnames = list(c(colnames(y),'total'),
+                             paste('h=',1:4,sep=''), estimates))
> for (k in (1:n.estimates)){
+   for (h in (1:4)){
+     mse[,h,k] = c(diag(z$forecastCov[[k]][h,]),
+                   sum(diag(z$forecastCov[[k]][h,])))
+}}
> mse = aperm(mse,c(2,3,1))
```

TABLE 1 In-sample (information) criteria of the estimated models.

	#par	port	like	aic	bic	gvc	rice	fpe
VAR	18	129.6	510.6	1057.2	1129.4	1058	1058.8	1057.2
BFTbic	6	144	531.3	1074.7	1098.7	1074.8	1074.9	1074.7
BFTbicML	6	148.7	528.8	1069.5	1093.6	1069.6	1069.7	1069.5
BFTaic	18	123.7	513.2	1062.4	1134.6	1063.2	1064.1	1062.4
BFTaicML	18	124.7	509.5	1055	1127.2	1055.9	1056.7	1055.1
ARMA11	18	249.3	566.4	1168.8	1241	1169.6	1170.5	1168.9
ARMA100	6	156.1	529.5	1070.9	1095	1071	1071.1	1070.9
ARMA110	12	138.6	519.4	1062.9	1111	1063.2	1063.6	1062.9
ARMA111	18	124.6	509.5	1055	1127.2	1055.9	1056.7	1055.1
ARMA211	24	121.4	506.1	1060.3	1156.5	1061.7	1063.3	1060.3

aic, Akaike information criterion; *bic*, Bayes information criterion; *fpe*, final prediction error; *gvc*, generalized cross validation; *like*, neg. log likelihood; *#par*, number of parameters; *port*, portmanteau test; *rice*, rice criterion.

TABLE 2 The (out-of-sample) mean squared errors of the estimated models for the 1-step ahead prediction.

	Consumption	Investment	Income	Total	rVAR
VAR	0.283	0.291	1.006	1.580	0.0%
BFTbic	0.357	0.348	0.988	1.693	−7.1%
BFTbicML	0.308	0.304	0.989	1.601	−1.3%
BFTaic	0.247	0.330	1.017	1.593	−0.8%
BFTaicML	0.260	0.299	0.987	1.546	2.2%
ARMA11	0.277	0.458	0.986	1.721	−8.9%
ARMA100	0.281	0.339	1.006	1.626	−2.9%
ARMA110	0.295	0.277	0.993	1.565	0.9%
ARMA111	0.260	0.300	0.989	1.549	2.0%
ARMA211	0.273	0.355	0.973	1.601	−1.3%

The column total is the sum of the MSE values for the three series (consumption, investment and income). The last column reports the percentage improvement as compared to the VAR model.

9 Discussion and notes

- The `ARMA211` model is the most complex model (24 free parameters) and thus it is no surprise that this model is optimal in terms of the likelihood.
- The models `BFTaicML`, `ARMA11`, `ARMA111` are obtained by optimizing the likelihood over essentially the same set of models (VARMA(1,1) or state space models with a state space dimension $s = 3$). Accordingly `BFTaicML`, `ARMA111` have the same likelihood value. However, `ARMA11` is much worse. This is an indication that the initial estimate is crucial for the ML estimation.
- The models `BFTaicML`, `ARMA111` are the best models with respect to the information criteria. Only the BIC criterion picks the more parsimonious model `BFTbicML`.
- The MSE values of the predictors are quite similar for most of the models (and time series). Therefore the ranking has to be interpreted with some care. For a careful analysis one should test whether the differences are "statistically significant," e.g., by a Diebold Mariano test.
- The models `BFTaicML`, `ARMA111` are also the best models in terms of out-of-sample prediction.
- By construction the ML estimates `BFTbicML`, `BFTaicML` yield better likelihood values than the corresponding initial estimates `BFTbic` and `BFTaic`. For the data considered here the ML estimates also give better predictions, i.e., there is a (small) performance gain.
- The ARMA11 model performs badly.
- Of course the above statements are not valid in general. The results heavily depend on the data considered.
- None of the above estimation schemes guarantees stable and miniphase models. So to be sure one should check the estimated models as described above.

9.1 Summary

In this contribution we provide an introduction to VARMA modeling of multivariate time series. The purpose of this contribution is threefold. We want to describe the structure of VARMA and state space modeling and the problems arising from this structure. Second we describe parameterizations and estimation methods and their properties. This includes both (real valued) parameter estimation and model selection. Third we describe actual computations using R and two appropriate R packages.

In econometrics the dominant approach is still VAR modeling. Despite the greater complexity of VARMA and state space estimation (e.g., more complicated parameter spaces, no explicit expressions for the ML estimates) and not yet fully developed R packages, as compared to VAR modeling, we think that VARMA or state space modeling of economic time series is an interesting and important alternative.

Acknowledgement

The authors thank Prof. Ruey Tsay for several useful comments.

References

Akaike, H., 1974. A new look at the statistical model identification. IEEE Trans. Autom. Control 19 (6), 716–723.

Anderson, T.W., 1951. Estimating linear restrictions on regression coefficients for multivariate normal distributions. Ann. Math. Stat. 22, 327–351.

Aoki, M., Havenner, A., 1991. State space modeling of multiple time series. Econ. Rev. 10 (1), 1–59.

Åström, K.J., Mayne, D.Q., 1982. A new algorithm for recursive estimation of controlled ARMA processes. In: Bekey, G.A., Savidis, G.W. (Eds.), Proceedings of the 6th IFAC Symposium on Identification and System Parameter Estimation. Pergamon Press, Oxford, pp. 175–179.

Bachman, G. Narici, L., 2000. Functional Analysis, (Reprint of the Academic Press, Inc., New York, 1966 edition), Dover Publ.; Mineola, NY, ISBN: 0486402517.

Bauer, D., 2005. Estimating linear dynamical systems using subspace methods. Economet. Theor. 21, 181–211. https://doi.org/10.1017/S0266466605050127.

Brockwell, P., Davis, R., 1991. Time series: theory and methods. In: Springer Series in Statistics, second ed. Springer-Verlag, New York.

Caines, E.P., 1988. Linear Stochastic Systems. John Wiley & Sons, New York.

Deistler, M., Scherrer, W., 2018. Modelle der Zeitreihenanalyse. Birkhäuser.

Deistler, M., Peternell, K., Scherrer, W., 1995. Consistency and relative efficiency of subspace methods. Automatica 31 (12), 1865–1875.

Gibson, S., Ninness, B., 2005. Robust maximum-likelihood estimation of multivariable dynamic systems. Automatica 41 (10), 1667–1682.

Gilbert, P., 2015. Brief User's Guide: Dynamic Systems Estimation. http://cran.r-project.org/web/packages/dse/vignettes/Guide.pdf.

Hannan, E.J., 1980. The estimation of the order of an ARMA process. Ann. Stat. 8, 1071–1081.

Hannan, E.J., Deistler, M., 2012. The statistical theory of linear systems. In: Classics in Applied Mathematics. SIAM, Philadelphia, (Originally published: John Wiley & Sons, New York, 1988).

Hannan, E.J., Rissanen, J., 1982. Recursive estimation of mixed autoregressive-moving average order. Biometrika 69, 81–94.

Hannan, E.J., Kavalieris, L., Mackisack, M., 1986. Recursive estimation of linear systems. Biometrika 73 (1), 119–133.

Ho, B., Kalman, R.E., 1966. Efficient construction of linear state variable models from input/output functions. Regelungstechnik 14, 545–548.

Kalman, R.E., 1963. Mathematical description of linear dynamical systems. J. Soc. Ind. Appl. Math. Ser. A Control 1 (2), 152–192.

Kalman, R.E., 1974. Algebraic geometric description of the class of linear systems of constant dimension. In: 8th Annual Princeton Conference on Information Sciences and Systems. Princeton University, Princeton, NJ.

Larimore, W.E., 1983. System identification, reduced order filters and modeling via canonical variate analysis. In: Rao, H.S., Dorato, P. (Eds.), Proc. 1983 Amer. Control Conference 2. IEEE Service Center, pp. 445–451. Piscataway, NJ.

Leeb, H., Pötscher, B.M., 2005. Model selection and inference: facts and fiction. Economet. Theor. 21 (1), 21–59.

Ljung, L., 1999. System Identification: Theory for the User, second ed. Prentice Hall.

Lütkepohl, H., 2005. Introduction to Multiple Time Series Analysis. Springer, Berlin.

McKelvey, T., Helmersson, A., Ribarits, T., 2004. Data driven local coordinates for multivariable linear systems and their application to system identification. Automatica 40 (9), 1629–1635.

ML Sysid TB, n.d.. Matlab System Identification Toolbox, R2017b. The MathWorks, Inc., USA. https://de.mathworks.com/products/sysid.html.

R Core Team, 2015. R: A Language and Environment for Statistical Computing. R Foundation for Statistical Computing, Vienna, Austria. https://www.R-project.org.

Reinsel, G.C., 1997. Elements of Multivariate Time Series Analysis. Springer.

Ribarits, T., Deistler, M., McKelvey, T., 2004. An analysis of the parametrization by data driven local coordinates for multivariable linear systems. Automatica 40, 789–803.

Rozanov, Y.A., 1967. Stationary Random Processes. Holden-Day, San Francisco.

Schwarz, G., 1978. Estimating the dimension of a model. Ann. Stat. 6 (2), 461–464. https://doi.org/10.1214/aos/1176344136.

Shibata, R., 1980. Asymptotically efficient selection of the order of the model for estimating parameters of a linear process. Ann. Stat. 8 (1), 147–164.

Shumway, R.H., Stoffer, D.S., 1982. An approach to time series smoothing and forecasting using the EM alogorithm. J. Time Ser. Anal. 3 (4), 253–264.

Söderström, T., Stoica, P., 1989. System Identification. Prentice Hall.

Tsay, R.S., 2014. Multivariate Time Series Analysis. John Wiley & Sons.

Tsay, R.S., 2015. MTS: All-Purpose Toolkit for Analyzing Multivariate Time Series (MTS) and Estimating Multivariate Volatility Models. R package version 0, p. 33. https://CRAN.R-project.org/package=MTS.

Van Overschee, P., De Moor, B., 1994. N4SID: Subspace algorithms for the identification of combined deterministic-stochastic systems. Automatica 30, 75–93.

Verhaegen, M., 1994. Identification of the deterministic part of MIMO state space models given in innovations form from input–output data. Automatica 30 (1), 61–74 (special issue: Statistical Signal Processing and Control).

Wold, H., 1954. A Study in the Analysis of Stationary Time Series, second ed. Almqvist and Wiksell, Uppsala.

This page intentionally left blank

Chapter 7

Multivariate GARCH models for large-scale applications: A survey

Kris Boudt[a,b,c,*], **Alexios Galanos**[d], **Scott Payseur**[d] **and Eric Zivot**[d,e]

[a]*Department of Economics, Ghent University, Ghent, Belgium*

[b]*Solvay Business School, Vrije Universiteit Brussel, Brussel, Belgium*

[c]*School of Business and Economics, Vrije Universiteit Amsterdam, Amsterdam, The Netherlands*

[d]*Amazon, Seattle, WA, United States*

[e]*University of Washington, Seattle, WA, United States*

[*]*Corresponding author: e-mail: kris.boudt@vub.be*

Abstract

This chapter provides a survey of various multivariate GARCH specifications that model the temporal dependence in the second moment of multivariate return series processes. The survey is focused on feasible multivariate GARCH models for large-scale applications, as well as on recent contributions in outlier-robust MGARCH analysis and the use of high-frequency returns or the score for covariance modeling. We discuss their likelihood-based estimation and application to forecasting and simulation with software implementations in the **R**-programming language.

Keywords: Comovement, Distribution, Time series, Volatility

1 Introduction

Many problems in both finance and economics require the specification and estimation of a time-varying covariance matrix for asset returns. Examples include portfolio allocation and risk management, derivatives pricing on more than one underlying contract, and contagion modeling of volatility shock transmission. A simple solution is to use the EWMA model of RiskMetrics (Mina et al., 2001). It predicts the next period's covariance as a weighted average of past "de-meaned" returns and uses exponential weighting to give more weight to the more recent return observations (Mina et al., 2001). Over the past two decades, it has become popular to use more flexible models for covariance prediction, while preserving the property that the conditional

Handbook of Statistics, Vol. 41. https://doi.org/10.1016/bs.host.2019.01.001

covariance matrix of the next period's asset return vector is a measurable function of the current and past returns. Most of these models fall in the class of Multivariate Generalized AutoRegressive Conditional Heteroskedasticity (**MGARCH**) models for modeling the multivariate dynamics of asset returns.

A realistic MGARCH model for asset returns should capture better the stylized facts of the presence of conditional heteroskedasticity, fat tails, and time variation in their comovement. It should also allow for flexible and feasible[a] estimation as the number of variables increases. It should allow for the modeling of covariance spillovers and feedbacks, with estimated coefficients which have an economic or financial interpretation. In practice, these ideal characteristics form a set of trade-offs which must be balanced based on the specific application to which the model is applied. For instance, flexible dynamics for a large number of assets usually leads to infeasible estimation, or what is called the curse of dimensionality. Feasible estimation on the other hand, based on some factor representation for instance, trades off a significant degree of the rich dynamics present in more fully parameterized models.

Multivariate GARCH models have been extensively reviewed in Bauwens et al. (2006), Engle (2009), Silvennoinen and Teräsvirta (2009), and Francq and Zakoian (2011). Zivot and Wang (2006), Sheppard (2009), Laurent (2013), and Ghalanos (2015b) discuss their implementation in S-PLUS, MATLAB®, Ox, and R. Our contribution is to provide a survey of feasible models and their applications based on existing implementations in the **R**-programming language. We also discuss recent advances in using either the score or high-frequency returns as drivers of the time variation of the MGARCH model parameters and using robust procedures to dampen the effect of outliers on the MGARCH predictions. Our survey is focused on the specification of the MGARCH model and the estimation using likelihood-based methods. While most of discussion is about the second moment, we also overview the use of MGARCH models for estimating higher order comoments. We do not discuss the specification of the conditional mean, nor discuss the techniques used for evaluation of MGARCH models, for which we refer the reader to (Laurent et al., 2012, 2013; Patton and Sheppard, 2009). Table 1 presents an overview of the main **R** packages currently available for MGARCH analysis of financial return series.

The remainder of this chapter is organized as follows: we begin with a short section on the generalization of univariate GARCH models to the multivariate domain, briefly reviewing the BEKK model which forms the foundation for factor and orthogonal factor models, followed by a review of multivariate distributions and the challenge of incorporating the skewness and fat tails in the multivariate dynamics while retaining feasibility of estimation. The key feasible models reviewed are those arising from linear combinations of univariate GARCH models, namely, the orthogonal and generalized

[a]We define **feasible** in this chapter in a broad computational sense for anything between 10 and 500 assets.

TABLE 1 Multivariate GARCH packages in **R**

Package	Models	Distributions	Features
gogarch	GOGARCH	Multivariate Normal	Max-Likelihood, Method of Moments, NLS and ICA, Prediction
rmgarch	CHICAGO, (a)(F)DCC	Multivariate Normal, Laplace, Student, maGH, Copula-Student	Max-Likelihood (2 step), Prediction, Simulation, Filtering, Testing
bayesDccGarch	DCC	Multivariate Normal	MCMC
ccgarch	E(CCC), E(DCC), E(STCC)	Multivariate Normal, Student	Estimation and Testing
GAS	GAS	Multivariate Normal, Student	Max-Likelihood, Prediction, Simulation, and Testing
lgarch	CC-log GARCH	Multivariate Normal	Estimation and Simulation
mgarchBEKK	BEKK, mGJR	Multivariate Normal	Estimation and Simulation
xdcclarge	cDCC	Multivariate Normal	Estimation (Composite Likelihood with shrinkage)

Note: The table provides a nonexhaustive list of packages and the models and features they support for the estimation of multivariate GARCH models in **R**. We would be remiss if we did not pay special tribute to the S+GARCH/FinMetrics module of S-PLUS which provided the first software implementation of MGARCH models over two decades ago (see Zivot and Wang (2006)), the **G@RCH** Ox package of Sébastien Laurent, and the **MFE** MATLAB® Toolbox of Kevin Sheppard.

orthogonal GARCH model, the nonlinear combination-type models forming the class of Generalized Dynamic Models (which includes the dynamic correlation models), and sections on the use of realized measures and the conditional score as drivers for the time variation of the conditional covariance matrix. Each section includes a short overview of existing **R** packages, while the illustration section and the Supplementary Material in the online version at https://doi.org/10.1016/bs.host.2019.01.001 provide a larger scale application using a number of different models to illustrate their use in a risk management context.

2 Multivariate generalization of GARCH models

The generalization of univariate GARCH models to the multivariate domain is conceptually simple, replacing the variance by the covariance matrix and

using an exterior product of the vector of returns. Consider a set of N assets whose log returns $\boldsymbol{r}_t$ are observed for T periods, with conditional mean $\boldsymbol{\mu}_t = \mathrm{E}[\boldsymbol{r}_t|\,\mathfrak{F}_{t-1}]$, where $\mathfrak{F}_{t-1}$ is the σ field generated by the past realizations of $\boldsymbol{r}_t$, i.e., $\mathfrak{F}_{t-1} = \sigma(\boldsymbol{r}_{t-1}, \boldsymbol{r}_{t-2}, \ldots)$

$$\begin{aligned} \boldsymbol{r}_t|\mathfrak{F}_{t-1} &= \boldsymbol{\mu}_t + \varepsilon_t \\ \varepsilon_t &= \boldsymbol{H}_t^{1/2}\boldsymbol{z}_t, \end{aligned} \tag{1}$$

with $\boldsymbol{H}_t$ being the $N \times N$ positive definite conditional covariance matrix of $\boldsymbol{r}_t$ and $\boldsymbol{z}_t$ an $N \times 1$ i.i.d. random vector with first and second moments:

$$\begin{aligned} \mathrm{E}[\boldsymbol{z}_t] &= 0 \\ \mathrm{Var}[\boldsymbol{z}_t] &= \boldsymbol{I}_N, \end{aligned} \tag{2}$$

with $\boldsymbol{I}_N$ denoting the identity matrix of order N. The conditional covariance matrix $\boldsymbol{H}_t$ of $\boldsymbol{r}_t$ may be defined as:

$$\begin{aligned} \mathrm{Var}(\boldsymbol{r}_t|\mathfrak{F}_{t-1}) &= \mathrm{Var}_{t-1}(\boldsymbol{r}_t) = \mathrm{Var}_{t-1}(\varepsilon_t) \\ &= \boldsymbol{H}_t^{1/2}\mathrm{Var}_{\mathrm{t}-1}(\boldsymbol{z}_t)\left(\boldsymbol{H}_{\mathrm{t}}^{1/2}\right)' \\ &= \boldsymbol{H}_t \end{aligned} \tag{3}$$

The literature on the dynamics governing $\boldsymbol{H}_t$ may be broadly divided into direct multivariate extensions, factor models, and linear combination of univariate GARCH models (Generalized Orthogonal GARCH) and nonlinear combination of univariate GARCH models (the broader class of Dynamic Correlation models). The usual trade-off of model parametrization and dimensionality clearly applies here, with the fully parameterized models offering the richest dynamics at the cost of increasing parameter size, making it unfeasible for modeling anything beyond a couple of assets. The need to invert the covariance matrix in many MGARCH parameterizations introduces estimation problems for large systems, as the eigenvalues of the covariance matrix decrease exponentially fast toward zero, even when the covariance is not singular.

A direct extension of univariate GARCH dynamics to the multivariate domain was proposed by Bollerslev et al. (1988), where each element of the conditional covariance matrix $\boldsymbol{H}_t$ is composed of linear combinations of the lagged errors and cross product errors and lagged values of $\boldsymbol{H}_t$. The VEC (p,q) model is defined as:

$$\mathrm{vech}(\boldsymbol{H}_t) = \mathbf{c} + \sum_{i=1}^{p} \boldsymbol{A}_i \mathrm{vech}\left(\varepsilon_{t-\mathrm{i}}\varepsilon_{t-\mathrm{i}}'\right) + \sum_{j=1}^{q} \boldsymbol{B}_j \mathrm{vech}\left(\boldsymbol{H}_{t-j}\right), \tag{4}$$

where $\boldsymbol{c}$ is the $N(N+1)/2 \times 1$ intercept, vech is the operator that stacks the lower triangular portion of the $N \times N$ symmetric matrix as an $N(N+1)/2$ vector, and matrices $\boldsymbol{A}_i$ and $\boldsymbol{B}_j$ are square of order $N(N+1)/2$, giving a total of

$\frac{1}{2}N^4 + N^3 + N^2 + \frac{1}{2}N$ variables! Because $\boldsymbol{H}_t$ is symmetric, vech($\boldsymbol{H}_t$) contains all the unique elements in $\boldsymbol{H}_t$. The richness of the model is immediately visible, as the variance of each individual asset is a function of its own squared errors and variances, all other squared errors and variances and all other cross lagged errors and covariances, and similarly modeled for the off-diagonal elements (covariances). There is obviously a high cost to modeling the full interaction of lags and cross lags and hence the contagion effect, where the (co)variance of an asset may be influenced by the lagged (co)variance of other assets. Moreover, the requirement that $\boldsymbol{H}_t$ be positive definite for all values of ε_t in the sample space is difficult to impose during estimation.

The Diagonal VEC (*DVEC*) model was suggested by the same authors to partly alleviate the dimensionality problem,[b] by foregoing the effect of cross lags on individual variances and covariances, modeling $\boldsymbol{A}_i$ and $\boldsymbol{B}_j$ as diagonal matrices. Additionally, the diagonal representation, usually expressed in terms of Hadamard products, also benefits from simpler conditions for imposing positive definiteness of $\boldsymbol{H}_t$, derived in Attanasio (1991), which is a drawback of the full VEC model for which such conditions are hard to arrive at.

To overcome the difficulties of imposing positive definiteness in the VEC model and the high dimensionality, while not giving up as much as the DVEC, the BEKK—Baba, Engle, Kratt, and Kroner—model of Engle and Kroner (1995) was proposed on the premise that comovements of financial assets are driven by a set of underlying factors (**K**). In terms of MGARCH categories, it lies somewhere between the direct extension VEC model, for which it is a special case,[c] and a class of factors models most of which can be expressed as special cases of the BEKK model (and discussed in Section 4). In the BEKK(p,q,K) model, the conditional covariance matrix $\boldsymbol{H}_t$ is governed by the following dynamics,

$$\boldsymbol{H}_t = \boldsymbol{C}'\boldsymbol{C} + \sum_{k=1}^{K}\sum_{j=1}^{q} \boldsymbol{A}'_{jk}\,\varepsilon_{t-j}\varepsilon'_{t-j}\boldsymbol{A}_{jk} + \sum_{k=1}^{K}\sum_{j=1}^{p} \boldsymbol{B}'_{jk}\boldsymbol{H}_{t-j}\boldsymbol{B}_{jk}, \tag{5}$$

where $\boldsymbol{C}$, $\boldsymbol{A}_{jk}$, and $\boldsymbol{B}_{jk}$ are $N \times N$ matrices, with $\boldsymbol{C}$ being upper triangular. A direct advantage of the BEKK model over the VEC model (4), is that positivity of $\boldsymbol{H}_t$ is easy to impose. The number of parameters is significantly less in the full BEKK model, being $\frac{5}{2}N^2 + \frac{1}{2}N$, and only about $\frac{5}{3}$ times bigger than the DVEC model. Unlike the DVEC model, the BEKK specification does model the dependence of conditional variances (covariances) subject to the lagged values of all other variances (covariances), hence capturing the spillover effect. However, the parameterization is difficult to understand. The quadratic form of the model means that certain sign restrictions are necessary to ensure identifiability, which for simple models such as when $K = 1$ and

[b]The DVEC requires $\frac{3}{2}(N^2 + N)$ parameters.
[c]In fact, for each BEKK model there is an equivalent VEC representation.

$p=q=1$ is a simple matter of ensuring the positivity of the upper diagonal elements of $\boldsymbol{A}_{11}$ and $\boldsymbol{B}_{11}$. Consistency of the Gaussian QML estimator of the BEKK model was proved by Jeantheau (1998) under the log-moment condition, asymptotic normality of the Quasi Maximum Likelihood (*QML*) estimates of the BEKK model was established by Comte and Lieberman (2003), while Hafner and Preminger (2009) established the asymptotic normality of the VEC model (in which the BEKK is nested) under the existence of sixth order moments.

The dynamics of both the VEC and BEKK models can be reduced to achieve dimensionality reduction gains, leading to several variants such as diagonal and scalar models, as well as the use of covariance targeting to reduce the number of parameters in the estimation of the intercept. In case of the BEKK (1,1,1) model, covariance targeting is achieved by setting:

$$\boldsymbol{C}'\boldsymbol{C}=\overline{\boldsymbol{H}}-\boldsymbol{A}'\overline{\boldsymbol{H}}\boldsymbol{A}-\boldsymbol{B}'\overline{\boldsymbol{H}}\boldsymbol{B}, \tag{6}$$

where $\overline{\boldsymbol{H}}$ is the unconditional covariance matrix of ε which may be consistently estimated by the sample covariance matrix. In order for $\boldsymbol{H}_t$ to be positive definite in the presence of covariance targeting, the eigenvalues of the intercept must be positive and checked during estimation. This is a highly nonlinear constraint which may lead to estimation problems and issues of convergence. Finally, covariance stationarity in the diagonal BEKK(p,q,1) models is simply a vectorized form of the scalar case so that the element-wise sum of the squared diagonal parameters is less than unity:

$$\sum_{j=1}^{p}a_{nn,j}^2+\sum_{j=1}^{q}b_{nn,j}^2<1, \tag{7}$$

for all $n=1,\ldots,N$. It would appear that covariance targeting for large-dimensional systems eliminates $N(N+1)/2$ parameters from the estimation, thus making it more feasible. However, this is only partly true. In the absence of covariance targeting, we can guarantee positive definiteness of the intercept, by construction, through $\boldsymbol{C}'\boldsymbol{C}$. With covariance targeting, the added constraint of positive definiteness provides for two possible avenues. The first one imposes a proper constraint by adding $N(N+1)/2$ parameters to the estimation so that the intercept, $\boldsymbol{\Omega}$, calculated through targeting, is constrained to be positive definite. The requirement for a matrix $\boldsymbol{M}$ to be positive definite is guaranteed if and only if there is a positive definite matrix $\boldsymbol{B}>0$ with $\boldsymbol{B}^2=\boldsymbol{M}$. The matrix $\boldsymbol{B}$ is called the "square root" of $\boldsymbol{M}$. This matrix $\boldsymbol{B}$ is unique, but only under the assumption $\boldsymbol{B}>0$. In terms of the optimization problem, we can include the following constraint to ensure the positive definiteness of the intercept Ω: $\boldsymbol{B}^2-\Omega=\boldsymbol{0}$. If $\boldsymbol{\Omega}$ has a "square root," then it is positive semidefinite. One therefore models the lower triangular part of $\boldsymbol{B}$ which creates an added $N(N+1)/2$ parameters in the optimization problem. Thus in this case, the full constraint reintroduces back into the model the same number of parameters eliminated because of covariance targeting in the first

place, which is possibly one of the reasons it has not been considered in many applications. The second approach, which does not introduce this constraint, involves the use of a global optimization approach since checking for positive and real eigenvalues as an "arbitrary" constraint introduces nonsmoothness and discontinuity in the likelihood, and is likely to lead to many local minima.

3 Multivariate distributions

While univariate GARCH models can easily be extended to have non-Normal distributions which capture asymmetry and fat tails, the concept of a multivariate distribution is far more complicated and forms a constraining element in the MGARCH data modeling process. Within financial applications the emphasis has mostly been on either the elliptical methodology, whereby the transition from the univariate to the multivariate domain has been achieved through the construction of densities that are quadratic form functions of the margins, or through copulas, where the dependency structure is separate from the marginal dynamics. A key step in the maximization of the likelihood function of a multivariate density with GARCH dynamics is to appropriately scale the data so that they are i.i.d.[d] This implies that a multivariate density with conditional mean $\boldsymbol{\mu}_t$ and conditional variance $\boldsymbol{H}_t$ can be scaled so that:

$$f(\mathbf{r}_t|\eta, \mathfrak{F}_{t-1}) = |\mathbf{H}_t|^{-1/2} g\left(\mathbf{H}_t^{-1/2}(\mathbf{r}_t - \boldsymbol{\mu}_t)|\eta\right), \tag{8}$$

where $g(\ldots)$ is the conditional density of the standardized errors and η may optionally represent asymmetry and shape (tail heaviness) parameters. An interesting property describing multivariate distribution is that of tail dependence, which describes the conditional probability of joint exceedance over a large threshold given that some components already exceed that threshold. Tail dependence indexes describe the amount of dependence in the upper right or lower left tail of the distribution and can be used to analyze return comovements for extreme random events. They can be defined and analyzed by using the copula of the distribution (see Chan and Li (2007)), which is discussed in Section 3.5.

A set of measures which capture the cross variation in asymmetry and tail behavior, generalizing the concept of covariance to higher comoments, are those of coskewess and cokurtosis, and defined as the third and fourth standardized cross central moments, respectively. For a vector of returns $\boldsymbol{r}$ with mean $\boldsymbol{\mu}$, the coskewness (M^3) and cokurtosis (M^4) can be represented as:

$$\begin{aligned} \boldsymbol{M}^3 &= E\left[(\boldsymbol{r}-\boldsymbol{\mu})(\boldsymbol{r}-\boldsymbol{\mu})'\otimes(\boldsymbol{r}-\boldsymbol{\mu})'\right] \\ \boldsymbol{M}^4 &= E\left[(\boldsymbol{r}-\boldsymbol{\mu})(\boldsymbol{r}-\boldsymbol{\mu})'\otimes(\boldsymbol{r}-\boldsymbol{\mu})'\otimes(\boldsymbol{r}-\boldsymbol{\mu})\right], \end{aligned} \tag{9}$$

[d]The weaker assumption that they are a martingale difference sequence with respect to the conditioning information leads to a quasi-likelihood approach.

where $\otimes$ is the kronecker product. These measures have proven to be particularly important in portfolio type applications (see for instance Kraus and Litzenberger (1976) and Harvey and Siddique (2000)). As shown in Section 4, at least one of the MGARCH models presented has closed form expressions for these measures, making it particularly flexible for portfolio modeling. Since it has long been established that the returns of financial assets exhibit characteristics such as fat tails and asymmetry, distributions which allow for such parameters may be important. Balancing the need for flexible distributions which allow for individual measures of asymmetry and tail dependence with parsimonious representations which do not suffer from the curse of dimensionality and are closed under linear affine transformations is a challenging proposition.

The class of elliptical distributions, introduced by Kelker (1970), may be considered as generalizations of the multivariate Normal distribution and therefore share many of its desirable properties, while also allowing for some tail dependence. Very generally, an elliptical distribution can be considered as an affine transformation of a spherical distribution, the latter being a distribution which is invariant under rotations and reflections. Within this class of distributions belong the multivariate Student and Laplace distributions, neither of which allow for asymmetry.[e] In the next subsections we consider four popular choices of multivariate distributions which have been used in MGARCH modeling; the multivariate Normal, Student, and Laplace distributions and the multivariate Generalized Hyperbolic which is a flexible distribution from the mean–variance mixture family, and nests the former three distributions and skewed variations of those as special cases. We also provide a section on the Copula distribution which provides a great deal of flexibility in the modeling of the margins seperately from the joint dynamics.

3.1 Multivariate Normal

Traditionally, because of its tractability and desirable features, the multivariate Normal distribution, uniquely determined by its mean and covariance, has dominated financial modeling. It possesses many desirable features such as invariance under affine linear transformations, infinite divisibility, self-decomposability, and formation of subsequences, making it ideal for the regressive and autoregressive modeling as well as portfolio modeling. It also forms a sufficient condition for the use of mean–variance analysis developed by Markowitz (1952) and used extensively in industry to this date. Even when the underlying data generating process is not conditionally multivariate

[e]Skewed versions of these distributions have been introduced in the literature, see, for example, Bauwens and Laurent (2005) for a version of a skewed multivariate Student distribution, and Bauwens and Laurent (2005), Kotz et al. (2002), Kozubowski and Podgórski (2001), and Arslan (2010) for skewed multivariate Laplace variants.

Normal, it will still yield consistent estimates of the MGARCH parameters, as shown by Bollen and Inder (2002) (see also Gourieroux (1997) for its asymptotic properties in the context of MGARCH), making it a rather forgiving distribution in terms of consistency in the presence of misspecification. However, the absence of tail dependence and asymmetry may lead to significant underestimation of extreme events, making it unsuitable for many financial applications.

A vector of N returns at time t, $\boldsymbol{r}_t$, with conditional mean $\boldsymbol{\mu}_t$ and conditional convariance matrix $\mathbf{H_t}$, follows a multivariate Normal distribution if $\boldsymbol{r}_t \sim \text{MN}(\boldsymbol{\mu}_t, \mathbf{H_t})$. Because of the affine linear transformation and scaling property of this distribution, the distribution of the errors $\boldsymbol{r}_t - \boldsymbol{\mu}_t = \varepsilon_t \sim \text{MN}(0, \mathbf{H_t})$ is also multivariate Normal with zero mean, and the scaled errors $\mathbf{H_t^{-1/2}}\varepsilon_t = \boldsymbol{z_t} \sim \text{MN}(0, \mathbf{I_N})$ are again multivariate Normal with identity matrix $\boldsymbol{I}_N$. The likelihood at time t of the errors is given by:

$$p_t(\varepsilon_t|\theta) = \frac{1}{(2\pi)^{N/2}|\mathbf{H_t}|^{1/2}} \exp\left(-\frac{1}{2}\varepsilon_t' \mathbf{H_t^{-1}} \varepsilon_\mathbf{t}\right) \tag{10}$$

All margins and conditionals of a multivariate Normal distribution are also multivariate Normal.

3.2 Multivariate Student

The multivariate Student distribution takes one extra parameter, the shape parameter ν, which is inherited from the derivation of this distribution as a mixture of a multivariate Normal and a Gamma. This gives rise to symmetric tail dependence[f] and also necessitates restrictions on its lower bounds in order to ensure existence of moments.[g] Formally, $\mathbf{H_t}$, with conditional mean $\boldsymbol{\mu}_t$, conditional covariance matrix $\mathbf{H_t}$, and conditional shape parameter ν, follows a multivariate Student distribution if $\boldsymbol{r}_t \sim \text{MT}(\boldsymbol{\mu}_t, \boldsymbol{\Omega}_t, \nu)$, where $\boldsymbol{\Omega}_t$ is a scale matrix such that $\mathbf{H_t} = \frac{\nu}{(\nu-2)}\boldsymbol{\Omega}_t$. The likelihood at time t of the errors $\varepsilon_t = \boldsymbol{r}_t - \boldsymbol{\mu}_t$ is given by:

$$p_t(\varepsilon_t|\theta) = \frac{\Gamma\left(\frac{1}{2}(\nu+N)\right)}{\Gamma\left(\frac{1}{2}\nu\right)\nu^{N/2}\pi^{N/2}|\boldsymbol{\Omega}_t|^{1/2}} \left[1 + \frac{1}{\nu}\varepsilon_t'\boldsymbol{\Omega}_t^{-1}\varepsilon_t\right]^{-\frac{1}{2}(\nu+N)}. \tag{11}$$

Similar to the Normal, and elliptical distributions in general, the distribution is invariant under a linear affine transformation so that if $\boldsymbol{r}$ is distributed as multivariate Student, then if $\boldsymbol{z} = \boldsymbol{A}\boldsymbol{y} + b$, $\boldsymbol{z} \sim t(\boldsymbol{A}\boldsymbol{\mu} + b, \boldsymbol{A}\boldsymbol{\Omega}\boldsymbol{A}', \nu)$ is a univariate

[f] Since multivariate Student distributions are radially symmetric, the upper and lower tail dependence are the same.

[g] $\nu > 1$ for existence of the mean, and $\nu > 2$ for existence of the covariance.

Student distribution with the same shape parameter. The fact that all affine transformations have the same shape parameter ν may be quite restrictive in practice. Extensions to this distribution have been proposed for instance by Bauwens and Laurent (2005) who generalize the univariate skew-Student of Fernández and Steel (1998) to the multivariate case, while allowing each margin to have a different asymmetry coefficient but common shape parameter. The square of the asymmetry parameter is nicely interpretable as the ratio of probability masses above and below the mode. In an empirical application using the time-varying correlation model of Tse and Tsui (2002) and four daily stock market indexes, they find empirical evidence of outperformance vs the multivariate Student. An interesting extension which allows for vector-valued shape parameters is proposed in Serban et al. (2007) and evaluated as part of horse race of MGARCH models with different multivariate distributions in Rossi and Spazzini (2010).

The multivariate Normal and Student distributions are available in the **R** package **mvtnorm**, and the **rmgarch** package in the context of DCC modeling, while the **sn** package implements the multivariate skew Student of Azzalini (2013).

3.3 Multivariate Laplace

The Laplace distribution has a special place alongside the Normal distribution, being stable under geometric rather than ordinary summation, thus making it suitable for stochastic modeling. In regression modeling, when the errors have a Laplace distribution, then the least absolute deviation estimate (*lad*) is also the maximum likelihood estimate, equivalent to the least squared deviation estimate when the errors have a Normal distribution. This can be easily inferred from the density function of the Laplace which differs mainly from the Normal by including a term for the mean absolute rather squared deviation of a random variable. It also arises as a special case in the Generalized Error distribution with shape parameter = 1, and the Geometric Stable distribution with stability parameter = 2, and zero skewness and location (also called the Linnik distribution with stability parameter = 2). Because it has tails heavier than the Normal distribution, it is more suitable for the modeling of financial returns. In the multivariate setting, the multivariate Laplace has been analyzed, among others, by Anderson (1992) as part of the multivariate Linnik distribution, Marshall and Olkin (1993) and Kalashnikov (1997) as a multivariate distribution generated by i.i.d. univariate Laplace margins, and Fernandez et al. (1995) as a natural generalization of the univariate model to N dimensions in the framework of the multivariate exponential power distribution.

There are numerous generalizations of univariate to multivariate Laplace distributions; we follow Kozubowski et al. (2013) who define a Generalized Laplace distribution as location-scale mixtures of normal distributions where $\boldsymbol{r}_t \sim \mathrm{ML}(\boldsymbol{\mu}_t, \mathbf{H}_t)$, with conditional mean $\boldsymbol{\mu}_t$ and conditional

covariance $\mathbf{H_t}$. The mixing distribution is the standard exponential. The likelihood at time t of the errors is given by:

$$p_t(\varepsilon_t|\theta) = \frac{2}{(2\pi)^{n/2}|\mathbf{H}|^{1/2}}\left(\frac{\varepsilon_t'\mathbf{H_t^{-1}}\varepsilon_\mathbf{t}}{2}\right)^{(2-n)/4} K_{(2-n)/4}\left(\sqrt{2\varepsilon_t'\mathbf{H_t^{-1}}\varepsilon_\mathbf{t}}\right), \quad (12)$$

where K is the modified Bessel function of the third kind. As in the case of the multivariate Student, the multivariate Laplace has the affine linear transformation property, and the margins of a multivariate Laplace are also Laplace. However, unlike the multivariate Normal, but similar to the multivariate Student, uncorrelatedness does not imply independence. Extensions to the multivariate Laplace have attempted to introduce asymmetry, as in Kozubowski and Podgórski (2001), which is propagated by the location vector making the distribution no longer location shift invariant and the distribution of the errors (i.e., centering by a constant) no longer belongs to the same family. An alternative skewed representation by Arslan (2010) does not suffer from this drawback. The multivatiate Laplace in the context of DCC model distributions is available in the **rmgarch R** package of Ghalanos (2015b).

3.4 Multivariate Generalized Hyperbolic distribution

The multivariate Generalized Hyperbolic distribution (MGH) arises as a special case of the normal mean–variance mixture distribution family which takes the following form:

$$\boldsymbol{r} \overset{d}{=} \boldsymbol{\mu} + W\gamma + \sqrt{W}\boldsymbol{A}\boldsymbol{Z}, \quad (13)$$

where $\boldsymbol{Z} \sim N_q(0, \boldsymbol{I}_q)$, $W \in \mathbb{R}_+^1$, $\boldsymbol{A} \in \mathbb{R}^{N\times q}$, and $\boldsymbol{\mu}$, $\gamma \in \mathbb{R}^N$. The basic premise behind this distribution is to introduce noise in the covariance matrix and mean vector of a multivariate Normal distribution through the mixing variable W. The vector-valued variable γ introduces asymmetry, and when it is equal to zero, $\boldsymbol{r}$ is distributed as a scale mixture of Normal distributions. Different mixing distributions for W give rise to different families of distributions. When the mixing variable W is Generalized Inverse Gaussian (GIG), the N-dimensional GH distribution of Barndorff-Nielsen (1977) arises, which allows for the modeling of multivariate data with some very desirable features such as the ability to model skewness individually for each dimension. Additionally, the distribution has the property of infinite divisibility (inherited from the GIG mixing distribution), and is closed under margining, conditioning, and linear affine transformations, and in the case of the NIG distribution is also closed under convolution when the margins have the same skew and shape parameters. Formally, the N-dimensional Generalized Hyperbolic distribution of the random vector $\boldsymbol{r} \in \mathbb{R}^N$, $\boldsymbol{r} \sim \mathrm{GH}(\lambda, \alpha, \boldsymbol{\mu}, \Delta, \delta, \boldsymbol{\beta})$, with

$(\lambda, \alpha, \delta) \in \mathbb{R}^1$ representing the shape parameters, $\boldsymbol{\beta} \in \mathbb{R}^N$ the asymmetry parameters, Δ the $N \times N$ scaling matrix, and $\boldsymbol{\mu} \in \mathbb{R}^N$ the location parameters, is given by:

$$\begin{aligned} f(\boldsymbol{r}) &= cK_{\lambda-\frac{N}{2}}\left(\alpha\sqrt{\delta^2 + (\boldsymbol{r}-\boldsymbol{\mu})'\Delta(\boldsymbol{r}-\boldsymbol{\mu})}\right)e^{\boldsymbol{\beta}'(\boldsymbol{r}-\boldsymbol{\mu})} \\ c &= \frac{\left(\alpha^2 - \boldsymbol{\beta}'\Delta\boldsymbol{\beta}\right)^{\lambda/2}}{(2\pi)^{\frac{N}{2}}\sqrt{|\Delta|}\alpha^{\lambda-\frac{N}{2}}\delta^{\lambda}K_{\lambda}\left(\delta\sqrt{\alpha^2 - \boldsymbol{\beta}'\Delta\boldsymbol{\beta}}\right)}, \end{aligned} \tag{14}$$

with the mixture representation given by the following:

$$\begin{aligned} \boldsymbol{r}|\mathbf{W} = & \quad w \sim N_N(\boldsymbol{\mu} + w\boldsymbol{\beta}\Delta, w\Delta) \\ & \mathbf{W} \sim \text{GIG}\left(\lambda, \delta^2, \alpha^2 - \boldsymbol{\beta}'\Delta\boldsymbol{\beta}\right) \end{aligned}, \tag{15}$$

where K_λ is the Bessel function of the third kind.

In the one-dimensional case, the distribution reduces to:

$$f(r) = \frac{\left(\alpha^2 - \beta^2\right)^{\lambda/2}}{\sqrt{2\pi}\alpha^{\lambda-1/2}\delta^{\lambda}K_{\lambda}\left(\delta\sqrt{\alpha^2-\beta^2}\right)} \frac{K_{\lambda-1/2}\left(\alpha\sqrt{\delta^2+(r-\mu)^2}\right)}{\left(\sqrt{\delta^2+(r-\mu)^2}\right)^{1/2-\lambda}} e^{\beta(r-\mu)}. \tag{16}$$

A number of different parameterizations exist for the multivariate and univariate case, and for GARCH modeling we seek to find one which is location scale invariant. An extensive review of the Generalized Hyperbolic distribution can be found in Prause (1999), and the **R** package **ghyp** provides both the univariate and multivariate representations with functions for easily switching between parameterizations. The univariate GH distribution has been applied in a number of different GARCH applications, and in Section 4 we describe how a large-dimensional problem can be reduced to the univariate estimation of the independent margins of an affine multivariate GH distribution.

3.5 Copula distributions

Copula functions were introduced by Sklar (1959) as a tool to connect disparate marginal distributions together to form a joint multivariate distribution. They were extensively used in survival analysis and the actuarial sciences for many years before being introduced in the finance literature by Frey and McNeil (2003) and Li (2000). They have since been very popular in investigating the dependence of financial time series of various assets classes and frequencies, an excellent reference for which is available in McNeil et al. (2015).

An N-dimensional copula $C(u_1, \ldots, u_N)$ is a distribution in the unit hypercube $[0,1]^N$ with uniform margins. Sklar (1959) showed that every joint

distribution F of the random vector $\boldsymbol{X}=(x_1,\ldots,x_N)$ with margins $F_1(x_1),\ \ldots,\ F_N(x_N)$ can be represented as:

$$F(x_1,\ldots,x_N)=C(F_1(x_1),\ldots,F_N(x_N)) \tag{17}$$

for a copula C, which is uniquely determined in $[0,1]^N$ for distributions F under absolutely continuous margins and obtained as:

$$C(u_1,\ldots,u_N)=F\left(F_1^{-1}(u_1),\ldots,F_N^{-1}(u_N)\right). \tag{18}$$

The density function may conversely be obtained as:

$$f(r_1,\ldots,r_N)=c(F_1(r_1),\ldots,F_N(r_N))\prod_{i=1}^{N}f_i(r_i), \tag{19}$$

where f_i are the marginal densities and c is the density function of the copula given by:

$$c(u_1,\ldots,u_N)=\frac{f\left(F_1^{-1}(u_1),\ldots,F_N^{-1}(u_N)\right)}{\prod_{i=1}^{N}f_i\left(F_i^{-1}(u_i)\right)}, \tag{20}$$

with F_i^{-1} being the quantile function of the margins. A key property of copulas is their invariance under strictly increasing transformation of the components of $\boldsymbol{r}$, so that, for example, the copula of the multivariate Normal distribution $F_N(\boldsymbol{\mu},\boldsymbol{H})$ is the same as that of $F_N(0,\boldsymbol{R})$ where $\boldsymbol{R}$ is the correlation matrix implied by the covariance matrix, and the same for the copula of the multivariate Student distribution reviewed in detail in Demarta and McNeil (2005). The density of the Normal copula, of the N-dimensional random vector $\boldsymbol{r}$ in terms of the correlation matrix $\boldsymbol{R}$, is then:

$$c(u;\boldsymbol{R})=\frac{1}{|\boldsymbol{R}|^{1/2}}e^{-\frac{1}{2}u'\left(\boldsymbol{R}^{-1}-\boldsymbol{I}_N\right)u}, \tag{21}$$

where $\boldsymbol{u}_i=\boldsymbol{\Phi}^{-1}(F_i(\boldsymbol{r}_i))$ for $i=1,\ldots,N$, representing the quantile of the probability integral transformed (*PIT*) values of $\boldsymbol{r}$. Because the Normal copula cannot account for tail dependence, the Student copula has been more widely used for modeling of financial assets. The density of the Student copula, of the N-dimensional random vector $\boldsymbol{r}$ in terms of the correlation matrix $\boldsymbol{R}$ and shape parameter ν, can be written as:

$$c(\boldsymbol{u};\boldsymbol{R},\nu)=\frac{\boldsymbol{\Gamma}\left(\frac{\nu+N}{2}\right)\left(\boldsymbol{\Gamma}\left(\frac{\nu}{2}\right)\right)^N\left(1+\nu^{-1}\boldsymbol{u}'\boldsymbol{R}^{-1}\boldsymbol{u}\right)^{-(\nu+N)/2}}{|\boldsymbol{R}|^{1/2}\left(\boldsymbol{\Gamma}\left(\frac{\nu+N}{2}\right)\right)^N\boldsymbol{\Gamma}\left(\frac{\nu}{2}\right)\prod_{i=1}^{N}\left(1+\frac{\boldsymbol{u}_i^2}{\nu}\right)^{-(\nu+1)/2}}, \tag{22}$$

where $\boldsymbol{u}_i={t_\nu}^{-1}(F(r_i;\nu))$, where t_ν^{-1} is the quantile function of the Student distribution with shape parameter ν.

While Pearson's product moment correlation totally characterizes the dependence structure in the multivariate Normal case, where zero correlation also implies independence, it can only characterize the ellipses of equal density when the distribution belongs to the elliptical class. In the latter case for instance, with a distribution such as the multivariate Student, the correlation cannot capture tail dependence determined by the shape parameter. Furthermore, it is not invariant under monotone transformations of original variables, making it inadequate in many cases. An alternative measure which does not suffer from this is Kendall's τ (see Kruskal (1958)) based on rank correlations which makes no assumption about the marginal distributions but depends only on the copula C. It is a pairwise measure of concordance calculated as:

$$\tau(r_i, r_j) = 4\int_0^1 \int_0^1 C(u_i, u_j)\, dC(u_i, u_j) - 1. \tag{23}$$

For elliptical distributions, Lindskog et al. (2003) proved that there is a one-to-one relationship between this measure and Pearson's correlation coefficient ρ given by:

$$\tau(r_i, r_j) = \left(1 - \sum_{x \in \mathbb{R}} \left(\mathbb{P}\{\boldsymbol{r}_i = r\}^2\right)\right) \frac{2}{\pi} \arcsin \rho_{ij}, \tag{24}$$

which under certain assumptions (such as in the case of the multivariate Normal) simplifies to $\frac{2}{\pi}\arcsin\rho_{ij}$.[h] Kendall's τ is also invariant under monotone transformations, making it rather more suitable when working with non-elliptical distributions.[i]

The univariate density estimation and subsequent PIT transformation of the margins provide for a great deal of flexibility, with the possibility of adopting a parametric, semiparametric, or empirical approach. The first method, whereby the margins and transformations are performed using a parametric density, was termed the Inference Functions for Margins by Joe (1997) who also established the asymptotic theory for it. The semiparametric method uses a distribution which couples together tails fitted by the generalized Pareto distribution[j] with a kernel-based interior and described in Davison and Smith (1990), and offers a rather flexible method for capturing fat tails

[h] Another popular measure is Spearman's correlation coefficient ρ_s which under Normality equates to $\frac{6}{\pi}\arcsin\frac{\rho_{ij}}{2}$, and it is usually very close in result to Kendall's measure.

[i] A useful application arises in the case of the multivariate Student distribution, where a maximum likelihood approach for the estimation of the correlation matrix $\boldsymbol{R}$ becomes computationally unfeasible for large dimensions. In this case, an alternative approach is to estimate the sample counterpart of Kendall's τ from the transformed margins and then translate that into the correlation matrix as detailed in (24), providing for a method of moments type estimator.

[j] For which a Probability Weighted Moment (*PWM*) approach exists which is quite robust (see, for example, Hosking et al. (1985)).

observed in practice.[k] Finally, the empirical approach, also called pseudo-likelihood, was investigated by Genest et al. (1995) and asymptotic properties established under the assumption that the sequence of $\boldsymbol{r}$ is i.i.d. The ability to estimate the model in two steps together with the option of choosing different distributions for each of the margins makes for a very computationally tractable and flexible system.

The extension of the static copula approach to dynamic models, and in particular GARCH, was investigated by Patton (2006) who extended and proved the validity of Sklar's theorem for the conditional case and discussed further in Section 5. The **copula R** package of Hofert et al. (2018) provides an extensive set of methods for working with commonly used elliptical, Archimedean, nested Archimedean, extreme-value and other copula families, as well as their rotations, mixtures, and asymmetrizations.

4 Generalized Orthogonal GARCH models

Factor ARCH (F-ARCH) models, originally introduced by Engle et al. (1990), and further discussed in Engle (2009), are based on the assumption that returns are generated by a set of unobserved underlying factors that are conditionally heteroscedastic, while the dependence framework is nondynamic as a consequence of large-scale estimation in a multivariate setting. The main advantage of this and related factor-type models is that it avoids estimating off-diagonal components of the MGARCH parameter matrices, thus avoiding the curse of dimensionality. In the F-ARCH model, the factors are assumed to be correlated which may be undesirable if it turns out that they represent genuinely different common components driving the returns. In the Orthogonal (O-) and Generalized Orthogonal (GO-) GARCH models it is instead assumed that the returns $\boldsymbol{r}_t$ are linked to a set of unobserved factors $\boldsymbol{f}_t$ through a linear invertible map $\boldsymbol{A}$. Consider a set of N assets whose returns $\boldsymbol{r}_t$ are observed for t periods, with conditional mean $E[\boldsymbol{r}_t|\,\Im_{t-1}]=\boldsymbol{\mu}_t$, as in (1). The orthogonal-type GARCH models map $\boldsymbol{r}_t-\boldsymbol{\mu}_t$ onto a set of uncorrelated factors $\boldsymbol{f}_t$ (or "structural errors"),

$$\boldsymbol{r}_t=\boldsymbol{\mu}_t+\varepsilon_t \quad t=1,\ldots,T \tag{25}$$

$$\varepsilon_t = \boldsymbol{A}\boldsymbol{f}_t. \tag{26}$$

Differences between the models are based on the specification of the linear map $\boldsymbol{A}$. In the O-GARCH model of Ding (1994) and Alexander (2001), $\boldsymbol{A}$ is an orthogonal matrix, estimated from unconditional information (full sample correlation matrix), so that $\boldsymbol{f}_t=\boldsymbol{A}'\varepsilon_t$ is the principal component vector with orthogonal factors and $\boldsymbol{A}=\boldsymbol{\Lambda}^{1/2}\boldsymbol{P}\boldsymbol{\Sigma}^{1/2}$ with $\boldsymbol{\Sigma}=\text{diag}\{\sigma_1,\ldots,\sigma_N\}$ calculated from sample information, $\boldsymbol{\Lambda}$ the $N\times N$ diagonal matrix of eigenvalues of the

[k]This is implemented in the **spd** package of Ghalanos (2012).

unconditional correlation matrix, and $\boldsymbol{P}$ the associated orthogonal eigenvectors.[1] The conditional covariance matrix, $\mathbf{H_t} \equiv \mathbf{E}\left[(\boldsymbol{r_t} - \boldsymbol{\mu_t})(\boldsymbol{r}_t - \boldsymbol{\mu_t})' \middle| \mathfrak{F}_{t-1}\right]$, of the returns is given by:

$$\boldsymbol{H}_t = \boldsymbol{A}\boldsymbol{V}_t\boldsymbol{A}', \tag{27}$$

where $\boldsymbol{V}_t = E_{t-1}(\boldsymbol{f}_t\boldsymbol{f}_t') = \text{diag}\left\{v_{f_{1,t}}^2, \ldots, v_{f_{t,N}}^2\right\}$ are the factor conditional variances which are assumed to follow univariate GARCH-type processes. A key benefit of this approach is that the number of factors can be restricted to be less than N, with a choice of heuristic or more complex methods for determining the cutoff dimension (see for instance Marchenko and Pastur (1967)). However, because one can always rediscover uncorrelated sources by certain statistical transformations, O-GARCH models suffer from identification issues in the presence of weakly correlated data as a result of using only unconditional information. As Van der Weide (2002) notes, orthogonal matrices form only a very small subset of all possible linear maps and indentification is only guaranteed when the variances of the transformed components are unique. In the GO-GARCH model, the linear map $\boldsymbol{A}$ is decomposed as:

$$\boldsymbol{A} = \boldsymbol{\Lambda}^{1/2}\boldsymbol{P}\boldsymbol{U}, \tag{28}$$

where $\boldsymbol{U}$ is an orthogonal matrix restricted to have determinant 1, and coincides with the O-GARCH model when $\boldsymbol{U}$ is the identity matrix. The calculation of $\boldsymbol{U}$ requires the use of conditional information, and while whitening is not sufficient for independence, it is nevertheless an important step in the preprocessing of the data in the search for independent factors, since by exhausting the second-order information contained in the sample covariance matrix it makes it easier to infer higher order information, reducing the problem to one of rotation (orthogonalization). The factors have the following specification:

$$\boldsymbol{f}_t = \boldsymbol{V}_t^{1/2}\boldsymbol{z}_t, \tag{29}$$

where $\boldsymbol{V}_t = E\left[\boldsymbol{f}_t\boldsymbol{f}_t' \middle| \mathfrak{F}_{t-1}\right]$ is a diagonal matrix with elements $(v_{f_{1,t}}^2, \ldots, v_{f_{t,N}}^2)$ which are the conditional variances of the factors, and $\boldsymbol{z}_t = (z_{1t}, \ldots, z_{Nt})'$. The random variable z_{it} is independent of z_{jt-s} $\forall j \neq i$ and $\forall s$, with $E[z_{it} | \mathfrak{F}_{t-1}] = 0$ and $E[z_i^2] = 1$, this implies that $E[\boldsymbol{f}_t | \mathfrak{F}_{t-1}] = 0$ and $E[\varepsilon_t | \mathfrak{F}_{t-1}] = 0$. The factor conditional variances, $v_{f_{i,t}}^2$, can be modeled as a GARCH-type process. The unconditional distribution of the factors is characterized by:

$$E[\boldsymbol{f}_t] = 0 \quad E[\boldsymbol{f}_t\boldsymbol{f}_t'] = \boldsymbol{I}_N \tag{30}$$

which, in turn, implies that:

$$E[\varepsilon_t] = 0 \quad E\left[\varepsilon_t\varepsilon_t'\right] = \boldsymbol{A}\boldsymbol{A}'. \tag{31}$$

[1]Where $\boldsymbol{P}$ satisfies $\boldsymbol{P}' = \boldsymbol{P}^{-1}$, $\boldsymbol{P}'\boldsymbol{P} = \boldsymbol{I}_N$, and $\boldsymbol{P}\boldsymbol{P}' = \boldsymbol{I}_N$.

It follows that the returns can be expressed as:

$$\boldsymbol{r}_t = \boldsymbol{\mu}_t + \boldsymbol{A}\boldsymbol{f}_t. \tag{32}$$

The conditional covariance matrix, $\mathbf{H_t}|\mathfrak{F}_{\mathbf{t}-1}$, of the returns is the same as in (27). In the original paper of of Van der Weide (2002), $\boldsymbol{U}$ was parameterized by means of rotation matrices with components of the Euler angles. This follows from the fact that every N-dimensional orthogonal matrix $\boldsymbol{U}$ with $\det(\boldsymbol{U}) = 1$ can be represented as a product of $\frac{N(N-1)}{2}$ rotation matrices ($\boldsymbol{U}$):

$$\boldsymbol{U} = \prod_{i>j} \boldsymbol{G}_{ij}(\theta_{ij}), \; -\pi \leqslant \theta_{ij} \leqslant \pi, \tag{33}$$

where $\boldsymbol{G}_{ij}(\theta_{ij})$ performs a rotation in the plane spanned by the ith and jth vectors of the canonical bases of $\mathbb{R}^N$ over the angle θ_{ij}. To illustrate this, consider the hypothetical example in Van der Weide (2002) where $\boldsymbol{U}_\theta$ has the following two-dimensional map:

$$\boldsymbol{U}_\theta = \begin{pmatrix} 1 & 0 \\ \cos\theta & \sin\theta \end{pmatrix} \tag{34}$$

where θ measures the degree to which the uncorrelated components are mapped in the same direction. When $\theta = 0$, the map is not invertible giving rise to perfect correlation between the observed variables, whereas when $\theta = 1/2\pi$ they are completely uncorrelated. Taking the conditional variance of the components to be $(v_{1,t}, v_{2,t})$, their ratio $z_t = \frac{v_{1,t}}{v_{2,t}}$ and strictly positive, then the correlation ρ_t can be expressed as:

$$\rho_t = \frac{1}{\sqrt{1 + z_t \tan^2\theta}}. \tag{35}$$

Since z_t will have finite lower and upper bounds in finite samples, then it follows that the conditional correlation ρ_t will also be bounded. The illustration also shows that even though $\boldsymbol{U}$ is constant, it still gives rise to time-varying correlations which increase on average when the components are mapped in the same direction.

A number of ways have been proposed to estimate $\boldsymbol{U}$. In Van der Weide (2002), a joint maximum likelihood approach was used to estimate all parameters in the model, making the procedure computationally unfeasible for anything other than a few assets. Alternative approaches such as nonlinear least squares and method of moments for the estimation of $\boldsymbol{U}$ have been proposed in van der Weide (2004) and Boswijk and van der Weide (2011), respectively. Alternatively, the matrix $\boldsymbol{U}$ can be estimated in a separate step by Independent Component Analysis (ICA) as in Broda and Paolella (2009) and Zhang and Chan (2009) which leads to fast estimation of very large systems. ICA is a computational method for separating multivariate mixed signals, $\boldsymbol{y} = [r_1, \ldots, r_N]'$, into additive statistically independent and non-Gaussian

components, $\boldsymbol{s}=[s_1,\ldots,s_N]'$, such that $\boldsymbol{y}=\boldsymbol{B}\boldsymbol{s}$. The independent source vector, $\boldsymbol{s}\in\mathbb{R}^N$, is assumed to be sampled from a joint distribution $f(\boldsymbol{s})$,

$$f(s_1,\ldots,s_N)=f(s_1)f(s_2)\ldots f(s_N), \tag{36}$$

where $\boldsymbol{s}$ is not directly observable, nor is the particular form of the individual distributions, $f(s_i)$, usually known.[m] This forms the key property of independence, namely, that the joint density of independent signals is simply the product of their margins. The estimate of the linear mixing matrix $\boldsymbol{B}$ can be obtained via estimation methods based on a choice of criteria for measuring independence which include the maximization of non-Gaussianity through measures such as kurtosis and negentropy, minimization of mutual information, likelihood, and infomax. This follows from the Central Limit Theorem which states that mixtures of independent variables tend to become more Gaussian in distribution when they are mixed linearly, hence maximizing non-Gaussianity leads to independent components (see Hyvarinen and Oja (2000) for more details).[n] The FastICA of Hyvarinen and Oja (2000) is a very efficient batch algorithm which can be used to estimate the components either one at a time by finding maximally non-Gaussian directions or in parallel by maximizing non-Gaussianity or the likelihood. It should be noted that since ICA is a linear noiseless model,[o] the implication for this two-stage estimation in the GO-GARCH model is that uncertainty plays no part in the derivation of $\boldsymbol{U}$ and hence does not affect the standard errors of the independent factors.

The fast estimation procedure of the GO-GARCH model proposed by Broda and Paolella (2009) and Zhang and Chan (2009) can be summarized as follows. First, the FastICA algorithm is applied to the whitened data $z_t=\widehat{\boldsymbol{\Sigma}}^{-1/2}\widehat{\varepsilon}_t$, where $\widehat{\Sigma}^{1/2}$ is obtained from the eigenvalue decomposition of the OLS residuals covariance matrix, returning an estimate of $\boldsymbol{f}_t$. Second, because of the assumption of independence, the likelihood function of the GO-GARCH model is greatly simplified so that the conditional log-likelihood function is expressed as the sum of the individual log-likelihoods derived from the

[m] If the distributions are known the problem reduces to a classical maximum likelihood parametric estimation.

[n] Estimation by minimization of the mutual information was first proposed by Comon (1994) who derived a fundamental connection between cumulants, negentropy, and mutual information. The approximation of negentropy by cumulants was originally considered much earlier in Jones and Sibson (1987), while the connection between infomax and likelihood was shown in Pearlmutter and Parra (1997), and the connection between mutual information and likelihood was explicitly discussed in Cardoso (2000).

[o] According to Hyvarinen and Oja (2000), this can be partially justified by the fact that most of the research on ICA has also concentrated on the noise-free model and it has been shown with overwhelming empirical support across a number of different disciplines to be a very good approximation to a more complex model with noise added. As the estimation of the noise-free model has proved to be a very difficult task in itself, the noise-free model may also be considered a tractable approximation of the more realistic noisy model.

conditional marginal densities of the factors, plus a term for the matrix $\boldsymbol{A}$, estimated in the first step by the ICA algorithm:

$$L(\widehat{\varepsilon}_t|\boldsymbol{\theta},\boldsymbol{A}) = T\log\left|\boldsymbol{A}^{-1}\right| + \sum_{t=1}^{T}\sum_{i=1}^{N}\log\left(F(\boldsymbol{f}_{it}|\,\theta_i)\right), \tag{37}$$

where θ is the vector of unknown parameters in the marginal densities, for some distribution F. In the model of Van der Weide (2002), this distribution is the multivariate Normal, whereas in Broda and Paolella (2009) the multivariate affine GH distribution (maGH) of Schmidt et al. (2006) is used which is an alternative nonelliptical representation of the GH distribution with independent margins allowed to take separate values for skew and shape.[p] Based on the parametrization of Schmidt et al. (2006), the vector of returns $\boldsymbol{r}_t$, which is expressed as a linear transformation of independent factors $\boldsymbol{f}_t \in \mathbb{R}^N$ as in (32), is conditionally maGH distributed

$$\boldsymbol{r}_t|\mathfrak{F}_{t-1} \sim maGH_N(\boldsymbol{\mu}_t, \mathbf{H_t}, \boldsymbol{\omega}), \tag{38}$$

where $\omega = (\omega_1, \ldots, \omega_N)'$ and $\omega_i = (\lambda_i, \alpha_i, \beta_i)'$ represent the conditional shape and skew parameter vectors, respectively. In the model of Broda and Paolella (2009), which they term the Conditionally Heteroscedastic Independent Component Analysis of Generalized Orthogonal GARCH (**CHICAGO**), the standardized random variables z_{it} are assumed to be conditionally distributed as a standardized GH as discussed in (16), with a location-scale invariant parameterization such as the (ρ, ζ) or (ξ, χ).[q] An interesting property of the GO-GARCH model, arising from its affine representation, is the ability to identify closed-form expressions for the conditional coskewness and cokurtosis of asset returns.[r] The conditional comoments of $\boldsymbol{r}_t$ of order 3 and 4 are represented as tensor matrices,

$$\begin{aligned} \boldsymbol{M}_t^3 &= \boldsymbol{A}\boldsymbol{M}_{f,t}^3(\boldsymbol{A}' \otimes \boldsymbol{A}'), \\ \boldsymbol{M}_t^4 &= \boldsymbol{A}\boldsymbol{M}_{f,t}^4(\boldsymbol{A}' \otimes \boldsymbol{A}' \otimes \boldsymbol{A}'), \end{aligned}, \tag{39}$$

where $\boldsymbol{M}_{f,t}^3$ and $\boldsymbol{M}_{f,t}^4$ are the $(N \times N^2)$ conditional third comoment matrix and the $(N \times N^3)$ conditional fourth comoment matrix of the factors, respectively. $\boldsymbol{M}_{f,t}^3$ and $\boldsymbol{M}_{f,t}^4$ are given by

[p] As Schmidt et al. (2006) point out, the margins of a random vector that is GH distributed are not mutually independent for some choice of the scaling matrix. Echoing similar observations, Ferreira and Steel (2006) developed a multivariate skew-Student density with independent margins.

[q] See Section 2.3.5 of the **rugarch R** package vignette of Ghalanos (2018) for details about the generalized hyperbolic distribution.

[r] It is possible to go beyond these moments, but the notation becomes cumbersome and the benefits are likely to be marginal.

$$\boldsymbol{M}_{f,t}^{3}=\left[\boldsymbol{M}_{1,f,t}^{3},\boldsymbol{M}_{2,f,t}^{3},\ldots,\boldsymbol{M}_{N,f,t}^{3}\right] \tag{40}$$

$$\boldsymbol{M}_{f,t}^{4}=\left[\boldsymbol{M}_{11,f,t}^{4},\boldsymbol{M}_{12,f,t}^{4},\ldots,\boldsymbol{M}_{1N,f,t}^{4}|\ldots|\boldsymbol{M}_{N1,f,t}^{4},\boldsymbol{M}_{N2,f,t}^{4},\ldots,\boldsymbol{M}_{NN,f,t}^{4}\right], \tag{41}$$

where $\boldsymbol{M}_{k,f,t}^{3}$, $k=1,\ldots,N$ and $\boldsymbol{M}_{kl,f,t}^{4}$, $k,l=1,\ldots,N$ are the $(N\times N)$ submatrices of $\boldsymbol{M}_{f,t}^{3}$ and $\boldsymbol{M}_{f,t}^{4}$, respectively, with elements

$$m_{ijk,}^{3}\boldsymbol{f}_{,t}=E\left[f_{i,t}f_{j,t}f_{k,t}|\,\mathfrak{F}_{t-1}\right]$$
$$m_{ijkl,}^{4}\boldsymbol{f}_{,t}=E\left[f_{i,t}f_{j,t}f_{k,t}f_{l,t}|\,\mathfrak{F}_{t-1}\right].$$

Since the factors f_{it} can be decomposed as $z_{it}\sqrt{h_{it}}$, and given the assumptions on z_{it}, then $E\left[f_{i,t}f_{j,t}f_{k,t}|\,\mathfrak{F}_{t-1}\right]=0$. It is also true that for $i\neq j\neq k\neq l$ $E\left[f_{i,t}f_{j,t}f_{k,t}f_{l,t}|\,\mathfrak{F}_{t-1}\right]=0$ and when $i=j$ and $k=l$,

$$E\left[f_{i,t}f_{j,t}f_{k,t}f_{l,t}|\,\mathfrak{F}_{t-1}\right]=v_{it}^{2}v_{kt}^{2}.$$

Thus, under the assumption of mutual independence, all elements in the conditional comoments matrices with at least three different indices are zero. Finally, standardizing the conditional comoments one obtains conditional coskewness and cokurtosis of $\boldsymbol{r}_t$,

$$\begin{aligned}\boldsymbol{S}_{ijk,t}&=\frac{m_{ijk,t}^{3}}{\left(h_{i,t}h_{j,t}h_{k,t}\right)},\\ \boldsymbol{K}_{ijkl,t}&=\frac{m_{ijkl,t}^{4}}{\left(h_{i,t}h_{j,t}h_{k,t}h_{l,t}\right)},\end{aligned} \tag{42}$$

where $\boldsymbol{S}_{ijk,t}$ represents the asset coskewness between elements i,j,k of $\boldsymbol{r}_t$, $h_{i,t}$ the standard deviation of $\boldsymbol{r}_{i,t}$, and in the case of $i=j=k$ represents the skewness of asset i at time t, and similarly for the cokurtosis tensor $\boldsymbol{K}_{ijkl,t}$. Two natural applications of return comoments matrices are Taylor-type utility expansions in portfolio allocation and higher moment news impact surfaces (see, for example, Chapter 5 of Jondeau et al. (2007)).[s]

An important question that can also be addressed in this framework is the determination of the portfolio conditional density, an issue of vital importance in risk management application. For instance, the N-dimensional NIG distribution which arises as a special case of the GH distribution when $\lambda=-0.5$ is closed under convolution and particularly suited to problems in portfolio and risk management where a weighted sum of assets is considered. However, when the distributional parameters α and β, representing skew and shape, are allowed to vary per asset, as will likely be the case unless restrictions are imposed, this property no longer holds and numerical methods such as that

[s]An interesting extension with time varying dynamics for the skew and shape parameters, following the extension initially proposed by Hansen (1994), is discussed in the IFACD model of Ghalanos et al. (2015).

of the Fast Fourier Transform (*FFT*) are needed to derive the weighted density by inversion of the characteristic function of the scaled parameters.[t] In the case of the NIG distribution, this is greatly simplified because of the representation of the modified Bessel function for the GIG shape index (λ) with value -0.5 which was derived in Barndorff-Nielsen and Bläsild (1981); otherwise the characteristic function of the GH involves the evaluation of the modified Bessel function with complex arguments, which complicates the inversion. Let R_t^p be the portfolio return formed from a set of allocations weights $\boldsymbol{w}_t$,

$$R_t^p = \boldsymbol{w}_t' \boldsymbol{r}_t = \boldsymbol{w}_t' \boldsymbol{m}_t + \left(\boldsymbol{w}_t' \boldsymbol{A} \boldsymbol{V}_t^{1/2} \right) \boldsymbol{z}_t, \tag{43}$$

where $\boldsymbol{V}_t^{1/2}$ is estimated from the GARCH dynamics of $\boldsymbol{f}_t$. The model allows to express the portfolio variance, skewness, and kurtosis in closed form,

$$\begin{aligned} \sigma_{p,t}^2 &= \boldsymbol{w}_t' \mathbf{H}_\mathbf{t} \boldsymbol{w}_t, \\ s_{p,t} &= \frac{\boldsymbol{w}_t' \boldsymbol{M}_t^3 (\boldsymbol{w}_t \otimes \boldsymbol{w}_t)}{\left(\boldsymbol{w}_t' \mathbf{H}_\mathbf{t} \boldsymbol{w}_t \right)^{3/2}}, \\ k_{p,t} &= \frac{\boldsymbol{w}_t' \boldsymbol{M}_t^4 (\boldsymbol{w}_t \otimes \boldsymbol{w}_t \otimes \boldsymbol{w}_t)}{\left(\boldsymbol{w}_t' \mathbf{H}_\mathbf{t} \boldsymbol{w}_t \right)^2}, \end{aligned} \tag{44}$$

where $\mathbf{H}_\mathbf{t}$ and $\boldsymbol{M}_t^3$ and $\boldsymbol{M}_t^4$ are derived in (27) and (39), respectively. The portfolio conditional density may be obtained via the inversion of the characteristic function through the FFT method as in Chen et al. (2007) or by simulation. Provided that $\boldsymbol{z}_t$ is a N-dimensional vector of innovations, marginally distributed as one-dimensional standardized GH, the density of weighted asset return, $w_{it} r_{it}$, is

$$w_{i,t} r_{i,t} = (w_{i,t} m_{i,t} + \overline{w}_{i,t} z_{i,t}) \sim GH_{\lambda_i} \left(\overline{w}_{i,t} \mu_i + w_{i,t} m_{i,t}, |\overline{w}_{i,t}| \delta_i, \frac{\alpha_i}{|\overline{w}_{i,t}|}, \frac{\beta_i}{|\overline{w}_{i,t}|} \right), \tag{45}$$

where $\overline{w}_t'$ is equal to $\overline{w}_t' \boldsymbol{A} \boldsymbol{V}_t^{1/2}$, and $\overline{w}_{i,t}$ is the ith element of $\overline{\boldsymbol{w}}_t$, $m_{i,t}$ the conditional mean of the ith underlying asset.[u] In order to obtain the density of the portfolio, the individual weighted densities of $z_{i,t}$ must be summed. The characteristic function of the portfolio return R_t^p is

$$\begin{aligned} \varphi_R(u) &= \prod_{i=1}^{n} \varphi_{\overline{w} Z_i}(u) \\ &= \exp \left(\mathrm{i} u \sum_{j=1}^{d} \overline{\mu}_j + \sum_{j=1}^{d} \left(\frac{\lambda_j}{2} \log \left(\frac{\gamma}{\upsilon} \right) + \log \left(\frac{\boldsymbol{K}_{\lambda_j} \left(\overline{\delta}_j \sqrt{\upsilon} \right)}{\boldsymbol{K}_{\lambda_j} \left(\overline{\delta}_j \sqrt{\gamma} \right)} \right) \right) \right), \end{aligned} \tag{46}$$

[t]This effectively means that the weighted density is not necessarily NIG distributed.
[u]Here, μ is the distributional location parameter in the $(\alpha, \beta, \delta, \mu)$ parameterization of the GH distribution. See Prause (1999) for details.

```
library(rmgarch)
data("dji30ret")
cl<-makeCluster(4)
spec<-gogarchspec(mean.model=list(model="AR"), lag=1,
variance.model=list(model="eGARCH", variance.targeting=TRUE),
distribution.model="manig", ica="fastica")
model<-gogarchfit(spec, data=dji30ret[,1:4], cluster=cl)
stopCluster(cl)
```

SNIPPET 1 GO-GARCH example.

where $\gamma = \overline{\alpha}_j^2 - \overline{\beta}_j^2$, $\upsilon = \overline{\alpha}_j^2 - \left(\overline{\beta}_j + \mathrm{i}u\right)^2$, and $\left(\overline{\alpha}_j, \overline{\beta}_j, \overline{\delta}_j, \overline{\mu}_j\right)$ are the scaled versions of the parameters $(\alpha_i, \beta_i, \delta_i, \mu_i)$ as shown in (45). The density may be accurately approximated by FFT as follows,

$$f_R(r) = \frac{1}{2\pi}\int_{-\infty}^{+\infty} e^{(-\mathrm{i}ur)}\varphi_R(u)\mathrm{d}u \approx \frac{1}{2\pi}\int_{-s}^{s} e^{(-\mathrm{i}ur)}\varphi_R(u)\mathrm{d}u. \tag{47}$$

Once the density is formed by FFT inversion of the characteristic function, distribution, quantile, and sampling functions can be created.

The **rmgarch** R package of Ghalanos (2015b) implements all methods and properties described in this section for working with the GO-GARCH model with either a multivariate Normal or multivariate affine GH distribution, while the **gogarch** package of Pfaff (2009) implements maximum likelihood, method of moments, nonlinear least squares, and ICA estimation methods under the multivariate Normal. As a simple illustration we show in code Snippet 1 how one can define a GO-GARCH model based on the ICA method in the **rmgarch** package and estimate it. Note the ability to use parallel computational resources (*makeCluster*) to estimate the univariate GARCH model of the independent margins following the estimation of the mixing matrix $\boldsymbol{A}$. Table 2 provides the methods and functions available for working with the model which include forecasting, filtering,[v] simulation, visual inference, and computation of the weighted density, among others. We refer the reader to the documentation of the package for more details, as well as the documentation of the **gogarch** package for examples and methods available.

5 Conditional correlation GARCH models

Conditional correlation models are founded on a decomposition of the conditional covariance matrix into conditional standard deviations and correlations, so that they may be expressed in such a way that the univariate and

[v]Filtering new data with an existing set of parameters is equivalent to a 1-step ahead forecast.

TABLE 2 rmgarch **GO-GARCH model functions and methods**

Functions/ methods	Description	Input classes
gogarchspec	Model specification	NA
gogarchfit	Model estimation	1
gogarchforecast	1- to *n*-ahead forecasts	2
gogarchsim	Simulation	2,3
gogarchfilter	1-ahead ahead filtering	2,3
convolution	Calculates the weight density by FFT	2,3,4,5,6
fitted	Conditional mean equation fitted/forecasted values	2,3
residuals	Conditional mean equation residuals	2,3
coef	Coefficients of model	2,3
show	Summary of output	2,3,4,5,6
nisurface	News impact surface	2,3
gportmoments	Geometric portfolio moments	2,3,4,5,6
rcor	Conditional correlations	2,3,4,5,6
rcov	Conditional covariance	2,3,4,5,6
sigma	Conditional margin volatilities	2,3,4,5,6
rcoskew	Conditional coskewness	2,3,4,5,6
rcokurt	Conditional cokurtosis	2,3,4,5,6
betacovar	Conditional covariance beta to a benchmark	2,3
betacoskew	Conditional coskewness beta to a benchmark	2,3
betacokurt	Conditional cokurtosis beta to a benchmark	2,3
dfft	FFT density function (interpolated)	7
pfft	FFT distribution function (interpolated)	7
qfft	FFT quantile function (interpolated)	7
nportmoments	First four conditional portfolio moments from FFT interpolated density using quadrature integration	7
gogarchroll	Rolling estimation/forecasting	1

Note: The table provides a list of the methods and functions for working with GO-GARCH models based on the ICA transformation in the **rmgarch** package. The input classes are as follows: 1 = goGARCHspec, 2 = goGARCHfit, 3 = goGARCHfilter, 4 = goGARCHforecast, 5 = goGARCHsim, 6 = goGARCHroll, 7 = goGARCHfft, NA = not a method.

multivariate dynamics may be separated. In the constant conditional correlation (CCC) model of Bollerslev (1990), the covariance matrix can be decomposed into

$$\boldsymbol{H}_t = \boldsymbol{D}_t \boldsymbol{R} \boldsymbol{D}_t = \rho_{ij} \sqrt{h_{iit} h_{jjt}}, \tag{48}$$

where $\boldsymbol{D}_t = \text{diag}\left(\sqrt{h_{11,t}}, \ldots, \sqrt{h_{NN,t}}\right)$, and $\boldsymbol{R}$ is the positive definite CCC matrix. The conditional variances, $h_{ii,t}$, which can be estimated separately, are univariate GARCH processes:

$$h_t = \omega + \sum_{i=1}^{p} \boldsymbol{A}_i \varepsilon_{t-i} \odot \varepsilon_{t-i} + \sum_{i=1}^{q} \boldsymbol{B}_i h_{t-i}, \tag{49}$$

where $\omega \in \mathbb{R}^N$, $\boldsymbol{A}_i$ and $\boldsymbol{B}_i$ are $N \times N$ diagonal matrices, and $\odot$ denotes the Hadamard operator. The conditions for the positivity of the covariance matrix $\boldsymbol{H}_t$ are that $\boldsymbol{R}$ is positive definite, and the elements of ω and the diagonal elements of the matrices $\boldsymbol{A}_i$ and $\boldsymbol{B}_i$ are positive. In the extended CCC model (E-CCC) of Jeantheau (1998), the assumption of diagonal elements on $\boldsymbol{A}_i$ and $\boldsymbol{B}_i$ was relaxed, allowing the past squared errors and variances of the series to affect the dynamics of the individual conditional variances, and hence providing for a much richer structure, albeit at the cost of an increase in parameters.

The decomposition in (48) allows the log-likelihood at each point in time (LL_t), in the multivariate Normal case, to be expressed as

$$\begin{aligned} LL_t &= \frac{1}{2}\left(\log(2\pi) + \log|\boldsymbol{H}_t| + \varepsilon_t' \boldsymbol{H}_t^{-1} \varepsilon_t\right) \\ &= \frac{1}{2}\left(\log(2\pi) + \log|\boldsymbol{D}_t \boldsymbol{R} \boldsymbol{D}_t| + \varepsilon_t' \boldsymbol{D}_t^{-1} \boldsymbol{R}^{-1} \boldsymbol{D}_t^{-1} \varepsilon_t\right) \\ &= \frac{1}{2}\left(\log(2\pi) + 2\log|\boldsymbol{D}_t| + \log|\boldsymbol{R}| + z_t' \boldsymbol{R}^{-1} z_t\right), \end{aligned} \tag{50}$$

where $z_t = \boldsymbol{D}_t^{-1} \varepsilon_t$. This can be described as a term ($\boldsymbol{D}_t$) for the sum of the univariate GARCH model likelihoods, a term for the correlation ($\boldsymbol{R}$), and a term for the covariance which arises from the decomposition.

The assumption of constant correlation may in practice be unrealistic. When this assumption does not hold, a class of models termed Dynamic Conditional Correlation (DCC), due to Engle (2002) and Tse and Tsui (2002), and discussed at length in Engle (2009), allow for the correlation matrix to be time varying with dynamics, such that

$$\boldsymbol{H}_t = \boldsymbol{D}_t \boldsymbol{R}_t \boldsymbol{D}_t, \tag{51}$$

where the time-varying correlation matrix, $\boldsymbol{R}_t$, must be constrained to be positive definite. The most popular of these DCC models, due to Engle (2002), achieves this constraint by modeling a proxy process, $\boldsymbol{Q}_t$ as:

$$\begin{aligned}\boldsymbol{Q}_t &= \overline{\boldsymbol{Q}} + a\left(\boldsymbol{z}_{t-1}\boldsymbol{z}_{t-1}' - \overline{\boldsymbol{Q}}\right) + b\left(\overline{\boldsymbol{Q}}_{t-1} - \overline{\boldsymbol{Q}}\right) \\ &= (1-a-b)\overline{\boldsymbol{Q}} + a\boldsymbol{z}_{t-1}\boldsymbol{z}_{t-1}' + b\boldsymbol{Q}_{t-1}\end{aligned}, \tag{52}$$

where a and b are nonnegative scalars controlling the reaction to shocks and persistence, respectively, with the condition that $a+b<1$ imposed to ensure stationarity and positive definiteness of $\boldsymbol{Q}_t$. $\overline{\boldsymbol{Q}}$ is the unconditional matrix of the standardized errors $\boldsymbol{z}_t$ which enters the equation via the covariance targeting part $(1-a-b)\overline{\boldsymbol{Q}}$, and $\boldsymbol{Q}_0$ is positive definite. The correlation matrix $\boldsymbol{R}_t$ is then obtained by rescaling $\boldsymbol{Q}_t$ such that,

$$\boldsymbol{R}_t = \operatorname{diag}(\boldsymbol{Q}_t)^{-1/2}\boldsymbol{Q}_t\operatorname{diag}(\boldsymbol{Q}_t)^{-1/2}. \tag{53}$$

The log-likelihood function in (49) can be decomposed more clearly into a volatility and correlation component by adding and subtracting $\varepsilon_t'\boldsymbol{D}_t^{-1}\boldsymbol{D}_t^{-1}\varepsilon_t = \boldsymbol{z}_t'\boldsymbol{z}_t$, so that:

$$\begin{aligned}LL &= \frac{1}{2}\sum_{i=1}^{T}\left(N\log(2\pi) + 2\log|\boldsymbol{D}_t| + \log|\boldsymbol{R}_t| + \boldsymbol{z}_t'\boldsymbol{R}_t^{-1}\boldsymbol{z}_t\right) \\ &= \frac{1}{2}\sum_{i=1}^{T}\left(N\log(2\pi) + 2\log|\boldsymbol{D}_t| + \varepsilon_t'\boldsymbol{D}_t^{-1}\boldsymbol{D}_t^{-1}\varepsilon_t\right) \\ &\quad -\frac{1}{2}\sum_{i=1}^{T}\left(\boldsymbol{z}_t'\boldsymbol{z}_t + \log|\boldsymbol{R}_t| + \boldsymbol{z}_t'\boldsymbol{R}_t^{-1}\boldsymbol{z}_t'\right) \\ &= LL_V(\theta_1) + LL_R(\theta_1,\theta_2),\end{aligned} \tag{54}$$

where $LL_V(\theta_1)$ is the volatility component with parameters θ_1, and $LL_R(\theta_1,\theta_2)$ the correlation component with parameters θ_1 and θ_2. In the Multivariate Normal and Laplace cases, the volatility component is the sum of the individual GARCH likelihoods which can be jointly maximized by separately maximizing each univariate model. In other distributions, such as the multivariate Student discussed in Section 3.2, where additional distributional parameters must be the same for all margins for that distribution to be closed under summation and affine linear transformations, the estimation must be performed in one step. The seperability of the likelihood into two parts, together with the use of covariance targeting, means that very large-scale systems may be estimated quickly and in parallel. However, as the number of variables grows, it becomes questionable whether the scalar model can adequately capture the complex dynamics of the underlying process.

In Cappiello et al. (2006), the scalar DCC is generalized to include asymmetry and lagged interactions in the form of the Asymmetric Generalized DCC (*AGDCC*) where the dynamics of $\boldsymbol{Q}_t$ are:

$$\begin{aligned}\boldsymbol{Q}_t = &\left(\overline{\boldsymbol{Q}} - \boldsymbol{A}'\overline{\boldsymbol{Q}}\boldsymbol{A} - \boldsymbol{B}'\overline{\boldsymbol{Q}}\boldsymbol{B} - \boldsymbol{G}'\overline{\boldsymbol{Q}}^{-}\boldsymbol{G}\right) + \boldsymbol{A}'\boldsymbol{z}_{t-1}\boldsymbol{z}_{t-1}'\boldsymbol{A} \\ &+ \boldsymbol{B}'\boldsymbol{Q}_{t-1}\boldsymbol{B} + \boldsymbol{G}'\boldsymbol{z}_t^{-}\boldsymbol{z}_t'^{-}\boldsymbol{G},\end{aligned} \tag{55}$$

where $\boldsymbol{A}$, $\boldsymbol{B}$, and $\boldsymbol{G}$ are the $N \times N$ parameter matrices, $\boldsymbol{z}_t^{-}$ are the zero-threshold standardized errors which are equal to $\boldsymbol{z}_t$ when less than zero and zero otherwise, $\overline{\boldsymbol{Q}}_t$ and $\overline{\boldsymbol{Q}}^{-}$ the unconditional matrices of $\boldsymbol{z}_t$ and $\boldsymbol{z}_t^{-}$, respectively. Because of its high dimensionality, restricted models have been used including the scalar, diagonal, and symmetric versions with the specifications nested being

- DCC: $\boldsymbol{G} = [0], \boldsymbol{A} = \sqrt{a}, \boldsymbol{B} = \sqrt{b}$
- ADCC: $\boldsymbol{G} = \sqrt{g}, \boldsymbol{A} = \sqrt{a}, \boldsymbol{B} = \sqrt{b}$
- GDCC: $\boldsymbol{G} = [0]$.

Covariance targeting in such high-dimensional models where the parameters are no longer scalars creates difficulties in imposing positive definiteness during estimation while at the same time guaranteeing a global optimum solution. More substantially, Aielli (2013) points out that the estimation of $\overline{\boldsymbol{Q}}_t$ as the empirical counterpart of the correlation matrix of $\boldsymbol{z}_t$ in the DCC model is inconsistent since $\mathrm{E}[\boldsymbol{z}_t\boldsymbol{z}_t] = \mathrm{E}[\boldsymbol{R}_t] \neq \boldsymbol{E}\left[\overline{\boldsymbol{Q}}_t\right]$. He proposes instead the *cDCC* model which includes a corrective step which eliminates this inconsistency, albeit at the cost of targeting which is no longer allowed. Whether the identified inconsistency is significant enough to merit widespread adoption is still an open question, since the elimination of the two-step approach also eliminates most of the advantages of using a DCC-type model over the BEKK, a point emphasized by Caporin and McAleer (2012) who questioned the merits of the DCC model over the BEKK model with covariance targeting which has more consistent properties.

Other notable DCC extensions have included the Smooth and double Smooth Transition Conditional Correlation models of Silvennoinen and Teräsvirta (2009) and the Regime Switching Dynamic Correlation of Pelletier (2006). An interesting compromise in the modeling of the dynamics in the AGDCC context was proposed by Billio et al. (2006) in terms of a block-diagonal structure so that the dynamics among groups of highly correlated securities is the same. The model may parsimoniously be represented as:

$$\boldsymbol{Q}_t = cc' + \sum_{j=1}^{P}\left(I_g a_j\right)\left(I_g a_j\right)' \odot \varepsilon_{t-j}\varepsilon_{t-j}' + \sum_{j=1}^{Q}\left(I_g b_j\right)\left(I_g b_j\right)' \odot \boldsymbol{Q}_{t-j}, \tag{56}$$

where I_g is the *assets* $\times$ *groups* logical matrix of group exclusive membership. This is a very flexible representation allowing a large range of representations, from a single group driving all dynamics (like the DCC), to each asset having its own group (like the GDCC). Unfortunately, without specialized restrictions correlation targeting is lost, but the model still remains feasible for a not too

large number of groups. Finally, Kroner and Ng (1998) formulated an omnibus model which nests the VEC, BEKK, F-GARCH, CCC, and DCC, and termed the Generalized Dynamic Covariance (*GDC*) Model:

$$\boldsymbol{H}_t = \boldsymbol{D}_t \boldsymbol{R}_t \boldsymbol{D}_t + \boldsymbol{\Phi} \odot \Theta_t, \tag{57}$$

where $\boldsymbol{D}_t = d_{ij,t}$, $d_{ii,t} = \sqrt{\theta_{ii,t}} \quad \forall i$, and $d_{ij,t} = 0 \quad \forall i \neq j$, $\odot$ is the Hadamard operator, $\boldsymbol{R}_t = \rho_{ij,t}$, and $\Theta = \theta_{ij,t}$ following BEKK dynamics as in (5). Depending on the parameter restrictions, various models arise such as the BEKK model when $\boldsymbol{R}$ is diagonal and $\boldsymbol{\Phi}$ with off-diagonal values of 1. Other restrictions, leading to other models, are given in Proposition 1 of Kroner and Ng (1998). The authors also describe in the same paper an asymmetric version of this model by adjusting the BEKK dynamics in $\theta_{ij,t}$ to incorporate an asymmetry term for the zero-threshold shocks, which is a natural generalization from such univariate models as the GJR-GARCH and T-GARCH of Glosten et al. (1993) and Zakoian (1994), respectively. Like in the case of the family GARCH model of Hentschel (1995) where comparison of nested models was made via the news impact curve of Engle and Ng (1993), the authors generalize the curve to a surface function providing for some revealing visual insights into the different multivariate dynamics.

An interesting extension, which makes use of the flexible decomposition of the covariance matrix given in (48), is in the use of copula distributions introduced in Section 3.5. The extension of the static copula approach to dynamic models, and in particular GARCH, was investigated by Patton (2006) who extended and proved the validity of Sklarś theorem for the conditional case. One simple direction is to introduce correlation dynamics to a copula distribution, with different marginal dynamics and distributions. Let the n-dimensional random vector of asset returns $\boldsymbol{r}_t = r_{it}, \ldots, r_{Nt}$ follow a copula GARCH model with joint distribution given by:

$$F(\boldsymbol{r}_t | \boldsymbol{\mu}_t, \boldsymbol{h}_t) = C(F_1(r_{1t} | \mu_{1t}, h_{1t}), \ldots, F_N(r_{Nt} | \mu_{Nt}, h_{Nt})) \tag{58}$$

where F_i, $i = 1, \ldots, N$ is the conditional distribution of the ith marginal series density, C is the N-dimensional copula. The conditional mean $\mathrm{E}[\mathrm{r}_{\mathrm{it}} | \Im_{\mathrm{t}-1}] = \mu_{\mathrm{it}}$ and the conditional variance h_{it} follows, for simplicity of exposition, a GARCH(1,1) process:

$$r_{it} = \mu_{it} + \varepsilon_{it}, \varepsilon_{it} = \sqrt{h_{it}} z_{it}, \tag{59}$$

$$h_{it} = \omega + \alpha_i \varepsilon_{t-1}^2 + \beta_i h_{it-1} \tag{60}$$

where z_{it} are i.i.d. random variables which conditionally follow some distribution with the requisite properties. Consider a Student copula with conditional density at time t is given by:

$$c_t(u_{it}, \ldots, u_{Nt} | \boldsymbol{R}_t, \nu) = \frac{f_t\left(F_i^{-1}(u_{it}|\nu), \ldots, F_i^{-1}(u_{Nt}|\nu) | \boldsymbol{R}_t, \eta\right)}{\prod_{i=1}^{N} f_i\left(F_i^{-1}(u_{it}|\nu)|\nu\right)}, \tag{61}$$

where $u_{it}=F_{it}(r_{it}|\mu_{it},h_{it},\xi_i)$ is the probability integral transform of each series by its conditional distribution F_{it} with any additional distributional parameters represented by ξ_i and estimated via the first-stage GARCH process, $F_i^{-1}(u_{it}|\nu)$ represents the quantile transformation of the uniform margins subject to the common shape parameter of the multivariate Student density, $f_t(.|\boldsymbol{R}_t,\nu)$ is the multivariate density of the Student distribution with conditional correlation $\boldsymbol{R}_t$ and shape parameter ν, and $f_i(.|\nu)$ is the univariate margins of the multivariate Student distribution with common shape parameter ν. The dynamics of $\boldsymbol{R}_t$ are assumed to follow an AGDCC model, though it is more common to use a restricted scalar DCC model for not too large a number of series. The joint density of the two-stage estimation is then given by:

$$f(\boldsymbol{r}_t|\boldsymbol{\mu}_t,\boldsymbol{h}_t,\boldsymbol{R}_t,\nu)=c_t(u_{it},\ldots,u_{Nt}|\boldsymbol{R}_t,\nu)\prod_{i=1}^{N}\frac{1}{\sqrt{h_{it}}}f_{it}(z_{it}|\xi_i) \tag{62}$$

where the likelihood is composed of a part due to the joint DCC copula dynamics and a part due to the first-stage univariate GARCH dynamics. A DCC-Student Copula model with Student margins was estimated by Ausin and Lopes (2010) using a Bayesian setup who used this to study a risk management application for the DAX and Dow Jones indices.

The **rmgarch R** package includes the CCC, DCC, aDCC, and Flexible DCC models with multivariate Normal, Laplace, and Student distributions, as well as the Copula Normal and Student. The **bayesDccGarch** of Fiorucci et al. (2016) provides a Bayesian estimation framework for DCC models described in Fioruci et al. (2014), and the **ccgarch** package of Nakatani (2008) has estimation, simulation, and testing funtions for the CCC, DCC, and extended CCC models and discussed in Nakatani and Teräsvirta (2009). As a simple illustration we show in code Snippet 2 how one can define a DCC model in the **rmgarch** package and estimate it. Note again the ability to use parallel computational

```
library(rmgarch)
data("dji30ret")
cl<-makeCluster(4)
uspec = multispec(list(
ugarchspec(mean.model=list(armaOrder=c(1,0)),variance.model=
   list(model="eGARCH")),
ugarchspec(mean.model=list(armaOrder=c(1,1)),variance.model=
   list(model="sGARCH")),
ugarchspec(mean.model=list(armaOrder=c(2,0)),variance.model=
   list(model="gjrGARCH")),
ugarchspec(mean.model=list(armaOrder=c(1,0)),variance.model=
   list(model="csGARCH"))
))
spec<-dccspec(uspec, model = "DCC", distribution="mvnorm")
model<-dccfit(spec, data=dji30ret[,1:4])
stopCluster(cl)
```

SNIPPET 2 DCC example.

TABLE 3 rmgarch **DCC model functions and methods**

Functions/ methods	Description	Input classes
dccspec	Model specification for the univariate GARCH models, conditional mean, joint dynamics, and distribution	NA
dccfit	Model estimation	1
dccforecast	1- to *n*-ahead forecasts	2
dccsim	Simulation	1,2
dccfilter	1-ahead ahead filtering	2,3
fitted	Conditional mean equation fitted/forecasted values	2,3,4,5,6
residuals	Conditional mean equation residuals	2,3
coef	Coefficients of model	2,3
show	Summary of output	2,3,4,5,6
nisurface	News impact surface	2,3
rcor	Conditional correlations	2,3,4,5,6
rcov	Conditional covariance	2,3,4,5,6
sigma	Conditional margin volatilities	2,3,4,5,6
dccroll	Rolling estimation/forecasting	1

Note: The table provides a list of the methods and functions for working with DCC models in the **rmgarch** package. The input classes are as follows: 1 = DCCspec, 2 = DCCfit, 3 = DCCfilter, 4 = DCCforecast, 5 = DCCsim, 6 = DCCroll, NA = not a method. Distributions allowed as Multivariate Normal, Laplace, and Student (QML), with a number of different options for the first-stage GARCH model dynamics, and the asymmetric or symmetric DCC, or the flexible DCC model of Billio et al. (2006) for the joint dynamics.

resources (*makeCluster*) to estimate the univariate GARCH models in the first-stage prior to the second-stage estimation of the joint dynamics. Table 3 provides the methods and functions available for working with the model which include forecasting, filtering, simulation, and visual inference, among others. We refer the reader to the documentation of the package for more details, and Section 7 of Engle et al. (1990) for the approximation to multistep ahead forecasting.

6 BIP and GAS MGARCH models

The estimated covariance updating equation of MGARCH models is mostly used as a filter to predict the conditional covariance based on the observed return series. The already discussed MGARCH models tend to use the same filter irrespective of the shape of the distribution function. This typically leads

to a large spike in the conditional covariance prediction following an extreme return realization. In order to dampen the effect of outliers on covariance predictions, Boudt and Croux (2010) and Boudt et al. (2013) recommend to use robust MGARCH filters that have the property of Bounded Innovation Propagation (BIP). The resulting model is called BIP-MGARCH for which they propose robust M-estimators under the assumption of elliptical innovations. The BIP-BEKK model corresponding to the BEKK model in (5) is given by:

$$\boldsymbol{H}_t = \boldsymbol{C}'\boldsymbol{C} + \sum_{k=1}^{K}\sum_{j=1}^{q} \boldsymbol{A}'_{jk}\widetilde{\varepsilon}_{t-j}\widetilde{\varepsilon}'_{t-j}\boldsymbol{A}_{jk} + \sum_{k=1}^{K}\sum_{j=1}^{p} \boldsymbol{B}'_{jk}\boldsymbol{H}_{t-j}\boldsymbol{B}_{jk}, \tag{63}$$

with

$$\widetilde{\varepsilon}_t = \varepsilon_t \sqrt{w\left(\varepsilon'_t \boldsymbol{H}_t^{-1}\varepsilon_t\right)}. \tag{64}$$

The weight function $w(\cdot)$ must be such that the effect of ε_t on $\boldsymbol{H}_t$ is bounded. Boudt and Croux (2010) use the following weight function

$$w(z) = \begin{cases} 1 & \text{if } z \le c_1 \\ 1-(1-c_1/z)^3 & \text{if } c_1 < z \le c_2 \\ (c_2/z)\left(1-(1-c_1/c_2)^3\right) & \text{else.} \end{cases} \tag{65}$$

They set the parameters c_1 and c_2 equal the 99% and 99.9% quantile of the distribution of the squared Mahalanobis Distances (MD) $\varepsilon'_t \boldsymbol{H}_t^{-1}\varepsilon_t$. The left panel of Fig. 1 shows this weight function in case the conditional distribution of ε_t is bivariate Normal (top plot) or Student t_4 (bottom plot). Note that only the observations with an extremely large MD are downweighted and that the weighting depends on the distributional assumption. In the right panel we also plot the function $w(z)z$, which is of interest since if z is the squared MD of $\widetilde{\varepsilon}_t$, then $w(z)z$ is the squared MD of ε_t. Note that the downweighting is such that the function $w(z)z$ is nondecreasing and bounded by $w(c_2)c_2$. The smoothness of the weight function is needed to avoid numerical problems in the parameter estimation.

The BIP-BEKK model can be seen as an ad hoc robustification of the BEKK filters. Similar BIP-DCC filters were proposed in Boudt et al. (2013). An elegant alternative to take the shape of the distribution function into account is the class of Generalized Autoregressive Score (GAS) and Dynamic Score models, proposed by Creal et al. (2013) and Harvey (2013) at Vrije Universiteit Amsterdam and Cambridge University, respectively. They developed a general framework to specifying the time-varying parameters of a conditional distribution function. The key feature of their framework is that the score of the conditional density function is used as the driver of the time variation in the parameters, making it possible to obtain the likelihood in closed form through a standard prediction error decomposition.

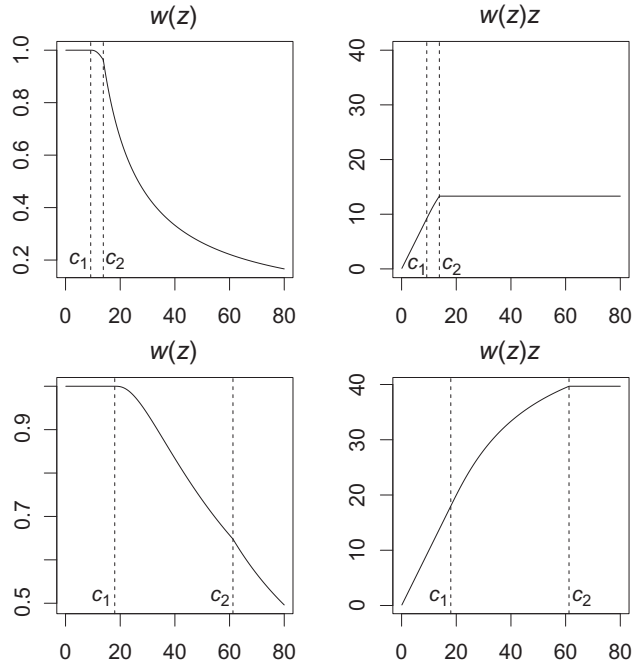

FIG. 1 Plot of the functions $w(z)$ and $w(z)z$ used in the bivariate BIP-BEKK model. The parameters c_1 and c_2 equal to the 99% and 99.9% quantiles of the squared MD under Gaussian (*upper panel*) and Student t_4 innovations (*lower panel*).

More formally, suppose that the variable of interest is $\mathbf{r_t}$ with conditional density function $f(\ldots)$. The conditional density depends on a vector of time-varying parameters denoted by $\theta_\mathbf{t} \in \mathbf{\Theta} \subseteq \Re^\mathbf{J}$. It typically contains the unique elements in the conditional covariance matrix $\boldsymbol{H}_t$, but may also consist of location and shape parameters, among others. Usually, the parameter space of $\theta_\mathbf{t}$ is restricted by various conditions, such as the requirement of positive definiteness for the conditional covariance. The standard solution under the GAS framework is to work with parameter transformations such that the parameter of interest $\theta_\mathbf{t}$ is in the parameter space. More precisely, when the unrestricted parameter vector is denoted by $\widetilde{\theta}_\mathbf{t} \in \Re^\mathbf{J}$, the GAS model uses a link function $\Lambda(\cdot)$ to map the transformed parameter $\widetilde{\theta}_\mathbf{t} \in \Re^\mathbf{J}$ into the parameter of interest $\theta_\mathbf{t}$. The evolution in the time-varying parameter vector $\widetilde{\theta}_\mathbf{t}$ is driven by the scaled score of the conditional density function, defined as:

$$\mathbf{s_t} = \mathbf{S_t} \frac{\partial \log \mathbf{f}(\mathbf{r_t}|\theta_\mathbf{t})}{\partial \widetilde{\theta}_\mathbf{t}},$$

where the matrix $\mathbf{S_t}$ is a $J \times J$ positive definite scaling matrix known at time t.[w] The quantity $\mathbf{s_t}$ indicates the direction to update the vector of parameters from $\theta_\mathbf{t}$, to $\theta_{\mathbf{t+1}}$, acting as a steepest ascent algorithm for improving the model's local fit given the current parameter position. This updating procedure resembles the well-known Newton–Raphson algorithm.

The updating equation for the unconstrained parameter vector is given by a linear function of the score, together with an autoregressive component:

$$\widetilde{\theta}_{\mathbf{t+1}} = \kappa + \mathbf{A}\mathbf{s_t} + \mathbf{B}\widetilde{\theta}_\mathbf{t}, \tag{66}$$

where κ, $\mathbf{A}$, and $\mathbf{B}$ are matrices of coefficients with proper dimensions.

A general implementation of univariate and multivarate GAS models can be found in the **R** package **GAS** (Ardia et al., 2018b; Catania et al., 2017). As a simple illustration we show in code Snippet 3 how one can define a multivariate GAS model in the **GAS** package and estimate it. Table 4 provides the methods and functions available for working with the model which include forecasting, filtering, simulation, and inference, among others. We refer the reader to the documentation of the package for more details.

The approach of specifying the time variation in all the distribution parameters jointly as a function of the conditional score is unfortunately not feasible in large-scale applications due to a curse of dimensionality. Creal et al. (2011) acknowledge this shortcoming and propose to use a time-varying copula specification in order to model the variances separately from the correlations. As in the DCC model of Engle (2002), it is then straightforward to combine the conditional variances and correlations into an estimate for the conditional covariance matrix $\boldsymbol{H}_t$. We illustrate this copula-approach next in the case of a GAS variance model assuming a Student t marginal distribution, and a GAS correlation model under the assumption of a bivariate t-copula specification.

In terms of modeling the conditional variance dynamics, we focus here on the case where a Student t distribution is assumed. Several GAS models exist

```
library("GAS")
data("dji30ret", package = "GAS")
mGASSpec <- MultiGASSpec(Dist = "mvt",
ScalingType = "Identity",
GASPar = list(scale = TRUE, correlation = TRUE))
model <- MultiGASFit(data = dji30ret[, 1:4],
GASSpec = mGASSpec)
```

SNIPPET 3 GAS example.

[w] Creal et al. (2013) suggest to set the scaling matrix S_t to a power $\gamma > 0$ of the inverse of the Information Matrix of $\widetilde{\theta}_t$ to account for the variance of the score. When $\gamma = 0$, $\mathbf{S_t}$ equals the identity matrix and there is no scaling. If $\gamma = 1$ (resp. $\gamma = \frac{1}{2}$), the conditional score is premultiplied by the inverse of (the square root of) its covariance matrix.

TABLE 4 GAS **multivariate GAS model functions and methods**

Functions/methods	Description	Input classes
MultiGASSpec	Model specification	NA
MultiGASFit	Model estimation	1
getFilteredParameters	1-ahead ahead filtering	1
ConfidenceBands	Confidence bands for the filtered parameters	1
getMoments	Extract conditional moments	1, 3, 5
getForecast	Extract parameter forecast	3, 5
LogScore	Extract log scores	3, 5
MultiGASFor	1- to *h*-ahead forecasts	2
MultiGASSim	Simulation	2
residuals	Conditional mean equation residuals	2
coef	Coefficients of model	2
show	Summary of output	3,4,5
summary	Summary of output	2
MultiGASRoll	Rolling estimation/forecasting	1

Note: The table provides a list of the methods and functions for working with multivariate GAS models in the **GAS** package. The input classes are as follows: 1 = mGASSpec, 2 = mGASFit, 3 = mGASFor, 4 = mGASSim, 5 = mGASRoll.

under this framework. The Beta-t-EGARCH model introduced by Harvey et al. (2008) uses the exponential function as link function. The Beta-t-GARCH model of Harvey et al. (2008) and the *t*-GAS model of Creal et al. (2013) use no transformation. The latter then leads to a GAS volatility model that is the close to the GARCH(1,1) model and for which the estimates are published on https://vlab.stern.nyu.edu. Under this model, the conditional variance for asset i with zero mean and ν_i degrees of freedom is given by:

$$h_{ii,t+1} = \omega_i + \alpha_i \frac{\nu_i + 3}{\nu_i} \left(\frac{\nu_i + 1}{\nu_i - 2 + \epsilon_{i,t}^2 / h_{ii,t}} \epsilon_{i,t}^2 - h_{ii,t} \right) + \beta_i h_{ii,t}. \tag{67}$$

Note that if $\nu_i = \infty$, the GAS-*t* volatility model collapses to a traditional GARCH model and the score has a quadratic impact on the conditional variance. This can be seen as well in Fig. 2, where we show the scaled score for various value of ν_i. Note that, the more fat-tailed the return distribution is, the more their effect on future variance is dampened due to to downweighting

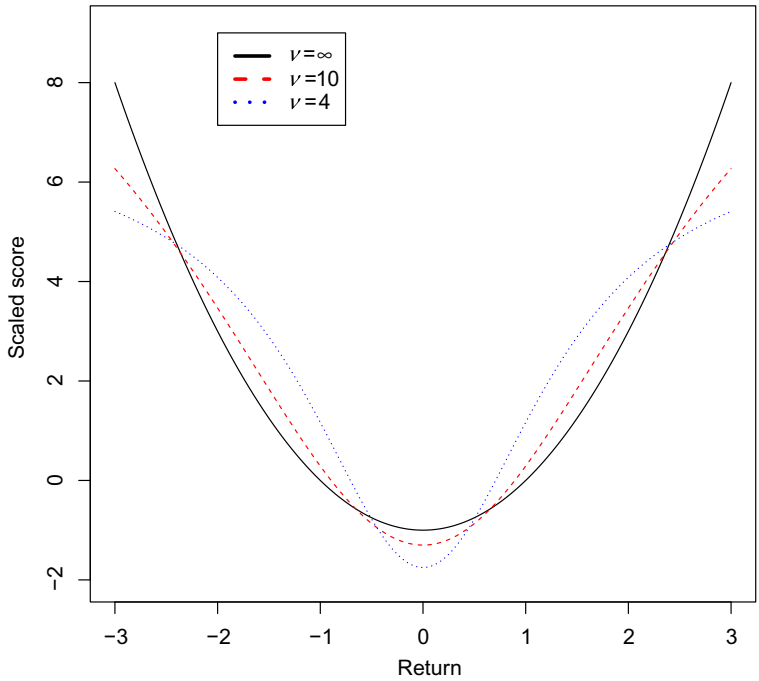

FIG. 2 Scaled score used as driver in the *t*-GAS conditional variance model for various values of ν_i when $h_{ii,t}=1$.

when ν_i is small. This is desired, since in case of a fat-tailed distribution, a large value of $\epsilon_{i,t}/h_{i,t}^{1/2}$ may as well be a tail realization and thus does not necessitate a substantial increase in the conditional variance.

We now discuss the GAS correlation model under the assumption of a bivariate *t*-copula specification. Besides the use of hyperspherical coordinate transformation, Creal et al. (2011) discuss also the approach to decompose the correlation matrix $\mathbf{R_t}$ as $\mathbf{\Delta_t^{-1}Q_t\Delta_t^{-1}}$, where $\mathbf{Q_t}$ is a symmetric positive definite matrix and $\mathbf{\Delta_t}$ is a diagonal matrix whose nonzero elements equal the square root of the diagonal elements of $\mathbf{Q_t}$. They then use the GAS framework in (66) to obtain a score-driven calibration of the time variation in $\text{vech}(\mathbf{Q_t})$. The score takes the shape of the student *t* copula into account, together with the surprise in the realized correlations compared to the predicted ones. We refer to Creal et al. (2011) for the general specification, and limit ourselves here to summarizing the discussion in Creal et al. (2011) regarding the bivariate case with fixed unit variance, correlation parameter ρ_t, and Gaussian copula. Then, the scaled score is given by:

$$s_t = \frac{2}{\left(1-\rho_t^2\right)^2}\left[\left(1+\rho_t^2\right)\left(y_{1,t}y_{2,t}-\rho_t\right)-\rho_t\left(y_{1,t}^2+y_{2,t}^2-2\right)\right]. \tag{68}$$

The first term increases the conditional correlation when the realized correlation exceeds the conditional correlation, while the second term attenuates this correlation increase in case of a large dispersion in the input vector. In fact, the correlation signal of $(1, 1)$ is much stronger than $(1/4, 4)$, even though their cross product is the same. Boudt et al. (2012) present similar expressions for the scaled score in case of the t-copula. The most important change is that, alike in the univariate case, the input data are then also downweighted when they have a large Mahalanobis distance. The more fat-tailed the t-copula is, the larger the downweighting in order to compensate for the fact that large deviations of realized correlations from predicted ones may be tail events and thus do not necessarily imply changes in the conditional correlation.

The application of the GAS framework to create MGARCH models is still an active field of research, as can be seen on the overview website at http://www.gasmodel.com/. A large number of authors have worked out the GAS dynamics in case of a more flexible distribution function, like the univariate skewed t distribution (see, e.g., Harvey and Sucarrat (2014) and Ardia et al. (2018a)), the generalized hyperbolic skewed t distribution (Lucas et al., 2014), or the Wishard distribution (Gorgi et al., 2019) and general finite mixture of distributions (Catania, 2016). Others have generalized the GAS specification to account for regime switches (Boudt et al., 2012; Catania, 2018). In terms of attempts at obtaining GAS models that are feasible in high dimensions, we refer the reader to Boudt et al. (2012) and Lucas et al. (2017) for an analysis assuming (block) equicorrelation, and to Creal et al. (2011, 2014) for the use of dynamic factors under the GAS framework.

7 MGARCH models using high-frequency returns

Availability of intraday return data has spawned a new class of conditional covariance models; most of which are correlaries to traditional multivariate approaches including MGARCH. These models contain more information about covariation between assets than traditional approaches due to frequent measurement throughout the trading day. They involve the extra, and often expensive (in terms of data acquisition), step of calculating a realized covariance matrix to replace the cross product of errors in the traditional GARCH equations. This data expense can be justified for those that need a relatively short response time; these realized covariance models are expected to perform well during market shocks where the level of volatility and correlation is subject to abrupt changes. In a less expensive option Payseur (2008) and Laurent et al. (2012) use realized covariance on a set of high-frequency data to evaluate which low-frequency option is the best for the proposed data and application; this work should be extended to include the Realized GARCH models in this section. The availability of **R** packages in the realm of conditional covariance modeling using intraday data is sparse; however, we summarize the popular realized models below and highlight packages that do exist. We hope that

this summary can help spur contributions to current packages or the creation of new packages.

First, some notation, all of the conditional covariance measures in this section require a realized covariance measure, $\boldsymbol{V}_{t-1}$, on day $t-1$ as an input for the conditional covariance calculation, $\boldsymbol{H}_t$, on day t. Various realized covariance measures exist. The standard one is the sum of outer products of all high-frequency returns in a day

$$\boldsymbol{V}_t = \boldsymbol{RC}_t^{(m)} = \sum_{i=1}^{m} \boldsymbol{r}_{t,i}\boldsymbol{r}'_{t,i}, \tag{69}$$

where m is an equally spaced subinterval within the trading day, such as 5 min or 10 s, $\boldsymbol{r}_{t,i}$ is a vector of asset returns for day t and intraday period i.

Andersen et al. (2003) show that, in the special case that prices are realization of a Brownian semimartingale diffusion, $\boldsymbol{RC}_t^{(m)}$ converges to the integrated covariance as the frequency of the intradata approaches infinity ($m \to \infty$); therefore, realized covariance provides an accurate measure of daily covariance. In practice, microstructure noise and jumps in the price level of an asset introduce bias in the diagonal elements of the covariance matrix—the variances. Covariance estimation of asset pairs over short time periods underestimates the degree of dependence between assess due to asynchronicity between asset observations, commonly known as the Epp's effect (Epps, 1979). Finally, the outer product of intraday returns does not necessarily lead to invertable positive semidefinite covariance matrices which are needed portfolio and risk managent.

Various methods address the above challenges, but unfortunately there is no "one-size-fits-all" method. Barndorff-Nielsen and Shephard (2004) use bipower covariation to create a jump-robust estimator. Boudt et al. (2011) use outlyingness covariation and Mancini and Gobbi (2012) use a threshold approach to create jump-robust esimates that also lead to positive semidefinite matrices. Zhang (2011) uses a two-timescale approach to eliminate microstructure bias, while Boudt and Zhang (2015) adds robustness to price jumps to this method. By averaging across different frequencies the estimator of Aït-Sahalia et al. (2005) yields an unbiased and positive semidefinate covariance matrix, while Barndorff-Nielsen et al. (2009, 2011) achieve the same by using with a kernel approach. All of the estimators above are available in **R** in the **highfrequency** package of Boudt et al. (2018). Further realized covariance kernel improvements by Hautsch et al. (2010), Lunde et al. (2016), and Boudt et al. (2017) are not yet included in this package.

Squaring intraday returns leaves out information from close of day $t-1$ to open of day t, this overnight period accounts for a large percentage of daily volatiliy. The literature presents two approaches in dealing with this issue. The first approach consists of adding a squared overnight return to the realized covariance, $(\boldsymbol{V}_{t-1}+\boldsymbol{\eta}_{t-2}\boldsymbol{\eta}'_{t-2})$ where $\boldsymbol{\eta}_{t-2}$ is the close-to-open return of

the previous day; and use close-to-close squared daily returns as the benchmark. A second approach is to model the open-to-close intraday covariance only; and benchmark the realized results to squared open-to-close return. These two approaches trade-off the lack of precision in estimating integrated covariance using $\boldsymbol{\eta}_{t-2}\boldsymbol{\eta}_{t-2}'$ with the information loss from ignoring overnight return. For the rest of this section we use $\boldsymbol{V}_{t-1}$ to represent both cases.

Estimated correctly, realized covariance provides an accurate measure of daily covariance. Below we discuss possible enhancements to MGARCH models that use $\boldsymbol{V}_{t-1}$ instead of the traditional daily squared return.

7.1 Realized BEKK

The general framework covered below follows from the BEKK(p,q,1) model (Eq. 5), with the de-meaned squared return, $\boldsymbol{\varepsilon}_{t-j}\boldsymbol{\varepsilon}_{t-j}'$, replaced by $\boldsymbol{V}_{t-j}$.

$$\boldsymbol{H}_t = \boldsymbol{C}\boldsymbol{C}' + \sum_{i=1}^{q} \boldsymbol{A}_i' \boldsymbol{V}_{t-i} \boldsymbol{A}_i + \sum_{j=1}^{p} \boldsymbol{B}_j' \boldsymbol{H}_{t-j} \boldsymbol{B}_j. \tag{70}$$

To reduce the number of parameters estimated all of the methods we cover use covariance targeting, with $\overline{\boldsymbol{H}}$ set to an estimate of the unconditional covariance.

$$\boldsymbol{H}_t = \overline{\boldsymbol{H}} + \sum_{i=1}^{q} \boldsymbol{A}_i' \boldsymbol{V}_{t-i} \boldsymbol{A}_i + \sum_{j=1}^{p} \boldsymbol{B}_j' \boldsymbol{H}_{t-j} \boldsymbol{B}_j. \tag{71}$$

The realized multivariate BEKK model is far from parsimonious so the approaches below further restrict $\boldsymbol{A}$ and $\boldsymbol{B}$ leading to realized multivariate scalar-BEKK models. To solve the problem of producing positive definite covariance matrices all of the dynamic models below use the Wishart distribution. The Wishart distribution is the distribution of the sample variance of independent zero-mean multivariate normal vectors (Wishart, 1928).

Restricting (70) yields the exponentially weighted moving average (EWMA) model. Fleming et al. (2001), De Pooter et al. (2008), and Bannouh et al. (2009) specify $\boldsymbol{H}_t$ using its lagged value and the realized variability of the previous day:

$$\boldsymbol{H}_t = \boldsymbol{\alpha} \exp(-\boldsymbol{\alpha}) \boldsymbol{V}_{t-1} + \exp(-\boldsymbol{\alpha}) \boldsymbol{H}_{t-1}, \tag{72}$$

where α is the decay parameter.

7.2 HEAVY

Barndorff-Nielsen et al. (2011) and Fleming et al. (2003) use the following realized scalar-BEKK(1,1,1) model with covariance targeting:

$$\boldsymbol{H}_t = (1 - \alpha_H - \beta_H)\overline{\boldsymbol{H}} + \alpha_H \boldsymbol{V}_{t-1} + \beta_H \boldsymbol{H}_{t-1}, \tag{73}$$

where $\alpha_H, \beta_H \geq 0$ and $\alpha_H + \beta_H \leq 1$. Noureldin et al. (2012) extend this approach by considering the joint prediction of the covariance matrix and the realized measure using a system of two equations. Eq. (73) referred to as the HEAVY-P equation is coupled with the HEAVY-V equation for predicting the realized measure. Denote the latter by $\boldsymbol{M}_t = E_{t-1}(\boldsymbol{V}_t)$. Then the HEAVY-V equation is

$$\boldsymbol{M}_t = (1 - \alpha_M - \beta_M)\overline{\boldsymbol{V}} + \alpha_M \boldsymbol{V}_{t-1} + \beta_M \boldsymbol{M}_{t-1}, \tag{74}$$

where $\alpha_M, \beta_M \geq 0$ and $\alpha_M + \beta_M \leq 1$. Note that the joint approach allows for multistep covariance forecasts.

The multivariate HEAVY model is currently implemented in Keven Sheppard's MFE toolbox in MATLAB® (Sheppard, 2013).

7.3 Realized DCC

Bauwens et al. (2012) include the realized covariance into the conditional covariance matrix by extending the DCC model of Engle (2002), as corrected by Aielli (2013). The cRDCC model of Bauwens et al. (2012) takes the following form:

$$\begin{aligned} \mathbf{H}_t &= \mathbf{D}_t \mathbf{R}_t \mathbf{D}_t \\ \mathbf{R}_t &= \operatorname{diag}(\mathbf{Q}_t)^{-1/2} \mathbf{Q}_t \operatorname{diag}(\mathbf{Q}_t)^{-1/2} \\ \mathbf{Q}_t &= (1 - \alpha - \beta)\overline{\mathbf{Q}} + \alpha \mathbf{P}^*_{t-1} + \beta \mathbf{Q}_{t-1}, \end{aligned} \tag{75}$$

where $\mathbf{P}^*_t = \operatorname{diag}(\mathbf{Q}_t)^{1/2} \mathbf{D}_t^{-1} \mathbf{V}_t \mathbf{D}_t^{-1} \operatorname{diag}(\mathbf{Q}_t)^{1/2}$ and $\mathbf{V}_t$ is the realized covariance measure. To reduce the number of parameters to be estimated, Bauwens et al. (2012) recommend to use correlation targeting, by replacing $\overline{\mathbf{Q}}$ by the mean of $\mathbf{P}^*_t$. Like the traditional (c)DCC model, the realized version can be estimated in two steps, where first univariate models are fitted to estimate the volatilities $\mathbf{D}_t$, which are then used to estimate the conditional correlation $\mathbf{R}_t$. Bollerslev et al. (2018) extend this model by accounting for leverage effects using realized semicorrelations.

7.4 Other approaches

There are other approaches for using realized covariance to forecast conditional variance that are not covered above. Callot et al. (2017) propose a LASSO approach that is promising for large-scale matrices and also is available in the `lassovar` R-package. The other approaches below are not covered in any R-Packages and mostly suffer from the curse of dimensionality. Bauer and Vorkink (2011) proposed a multivariate heterogenous autoregressive, or HAR (Corsi, 2009), model based on the log-matrix covariance specification of Chiu et al. (1996). Golosnoy et al. (2012) extend the work of Gourieroux et al. (2009) with a Conditional Autoregressive Wishart (CAW) analysis of realized covariance matrices. Chiriac and Voev (2011) use a multivariate ARFIMA model to forecast realized covariances.

8 Illustration

We conclude this overview with a simple illustration of using MGARCH models for portfolio analysis in **R**. For detailed horse races among MGARCH models, we refer to Laurent et al. (2012), Santos et al. (2013), and Laurent et al. (2013), among others.

The illustration involves estimating MGARCH models and using those estimates to characterize the conditional distribution of the return of a portfolio invested in 10 Vanguard mutual funds, which were chosen as a representative sample of a diversified portfolio with historical data going back at least as far as 2000. Besides four equity funds (Vanguard 500 (VFINX), Vanguard European (VEURX), Vanguard Pacific (VPACX), and Vanguard Emerging Mkts (VEIEX) Stock Index Investor) and two bonds funds (Vanguard Long-Term Bond Index Investor (VBLTX) and Vanguard Long-Term Investment-Grade Investor (VWESX)), we have funds active in real estate (VGSIX), energy (VGEN), precious metals, and mining (VGPM) and alternatives (Vanguard Market Neutral Investor (VMNFX)).

Daily January 2000–July 2018 return data were downloaded from Yahoo Finance using the **quantmod** package of Peterson and Carl (2018). We then use **PerformanceAnalytics** of Ryan and Ulrich (2018) to obtain a first visualization of the data. Fig. 3 shows on the diagonal the univariate distribution of the daily ETF returns. We clearly see cross-sectional heterogeneity in the scale and shape of the distribution, as well as large differences in the correlation between the ETFs. Additional rolling window plots indicate time-varying volatility and correlations.

Henceforth, we split our sample into an estimation sample and evaluation sample. The estimation period ranges from 2000-01-04 to 2006-12-31, while the out-of-sample forecast period ranges from 2007-01-03 to 2018-07-30, and thus includes the global financial crisis of 2007/2008. In practice we recommend to reestimate frequently the model, as compared to a simple split-sample evaluation. Reestimating the model allows to account for changes in relationships. However, for this simple exposition we fix the estimated parameters during the prediction period.

We estimate the following set of models:

- Normal GARCH-DCC model of Engle (2002) using the **rmgarch** package of Ghalanos (2015a).
- skewed t GAS-DCC model using the **GAS** package of Ardia et al. (2018b) for the estimation of the conditional variance, and the **ccgarch** package of Nakatani (2014) for the estimation of the DCC correlations.
- The CHICAGO model, consisting of a GO-GARCH model with multivariate affine NIG distributions (as in Broda and Paolella (2009)) and the component GARCH(1,1) model of Engle and Lee (1999), using the **rmgarch** package of Ghalanos (2015a). The component GARCH(1,1) models allows to capture permanent and transitory components of the underlying volatility dynamics.

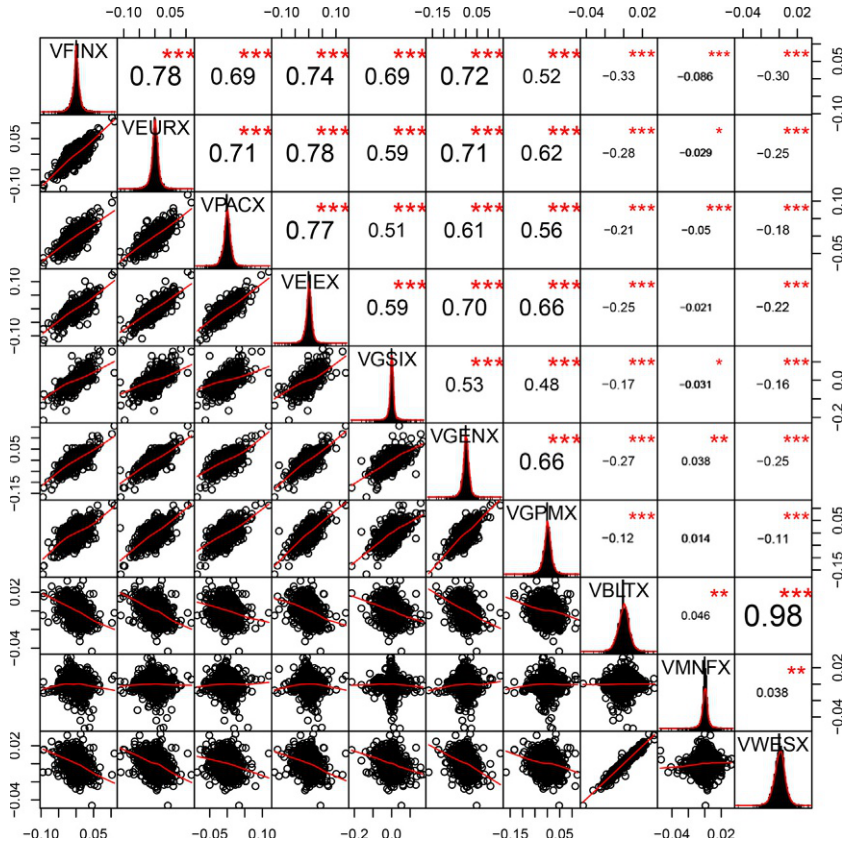

FIG. 3 Univariate and bivariate distribution plots of the daily 2000–2012 returns of the ETFs with tickers VFINX (US stocks), VEURX (European stocks), VPACX (Pacific stocks), VEIEX (Emerging Mkts stocks), VGSIX (Real Estate), VGENX (Energy), VGPMX (Precious Metals and Mining), VBLTX (Long-Term Bond), VMNFX (Market Neutral), VWESX (Long-Term Investment-Grade).

For each of the models, we only include a constant in the conditional mean equation. It is unlikely that any of these three methods correspond to the true data generating process. Instead, as noted, e.g., in Caporin and McAleer (2013), we should consider the obtained correlations as filters of the true conditional correlations.

Once the DCC parameters have been estimated, it is straightforward to compute the corresponding correlations over the test period. As an example, we plot in Fig. 4 the obtained conditional correlation between US equity returns (VFINX), on the one hand, and bonds, real estate, and equity neutral funds on the other hand. Note the negative correlation between bond and equity returns over the evaluation period, while equity and real estate have a strong positive correlation over the period. The equity fund returns and the equity market neutral have a correlation that fluctuates around zero.

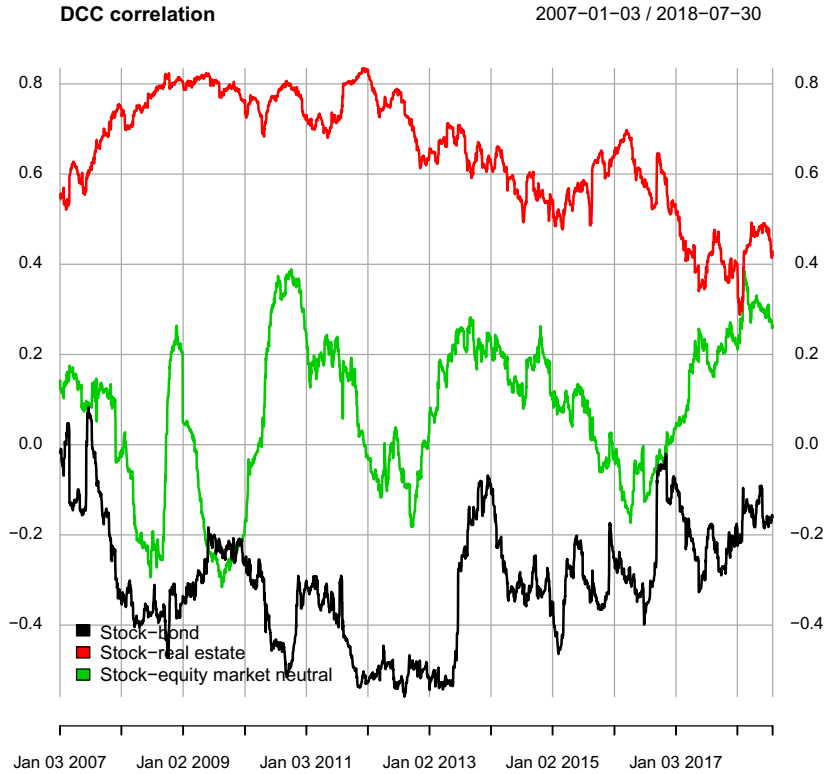

FIG. 4 Forecasted stock–bond, stock–real estate, and stock–equity market neutral correlations.

The dynamics in the conditional correlations are driven by the shocks in the underlying asset returns. This propagation can be visualized using a news impact curve, as depicted in Fig. 5 for the case of the bond–equity correlation. Correlations rise (resp. fall) in case of large returns of the same (resp. opposite) sign.

The use of the normal GARCH-DCC model and the CHICAGO model in **R** has the advantage of directly yielding a complete characterization of the conditional portfolio return distribution. The joyplots in Fig. 6 show the time-varying conditional distribution of the daily returns for the portfolio that is equally weighted in the 10 ETFs.[x]

We clearly see a strong time variation in the fitted distribution function, with a time variation that is aligned with normal and market turbulent regimes. Note in particular the large expansion in the portfolio variance in 2008, and the presence of volatility clusters.

[x]The term "joyplot" was coined on twitter in April 2017 by Jenny Bryan as a series of statistical data graphed in such a way that they resemble the cover artwork of the Joy Division's Unknown Pleasures album.

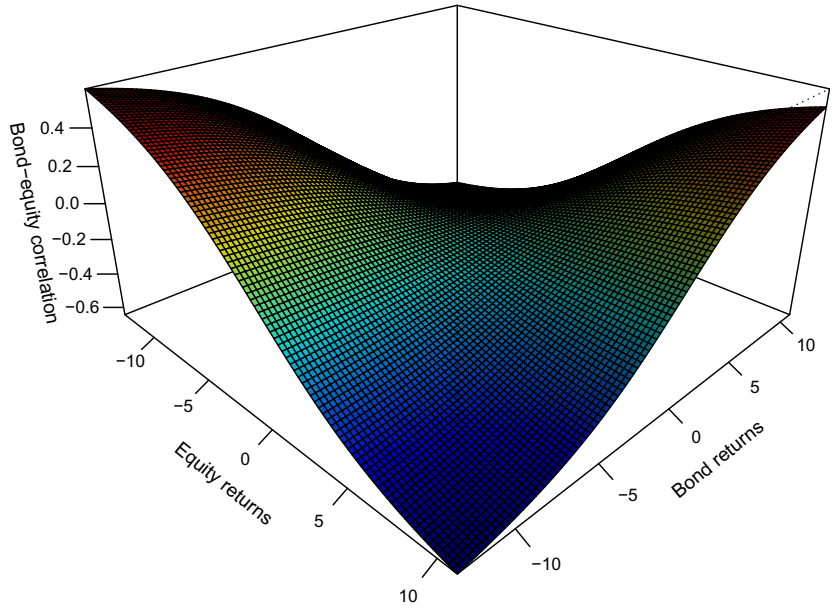

FIG. 5 News impact curve for bond–equity correlations obtained using the DCC model.

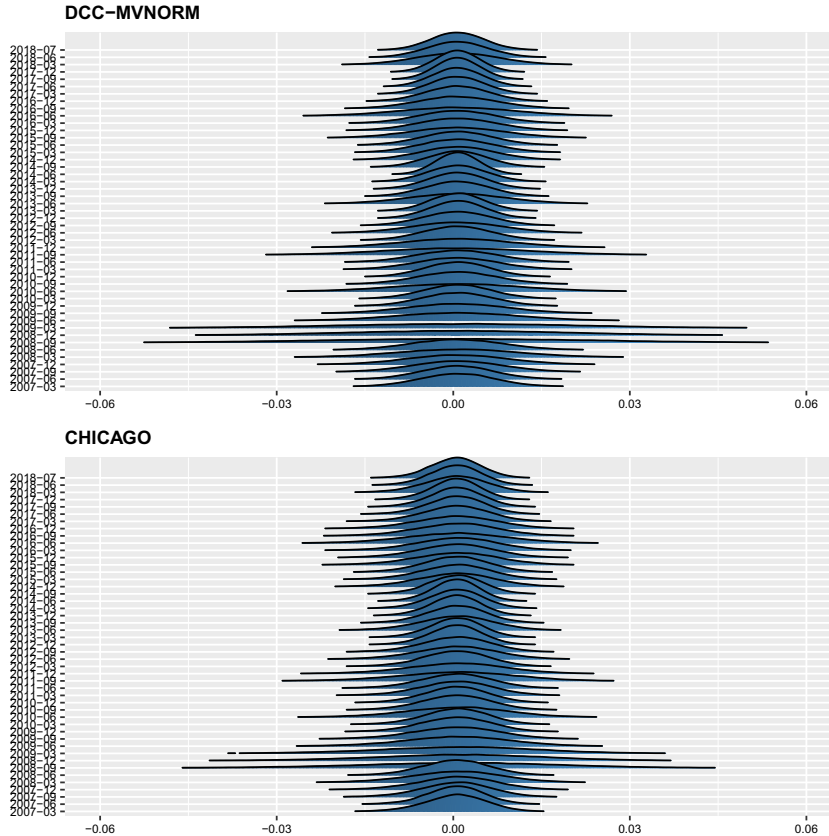

FIG. 6 Forecast of conditional portfolio densities at quarterly intervals.

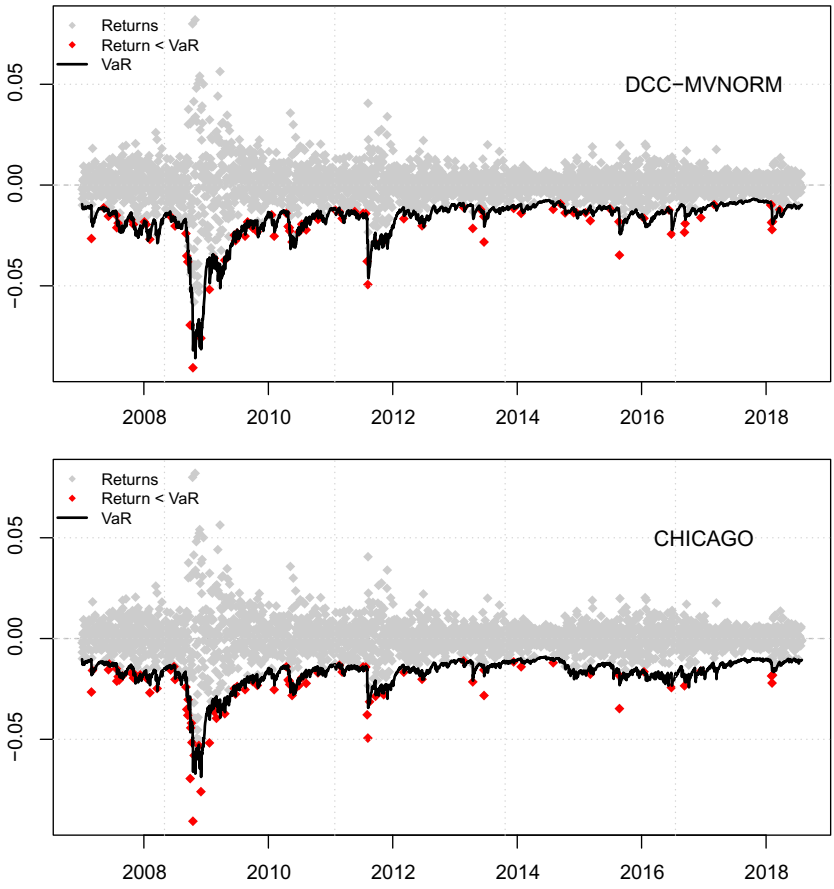

FIG. 7 Daily 1% Value-at-Risk predictions over the period January 2007–July 2018.

The MGARCH specification matters as can be seen in the disagreement between the Normal GARCH-DCC and the CHICAGO distributions in Fig. 6. This also leads to substantial differences in downside risk assessments, for which the 1% and 5% Value-at-Risk risk measure is a standard (in addition to expected shortfall; Jorion (1997)). They correspond to the 1% and 5% quantile of the return distribution.

Fig. 7 shows the daily 1% Value-at-Risk predictions for the equally weighted portfolio obtained using the Normal GARCH-DCC and the CHICAGO MGARCH models. We clearly see that the time-varying covariance matrix is an important driver of the time variation in downside risk as measured by the 1% Value-at-Risk. The red diamonds in the graph indicate the so-called exceedances where the observed return is more negative than the predicted Value-at-Risk. Further statistical tests, such as the test for

correct unconditional coverage (UC) proposed by Kupiec (1995) and test for correct conditional coverage (CC) proposed by Christoffersen (1998), can be used to evaluate whether the risk model is reliable. As already mentioned, a detailed evaluation of MGARCH models is beyond the scope of this illustration, and would require a rolling estimation approach, with period reestimation of the model, instead of the simple split-sample setup used in this illustration. In the Supplementary Material in the online version at https://doi.org/10.1016/bs.host.2019.01.001, we provide example code to do so for the DCC and CHICAGO model.

9 Conclusion

The joint analysis of more than 10 financial return series is a challenging empirical task. It needs to take into account the nonnormality of the return series, their time-varying volatility and correlation, and avoid the curse of dimensionality while preserving sufficient flexibility in order to avoid severe model misspecification. This chapter has provided an overview of recent advances in the field, and, when available, their implementation in **R** packages.

We end the overview chapter on feasible MGARCH models with a call for more research on feasible MGARCH models. Our field experience is that most of the large financial institutions are not willing to put a flexible MGARCH model into production. They still base their risk models on the EWMA model or some modification of it, since the simple EWMA model tends to be more reliable than the flexible ones in practice. An important research direction is thus to put reliability as a primary objective when developing the next generation of flexible MGARCH models. We believe that publishing the code using open source software like **R** will help in achieving this objective of reliability, since open sourcing allows code to be more actively vetted by a large community whose feedback helps in developing better models.

References

Aielli, G.P., 2013. Dynamic conditional correlation: on properties and estimation. J. Bus. Econ. Stat. 31 (3), 282–299.

Aït-Sahalia, Y., Mykland, P.A., Zhang, L., 2005. How often to sample a continuous-time process in the presence of market microstructure noise. Rev. Financ. Stud. 18 (2), 351–416.

Alexander, C., 2001. Orthogonal garch. In: Mastering Risk, vol. 2. Financial Times–Prentice Hall, pp. 21–38.

Andersen, T., Bollerslev, T., Diebold, F., Labys, P., 2003. Modeling and forecasting realized volatility. Econometrica 71, 579–625.

Anderson, D., 1992. A multivariate linnik distribution. Statist. Probab. Lett. 14 (4), 333–336.

Ardia, D., Boudt, K., Catania, L., 2018a. Downside risk evaluation with the R package GAS. R J. https://journal.r-project.org/archive/2018/RJ-2018-064/index.html. forthcoming.

Ardia, D., Boudt, K., Catania, L., 2018b. Generalized autoregressive score models in R: the GAS package. J. Stat. Softw. 88 (6), 1–28.

Arslan, O., 2010. An alternative multivariate skew Laplace distribution: properties and estimation. Stat. Pap. 51 (4), 865–887.
Attanasio, O., 1991. Risk, time-varying second moments and market efficiency. Rev. Econ. Stud. 58 (3), 479–494.
Ausin, M., Lopes, H., 2010. Time-varying joint distribution through copulas. Comput. Stat. Data Anal. 54 (11), 2383–2399.
Azzalini, A., 2013. The Skew-Normal and Related Families. vol. 3. Cambridge University Press.
Bannouh, K., van Dijk, D., Martens, M., 2009. Range-based covariance estimation using high-frequency data: the realized co-range. J. Financ. Economet. 7, 341–372.
Barndorff-Nielsen, O., 1977. Exponentially decreasing distributions for the logarithm of particle size. Proc. R. Soc. London A 353 (1674), 401–419.
Barndorff-Nielsen, O., Bläsild, P., Taillie, C., Patil, G., Baldessari, B., 1981. Hyperbolic distributions and ramifications: contributions to theory and application. In: Statistical Distributions in Scientific Work. vol. 4. Reidel, pp. 19–44.
Barndorff-Nielsen, O.E., Shephard, N., 2004. Power and bipower variation with stochastic volatility and jumps. J. Financ. Economet. 2, 1–37.
Barndorff-Nielsen, O.E., Hansen, P.R., Lunde, A., Shephard, N., 2009. Realized kernels in practice: trades and quotes. Econ. J. 12, C1–C32.
Barndorff-Nielsen, O.E., Hansen, P.R., Lunde, A., Shephard, N., 2011. Multivariate realised kernels: consistent positive semi-definite estimators of the covariation of equity prices with noise and non-synchronous trading. J. Econ. 162, 149–169.
Bauer, G.H., Vorkink, K., 2011. Forecasting multivariate realized stock market volatility. J. Econ. 160 (1), 93–101.
Bauwens, L., Laurent, S., 2005. A new class of multivariate skew densities, with application to generalized autoregressive conditional heteroscedasticity models. J. Bus. Econ. Stat. 23 (3), 346–354.
Bauwens, L., Laurent, S., Rombouts, J.V., 2006. Multivariate garch models: a survey. J. Appl. Econ. 21 (1), 79–109.
Bauwens, L., Storti, G., Violante, F., 2012. Dynamic conditional correlation models for realized covariance matrices. Working Paper.
Billio, M., Caporin, M., Gobbo, M., 2006. Flexible dynamic conditional correlation multivariate garch models for asset allocation. Appl. Financ. Econ. Lett. 2 (2), 123–130.
Bollen, B., Inder, B., 2002. Estimating daily volatility in financial markets utilizing intraday data. J. Empir. Financ. 9 (5), 551–562.
Bollerslev, T., 1990. Modelling the coherence in short-run nominal exchange rates: a multivariate generalized arch model. Rev. Econ. Stat. 72 (3), 498–505.
Bollerslev, T., Engle, R., Wooldridge, J., 1988. A capital asset pricing model with time-varying covariances. J. Polit. Econ. 96 (1), 116.
Bollerslev, T., Patton, A.J., Quaedvlieg, R., 2018. Multivariate leverage effects and realized semi-covariance garch models. J. Econ. forthcoming.
Boswijk, P., van der Weide, R., 2011. Method of moments estimation of go-garch models. J. Econ. 163 (1), 118–126.
Boudt, K., Croux, C., 2010. Robust M-estimation of multivariate garch models. Comput. Stat. Data Anal. 54 (11), 2459–2469.
Boudt, K., Zhang, J., 2015. Jump robust two time scale covariance estimation and realized volatility budgets. Quant. Finan. 15 (6), 1041–1054.
Boudt, K., Croux, C., Laurent, S., 2011. Robust estimation of intraweek periodicity in volatility and jump detection. J. Empir. Financ. 18, 353–367.

Boudt, K., Danielsson, J., Koopman, S.J., Lucas, A., 2012. Regime switches in volatility and correlation of financial institutions. National Bank of Belgium Working Paper. No. 227.

Boudt, K., Danielsson, J., Laurent, S., 2013. Robust forecasting of dynamic conditional correlation garch models. Int. J. Forecast. 29 (2), 244–257.

Boudt, K., Laurent, S., Lunde, A., Quaedvlieg, R., Sauri, O., 2017. Positive semidefinite integrated covariance estimation, factorizations and asynchronicity. J. Econ. 196 (2), 347–367.

Boudt, K., Cornelissen, J., Payseur, S., 2018. Highfrequency: tools for high-frequency data analysis. R package version 0.5.3. https://CRAN.R-project.org/package=highfrequency

Broda, S., Paolella, M., 2009. Chicago: a fast and accurate method for portfolio risk calculation. J. Financ. Economet. 7 (4), 412.

Callot, L., Kock, A., Medeiros, M., 2017. Modeling and forecasting large realized covariance matrices and portfolio choice. J. Appl. Econ. 32 (1), 140–158.

Caporin, M., McAleer, M., 2012. Do we really need both BEKK and DCC? A tale of two multivariate GARCH models. J. Econ. Surv. 26 (4), 736–751.

Caporin, M., McAleer, M., 2013. Ten things you should know about the dynamic conditional correlation representation. Econometrics 1 (1), 115–126.

Cappiello, L., Engle, R., Sheppard, K., 2006. Asymmetric correlations in the dynamics of global equity and bond returns. J. Financ. Economet. 4 (4), 537–572.

Cardoso, J., 2000. Entropic contrasts for source separation: geometry and stability. In: Haykin, S. (Ed.), Unsupervised Adaptive Filters. John Wiley & Sons, pp. 139–190.

Catania, L., 2016. Dynamic adaptive mixture models. arXiv. preprint arXiv:1603.01308.

Catania, L., 2018. Switching-gas copula models for systemic risk assessment. J. Appl. Econ. forthcoming.

Catania, L., Boudt, K., Ardia, D., 2017. GAS: Generalised Autoregressive Score Models. R package version 0.2.5. https://cran.r-project.org/package=GAS.

Chan, Y., Li, H., 2007. Tail Dependence for Multivariate t-Distributions and Its Monotonicity. Technical Report 2007-6. http://www.math.wsu.edu/TRS/.

Chen, Y., Härdle, W., Spokoiny, V., 2007. Portfolio value at risk based on independent component analysis. J. Comput. Appl. Math. 205 (1), 594–607.

Chiriac, R., Voev, V., 2011. Modelling and forecasting multivariate realized volatility. J. Appl. Econ. 26 (6), 922–947.

Chiu, T.Y.M., Leonard, T., Tsui, K.-W., 1996. The matrix-logarithmic covariance model. J. Am. Stat. Assoc. 91 (433), 198–210.

Christoffersen, P.F., 1998. Evaluating interval forecasts. Int. Econ. Rev. 39 (4), 841–862.

Comon, P., 1994. Independent component analysis, a new concept? Signal Process. 36 (3), 287–314.

Comte, F., Lieberman, O., 2003. Asymptotic theory for multivariate garch processes. J. Multivar. Anal. 84 (1), 61–84.

Corsi, F., 2009. A simple approximate long-memory model of realized volatility. J. Financ. Economet. 7 (2), 174–196. https://doi.org/10.1093/jjfinec/nbp001.

Creal, D., Koopman, S.J., Lucas, A., 2011. A dynamic multivariate heavy-tailed model for time-varying volatilities and correlations. J. Bus. Econ. Stat. 29 (4), 552–563.

Creal, D., Koopman, S.J., Lucas, A., 2013. Generalized autoregressive score models with applications. J. Appl. Econ. 28 (5), 777–795.

Creal, D., Schwaab, B., Koopman, S.J., Lucas, A., 2014. Observation-driven mixed-measurement dynamic factor models with an application to credit risk. Rev. Econ. Stat. 96 (5), 898–915.

Davison, A., Smith, R., 1990. Models for exceedances over high thresholds. J. R. Stat. Soc. B. Methodol. 52 (3), 393–442.

Demarta, S., McNeil, A., 2005. The t copula and related copulas. Int. Stat. Rev. 73 (1), 111–129.
De Pooter, M., Martens, M.P., Van Dijk, D.J., 2008. Predicting the daily covariance matrix for S&P 100 stocks using intraday data—but which frequency to use? Econ. Rev. 27, 199–229.
Ding, Z., 1994. Time Series Analysis of Speculative Returns (Ph.D. Dissertation). University of California, San Diego, CA.
Engle, R., 2002. Dynamic conditional correlation. J. Bus. Econ. Stat. 20 (3), 339–350.
Engle, R., 2009. Anticipating Correlations: A New Paradigm for Risk Management. Princeton University Press.
Engle, R.F., Kroner, K.F., 1995. Multivariate simultaneous generalized arch. Economet. Theor. 11 (1), 122–150.
Engle, R.F., Lee, G., 1999. A long-run and short-run component model of stock return volatility. In: Engle, R.F., White, H. (Eds.), Cointegration, Causality, and Forecasting: A Festschrift in Honour of Clive WJ Granger. Oxford University Press, Oxford, pp. 475–497.
Engle, R., Ng, V., 1993. Measuring and testing the impact of news on volatility. J. Financ. 48 (5), 1749–1778.
Engle, R.F., Ng, V.K., Rothschild, M., 1990. Asset pricing with a factor arch covariance structure: empirical estimates for treasury bills. J. Econ. 45 (1–2), 213–237.
Epps, T.W., 1979. Comovements in stock prices in the very short run. J. Am. Stat. Assoc. 74, 291–298.
Fernández, C., Steel, M.F., 1998. On Bayesian modeling of fat tails and skewness. J. Am. Stat. Assoc. 93 (441), 359–371.
Fernandez, C., Osiewalski, J., Steel, M., 1995. Modeling and inference with v-spherical distributions. J. Am. Stat. Assoc. 90 (432), 1331–1340.
Ferreira, J., Steel, M., 2006. A constructive representation of univariate skewed distributions. J. Am. Stat. Assoc. 101 (474), 823–829.
Fiorucci, J.A., Ehlers, R.S., Louzada, F., Fiorucci, M.J.A., 2016. Package 'bayesdccgarch'. Technical Report. Available at http://cran.r-project.org/web/packages/bayesDccGarch.
Fioruci, J.A., Ehlers, R.S., Andrade Filho, M.G., 2014. Bayesian multivariate garch models with dynamic correlations and asymmetric error distributions. J. Appl. Stat. 41 (2), 320–331.
Fleming, J., Kirby, C., Ostdiek, B., 2001. The economic value of volatility timing. J. Financ. 56, 329–352.
Fleming, J., Kirby, C., Ostdiek, B., 2003. The economic value of volatility timing using realized volatility. J. Financ. Econ. 67, 473–509.
Francq, C., Zakoian, J.-M., 2011. GARCH Models: Structure, Statistical Inference and Financial Applications. John Wiley & Sons.
Frey, R., McNeil, A.J., 2003. Dependent defaults in models of portfolio credit risk. J. Risk 6, 59–92.
Genest, C., Ghoudi, K., Rivest, L., 1995. A semiparametric estimation procedure of dependence parameters in multivariate families of distributions. Biometrika 82 (3), 543–552.
Ghalanos, A., 2012. spd: The Semi Parametric Distribution. Technical Report. Available at http://cran.r- project.org/web/packages/spd.
Ghalanos, A., 2015a. rmgarch: Multivariate GARCH Models. R package version 1.3-0.https://cran.r-project.org/package=rmgarch.
Ghalanos, A., 2015b. The rmgarch Models: Background and Properties (version 1.3-0). Technical Report. Available at http://cran.r-project.org/web/packages/rmgarch.
Ghalanos, A., 2018. Introduction to the rugarch Package (version 1.3-1). Technical Report. Available at http://cran.r-project.org/web/packages/rugarch.
Ghalanos, A., Rossi, E., Urga, G., 2015. Independent factor autoregressive conditional density model. Econ. Rev. 34 (5), 594–616.

Glosten, L., Jagannathan, R., Runkle, D., 1993. On the relation between the expected value and the volatility of the nominal excess return on stocks. J. Financ. 48 (5), 1779–1801.
Golosnoy, V., Gribisch, B., Liesenfeld, R., 2012. The conditional autoregressive Wishart model for multivariate stock market volatility. J. Econ. 167 (1), 211–223.
Gorgi, P., Hansen, P., Janus, P., Koopman, S., 2019. Realized wishart-garch: a score-driven multi-asset volatility model. J. Financ. Economet. 17 (1), 1–32.
Gourieroux, C., 1997. ARCH Models and Financial Applications. Springer.
Gourieroux, C., Jasiak, J., Sufana, R., 2009. The Wishart autoregressive process of multivariate stochastic volatility. J. Econ. 150 (2), 167–181.
Hafner, C., Preminger, A., 2009. Asymptotic theory for a factor garch model. Economet. Theor. 25 (2), 336–363.
Hansen, B., 1994. Autoregressive conditional density estimation. Int. Econ. Rev. 35 (3), 705–730.
Harvey, A.C., 2013. Dynamic Models for Volatility and Heavy Tails: With Applications to Financial and Economic Time Series. Cambridge University Press.
Harvey, C.R., Siddique, A., 2000. Conditional skewness in asset pricing tests. J. Financ. 55 (3), 1263–1295.
Harvey, A.C., Sucarrat, G., 2014. Egarch models with fat tails, skewness and leverage. Comput. Stat. Data Anal. 76, 320–338.
Harvey, A., Chakravarty, T., et al., 2008. Beta-t-(e) garch. Technical Report. Faculty of Economics, University of Cambridge.
Hautsch, N., Kyj, L.M., Oomen, R.C.A., 2010. A blocking and regularization approach to high-dimensional realized covariance estimation. J. Appl. Econ. 27 (4), 625–645.
Hentschel, L., 1995. All in the family nesting symmetric and asymmetric garch models. J. Financ. Econ. 39 (1), 71–104.
Hofert, M., Kojadinovic, I., Maechler, M., Yan, J., 2018. Copula: Multivariate Dependence With Copulas. Technical Report.
Hosking, J.R., Wallis, J.R., Wood, E.F., 1985. Estimation of the generalized extreme-value distribution by the method of probability-weighted moments. Technometrics 27 (3), 251–261.
Hyvarinen, A., Oja, E., 2000. Independent component analysis: algorithms and applications. Neural Netw. 13 (4–5), 411–430.
Jeantheau, T., 1998. Strong consistency of estimators for multivariate arch models. Economet. Theor. 14 (1), 70–86.
Joe, H., 1997. Multivariate Models and Dependence Concepts. vol. 73. Chapman & Hall/CRC.
Jondeau, E., Poon, S.-H., Rockinger, M., 2007. Financial Modeling Under Non-Gaussian Distributions. Springer Science & Business Media.
Jones, M., Sibson, R., 1987. What is projection pursuit? J. R. Stat. Soc. Ser. A 150 (1), 1–37.
Jorion, P., 1997. Value at Risk. McGraw-Hill, New York. https://doi.org/10.1036/0071464956.
Kalashnikov, V., 1997. Geometric Sums: Bounds for Rare Events With Applications: Risk Analysis, Reliability, Queueing. Springer.
Kelker, D., 1970. Distribution theory of spherical distributions and a location-scale parameter generalization. Sankhya Indian J. Stat. 32 (4), 419–430.
Kotz, S., Kozubowski, T., Podgórski, K., 2002. Maximum likelihood estimation of asymmetric Laplace parameters. Ann. Inst. Stat. Math. 54 (4), 816–826.
Kozubowski, T., Podgórski, K., 2001. Asymmetric Laplace laws and modeling financial data. Math. Comput. Model. 34 (9), 1003–1021.
Kozubowski, T.J., Podgórski, K., Rychlik, I., 2013. Multivariate generalized Laplace distribution and related random fields. J. Multivar. Anal. 113, 59–72.

Kraus, A., Litzenberger, R.H., 1976. Skewness preference and the valuation of risk assets. J. Financ. 31 (4), 1085–1100.
Kroner, K., Ng, V., 1998. Modeling asymmetric comovements of asset returns. Rev. Financ. Stud. 11 (4), 817.
Kruskal, W., 1958. Ordinal measures of association. J. Am. Stat. Assoc. 53 (284), 814–861.
Kupiec, P.H., 1995. Techniques for verifying the accuracy of risk measurement models. J. Deriv. 3 (2), 73–84.
Laurent, S., 2013. G@RCH 7.0: Estimating and Forecasting Arch Models. Timberlake Consultants Ltd., London.
Laurent, S., Rombouts, J.V., Violante, F., 2012. On the forecasting accuracy of multivariate garch models. J. Appl. Econ. 27 (6), 934–955.
Laurent, S., Rombouts, J.V., Violante, F., 2013. On loss functions and ranking forecasting performances of multivariate volatility models. J. Econ. 173 (1), 1–10.
Li, D., 2000. On default correlation: a copula function approach. J. Fixed Income 9 (4), 43–54.
Lindskog, F., Mcneil, A., Schmock, U., 2003. Kendall's tau for elliptical distributions. In: Credit Risk: Measurement, Evaluation and Management. Physica-Verlag, pp. 149–156.
Lucas, A., Schwaab, B., Zhang, X., 2014. Conditional euro area sovereign default risk. J. Bus. Econ. Stat. 32 (2), 271–284.
Lucas, A., Schwaab, B., Zhang, X., 2017. Modeling financial sector joint tail risk in the euro area. J. Appl. Econ. 32 (1), 171–191.
Lunde, A., Shephard, N., Sheppard, K., 2016. Econometric analysis of vast covariance matrices using composite realized kernels and their application to portfolio choice. J. Bus. Econ. Stat. 34 (4), 504–518.
Mancini, C., Gobbi, F., 2012. Identifying the covariation between the diffusion parts and the co-jumps given discrete observations. Economet. Theor. 28 (2), 249–273.
Marchenko, V.A., Pastur, L.A., 1967. Distribution of eigenvalues for some sets of random matrices. Matematicheskii Sbornik 114 (4), 507–536.
Markowitz, H., 1952. Portfolio selection. J. Financ. 7 (1), 77–91.
Marshall, A., Olkin, I., 1993. Maximum likelihood characterizations of distributions. Stat. Sin. 3, 157–171.
McNeil, A.J., Frey, R., Embrechts, P., 2015. Quantitative Risk Management: Concepts, Techniques and Tools. Princeton University Press.
Mina, J., Xiao, J.Y., et al., 2001. Return to Riskmetrics: The Evolution of a Standard. 1, RiskMetrics Group, pp. 1–11.
Nakatani, T., 2008. Package ccgarch. Technical Report. Available at http://cran.r-project.org/web/packages/ccgarch.
Nakatani, T., 2014. ccgarch: An R Package for Modelling Multivariate GARCH Models With Conditional Correlations. R package version 0.2.3. http://CRAN.r-project/package=ccgarch.
Nakatani, T., Teräsvirta, T., 2009. Testing for volatility interactions in the constant conditional correlation garch model. Econ. J. 12 (1), 147–163.
Noureldin, D., Shephard, N., Sheppard, K., 2012. Multivariate high-frequency-based volatility (HEAVY) models. J. Appl. Econ. 27 (6), 907–933.
Patton, A., 2006. Modelling asymmetric exchange rate dependence. Int. Econ. Rev. 47 (2), 527–556.
Patton, A.J., Sheppard, K., 2009. Evaluating volatility and correlation forecasts. In: Handbook of Financial Time Series. Springer, In, pp. 801–838.
Payseur, S., 2008. Essays in Realized Covariance Matrix Estimation (Ph.D. Dissertation). University of Washington.

Pearlmutter, B.A., Parra, L.C., 1997. Maximum likelihood blind source separation: a context-sensitive generalization of ICA. Adv. Neural Inf. Process. Syst. 9, 613–619.

Pelletier, D., 2006. Regime switching for dynamic correlations. J. Econ. 131 (1–2), 445–473.

Peterson, B.G., Carl, P., 2018. PerformanceAnalytics: Econometric Tools for Performance and Risk Analysis. R package version 1.5.2. https://CRAN.R-project.org/package=PerformanceAnalytics.

Pfaff, B., 2009. Package gogarch. Technical Report. Available at http://cran.r-project.org/web/packages/gogarch.

Prause, K., 1999. The Generalized Hyperbolic Model: Estimation, Financial Derivatives, and Risk Measures. PhD Thesis, University of Freiburg.

Rossi, E., Spazzini, F., 2010. Model and distribution uncertainty in multivariate garch estimation: a Monte Carlo analysis. Comput. Stat. Data Anal. 54 (11), 2786–2800.

Ryan, J.A., Ulrich, J.M., 2018. quantmod: Quantitative Financial Modelling Framework. R package version 0.4-13. https://CRAN.R-project.org/package=quantmod.

Santos, A.A., Nogales, F.J., Ruiz, E., 2013. Comparing univariate and multivariate models to forecast portfolio value-at-risk. J. Financ. Economet. 11 (2), 400–441.

Schmidt, R., Hrycej, T., Stützle, E., 2006. Multivariate distribution models with generalized hyperbolic margins. Comput. Stat. Data Anal. 50 (8), 2065–2096.

Serban, M., Brockwell, A., Lehoczky, J.P., Srivastava, S., 2007. Modelling the dynamic dependence structure in multivariate financial time series. J. Time Ser. Anal. 28 (5), 763–782.

Sheppard, K., 2009. MFE MATLAB Function Reference Financial Econometrics. University of Oxford.

Sheppard, K., 2013. MFE Toolbox. University of Oxford. https://www.kevinsheppard.com/MFE_Toolbox#Last_Updated.

Silvennoinen, A., Teräsvirta, T., 2009. Multivariate garch models. In: Mikosch, T., Kreiß, J.P., Davis, R., Andersen, T. (Eds.), Handbook of Financial Time Series. Springer, pp. 201–229.

Sklar, A., 1959. Fonctions de répartition à n dimensions et leurs marges. 8, Publ. Inst. Statist. Univ. Paris, p. 11 1.

Tse, Y., Tsui, A., 2002. A multivariate generalized autoregressive conditional heteroscedasticity model with time-varying correlations. J. Bus. Econ. Stat. 20 (3), 351–362.

Van der Weide, R., 2002. Go-garch: a multivariate generalized orthogonal garch model. J. Appl. Econ. 17 (5), 549–564.

van der Weide, R., 2004. Wake me up before you go-garch. In: Computing in Economics and Finance. Society for Computational Economics.

Wishart, J., 1928. The generalized product moment distribution in samples from multinomial population. Biometrika 20, 32–52.

Zakoian, J., 1994. Threshold heteroskedastic models. J. Econ. Dyn. Control. 18 (5), 931–955.

Zhang, L., 2011. Estimating covariation: Epps effect, microstructure noise. J. Econ. 160 (1), 33–47.

Zhang, K., Chan, L., 2009. Efficient factor garch models and factor-dcc models. Quant. Finan. 9 (1), 71–91.

Zivot, E., Wang, J., 2006. Modelling financial time series with S-PLUS. Springer.

Part III

Miscellaneous Topics

This page intentionally left blank

Chapter 8

Modeling fractional responses using R

Joaquim Jose Santos Ramalho*
Department of Economics and BRU-IUL, Instituto Universitário de Lisboa (ISCTE-IUL), Lisboa, Portugal
Corresponding author: e-mail: jjsro@iscte-iul.pt

Abstract
Often, the dependent variable in regression models has a fractional nature, being bounded in the unit interval. Several variants of cross-sectional and panel data fractional regression models have recently been proposed. This chapter shows how to estimate most of those models by using three R packages: `frm`, `frmhet`, and `frmpd`.

Keywords: Fractional responses, Heterogeneity, Endogeneity, Panel data, Exponential regression, Two-part models, Specification tests, Partial effects, Prediction, Smearing estimator, R

1 Introduction

Fractional responses, i.e., variables bounded by 0 and 1, are a very common type of dependent variable in econometric models. Examples of such variables include firm market shares, proportion of debt in the financing mix of firms, fraction of land allocated to agriculture, and proportion of exports in total sales and data envelopment analysis efficiency scores. The bounded nature of these variables and, in some cases, the possibility of nontrivial probability mass accumulating at one or both the boundaries imply that specific econometric methodology has to be applied in this context.

Formal models for fractional response variables were first suggested by Papke and Wooldridge (1996). They developed the now commonly called fractional regression models, which require only the specification of the conditional mean of the response variable and are estimated by quasi-maximum likelihood (QML). Recently, several extensions have been proposed in the literature. Ramalho et al. (2011) suggested alternative specifications for Papke and Wooldridge (1996) conditional mean models, the use of two-part models in cases where a significant proportion of observations at either 0 or 1 are present

Handbook of Statistics, Vol. 41. https://doi.org/10.1016/bs.host.2018.11.008

and various specification tests for both types of models. Ramalho and Ramalho (2017) developed a new class of exponential-fractional estimators that are robust to neglected heterogeneity and accommodate endogenous explanatory variables. These estimators were extended to a panel data framework by Ramalho et al. (2018), a setting which has also been considered by Papke and Wooldridge (2008).

Given the very recent developments in this research field, most econometric software still does not possess simple, canned commands for applying most of the proposed models and tests. For this reason, I have written three R packages (`frm`, `frmhet`, and `frmpd`), which are available on the Comprehensive R Archive Network (https://cran.r-project.org), that allow easy implementation of the most popular features of fractional regression models. The main aim of this chapter is to explain in detail how to use those packages in practice. To this end, I use the dataset *RRC2018iv.txt*, which, in addition to the financial information of 620 Portuguese firms for the 2007–2011 period considered by Ramalho et al. (2018), includes two generated variables that will be used as instruments in some of the examples provided in the chapter. The number of observations per firm ranges from one to five, yielding an unbalanced panel of 1843 observations. The aim is to model the proportion of debt in firms' capital structure. The dependent variable is *Leverage* and the explanatory variables are *Growth*, *Size*, *Profitability*, and *Tangibility*. The file also contains the variables *Ident*, which identifies the firms, *Year*, which indicates the year of the observation, and *ProfitIVa* and *ProfitIVb*, the two generated instrumental variables. The following R code fragment will load the dataset in R and allows to treat each column name as a vector representing the aforementioned variables:

```
data <- read.table("http://home.iscte-iul.pt/~jjsro/data_code
        /RRC2018iv.txt",header=TRUE)
attach(data)
```

We may check that *Leverage* is bounded by 0 and 1, with 16.2% of observations at the lower bound, but no observations at the upper bound:

```
> summary(Leverage)
     Min.   1st Qu.   Median     Mean   3rd Qu.     Max.
0.00000  0.02502  0.18368  0.23613  0.39118  0.98170
> mean(Leverage==0)
[1] 0.1622355
```

The outline of the remaining sections in this chapter is as follows. Section 2 reviews in a comprehensive way the base econometric models for fractional responses, discussing their estimation and evaluation and showing how to calculate partial effects and making prediction. Section 3 introduces the

exponential-fractional estimators, which are extended for a panel data setting in Section 4. In all sections, for each topic I first present, at a theoretical level, the relevant econometric models and tests and then immediately show how to implement them using R.

2 The base case: Cross-sectional data and no unobserved heterogeneity

This section deals with the most common econometric methods for fractional responses in a cross-sectional framework. Therefore, we only need to load the package `frm` and we will work only with 2007 data:

```
library(frm)
Y <- Leverage[Year==2007]
X <- cbind(Growth, Size, Profitability, Tangibility)
X <- X[Year==2007,]
```

2.1 Conditional mean models

Let y_i denote the fractional response variable, defined on the interval $[0,1]$, to be explained for individual i, $i = 1, \ldots, N$, and let x_i denote a k-vector of explanatory variables. The standard fractional regression model used in the cross-sectional context is defined by the following conditional expectation:

$$E(y_i|x_i) = G(x_i\theta), \tag{1}$$

where θ is the vector of parameters of interest and $G(\cdot)$ is a (nonlinear) function bounded on the unit interval. Possible choices for $G(x_i\theta)$ are the following functional forms: $\frac{e^{x_i\theta}}{1+e^{x_i\theta}}$ (logit), $\Phi(x_i\theta)$ (probit), $e^{-e^{-x_i\theta}}$ (loglog), $1-e^{-e^{x_i\theta}}$ (cloglog), and $\frac{1}{2}+\frac{1}{\pi}\arctan(x_i\theta)$ (cauchit).

Papke and Wooldridge (1996) proposed to estimate the model defined by Eq. (1) by QML based on the Bernoulli log-likelihood function. Under suitable regularity conditions and assuming that (1) holds, the resultant QML estimator $\hat{\theta}$ is consistent and asymptotically normal and is efficient in a class of estimators containing all linear exponential family-based QML and weighted nonlinear least squares estimators. The asymptotic distribution of the QML estimator is given by

$$\sqrt{N}\left(\hat{\theta}-\theta_0\right) \xrightarrow{d} N(0,V), \tag{2}$$

where $V = A^{-1}BA^{-1}$, with $A = E[-\nabla_{\theta\theta'}LL(\theta)]$ and $B = E[\nabla_\theta LL(\theta)\nabla_{\theta'}LL(\theta)]$. Consistent estimators for A and B are given by $\hat{A} = N^{-1}\sum_{i=1}^{N}\hat{g}_i^2 x_i' x_i \left[\hat{G}_i\left(1-\hat{G}_i\right)\right]^{-1}$ and $\hat{B} = N^{-1}\sum_{i=1}^{N}\hat{u}_i^2\hat{g}_i^2 x_i' x_i \left[\hat{G}_i\left(1-\hat{G}_i\right)\right]^{-2}$, respectively, where $\hat{G}_i = G\left(x_i\hat{\theta}\right)$, $\hat{u}_i = y_i - \hat{G}_i$, and $\hat{g}_i = g\left(x_i\hat{\theta}\right) = \partial G\left(x_i\hat{\theta}\right)/\partial\left(x_i\hat{\theta}\right)$.

Using R, estimates for θ may be obtained by using the command `glm`, which is included in the base installation of R. However, here I consider the package `frm`, which includes many other features that are specific for fractional regression models. The main estimation command of this package has also the name `frm`, which is an extension of the command `glm`. To estimate a logit fractional regression model, the following code may be applied:

```
> frm(y=Y,x=X,linkfrac="logit")

*** Fractional logit regression model ***

               Estimate Std. Error t value Pr(>|t|)
INTERCEPT     -1.327157   0.798204  -1.663   0.096 *
Growth         0.004894   0.003752   1.304   0.192
Size           0.001633   0.070879   0.023   0.982
Profitability -3.374131   0.959341  -3.517   0.000 ***
Tangibility    1.111818   0.368599   3.016   0.003 ***

Note: robust standard errors

Number of observations: 379
R-squared: 0.067
```

By default, the variance of each regression coefficient is computed in a robust way and is based on the expected information matrix, as in (2). We may add the option `var.eim=F` if, instead, the observed information matrix is to be considered and the option `var.type` to use other variance formulas, such as `cluster` to compute a cluster–robust estimator. The R^2 measure provided in the output is calculated as the square of the correlation coefficient between the actual and fitted values of the dependent variable. To implement any of the other four alternative functional forms mentioned above one just needs to change the option `linkfrac` to the name of the desired specification.

2.2 Two-part models

In economics and other social sciences, it is common to observe a substantial proportion of limit values in samples of fractional data. However, most samples cluster only at zero or one and not at both boundaries simultaneously. Examples include the proportion of exports in total sales (Wagner, 2003), the proportion of deaths caused by traffic accidents across districts (Ospina and Ferrari, 2012), employer 401(k) contribution match rates (Papke and Wooldridge, 1996) and data envelopment analysis efficiency scores (Ramalho et al., 2010). In the first two examples samples cluster at zero but there are no observations at one, and in the last two examples it occurs the opposite.

Although the models discussed in the previous section can be used in the presence of boundary values of fractional responses, this may not be the best

option when the number of corner observations is large. For the most common case, the observation of extreme values at only one of the boundaries, a better approach may be the use of a two-part fractional regression model (Ramalho et al., 2011). This model first uses a binary regression model to explain the probability of observing a specific boundary value (0 or 1) and then uses a conditional mean model (one of those described in the previous section) to explain the value observed for the remaining values of the fractional response.

To simplify the exposition, next we focus on the case where we have limit observations only at zero.[a] The two-part fractional regression model may be expressed as

$$\begin{aligned} E(y_i|x_i) &= \Pr(y_i > 0|x_{ib}) \cdot E\left(y_i|x_{if}, y_i > 0\right) \\ &= G_b(x_{ib}\theta_b) \cdot G_f\left(x_{if}\theta_f\right), \end{aligned} \tag{3}$$

where x_{ib} and x_{if} are the explanatory variables used in the binary and fractional components of the model, respectively, θ_b and θ_f are vectors of variable coefficients and $G_b(\cdot)$ and $G_f(\cdot)$ may be specified in exactly the same way as the $G(\cdot)$ function considered in the previous section, since both must be also bounded by 0 and 1. The two components of (3) are assumed to be independent and hence estimated separately: while the binary component is estimated by maximum likelihood (ML) using the whole sample, the fractional component is estimated by QML using only the subsample of nonzero observations. Clearly, the adaptation of this formulation for the case where the boundary value observed with a nontrivial probability is 1 is straightforward.

As calculated before, the *Leverage* variable has 16.2% of observations at 0, but no observations at 1. Therefore, a natural candidate for modeling that variable is the two-part model just described. With `frm`, we can use any combination of functional forms for the binary and fractional parts of the model. For example, for estimating a two-part model based on cauchit binary and loglog fractional specifications, we may use the following code:

```
> frm(y=Y,x=X,type="2P",linkbin="cauchit",linkfrac="loglog")

*** Binary component of a two-part model - cauchit ***

               Estimate Std. Error t value Pr(>|t|)
INTERCEPT      4.680859   2.047746   2.286   0.022 **
Growth         0.008303   0.009789   0.848   0.396
Size          -0.320178   0.169118  -1.893   0.058 *
Profitability -7.873038   2.180580  -3.611   0.000 ***
Tangibility    5.484858   1.710037   3.207   0.001 ***
```

[a]Actually, we may also have observations at one, but it is assumed that they are generated by the same mechanism as the other nonzero values of the response variable.

```
Number of observations: 379
R-squared: 0.1

*** Fractional component of a two-part model - loglog ***

               Estimate Std. Error t value Pr(>|t|)
INTERCEPT     -0.692913   0.421335  -1.645   0.100
Growth         0.003689   0.002121   1.739   0.082 *
Size           0.037108   0.037704   0.984   0.325
Profitability -1.366118   0.535941  -2.549   0.011 **
Tangibility    0.425081   0.209386   2.030   0.042 **

Note: robust standard errors

Number of observations: 310
R-squared: 0.043

*** Two-part model - binary cauchit + fractional loglog ***

R-squared: 0.073
```

To estimate this two-part model, we needed to add the options `type="2P"` (the default option is `"1P"`, which indexes the standard fractional conditional mean models of the previous section) and `linkbin`. If we were interested only in the binary or the fractional component of the two-part model, then we could have used the option `type="2Pbin"` or `"2Pfrac"`, respectively, dropping from the command line the options `"linkfrac"` (in the former case) or `"linkbin"` (in the latter case).

Three R^2 measures are provided, one for each part of the model and a global statistic based on the predicted values of y_i according to Eq. (3). Note that the variance of each regression coefficient in the binary model is computed, by default, in an efficient way ($V = A^{-1}$), since this model is estimated by ML. Also by default the same covariates are used in each part of the model. To use a different set of explanatory variables in the fractional specification we would need to specify the option `x2`. For example, we may reestimate the previous model without the covariates that are nonsignificant at the 10% level:

```
> frm(y=Y,x=X[,2:4],x2=X[,c(1,3:4)],type="2P",linkbin=
        "cauchit",linkfrac="loglog")

*** Binary component of a two-part model - cauchit ***
```

```
               Estimate Std. Error t value Pr(>|t|)
INTERCEPT      5.037010   1.987137   2.535   0.011 **
Size          -0.347119   0.164126  -2.115   0.034 **
Profitability -7.210996   2.044513  -3.527   0.000 ***
Tangibility    5.295094   1.662974   3.184   0.001 ***

Number of observations: 379
R-squared: 0.096

*** Fractional component of a two-part model - loglog ***

               Estimate Std. Error t value Pr(>|t|)
INTERCEPT     -0.285641   0.077475  -3.687   0.000 ***
Growth         0.003694   0.002096   1.762   0.078 *
Profitability -1.301778   0.542970  -2.398   0.017 **
Tangibility    0.406902   0.208460   1.952   0.051 *

Note: robust standard errors

Number of observations: 310
R-squared: 0.04

*** Two-part model - binary cauchit + fractional loglog  ***

R-squared: 0.072
```

Finally, note that by default the `frm` command with the option `type="2P"`, `"2Pbin"`, or `"2Pfrac"` assumes that the relevant boundary value for defining the two-part model is zero. If, in contrast, that value is one, then we can use the option `inflation=1` (the default option is `inflation=0`).

2.3 Partial effects

In nonlinear regression models, such as the ones considered in this chapter, the magnitude of the regression coefficients cannot be compared across models based on different functional forms. However, it is relatively easy to calculate partial effects with a meaningful and comparable interpretation. For conditional mean models (from now on called "one-part" models), the average effect on y of a unitary change in x_j is given by:

$$\frac{\partial E(y_i|x_i)}{\partial x_{ij}} = \theta_j g(x_i\theta), \tag{4}$$

where $g(x_i\theta)$ is given by $G(x_i\theta)[1-G(x_i\theta)]$ (logit), $\phi(x_i\theta)$ (probit), $e^{-x_i\theta}G(x_i\theta)$ (loglog), $e^{x_i\theta}[1-G(x_i\theta)]$ (cloglog) or $\frac{1}{\pi}\frac{1}{(x_i\theta)^2+1}$ (cauchit). For two-part models, partial effects are given by:

$$\frac{\partial E(y_i|x_i)}{\partial x_{ij}} = \frac{\partial \Pr(y_i > 0|x_{ib})}{\partial x_{ij}} \cdot E\left(y_i|x_{if}, y_i > 0\right) + \frac{\partial E\left(y_i|x_{if}, y_i > 0\right)}{\partial x_{ij}} \cdot \Pr(y_i > 0|x_{ib})$$
$$= \theta_{bj} g_b(x_i\theta) \cdot G_f\left(x_{if}\theta_f\right) + \theta_{fj} g_f(x_i\theta) \cdot G_b(x_{ib}\theta_b), \tag{5}$$

where $g_b(x_i\theta) = \partial G_b(x_i\theta)/\partial(x_i\theta)$ and $g_f(x_i\theta) = \partial G_f(x_i\theta)/\partial(x_i\theta)$. Note that in this case the independent analysis of the partial effects

$$\frac{\partial \Pr(y_i > 0|x_{ib})}{\partial x_{ij}} = \theta_{bj} g_b(x_i\theta) \tag{6}$$

and

$$\frac{\partial E\left(y_i|x_{if}, y_i > 0\right)}{\partial x_{ij}} = \theta_{fj} g_f(x_i\theta) \tag{7}$$

may be also of interest.

Because the partial effects depend on the value of the explanatory variables, in empirical work it is customary to measure them in two main ways: (i) replacing x_i by some specific values, such as the mean of each covariate or the characteristics of a particular individual (conditional partial effect); or (ii) calculating the average partial effect of all sampling units (average partial effect).

The package `frm` contains the command `frm.pe` that allows easy calculation of both conditional and average partial effects based on formulas (4)–(7) and reports not only their estimated values but also standard errors and statistical significance. In all cases, we first need to estimate the relevant model using the appropriate `type` option (`1P` for (4), `2P` (5), `2Pbin` (6), and `2Pfrac` (7)) and store the results as an R object. For example, to estimate average partial effects for a one-part fractional regression model, the following code may be used (note the use of the option `table=F` in the command `frm` to suppress the output):

```
> res <- frm(y=Y,x=X,linkfrac="logit",table=FALSE)
> frm.pe(res,APE=TRUE,CPE=FALSE)

*** Average partial effects ***

Fractional logit model

              Estimate Std. Error t value Pr(>|t|)
Growth          0.0009     0.0007   1.302   0.193
Size            0.0003     0.0125   0.023   0.982
Profitability  -0.5969     0.1686  -3.541   0.000 ***
Tangibility     0.1967     0.0643   3.058   0.002 ***
```

In the calculation of the standard errors it is taken into account the option that was previously chosen for estimating the model.

Conditional partial effects require the use of the option `at`. We may choose to evaluate the covariates at their mean or median values or provide a numeric vector containing specific values for each explanatory variable, as follows:

```
> frm.pe(res,APE=FALSE,CPE=TRUE,at="median")

*** Conditional partial effects ***

Fractional logit model

              Estimate Std. Error t value Pr(>|t|)
Growth          0.0009     0.0007   1.301   0.193
Size            0.0003     0.0126   0.023   0.982
Profitability  -0.6000     0.1696  -3.537   0.000 ***
Tangibility     0.1977     0.0652   3.032   0.002 ***

------------------
Note: covariates evaluated at median (or mode, for dummies)
        values

> frm.pe(res,APE=FALSE,CPE=TRUE,at=c(0,7,0.1,0.5))

*** Conditional partial effects ***

Fractional logit model

              Estimate Std. Error t value Pr(>|t|)
Growth          0.0009     0.0007   1.302   0.193
Size            0.0003     0.0133   0.023   0.982
Profitability  -0.6331     0.1789  -3.538   0.000 ***
Tangibility     0.2086     0.0688   3.032   0.002 ***

------------------
Note: covariates evaluated at the following values:

       Growth          Size Profitability   Tangibility
          0.0           7.0           0.1           0.5
```

For the other three types of partial effects the procedure is similar. In the case of the overall partial effects in two-part models, the current version of `frm.pe` requires both the binary and fractional specifications to use the same covariates.

2.4 Specification tests

The crucial assumption underlying all fractional regression models discussed in this chapter is the correct specification of the conditional mean of y, that is the $G(\cdot)$ function in (1) and the $G_b(\cdot)$ and $G_f(\cdot)$ functions in (5). Therefore, it is important to test the statistical validity of this assumption. To this end, there are three main tests that may be applied: (i) a RESET-type test, which was proposed by Papke and Wooldridge (1996); (ii) the goodness-of-functional form (GOFF) tests developed by Ramalho et al. (2011) and Ramalho et al. (2014); and (iii) the P test for general nonnested hypotheses proposed by Davidson and MacKinnon (1981), which was adapted to the fractional framework by Ramalho et al. (2011).

The RESET test is based on the fact that, using standard approximation results for polynomials, any index model of the form $L(x_i\theta)$ can be approximated by the model $M\left[x_i\theta + \sum_{j=1}^{J}\phi_j(x_i\theta)^{j+1}\right]$ for J large enough, where $L(\cdot)$ and $M(\cdot)$ are any mathematical functions. Thus, to test the functional form assumed for $G(\cdot)$ in one-part models, we may consider the generalized specification

$$E(y_i|x_i) = G\left[x_i\theta + \sum_{j=1}^{J}\phi_j\left(x_i\hat{\theta}\right)^{j+1}\right] \tag{8}$$

and test for H_0: $\phi = 0$, where ϕ is a J-dimensional vector. In practice, typically, $J \leq 3$. This test may be also used to assess the suitability of the functional form assumed in the separate components of two-part models: we just need to replace $G(\cdot)$ by $G_b(\cdot)$ or $G_f(\cdot)$ in Eq. (8).

The GOFF tests have three main variants, each one based on a different generalization of $G(\cdot)$. The generalizations underlying the GOFF1 and GOFF2 versions are, respectively, the following:

$$E(y_i|x_i) = G(x_i\theta)^{\alpha} \tag{9}$$

and

$$E(y_i|x_i) = 1 - [1 - G(x_i\theta)]^{\alpha}, \tag{10}$$

where $\alpha > 0$. Both (9) and (10) induce (complementary forms of) asymmetry in $G(\cdot)$ and reduce to this function when $\alpha = 1$. Therefore, to test the functional form assumed for $G(\cdot)$, we may test for H_0: $\alpha = 1$. The third variant, called generalized GOFF (GGOFF) test, is based on an alternative generalization of $G(\cdot)$ that is a mixture of both (9) and (10) and allows not only a wider variety of asymmetric forms for $E(y_i|x_i)$, but also for many different symmetric shapes:

$$E(y_i|x_i) = \lambda G(x_i\theta)^{\alpha_1} + (1-\lambda)\{1 - [1 - G(x_i\theta)]^{\alpha_2}\}, \tag{11}$$

where $0 < \lambda < 1$ and $\alpha_1, \alpha_2 > 0$. In this case, the hypothesis to be tested is H_0: $\alpha_1 = \alpha_2 = 1$. As for RESET, all GOFF tests may be straightforwardly

adapted to separately testing the specifications assumed for $G_b(\cdot)$ and $G_f(\cdot)$ in the binary and fractional components of two-part models.

While the RESET and GOFF tests may only be applied to assess the suitability of the functional form assumed in one-part models or in the separate components of two-part models, the P test may be used not only for that but also to test the full specification of two-part models against both one-part models and other two-part models, and vice versa. Suppose that $L(\cdot)$ and $M(\cdot)$ are competing specifications for $E(y_i|x_i)$, each representing a one-part or a two-part model. Following Ramalho et al. (2011), testing the suitability of $L(x_i\theta)$ after taking into account the information provided by the alternative specification $M(x_i\eta)$ corresponds to test for H_0: $\delta_2 = 0$ in the auxiliary regression

$$\left[y - L\left(x_i\hat{\theta}\right)\right] = l\left(x_i\hat{\theta}\right)x_i\delta_1 + \delta_2\left[M(x_i\hat{\eta}) - L\left(x_i\hat{\theta}\right)\right] + error, \qquad (12)$$

where $\hat{\theta}$ and $\hat{\eta}$ are previously obtained by estimating separately each model and $l\left(x_i\hat{\theta}\right) = \partial L\left(x_i\hat{\theta}\right)/\partial\left(x_i\hat{\theta}\right)$. To perform the opposite test, i.e., testing the suitability of $M(x_i\theta)$ after taking into account the information provided by $L(x_i\eta)$, we just need to reverse the roles of the two models in regression (12).

To apply the three classes of tests, we may use the commands `frm.reset`, `frm.ggoff`, and `frm.ptest` included in the package `frm`. In all cases, LM and Wald versions of the tests are available. When testing the binary component of two-part models, LR versions of the RESET and GOFF tests are also available.

To illustrate the application of the tests, suppose that we are considering the following alternative models: logit one-part model, cauchit one-part model, and binary cauchit + fractional probit two-part model. First, we need to estimate all models. In the case of the two-part model, we need to estimate it not only separately (options `type="2Pbin"` and `type="2Pfrac"`) but, because of the P test, also as a whole (option `type="2P"`):

```
logit1 <- frm(y=Y,x=X,type="1P",linkfrac="logit",table=FALSE)
cauchit1 <- frm(y=Y,x=X,type="1P",linkfrac="cauchit",table=
        FALSE)
cauchit2b <- frm(y=Y,x=X,type="2Pbin",linkbin="probit",table=
        FALSE)
probit2f <- frm(y=Y,x=X,type="2Pfrac",linkfrac="probit",table=
        FALSE)
caupro2 <- frm(y=Y,x=X,type="2P",linkbin="cauchit",linkfrac=
        "probit",table=FALSE)
```

Then, we apply LM and Wald versions of both the RESET test based on 1 (number 2 of the vector defined by `lastpower.vec`) and 2 (number 3 of the same vector) fitted powers and the GOFF tests to the two one-part models:

```
> frm.reset(logit1,lastpower.vec=c(2:3),version=c("Wald","LM"))

*** RESET test ***

H0:  Fractional logit model

Version Statistic p-value
  LM(2)     2.375   0.123
Wald(2)     1.781   0.182
  LM(3)     2.452   0.293
Wald(3)     2.237   0.327

> frm.ggoff(logit1,version=c("Wald","LM"))

*** GGOFF test ***

H0:  Fractional logit model

 Test Version Statistic p-value
GOFF1      LM     2.442   0.118
GOFF1    Wald     1.919   0.166
GOFF2      LM     2.425   0.119
GOFF2    Wald     2.111   0.146
GGOFF      LM     2.445   0.294
GGOFF    Wald     2.059   0.357

> frm.reset(cauchit1,lastpower.vec=c(2:3),version=c("Wald",
        "LM"))

*** RESET test ***

H0:  Fractional cauchit model

Version Statistic p-value
  LM(2)     4.676   0.031 **
Wald(2)     3.576   0.059 *
  LM(3)     4.856   0.088 *
Wald(3)     3.754   0.153

> frm.ggoff(cauchit1,version=c("Wald","LM"))

*** GGOFF test ***

H0:  Fractional cauchit model

 Test Version Statistic p-value
GOFF1      LM     4.229   0.040 **
GOFF1    Wald     3.292   0.070 *
GOFF2      LM     3.481   0.062 *
GOFF2    Wald     2.822   0.093 *
GGOFF      LM     5.238   0.073 *
GGOFF    Wald     3.920   0.141
```

Clearly, the cauchit specification does not seem appropriate.

The same tests are now applied separately to the individual components of the two-part model (in the binary case, we also calculate an LR version of the tests):

```
> frm.reset(cauchit2b,lastpower.vec=c(2:3),version=c("Wald",
        "LM","LR"))

*** RESET test ***

H0:  Binary probit component of a two-part model

Version Statistic p-value
  LM(2)     1.117   0.291
  LR(2)     1.060   0.303
Wald(2)     1.061   0.303
  LM(3)     2.850   0.241
  LR(3)     2.208   0.332
Wald(3)     2.946   0.229

> frm.ggoff(cauchit2b,version=c("Wald","LM","LR"))

*** GGOFF test ***

H0:  Binary probit component of a two-part model

 Test Version Statistic p-value
GOFF1      LM     1.256   0.262
GOFF1      LR     1.180   0.277
GOFF1    Wald     1.193   0.275
GOFF2      LM     0.777   0.378
GOFF2      LR     0.760   0.383
GOFF2    Wald     0.721   0.396
GGOFF      LM     3.326   0.190
GGOFF      LR     2.747   0.253
GGOFF    Wald     3.452   0.178

> frm.reset(probit2f,lastpower.vec=c(2:3),version=c("Wald",
        "LM"))

*** RESET test ***

H0:  Fractional probit component of a two-part model

Version Statistic p-value
  LM(2)     0.624   0.430
Wald(2)     0.613   0.434
  LM(3)     0.625   0.732
Wald(3)     0.613   0.736
```

```
> frm.ggoff(probit2f,version=c("Wald","LM"))

*** GGOFF test ***

H0:  Fractional probit component of a two-part model

 Test Version Statistic p-value
GOFF1      LM     0.625   0.429
GOFF1    Wald     0.613   0.434
GOFF2      LM     0.623   0.430
GOFF2    Wald     0.613   0.434
GGOFF      LM     0.625   0.732
```

In this case, there is no statistical evidence against the conditional mean assumptions made.

Finally, we test the logit one-part model against the estimated two-part model using the P test (Wald version):

```
> frm.ptest(logit1,caupro2,version="Wald")

*** P test ***

H0:  Fractional logit model
H1:  Binary cauchit + Fractional probit two-part model

Version Statistic p-value
   Wald     1.834   0.067 *

H0:  Binary cauchit + Fractional probit two-part model
H1:  Fractional logit model

Version Statistic p-value
   Wald     1.150   0.251
```

At a 10% significance level, the correct specification of the one-part model is rejected, while the two-part model seems to be a suitable specification for our data.

3 Linearized- and exponential-fractional estimators

The previous models do not allow for omitted covariates, be they correlated (endogeneity) or not (neglected heterogeneity) with the included regressors. In this section we still focus on a cross-sectional framework, but consider alternative regression models where those issues may be present. The relevant R package for this section is `frmhet`:

```
library(frmhet)
```

3.1 Framework

Economic theory often postulates that a response variable depends on both observed and unobserved variables. However, the econometric models described above assume that all relevant variables are observed. Because it is not easy to work with unobservables in the framework of Eq. (1), Ramalho and Ramalho (2017) consider alternatively the structural model

$$y_i = G(x_i\theta + u_i), \tag{13}$$

where u_i denotes the unobservables. In this model, observed and omitted variables are treated in a similar manner and we have what Heckman (2000) calls a "well-posed economic model" where "all of the input processes, observed and unobserved by the analyst, and their relationship to outputs" are specified.

From (13), it follows that

$$E(y_i|x_i) = E_u[G(x_i\theta + u_i)] = \int_U G(x_i\theta + u_i) f(u_i|x_i) du_i, \tag{14}$$

where $E_u[\cdot]$ denotes expectation with respect to the conditional distribution of u and U and $f(u_i|x_i)$ denote, respectively, the sample space and the conditional (on the observables) density of u. Clearly, in this setting conditioning on x_i does not remove the dependency of the model on the unobservables and the QML estimator based on Eq. (1) will no longer be consistent, even if x_i and u_i are not correlated.

To overcome the inconsistency of standard estimators in this context, and to avoid making distributional assumptions as seems to be required by (14), Ramalho and Ramalho (2017) propose rewriting model (13) in such a way that observed and unobserved covariates become additively separable. Their proposal requires the $G(x_i\theta + u_i)$ function to be decomposed as $G_1[\exp(x_i\theta + u_i)]$ such that (13) may be re-written as:

$$y_i = G_1[\exp(x_i\theta + u_i)], \tag{15}$$

where $G_1(\cdot)$ is an invertible function. Let $H_1 = G_1(\cdot)^{-1}$. Then, from (15) it follows that:

$$H_1(y_i) = \exp(x_i\theta + u_i). \tag{16}$$

This equation is the basis for the so-called exponential-fractional regression model (EFRM) proposed by Ramalho and Ramalho (2017). The EFRM includes as particular cases the logit ($H_1(y_i) = \frac{y_i}{1-y_i}$) and cloglog ($H_1(y_i) = -\ln(1-y_i)$) models. The other specifications considered in the previous section (probit, loglog, and cauchit) cannot be used in this framework, because they cannot be decomposed as in (15).

An alternative to the EFRM is the linearized-fractional regression model (LFRM), which is given by:

$$H(y_i) = x_i\theta + u_i, \tag{17}$$

where $H = G(\cdot)^{-1}$. The LFRM applies to all previously discussed specifications. For the logit and cloglog models, $H(y_i) = \ln H_1(y_i)$. For the models where $H_1(y_i)$ does not exist, $H(y_i) = \Phi^{-1}(y_i)$ (probit), $-\ln[-\ln(y_i)]$ (loglog), or $\tan[\pi(y_i - 0.5)]$ (cauchit).

The main advantage of LFRM over EFRM is its simplicity: Eq. (17) represents a linear regression model and, with exogenous regressors, may be estimated by OLS, while Eq. (16) represents an exponential regression model and is typically estimated by QML. On the other hand, while the LFRM is not defined for both the boundary values of the fractional response, the EFRM accommodates the value zero of y_i. Since, as discussed in Section 2.2, most samples of fractional responses include observations at (only) one of the limits, this is a very important advantage of the EFRM relative to the LFRM.[b]

3.2 Neglected heterogeneity

Assume that u_i and x_i are uncorrelated. In particular, without any loss of generality, provided that x_i contains a constant term, assume that $E[\exp(u_i)|x_i] = 1$. Under this assumption, and after transforming the dependent variable as indicated above, the EFRM (16) may be estimated as a standard exponential regression model using, as in Santos Silva and Tenreyro (2006), Poisson- or Exponential-based QML. Because in the next section we allow for endogenous regressors and Poisson-QML is not well suited to deal with endogenous covariates in models with multiplicative heterogeneity (Windmeijer and Santos Silva, 1997), package `frmhet` considers only Exponential-based QML estimation.

To estimate a logit fractional regression model allowing for neglected heterogeneity, the command `frmhet`, with the option `type="GMMx"`, is applied

```
> frmhet(y=Y,x=X,type="GMMx",link="logit")

*** Fractional logit regression model ***
*** Estimator: GMMx

               Estimate Std. Error t value Pr(>|t|)
INTERCEPT     -0.884512   1.242392  -0.712   0.477
Growth         0.008382   0.005555   1.509   0.131
Size           0.025152   0.104372   0.241   0.810
Profitability -4.276801   1.370205  -3.121   0.002 ***
Tangibility    1.151494   0.456426   2.523   0.012 **

Note: robust standard errors

Number of observations: 379
```

[b]Note that, if needed, we can redefine the response variable and model its complementary, which means the EFRM is applicable irrespective of the (original) boundary value that is observed with a nonzero probability.

Note that the dependent variable is introduced in its original fractional form, since the transformation $H_1(y_i)$ is automatically applied by the command according to the selected specification. While in the `frm` package the model functional form was specified using the option `linkfrac`, here the corresponding definition is simply `link`, since `frmhet` is only implemented for conditional mean models. By default, the variance is calculated in a robust way, but a cluster–robust option is also available.

Ramalho and Ramalho (2017) designated the Exponential-based QML estimator by GMMx because the EFRM (16) may be also represented in a moment condition form that is particularly useful for dealing with endogeneity. Dividing both sides of (16) by $\exp(x_i\theta)$ and subtracting one, the EFRM may be equivalently expressed as:

$$\frac{H_1(y_i)}{\exp(x_i\theta)} - 1 = \exp(u_i) - 1. \tag{18}$$

Under the conditions stated above, it follows that

$$E\left[\frac{H_1(y_i)}{\exp(x_i\theta)} - 1 \middle| x_i\right] = 0, \tag{19}$$

which may be used to generate a set of moment conditions and allow estimation of θ. In particular, GMMx is based on the following orthogonality conditions:

$$E\left\{x_i'\left[\frac{H_1(y_i)}{\exp(x_i\theta)} - 1\right]\right\} = 0, \tag{20}$$

which correspond to the first-order conditions defining Exponential-based QML estimators.

Using the command `frmhet` with the option `type="LINx"`, we may also estimate the LFRM (17). In this case, because y_i is zero for some firms, we obtain an error message in the application of the command:

```
> frmhet(y=Y,x=X,type="LINx",link="logit")
Error in frmhet(y = Y, x = X, type = "LINx", link = "logit") :
0/1 values for the response variable: LIN estimators require
        adjustment
```

To overcome this situation, we may add an arbitrary constant to all observations of y_i (for example, 0.001 or 0.000001) or we may drop observations with $y_i = 0$, as follows:

```
> frmhet(y=Y,x=X,type="LINx",link="logit",adjust=0.001)

*** Fractional logit regression model ***
*** Estimator: LINx
```

```
*** Adjustment: 0.001 added to all observations

               Estimate Std. Error t value Pr(>|t|)
INTERCEPT     -0.860114   1.566791  -0.549   0.583
Growth         0.007329   0.007316   1.002   0.316
Size          -0.165517   0.140634  -1.177   0.239
Profitability -6.940474   1.757618  -3.949   0.000 ***
Tangibility    2.671571   0.681805   3.918   0.000 ***

Note: robust standard errors

Number of observations: 379

> frmhet(y=Y,x=X,type="LINx",link="logit",adjust=0.000001)

*** Fractional logit regression model ***
*** Estimator: LINx
*** Adjustment: 1e-06 added to all observations

                Estimate Std. Error t value Pr(>|t|)
INTERCEPT       0.413200   2.990148   0.138   0.890
Growth          0.008621   0.014070   0.613   0.540
Size           -0.411956   0.270661  -1.522   0.128
Profitability -13.044430   3.628739  -3.595   0.000 ***
Tangibility     4.735408   1.270253   3.728   0.000 ***

Note: robust standard errors

Number of observations: 379

> frmhet(y=Y,x=X,type="LINx",link="logit",adjust="drop")

*** Fractional logit regression model ***
*** Estimator: LINx
*** Adjustment: all boundary observations dropped

               Estimate Std. Error t value Pr(>|t|)
INTERCEPT     -1.746243   1.266901  -1.378   0.168
Growth         0.010851   0.006185   1.754   0.079 *
Size          -0.002671   0.117436  -0.023   0.982
Profitability -2.825326   1.631900  -1.731   0.083 *
Tangibility    1.389850   0.517068   2.688   0.007 ***

Note: robust standard errors

Number of observations: 310
```

Note how the magnitude of the parameter estimates is highly sensitive to the constant added to y_i and how dropping observations with $y_i = 0$ may lead to different conclusions in terms of the significance of the variables, which is in accordance with the Monte Carlo results obtained by Ramalho and Ramalho (2017). Clearly, the LFRM should be avoided when there are boundary observations for the response variable.

The RESET test may be implemented in this context using the command `frmhet.reset`. For example, for computing the GMMx estimator, we may use a logit or a cloglog specification. Based on a RESET test with two fitted powers, the former functional form is preferable:

```
> res <- frmhet(y=Y,x=X,type="GMMx",link="logit",table=FALSE)
> frmhet.reset(res,lastpower.vec=3,version="Wald")

*** RESET test ***
Fractional logit regression model

H0:  Estimator: GMMx

Version Statistic p-value
Wald(3)     2.412   0.299

> res <- frmhet(y=Y,x=X,type="GMMx",link="cloglog",table=FALSE)
> frmhet.reset(res,lastpower.vec=3,version="Wald")

*** RESET test ***
Fractional cloglog regression model

H0:  Estimator: GMMx

Version Statistic p-value
Wald(3)   740.519   0.000 ***
```

3.3 Endogenous regressors

Now, we assume that one or more explanatory variables are endogenous. Let z_i denote an s-vector of instrumental variables, including the exogenous explanatory variables. The estimators of the previous section may be straightforwardly extended to deal with endogeneity. For example, Ramalho and Ramalho (2017) GMMz estimator is defined by the following moment conditions, which is an adaptation of (20):

$$E\left\{z_i'\left[\frac{H_1(y_i)}{\exp(x_i\theta)} - 1\right]\right\} = 0. \tag{21}$$

Similarly, we may define a LINz estimator, which in this case is also computed as a GMM estimator, based on:

$$E\left\{z_i'[H(y_i) - x_i\theta]\right\} = 0. \tag{22}$$

Suppose that *Profitability* is the endogenous variable and *ProfitIVa* the instrumental variable. Then, the logit version of the GMM estimator may be obtained as follows:

```
> Z <- cbind(Growth, Size, Tangibility, ProfitIVa)
> Z <- Z[Year==2007,]
> frmhet(y=Y,x=X,z=Z,type="GMMz",link="logit")
ALGORITHM DID NOT CONVERGE
```

However, the optimization algorithm behind `frmhet` did not converge in this example. Whenever this happens, there are two main alternatives that we may try to obtain the estimates. The first is simply to provide a numeric vector containing starting values for the parameters to be optimized, using the option `start`:

```
> frmhet(y=Y,x=X,z=Z,type="GMMz",link="logit",start=c(-1,0,0,
        -8,3))

*** Fractional logit regression model ***
*** Estimator: GMMz

                Estimate Std. Error t value Pr(>|t|)
INTERCEPT      -2.240221   1.485953  -1.508   0.132
Growth          0.024997   0.019006   1.315   0.188
Size            0.178961   0.166608   1.074   0.283
Profitability -11.168674   6.182291  -1.807   0.071 *
Tangibility     1.648019   0.947288   1.740   0.082 *

Note: robust standard errors

Number of observations: 379
```

The other alternative is to change some of the control parameters used by `nlminb`, the R command on which `frmhet` is based. Below, we change the maximum number of iterations and evaluations of the objective function allowed:

```
> frmhet(y=Y,x=X,z=Z,type="GMMz",link="logit",control=
        list(iter.max=300,eval.max=400))

*** Fractional logit regression model ***
*** Estimator: GMMz
```

```
                Estimate Std. Error t value Pr(>|t|)
INTERCEPT      -2.240220   1.485953  -1.508   0.132
Growth          0.024997   0.019006   1.315   0.188
Size            0.178961   0.166608   1.074   0.283
Profitability -11.168673   6.182291  -1.807   0.071 *
Tangibility     1.648019   0.947288   1.740   0.082 *

Note: robust standard errors

Number of observations: 379
```

The results obtained in both cases are virtually identical. See the help file for `nlmimb` to find all control parameters that may be changed.[c]

When the number of instruments is larger than the number of endogenous covariates, then Hansen (1982) *J* test statistic of overidentifying moment conditions is also reported in the output of `frmhet`. For example, if we use also *ProfitIVb* as instrument, we find that we cannot reject the exogeneity of the variables contained in z_i:

```
> Z <- cbind(Growth, Size, Tangibility, ProfitIVa, ProfitIVb)
> Z <- Z[Year==2007,]
> frmhet(y=Y,x=X,z=Z,type="GMMz",link="logit")

*** Fractional logit regression model ***
*** Estimator: GMMz

                Estimate Std. Error t value Pr(>|t|)
INTERCEPT      -1.227499   1.029194  -1.193   0.233
Growth          0.006670   0.006838   0.975   0.329
Size            0.039852   0.090423   0.441   0.659
Profitability -2.410194   2.759632  -0.873   0.382
Tangibility     1.102152   0.462752   2.382   0.017 **

Note: robust standard errors

Number of observations: 379

J test of overidentifying moment conditions: 0.6598782
        (p-value: 0.4166029 )
```

[c]The `frm` command considered in Section 2 is based on the R command `glm`. If we find convergence problems with `frm`, we may use a similar procedure to change the control parameters used by `glm`.

One important feature of GMMz and LINz estimators is that they do not require the specification of a reduced form model for the endogenous explanatory variable, which is in clear contrast to most instrumental variables that have been proposed for nonlinear regression models. However, if we have information about the reduced form of the endogenous covariate, then a potentially more efficient estimator may be constructed, based on Wooldridge (2015) control function approach.

Denote by x_{i2} the set of k_2 endogenous explanatory variables and assume that a linear reduced form with additive disturbances v can be found for the monotonic transformation $S(x_{i2})$:

$$S(x_{i2}) = z_i\pi + v_i, \tag{23}$$

where π is an $s \times k_2$ matrix of reduced form parameters. Assume also that

$$u_i = v_i\rho + \epsilon_i, \tag{24}$$

where v_i is independent of both z_i and ϵ_i. Replacing u_i in (16) by the right-hand side of (24) and v_i by its OLS estimate $\hat{v}_i = S(x_{i2}) - z_i\hat{\pi}$, and repeating all previous procedures that led to Eq. (20), we obtain a new set of moment conditions,

$$E\left\{w_i'\left[\frac{H_1(y_i)}{\exp(x_i\theta + \hat{v}_i\rho)} - 1\right]\right\} = 0, \tag{25}$$

where $w_i' = (x_i, \hat{v}_i)'$. Ramalho and Ramalho (2017) designated the GMM estimator of θ and ρ based on (25) by GMMxv. A LINxv estimator may also be constructed in a similar way, based on:

$$E\left\{w_i'[H_1(y_i) - (x_i\theta + \hat{v}_i\rho)]\right\} = 0. \tag{26}$$

To use the command `frmhet` to obtain GMMxv or LINxv estimators, we may apply command lines similar to those illustrated above for GMMz, but with the option `type="GMMxv"` or `"LINxv"` and, in addition, we need to indicate, through the option `var.endog`, the name of the vector that contains the values of the endogenous covariate (or of some transformation of it, in case the $S(\cdot)$ function in Eq. (23) is not linear).[d] Continuing the example above, suppose that we assume a linear form for $S(\cdot)$. Thus, the GMMxv is implemented as follows:

```
> Profit2007 <- Profitability[Year==2007]
> frmhet(y=Y,x=X,z=Z,var.endog=Profit2007,type="GMMxv",
        link="logit",control=list(iter.max=300,eval.max=400))

*** Fractional logit regression model ***
*** Estimator: GMMxv
```

[d]Note that the current `GMMxv/LINxv` versions of `frmhet` only allow for one endogenous explanatory variable. In contrast, `GMMz/LINz` allow for multiple endogenous covariates.

```
               Estimate Std. Error t value Pr(>|t|)
INTERCEPT     -0.950287   1.114195  -0.853   0.394
Growth         0.009339   0.005580   1.674   0.094 *
Size           0.040577   0.089653   0.453   0.651
Profitability -5.754944   3.423699  -1.681   0.093 *
Tangibility    1.097722   0.484070   2.268   0.023 **
vhat           2.096154   4.389900   0.477   0.633

Reduced form:
               Estimate Std. Error t value Pr(>|t|)
Z_INTERCEPT    0.000813   0.036972   0.022   0.982
Z_Growth       0.000755   0.000204   3.694   0.000 ***
Z_Size         0.004382   0.003271   1.340   0.180
Z_Tangibility -0.027189   0.019124  -1.422   0.155
Z_ProfitIVa    0.155413   0.028526   5.448   0.000 ***
Z_ProfitIVb    0.243621   0.030669   7.943   0.000 ***

Note: robust standard errors

Number of observations: 379
```

Note that for this class of estimators, the output includes also the results of the estimation of the reduced form model.

If, for example, an exponential form is assumed for $S(\cdot)$, then we must first create the exponential transformation of *Profitability*:

```
> Profit2007exp <- exp(Profit2007)
> frmhet(y=Y,x=X,z=Z,var.endog=Profit2007exp,type="GMMxv",
        link="logit",control=list(iter.max=300,eval.max=400))

*** Fractional logit regression model ***
*** Estimator: GMMxv

               Estimate Std. Error t value Pr(>|t|)
INTERCEPT     -0.930133   1.134409  -0.820   0.412
Growth         0.009246   0.005557   1.664   0.096 *
Size           0.038834   0.090336   0.430   0.667
Profitability -5.774066   3.441828  -1.678   0.093 *
Tangibility    1.103188   0.481986   2.289   0.022 **
vhat           1.811518   3.791412   0.478   0.633

Reduced form:
               Estimate Std. Error t value Pr(>|t|)
Z_INTERCEPT    1.003007   0.042115  23.816   0.000 ***
Z_Growth       0.000859   0.000237   3.630   0.000 ***
Z_Size         0.004450   0.003725   1.195   0.232
Z_Tangibility -0.030065   0.021838  -1.377   0.169
```

```
Z_ProfitIVa     0.177334   0.033847   5.239   0.000 ***
Z_ProfitIVb     0.279039   0.038986   7.157   0.000 ***

Note: robust standard errors

Number of observations: 379
```

The *xv*-type estimators require heavier assumptions than the *z*-type estimators, namely the correctness of both Eqs. (23) and (24), but have the attractive feature of providing a simple test for endogeneity. Indeed, such a test may be implemented as a test for the significance of the parameter ρ associated to $\hat{v}_i$ (denoted by `vhat` in the output). In this example, we fail to reject the null hypothesis of exogeneity of *Profitability*.

Finally, there is a seventh estimator that may be obtained using `frmhet`: QMLxv. This estimator, proposed by Wooldridge (2005), is based on the original formulation of conditional mean models described in Section 2.1, but uses a control function approach similar to that described above to deal with endogeneity issues. However, unlike GMMxv and LINxv, it does not allow for other sources of heterogeneity. To implement the QMLxv estimator, simply use the option `type="QMLxv"` in `frmhet`.

3.4 Smearing estimation of partial effects

So far, the discussion concerning EFRM and LFRM has been focused on the estimation of the parameters that appear in the structural model (13). As in the standard case, we may be also interested in computing partial effects in order to measure how unitary changes in x_{ij} affect the value of y_i. Unfortunately, calculating partial effects in the framework of transformation regression models is not trivial. On the one hand, differentiating the base model (13) in order to x_{ij} would give a simple expression for a partial effect conditional on both observables and unobservables. However, since there are no interesting values to plug-in for u_i, what we need are partial effects conditional only on observables, $E(y_i|x_i)$. On the other hand, computing partial effects directly from (16) or any of the other transformation regression models considered above allows the analysis to be conditional only on observables, since the unobservables are additively separable. However, what we get directly is the the effect of a unitary change in x_{ij} on $E[H_1(y_i)|x_i]$, instead of the desirable quantity $E(y_i|x_i)$.

Duan (1983) suggested the so-called smearing technique that allows to estimate partial effects on $E(y_i|x_i)$ after estimating a transformation regression model. His technique was adapted by Ramalho and Ramalho (2017) for EFRM and LFRM and requires that the dependence between observables and unobservables, if any, is restricted to the conditional mean. Following these authors, from (14) we may compute partial effects conditional only on observables using the following expression:

$$\frac{\partial E(y_i|x_i)}{\partial x_{ij}} = \theta_j \int_U g(x_i\theta + u_i) f(u_i|x_i) du_i. \tag{27}$$

This expression still depends on u_i. To remove this dependency, Ramalho and Ramalho (2017) used a two-step procedure. First, θ is estimated by any of the GMM/LIN estimators described above and the residuals $\hat{u}_i$ for all sampling units are obtained, where $\hat{u}_i = \frac{H_1(y_i)}{\exp(x_i\hat{\theta})} - 1$ (GMMx/GMMz), $\hat{u}_i = \frac{H_1(y_i)}{\exp(x_i\hat{\theta} + \hat{v}_i\hat{\rho})} - 1$ (GMMxv), $\hat{u}_i = H_1(y_i) - x_i\hat{\theta}$ (LINx/LINz) or $\hat{u}_i = H_1(y_i) - (x_i\hat{\theta} + \hat{v}_i\hat{\rho})$ (LINxv). Then, for individual i, the partial effects are estimated using:

$$\left[\widehat{\frac{\partial E(y_i|x_i)}{\partial x_{ij}}}\right] = \frac{1}{N}\hat{\theta}_j \sum_{m=1}^{N} g\left(x_i\hat{\theta} + \hat{u}_{im}\right), \tag{28}$$

where the unknown error distribution is estimated by the empirical distribution of the GMM or LIN residuals calculated in step 1. Note that this calculation has to be made independently for each sampling unit.

It is possible to calculate partial effects using the command `frmhet.pe` contained in package `frmhet`. This command works in a similar way to the command `frm.pe` described in Section 2.3, but has an additional option called `smearing`. By default, `smearing=T` and formula (28) is used. For comparison purposes, we may define `smearing=F`, in which case the naive estimator $\left[\widehat{\frac{\partial E(y_i|x_i)}{\partial x_{ij}}}\right] = \hat{\theta}_j g\left(x_i\hat{\theta}\right)$ is calculated. For example:

```
> res <- frmhet(y=Y,x=X,type="GMMx",link="logit",table=FALSE)
> frmhet.pe(res,smearing=TRUE,APE=TRUE,CPE=FALSE)

*** Average partial effects (conditional only on observables,
        based on the smearing estimator)

Fractional logit regression model
Estimator: GMMx


              Estimate Std. Error t value Pr(>|t|)
Growth          0.0010     0.0007   1.541   0.123
Size            0.0031     0.0128   0.242   0.809
Profitability  -0.5265     0.1658  -3.176   0.001 ***
Tangibility     0.1418     0.0548   2.587   0.010 ***

> frmhet.pe(res,smearing=FALSE,APE=TRUE,CPE=FALSE)

*** Average partial effects (conditional on both observables
        and unobservables, with error term = 0)
```

```
Fractional logit regression model
Estimator: GMMx


             Estimate Std. Error t value Pr(>|t|)
Growth         0.0019     0.0012   1.548   0.122
Size           0.0057     0.0236   0.242   0.809
Profitability -0.9688     0.2995  -3.235   0.001 ***
Tangibility    0.2608     0.0997   2.616   0.009 ***
```

Notice how the estimated partial effects are substantially different in each case, confirming the importance of using a smearing estimator in this context.

4 Panel data estimators

All estimators discussed so far were developed assuming the availability of cross-sectional data. Specific estimators for panel data fractional regression models may also be constructed. In this section, we review the main panel data estimators for fractional responses and show how to obtain them using the `frmpd` package and the full dataset:

```
library(frmpd)
Y <- Leverage
X <- cbind(Growth, Size, Profitability, Tangibility)
```

4.1 Framework

In a panel data setting, it is common to include time-invariant unobserved heterogeneity in the regression model. Let α_i denote those individual effects. As in the previous section, it is not straightforward to work with α_i in the framework of Eq. (1), but it is possible to use a similar methodology that allows easier handling of not only individual effects, but also of time-varying unobservables (φ_{it}). In particular, Ramalho et al. (2018) proposed using the following structural model:

$$y_{it} = G(x_{it}\theta + \alpha_i + \varphi_{it}), \tag{29}$$

$t = 1, \ldots T$.

The $G(\cdot)$ function in (29) has to have exactly the same properties of that considered in Eq. (13), which implies that the estimators described in this section are also only available for logit and cloglog models. Using similar arguments to those used to derive Eqs. (16) and (18), we now have the following transformation regression models:

$$H_1(y_{it}) = \exp(x_{it}\theta + \alpha_i + \varphi_{it}) \tag{30}$$

and, depending on the assumptions made on α_i,

$$\frac{H_1(y_{it})}{\exp(x_{it}\theta)} - 1 = \exp(\alpha_i + \varphi_{it}) - 1 \tag{31}$$

or

$$\frac{H_1(y_{it})}{\exp(\alpha_i + x_{it}\theta)} - 1 = \exp(\varphi_{it}) - 1. \tag{32}$$

Based on these equations, Ramalho et al. (2018) proposed six alternative panel data GMM estimators, which differ on the assumptions about the correlation between α_i, φ_{it} and the covariates. Indeed, some estimators allow for α_i and x_{it} to be correlated, while others require them to be not correlated. Regarding φ_{it}, we may have the following cases:

- strict exogeneity: $E[\exp(\varphi_{it})|\alpha_i, x_{i1}, \ldots, x_{iT}] = 1$
- weak exogeneity: $E[\exp(\varphi_{it})|\alpha_i, x_{i1}, \ldots, x_{it}] = 1$
- contemporaneous exogeneity: $E[\exp(\varphi_{it})|\alpha, x_{it}] = 1$
- contemporaneous endogeneity: $E[\exp(\varphi_{it})|\alpha, x_{it}] \neq 1$

In all cases, α_i and φ_{it} are assumed to be not correlated and the latter not to be serially correlated. The estimators next discussed may be used with both balanced and unbalanced data.[e]

4.2 Pooled random and fixed effects estimators

There are two simple pooled estimators that may be used in this framework. The first is a 'pooled random effects' estimator (GMMpre). Assuming that x_{it} and α_i are independently distributed, we may treat $\alpha_i + \varphi_{it}$ as a single error term and use Eq. (31) as basis for consistent estimation of θ. Therefore, assuming contemporaneous exogeneity for x_{it}, the GMMpre estimator is simply the pooled GMMx estimator defined in (20). In case of endogenous x_{it}, GMMpre corresponds to the pooled GMMz estimator defined by (21).

The second estimator is a "pooled fixed effects" (GMMpfe) estimator, allowing x_{it} and α_i to be correlated and interpreting α_i as a vector of individual-specific intercepts to be estimated simultaneously with θ. From Eq. (32), estimates for θ and α_i may be obtained by GMM estimation based on:

$$E\left\{(x_{it}, \alpha_i)' \left[\frac{H_1(y_i)}{\exp(x_{it}\theta + \alpha_i)} - 1\right]\right\} = 0. \tag{33}$$

[e]Note that the `frmpd` package does not allow for NA values. If you have missing data for some variables in some years for some individuals, your database should include only the individuals/years for which all variables were observed. Lines with missing data should be removed before using `frmpd`.

As shown by Dhaene and Jochmans (2017), there is no incidental parameters problem in this case. Moreover, see Ramalho et al. (2018), when there are no boundary values estimation may be based on the alternative set of moment conditions

$$E\left\{x_{it}'\left[\frac{H_1(y_{it})}{\exp(x_{it}\theta)}\overline{\left[\frac{H_1(y_i)}{\exp(x_{it}\theta)}\right]}^{-1}-1\right]\right\}=0\,, \tag{34}$$

where $\overline{\lambda}$ is the mean over t of λ_{it} and the individual effects do not need to be estimated. Consistency of the GMMpfe estimator requires a strict exogeneity assumption for x_{it}. It is also possible to deal with endogenous regressors replacing x_{it} by z_{it} in (33) or (34). In this case, we may use lags and/or leads of the regressors as instruments or use external instruments.

Under exogeneity, the GMMpre estimator may be obtained using the `frmpd` package as follows:

```
> frmpd(id=Ident,time=Year,y=Y,x=X,x.exogenous=TRUE,type=
        "GMMpre",link="logit")

*** Fractional logit regression model ***
*** Estimator: GMMpre
*** Exogeneity: TRUE
*** Use first lag of instruments: FALSE

               Estimate Std. Error t value Pr(>|t|)
INTERCEPT     -1.085236   0.892587  -1.216   0.224
Growth         0.002032   0.003004   0.676   0.499
Size           0.063387   0.083947   0.755   0.450
Profitability -4.293260   1.163781  -3.689   0.000 ***
Tangibility    0.712524   0.331692   2.148   0.032 **

Note: cluster standard errors

Number of observations (initial): 1843
Number of observations (for estimation): 1843
Number of cross-sectional units (initial): 620
Number of cross-sectional units (for estimation): 620
Average number of time periods per cross-sectional unit
        (initial): 2.972581
Average number of time periods per cross-sectional unit
        (for estimation): 2.972581
```

By default, the variance of the parameter estimates is estimated in a cluster–robust way, but a simpler robust estimator of the type considered in the

previous sections is also available by setting `var.type="robust"`. Both variance estimators may be bootstrapped using the option `bootstrap=TRUE`. For example, using a cluster–robust estimator based on 100 bootstrap replications the following results are obtained (`set.seed(300)` ensures that you obtain the results below):[f]

```
> set.seed(300)
> frmpd(id=Ident,time=Year,y=Y,x=X,x.exogenous=TRUE,type=
        "GMMpre",link="logit",bootstrap=TRUE,B=100,control=
        list(iter.max=300,eval.max=400))
1111111111111111111111111111111111111111111111111111
1111111111111111111111111111111111111111111111111111

*** Fractional logit regression model ***
*** Estimator: GMMpre
*** Exogeneity: TRUE
*** Use first lag of instruments: FALSE

               Estimate Std. Error t value Pr(>|t|)
INTERCEPT     -1.085236   0.901134  -1.204   0.228
Growth         0.002032   0.003459   0.587   0.557
Size           0.063387   0.083618   0.758   0.448
Profitability -4.293260   1.290966  -3.326   0.001 ***
Tangibility    0.712524   0.344081   2.071   0.038 **

Note: bootstrap standard errors

Number of observations (initial): 1843
Number of observations (for estimation): 1843
Number of cross-sectional units (initial): 620
Number of cross-sectional units (for estimation): 620
Average number of time periods per cross-sectional unit
        (initial): 2.972581
Average number of time periods per cross-sectional unit
        (for estimation): 2.972581
```

With contemporaneous endogeneity, we need to change the option `x.exogenous` to `FALSE` and provide the name of the matrix containing the instruments. In the following example, we consider again *ProfitIVa* and *ProfitIVb* as instruments for *Profitability*:

[f]Whenever a bootstrap replication is concluded, a number is reported: 1 means that convergence in estimation was achieved; 0 means the opposite, meaning that at the end less than B bootstrap replications were taken into account in the estimation of the parameter variance.

```
> Z <- cbind(Growth, Size, Tangibility, ProfitIVa, ProfitIVb)
> frmpd(id=Ident,time=Year,y=Y,x=X,z=Z,x.exogenous=FALSE,type=
        "GMMpre",link="logit")

*** Fractional logit regression model ***
*** Estimator: GMMpre
*** Exogeneity: FALSE
*** Use first lag of instruments: FALSE

               Estimate Std. Error t value Pr(>|t|)
INTERCEPT     -1.624264   0.923614  -1.759   0.079 *
Growth         0.005763   0.003608   1.597   0.110
Size           0.124290   0.094188   1.320   0.187
Profitability -7.193148   2.235043  -3.218   0.001 ***
Tangibility    0.774943   0.337719   2.295   0.022 **

Note: cluster standard errors

Number of observations (initial): 1843
Number of observations (for estimation): 1843
Number of cross-sectional units (initial): 620
Number of cross-sectional units (for estimation): 620
Average number of time periods per cross-sectional unit
        (initial): 2.972581
Average number of time periods per cross-sectional unit
        (for estimation): 2.972581

J test of overidentifying moment conditions: 0.1725407
        (p-value: 0.6778636)
```

To obtain the GMMpfe estimator, it is only necessary to change the option `type` to `"GMMpfe"`. Note that if the sample contains the value zero, the optimization process will tend to be very time-consuming for large N, since in such a case estimation will be based on Eq. (33) and there are $N + k$ parameters to be estimated.

In all cases, a set of time dummies may be automatically added to the model by defining the option `tdummies=TRUE`.

4.3 Fixed effects estimators based on quasi- and mean-differences

Because model (30) is basically the expression of an exponential regression model, the quasi- and mean-difference transformations commonly applied to exponential models to eliminate fixed effects may also be applied to Eq. (30). In particular, Ramalho et al. (2018) adapted to the fractional framework three estimators that use alternative transformations.

The first estimator is based on the following set of moment conditions

$$\left\{x'_{i,t-1}\left[\frac{H_1(y_{it})}{\exp(x_{it}\theta)}-\frac{H_1(y_{i,t-1})}{\exp(x_{i,t-1}\theta)}\right]\right\}=0 \tag{35}$$

and assumes weak exogeneity. Ramalho et al. (2018) denoted this estimator by GMMww, because in the exponential regression case it was originally proposed by Wooldridge (1997) and Windmeijer (2000). This is the most flexible estimator, because x_{it} may contain endogenous covariates and lagged values of the response variable. As instruments, lagged values of $x_{i,t-1}$ or external instruments ($z_{i,t-1}$) may be used.

The second estimator, based on Chamberlain (1992) proposal for the exponential regression model, is denoted by GMMc and uses:

$$\left\{x'_{i,t-1}\left[\frac{\exp(x_{i,t-1}\theta)}{\exp(x_{it}\theta)}H_1(y_{it})-H_1(y_{i,t-1})\right]\right\}=0. \tag{36}$$

Again, the weak exogeneity assumption is enough for consistent estimation of θ and x_{it} may contain lagged dependent variables. However, this estimator cannot be used when some explanatory variable is endogenous.

Finally, the third estimator (GMMbgw) is based on the mean differenced transformation used by Blundell et al. (2002) for exponential models. The moment conditions derived by Ramalho et al. (2018) for fractional regression models are given by:

$$\left\{x'_{it}\left[H_1(y_{it})-\frac{\overline{H_1(y_i)}}{\overline{\exp(x_i\theta)}}\exp(x_{it}\theta)\right]\right\}=0. \tag{37}$$

This estimator requires strict exogeneity and cannot be applied with endogenous covariates and lagged dependent variables.

All three estimators are easily obtained with the `frmpd` package. The procedure is similar to that illustrated before for the GMMpre estimator, the only difference being the option defined for `type`: `GMMww`, `GMMc`, or `GMMbgw`. Note that in all cases you should define the options `y`, `x`, and, if needed, `z` as, respectively, y_{it}, x_{it}, and z_{it}: `GMMww`, `GMMc`, and `GMMbgw` will apply the appropriate lags whenever appropriate. Continuing the preceding example, the GMMww estimator could be obtained as follows:

```
> frmpd(id=Ident,time=Year,y=Y,x=X,z=Z,x.exogenous=FALSE,type=
        "GMMww",link="logit")

*** Fractional logit regression model ***
*** Estimator: GMMww
*** Exogeneity: FALSE
*** Use first lag of instruments: TRUE
```

```
                Estimate Std. Error t value Pr(>|t|)
Growth          0.003760   0.003040   1.237   0.216
Size            0.015257   0.254081   0.060   0.952
Profitability -10.847346   3.194634  -3.395   0.001 ***
Tangibility     0.497086   0.856810   0.580   0.562

Note: cluster standard errors

Number of observations (initial): 1843
Number of observations (for estimation): 1157
Number of cross-sectional units (initial): 620
Number of cross-sectional units (for estimation): 441
Average number of time periods per cross-sectional unit
        (initial): 2.972581
Average number of time periods per cross-sectional unit
        (for estimation): 2.623583

J test of overidentifying moment conditions: 0.6223896
        (p-value: 0.4301607)
```

In this illustration, the first lag of *ProfitIVa* and *ProfitIVb* (included in the matrix Z) were used as instruments for the first lag of *Profitability* (included in the matrix X).

4.4 Correlated random effects estimators

Finally, a "correlated random effects" estimator (GMMcre) similar in spirit to the QML estimators proposed by Papke and Wooldridge (2008) and Wooldridge (2010) may be constructed. These estimators use flexible functional forms for representing the relationship between α_i and x_{it}. For example, assuming balanced panel data, α_i may be modeled as a linear function of all exogenous variables:

$$\alpha_i = \psi_0 + \overline{z}_i \psi_1 + \eta_i, \tag{38}$$

where $\overline{z}_i \equiv T^{-1} \sum_{t=1}^{T} z_{it}$, $\psi = (\psi_0, \psi_1)$ is a vector of parameters to be estimated and η_i is a disturbance term that is uncorrelated with all the other variables and error terms. Plugging (38) into (30), it follows that[g]

$$\frac{H_1(y_{it})}{\exp(x_{it}\theta + \psi_0 + \overline{z}_i \psi_1)} - 1 = \exp(\eta_i + \varphi_{it}) - 1 \tag{39}$$

and, therefore, GMMcre is based on the following moment conditions:

[g]Without loss of generality, since (38) contains a constant term, it is assumed that $E[\exp(\eta_i)] = 1$.

$$E\left\{(x_{it},1,\overline{z}_i)'\left[\frac{H_1(y_i)}{\exp(x_{it}\theta+\psi_0+\overline{z}_i\psi_1)}-1\right]\right\}=0. \tag{40}$$

A similar analysis may be carried out in the case of unbalanced data, but based on

$$\alpha_i=\sum_{r=2}^{T}\delta_{T_i,r}\psi_{0r}+\sum_{r=2}^{T}\delta_{T_i,r}\overline{z}_i\psi_{1r}+\eta_i, \tag{41}$$

where T_i is the number of observations available for individual i and $\delta_{T_i,r}$ is a dummy variable which is equal to unity if $T_i = r$ and data exist on the full set of variables. See Wooldridge (2010) for details and alternative expressions for α_i.

An example of the application of the GMMcre estimator is the following:

```
> frmpd(id=Ident,time=Year,y=Y,x=X,x.exogenous=TRUE,type=
        "GMMcre",link="logit",control=list(iter.max=1000,
        eval.max=2000))

*** Fractional logit regression model ***
*** Estimator: GMMcre
*** Exogeneity: TRUE
*** Use first lag of instruments: FALSE

                      Estimate Std. Error t value Pr(>|t|)
Growth                0.001147   0.001602   0.716   0.474
Size                 -0.398295   0.281773  -1.414   0.157
Profitability        -1.499670   0.919217  -1.631   0.103
Tangibility          -0.458599   0.665933  -0.689   0.491
INTERCEPT_2           0.976489   1.712421   0.570   0.569
INTERCEPT_3           0.459833   1.896913   0.242   0.808
INTERCEPT_4          -2.202255   1.216480  -1.810   0.070 *
INTERCEPT_5          -2.472313   1.653173  -1.495   0.135
Growth_mean_2         0.005858   0.010211   0.574   0.566
Growth_mean_3         0.005029   0.011575   0.435   0.664
Growth_mean_4         0.013685   0.011367   1.204   0.229
Growth_mean_5        -0.005388   0.020946  -0.257   0.797
Size_mean_2           0.257174   0.321049   0.801   0.423
Size_mean_3           0.317949   0.343350   0.926   0.354
Size_mean_4           0.472391   0.301710   1.566   0.117
Size_mean_5           0.609260   0.261945   2.326   0.020 **
Profitability_mean_2 -7.343594   2.511484  -2.924   0.003 ***
Profitability_mean_3 -4.417342   2.403205  -1.838   0.066 *
Profitability_mean_4  1.293648   2.125925   0.609   0.543
Profitability_mean_5 -8.953168   2.665799  -3.359   0.001 ***
Tangibility_mean_2    2.242842   0.955192   2.348   0.019 **
Tangibility_mean_3    1.077881   1.284365   0.839   0.401
Tangibility_mean_4    3.051096   0.969771   3.146   0.002 ***
Tangibility_mean_5    1.295880   1.119023   1.158   0.247
```

```
Note: cluster standard errors

Number of observations (initial): 1843
Number of observations (for estimation): 1681
Number of cross-sectional units (initial): 620
Number of cross-sectional units (for estimation): 458
Average number of time periods per cross-sectional unit
        (initial): 2.972581
Average number of time periods per cross-sectional unit
        (for estimation): 3.670306
```

Papke and Wooldridge (2008) correlated random effects QML estimator, which requires a probit specification, is also available in the `frmpd` package by defining the option `type=QMLcre` (and, naturally, `link="probit"`).

5 Future developments

The three packages described in this chapter allow practitioners to apply a range of different methodologies to the analysis of responses variables that represent a *single* proportion. However, sometimes, the joint behavior of a *multivariate* fractional variable is of interest. Models for the multivariate case have been analyzed by Mullahy (2015) and were further developed by Murteira and Ramalho (2016). Some of the proposed models are relatively complex and, to the best of our knowledge, there is no R package available for estimating them. The developing of such an R package is in my research agenda for the near future.

Acknowledgments

Financial support from Fundacao para a Ciencia e a Tecnologia (grant UID/GES/00315/2013) is gratefully acknowledged. The author also thanks Esmeralda A. Ramalho and Jose Dias Curto for their valuable comments and suggestions.

References

Blundell, R., Griffith, R., Windmeijer, F.A.G., 2002. Individual effects and dynamics in count data models. J. Economet. 108, 113–131.

Chamberlain, G., 1992. Comment: sequential moment restrictions in panel data. J. Bus. Econ. Stat. 10 (1), 20–26.

Davidson, R., MacKinnon, J.G., 1981. Several tests for model specification in the presence of alternative hypotheses. Econometrica 49 (3), 781–793.

Dhaene, G., Jochmans, K., 2017. Profile-score adjustments for incidental-parameter problems. Tech. rep.

Duan, N., 1983. Smearing estimate: a nonparametric retransformation method. J. Am. Stat. Assoc. 78 (383), 605–610.

Hansen, L.P., 1982. Large sample properties of generalised method of moments estimators. Econometrica 50 (4), 1029–1054.
Heckman, J.J., 2000. Causal parameters and policy analysis in economics: a twentieth century retrospective. Q. J. Econ. 115 (1), 45–97.
Mullahy, J., 2015. Regression estimation of econometric share models. J. Economet. Methods 4 (1), 71–100.
Murteira, J.M.R., Ramalho, J.J.S., 2016. Regression analysis of multivariate fractional data. Economet. Rev. 35 (4), 515–552.
Ospina, R., Ferrari, S.L.P., 2012. A general class of zero-or-one inflated beta regression models. Comput. Stat. Data Anal. 56 (6), 1609–1623.
Papke, L., Wooldridge, J., 1996. Econometric methods for fractional response variables with an application to 401(k) plan participation rates. J. Appl. Economet. 11 (6), 619–632.
Papke, L., Wooldridge, J., 2008. Panel data methods for fractional response variables with an application to test pass rates. J. Economet. 145 (1-2), 121–133.
Ramalho, E.A., Ramalho, J.J.S., 2017. Moment-based estimation of nonlinear regressionmodels with boundary outcomes and endogeneity, with applications to nonnegative and fractional responses. Economet. Rev. 36 (4), 397–420.
Ramalho, E.A., Ramalho, J.J.S., Henriques, P.D., 2010. Fractional regression models for second stage DEA efficiency analyses. J. Prod. Anal. 34 (3), 239–255.
Ramalho, E.A., Ramalho, J.J.S., Murteira, J.M.R., 2011. Alternative estimating and testing empirical strategies for fractional regression models. J. Econ. Surv. 25 (1), 19–68.
Ramalho, E.A., Ramalho, J.J.S., Murteira, J.M.R., 2014. A generalized goodness-of-functional form test for binary and fractional regression models. Manch. Sch. 82 (4), 488–507.
Ramalho, E.A., Ramalho, J.J.S., Coelho, L.M.S., 2018. Exponential regression of fractional-response fixed-effects models with an application to firm capital structure. J. Economet. Methods 7 (1), article 20150019.
Santos Silva, J.M.C., Tenreyro, S., 2006. The log of gravity. Rev. Econ. Stat. 88 (4), 641–658.
Wagner, J., 2003. Unobserved firm heterogeneity and the size-exports nexus: evidence from German panel data. Rev. World Econ. 139 (1), 161–172.
Windmeijer, F., 2000. Moment conditions for fixed effects count data models with endogenous regressors. Econ. Lett. 68 (1), 21–24.
Windmeijer, F.A.G., Santos Silva, J.M.C., 1997. Endogeneity in count data models: an application to demand for health care. J. Appl. Economet. 12, 281–294.
Wooldridge, J.M., 1997. Multiplicative panel data models without the strict exogeneity assumption. Economet. Theor. 13, 667–678.
Wooldridge, J.M., 2005. Unbserved heterogeneity and estimation of average partial effects. In: Andrews, D.W.K., Stock, J.H. (Eds.), Identification and Inference for Econometric Models. Cambridge University Press, New York, pp. 27–55. (Chapter 3).
Wooldridge, J.M., 2010. Correlated random effects models with unbalanced panels. Tech. rep.
Wooldridge, J.M., 2015. Control function methods in applied econometrics. J. Hum. Resour. 50, 420–445.

This page intentionally left blank

Chapter 9

Quantitative game theory applied to economic problems

Sebastián Cano-Berlanga*,[a], José-Manuel Giménez-Gómez[b] and Cori Vilella[b]

[a]Universitat Autònoma de Barcelona and CREIP, Catalonia, Spain

[b]Universitat Rovira i Virgili and CREIP, Tarragona, Spain

*Corresponding author: e-mail: sebastian.cano@uab.cat

Abstract

The main purpose of the present chapter is to introduce the reader to game theory through R. Specifically, it focuses on cooperative games with transferable utility and it introduces well-known punctual solutions, the voting power index and the claims problems. Furthermore, we introduce a modern application of cooperative game theory to the marketing field, where we develop a framework to distribute revenues among Internet selling channels. For the sake of comprehensiveness, after theoretical explanation, the reader may find the R code to execute the examples.

Keywords: Cooperative game, Shapley value, Nucleolus, Claims problem, Bankruptcy, Attribution model, R package game theory

1 Introduction

There exists a large number of social and economic situations where the agents are strategic dependent, i.e., each agent's outcome is influenced by the other agents' decision. The economics field that studies such situations is game theory, whose influence in economic modeling is extraordinary. Specifically, the current chapter focuses on the situations where cooperation among agents is necessary and mutually beneficial, such as the formation of a cartel among companies, or the financial support through *crowdfunding*. By doing so, we use the cooperative game theory, which not only models the cooperation among agents (in terms of gains and costs) but also provides solutions to determine the way to share the benefit obtained from the cooperation among agents. These situations, where collaboration and conflict of interest arise naturally, are called *games*, and the agents, *players*, who may be individuals, nations, political parties, associations, companies, etc.

Handbook of Statistics, Vol. 41. https://doi.org/10.1016/bs.host.2018.11.003

It is noteworthy that the importance of the role to be developed by game theory is to provide, from a quantitative rather than a qualitative point of view, the objective tools that promote the cooperation and solve potential conflict. Specifically, the analysis of such situations from a formal and an axiomatic point of view becomes essential as the complexity of the situation and the assets to be distributed increases.

The current chapter presents some applications of game theory to economic problems, following Cano-Berlanga et al. (2017b). First, Section 2 provides some basics about cooperative game theory and the main solutions, through the computation of some real examples. Second, Section 3 applies the game theory analysis to a marketing problem. Specifically, it solves the case of distributing sales revenues among all the channels that induce the purchasing process. Finally, Section 4 analyzes the conflicting claims problems through the most used solutions and gives some real examples. The examples of this chapter are driven by the R package *Gametheory*.

2 Cooperative game theory

Game theory is the discipline that studies how agents make strategic decisions. It was initially developed in economics to understand a large collection of economic behaviors, including firms, markets, and consumers. Specifically, a game is the mathematical formalization of such conflicts, originated by Antoine Cournot (1801–1877) in 1838 with his solution of the Cournot duopoly.

Modern game theory begins with the publication of the book "Theory of Games and Economic Behavior" written by von Neumann and Morgenstern (1944), who considered cooperative games with several players. Indeed, according to Maschler (1979) after this initial point, game theory was developed extensively in the 1950s by numerous authors. Later on, the application field of game theory was not unique to economics and we may find game theory in social network formation, behavioral economics, ethical behavior and biology, among others.

Game theory is divided into two branches, called the noncooperative and the cooperative. These two branches differ in how they formalize interdependence among the players. In noncooperative game theory, a game is a detailed model of all the moves available to the players. In contrast, cooperative game theory abstracts away from this level of detail and describes only the outcomes that result when the players come together in different combinations.

Formally, in a cooperative game we have a finite set of players $N = \{1,2,\ldots,n\}$, which can be grouped into 2^N subsets of N, called *coalitions*. Coalitions are represented in capital letters and the corresponding lower case letter will represent the number of players in the coalition; so, the coalition S has s players. In addition, the coalition without players is called the empty coalition and it is represented by $\varnothing$. The game assigns to each coalition a real

value $v(S)$, the *worth of* S. Usually, this number can be interpreted as a measure of what the coalition can achieve on its own. Formally,

Definition 1. A cooperative game, or simply a *game*, is a pair (N, v), where N is a finite set of players and v is a function, the characteristic function

$$v : 2^N \to \mathbb{R}$$

where $v(S) \in \mathbb{R}$ for all $S \subseteq 2^N$, and $v(\varnothing) = 0$. Let $\mathcal{G}^N$ denote the class of all cooperative games with player set N.

During many years of study and developing of cooperative game theory, most of the results obtained have been based on the fulfillment of some properties by the characteristic function. We present below three of the properties considered as basic requirements.

Monotonicity: If the number of players in the coalition increases, the benefits should not decrease.
Superadditivity: The union of coalitions with no common players is beneficial
Convexity: The higher the coalition, the higher each player's marginal contribution.

Cooperative game theory centers its interest on particular sets of strategies known as "solution concepts" or "equilibria" based on what is required by norms of (ideal) rationality. Among the several types of games, this chapter focuses on *cooperative games with transferable utility*.

Specifically, a coalitional game with transferable utility involving a set of agents, hereinafter a cooperative game, can be described as a function that associates with each group of agents (or coalition), a real number which is denoted as the worth of the coalition. If a coalition forms, then it can divide its worth in any possible way among its members. This is possible if money is available as a medium of exchange, and if each player's utility for money is linear (see Aumann, 1960).

Furthermore, A solution on cooperative games is a correspondence that associates with each game a nonempty set of payoff vectors in $\mathbb{R}^N$ whose coordinates add up to $v(N)$. One of the most important solutions is the core and it selects for each game all the payoff vectors such that no coalition could simultaneously provide a higher payoff to each of its members. The Core is a multivalued solution but the ones we present here, the Shapley value (Shapley, 1953) and the *nucleolus* (Schmeidler, 1969) are point solutions.

2.1 The core

The concept of the *core*, one of the most important solution concepts in cooperative game theory, is the one that selects for each game all the payoff vectors

such that no coalition could simultaneously provide a higher payoff to each of its members. It is based on the idea of imputations. Formally, for each game $(N,v) \in \mathcal{G}^N$ the *imputations set* is $I(v) = \{x \in \mathbb{R}^n \mid x_i \geq v(\{i\}), \sum_{i \in N} x_i = v(N)\}$ the set of allocations that are individually rational and efficient.

Formally, the core of a game, which was formulated by Edgeworth (1881) and first defined into game theory by Gillies (1953), is the set of those imputations where each coalition gets at least its worth, that is $C(v) := \{x \in I(v) \mid \sum_{i \in S} \geq v(S) \text{ for all } S \subseteq N\}$.

We may say that the allocations in the core are stable allocations since no coalition S will have incentives to deviate in order to obtain a strictly better payoff than what a core element assigns.

The core of a cooperative game is a polyhedric set, closed and bounded in $\mathbb{R}^n$. In particular it is a convex polyhedron, i.e., it may be an empty set or if it is nonempty it may have either a single point or infinite elements.

Finally, it is noteworthy that the core is a multivalued solution, but the ones presented next are single-valued.

2.2 The Shapley value

The *Shapley value* (Shapley, 1953) is based on the following idea: consider players arriving one at each time. Calculate for each player the amount by which his arrival increases the worth of the coalition consisting of all the players who arrived before him. We call this difference the marginal contribution of the player to the coalition. Therefore, if he is the first to arrive, his contribution is simply his own worth. Suppose that all orders of arrivals are equally like, then his payoff is the average of his contributions for all possible arrival orders. Formally, given a game $(N,v) \in \mathcal{G}^N$, for each $i \in N$ and each $S \subset N$, we call the *marginal contribution of an agent i to the coalition S*, $v(S \cup \{i\}) - v(S)$.

According to this solution the worth of the grand coalition is distributed assuming that all orders of agents' arrivals to the grand coalition are equally probable and in each order, each agent gets his marginal contribution from the coalition that he joins. Therefore, for each $(N,v) \in \mathcal{G}^N$, the *Shapley value*, $\phi(v)$, associates to each $i \in N$, the amount $\phi_i(v) = \sum_{S \subseteq N \setminus \{i\}} [(s!(n-s-1)!)/n!]$ $(v(S \cup \{i\}) - v(S))$.

Usually, assigning the marginal contribution to each player is not efficient, and, therefore, it is not a solution. In order to avoid the problem of inefficiency, we may consider that players join the coalition following a certain ordering, and, then, consider each players' marginal contribution. Note that this distribution, called the vector of marginal contributions, is efficient, but depends on an arbitrary ordering of the players. This distribution is called the marginal contribution vector associated with θ and we denote it by $m^\theta(v)$.

Definition 2. Let $m^{\theta}(v) \in \mathcal{R}^n$ be the *vector of marginal contributions* associated to an ordering $\theta = (i_1, \ldots, i_n)$, where, for each player $i \in N$,

$$\begin{aligned} m^{\theta}_{i_1}(v) &:= v(i_1), \\ m^{\theta}_{i_2}(v) &:= v(i_1, i_2) - v(i_1), \\ &\cdots \\ m^{\theta}_{i_n}(v) &:= v(i_1, \ldots i_n) - v(i_1, \ldots i_{n-1}). \end{aligned}$$

Obviously, up to this point, the ordering determines the marginal contributions, so the revenues obtained. In order to solve this arbitrariness of the players, Shapley (1953) proposes a distribution that considers all possible orderings. Specifically, he assumes that each ordering has the same probability of being considered. Therefore, he considers the average of all the marginal contributions according to all possible orderings (see Table 1).

Definition 3. Let (N, v) be a game, the *Shapley value* of this game, $\phi(v) = (\phi_1(v), \ldots \phi_n(v))$, for each player $i \in N$, is defined as,

$$\phi_i(v) := \frac{1}{n!} \sum_{i \in S_n} m^{\theta}_i(v)$$

where the summation is applied on the set S_n of all the orderings, and $m^{\theta}(v)$ is the associated vector of marginal contributions.

TABLE 1 Formal computation of the Shapley value for the three-players game

Arrival ordering, θ	$m^{\theta}_1(v)$	$m^{\theta}_2(v)$	$m^{\theta}_3(v)$
$\theta = (1, 2, 3)$	$v(1)$	$v(12) - v(1)$	$v(123) - v(12)$
$\theta = (1, 3, 2)$	$v(1)$	$v(123) - v(13)$	$v(13) - v(1)$
$\theta = (2, 1, 3)$	$v(12) - v(2)$	$v(2)$	$v(123) - v(12)$
$\theta = (2, 3, 1)$	$v(123) - v(23)$	$v(2)$	$v(23) - v(2)$
$\theta = (3, 1, 2)$	$v(13) - v(3)$	$v(123) - v(13)$	$v(3)$
$\theta = (3, 2, 1)$	$v(123) - v(23)$	$v(23) - v(3)$	$v(3)$
$\phi_i(v)$	$\frac{1}{3!}\sum m^{\theta}_1(v)$	$\frac{1}{3!}\sum m^{\theta}_2(v)$	$\frac{1}{3!}\sum m^{\theta}_3(v)$

The rows represent the players' arrival ordering (θ) and the columns their marginal contribution (m^{θ}_i).

Note that the Shapley value selects an efficient allocation, always exists and it is unique. That is, whatever the characteristics of the game is, we can always compute it, and the result is a univalued distribution. Furthermore, if the game is supperaditive, then the Shapley value is individual rational; and, if the game is convex, the Shapley value belongs to the set of solutions whose allocation cannot be improved by any group of players, known as the Core of the game (Gillies, 1953; Shapley, 1953). Consequently, we can argue that the allocation proposed by the Shapley value is somewhat stable, since no player or group of players could improve on it. Furthermore, it is noteworthy that the Shapley value considers the following two key features:

Marginal contribution: the Shapley value does not share the individual revenue according to the worth of the individual coalitions, it measures the individual contribution of each player to each coalition. Therefore, the players are awarded by their contribution in each of the possible cases.
Temporal sequence: the ordering in which the player joint to the coalition is a conflict issue. For avoiding it, the Shapley value computes each player's marginal contribution taking into the account all the possible orderings.

Finally, Shapley (1953) shows that this solution is the unique solution that jointly satisfies the following properties:

- Efficiency: The Shapley value distributes all gains or costs among players.
- Symmetry: If two players make equal contributions to the game, i.e., if they are substitutes, they should receive the same amount.
- Dummy player: If a player does not provide any additional benefit to the other players, he should not receive any additional payment. In terms of the game if the player's marginal contribution is equal to zero, then he must receive an allocation equal to his individual worth.
- Additivity: The player's allocation for a sum of two games is the sum of the player's allocations for each individual game.

In order to illustrate we first take the example proposed by Lemaire (1991) where three individuals can collaborate by investing in common funds. This particular game is defined as follows

$$
\begin{aligned}
v(1) &= 46125.0 \\
v(2) &= 17437.5 \\
v(3) &= 5812.5 \\
v(12) &= 69187.5 \\
v(13) &= 53812.5 \\
v(23) &= 30750.0 \\
v(123) &= 90000.0
\end{aligned}
$$

With this data we can compute the Shapley value. At this point we would like to highlight that the way of getting any solution is a two-step process. We always proceed following the same scheme:

Setup the game: this is done by using the command `DefineGame()`. The user needs to provide the number of players and coalitional values.
Solution choice: the user can obtain the desired solution by applying any TU-Game method over the defined game.

To start the process we introduce the worth of the coalitions with the command `DefineGame`. After that, *R* is ready to perform the Shapley value solution, which returns the following output:

```
> COALITIONS <- c(46125,17437.5,5812.5,69187.5,53812.5,
    30750,90000)
> LEMAIRE<-DefineGame(3,COALITIONS)
> summary(LEMAIRE)
```

```
Characteristic form of the game

Number of agents: 3

Coalition Value(s)

  Value
1   46125.0
2   17437.5
3    5812.5
12  69187.5
13  53812.5
23  30750.0
123 90000.0
```

```
> NAMES <- c("Investor 1","Investor 2","Investor 3")
> LEMAIRESHAPLEY <- ShapleyValue(LEMAIRE,NAMES)
> summary(LEMAIRESHAPLEY)
```

```
Shapley value for the given game

      Shapley value
Investor 1     51750
Investor 2     25875
Investor 3     12375
```

2.3 The nucleolus

As we have seen the Shapley value is an attractive single-valued solution with nice properties that takes into account the marginal contributions of the players, but we cannot ensure that in belongs to the core. Belonging to the core very important in cooperative games since the core selects the distributions preserving the cooperation. In contrast, the nucleolus will always select an imputation in the core if the core is nonempty. The underlying idea in the *nucleolus* (Schmeidler, 1969) consists in observing the dissatisfaction level of the coalitions on a proposed distribution of the worth of the grand coalition.

To introduce this solution, some additional notation is needed. For each $(N,v)\in\mathcal{G}^N$, $x\in\mathbb{R}^n$ and each coalition $S\subseteq N, e(x,S)=v(S)-\sum_{i\in S}x_i$ is the excess of coalition S with respect to x and represents a measure of dissatisfaction of such a coalition at x. This difference indicates how well or badly a given coalition is treated. Note that a payoff vector belongs to the core if all this differences are greater or equal than zero. The vector $e(x)=\{e(x,S)\}_{S\subseteq N}$ provides the excesses of all coalitions to x. Given $x\in\mathbb{R}^n, \theta(x)$ is the vector that results from x by permuting the coordinates in a decreasing order, $\theta_1(x)\geq\theta_2(x)\geq\cdots\geq\theta_n(x)$. Finally, $\leq_L$ stands for the lexicographic order, that is, given $x,y\in\mathbb{R}^n$, $x\leq_L y$ if there is $k\in N$ such that for all $j\leq k$, $x_j=y_j$ and $x_{k+1}\leq y_{k+1}$.

The nucleolus looks for an individually rational distribution of the worth of the grand coalition in which the maximum dissatisfaction is minimized. Formally, for each $(N,v)\in\mathcal{G}^N$, the *nucleolus* γ^{nu} is the vector $\gamma^{nu}(v)=x\in I(v)$ such that $\theta(e(x))\leq_L\theta(e(y))$ for all $y\in I(v)$. That is, it selects the element in the core, if this is nonempty, that lexicographically minimizes the vector of nonincreasing ordered excesses of coalitions. In order to compute this solution we consider the following linear programming model, which looks for an imputation that minimizes the maximum excess ε among all coalitions. Formally,

$$\begin{aligned}&\min_x\ \varepsilon\\&\text{subject to } v(S)-\sum\nolimits_{i\in S}x_i\leq\varepsilon,\quad \forall S\subset N, S\neq\varnothing\\&\sum\nolimits_{i\in N}x_i=v(N)\\&\varepsilon\in\mathbb{R},\ \ x_j\in\mathbb{R};\forall j\in N\end{aligned}$$

In order to calculate the nucleolus solution we simply apply the command `Nucleolus()` over the formerly defined game. Once the instruction is executed, the computer returns the first linear program to check that everything is running smoothly, followed by a large output related to the optimization process as follows.

```
> LEMAIRENUCLEOLUS<-Nucleolus(LEMAIRE)
```

```
Model name: Nucleolus of a gains game
          C1    C2    C3    C4
Minimize   0     0     0    -1
R1         0     0     0     1   >=         0
R2         1     0     0    -1   >=     46125
R3         0     1     0    -1   >=   17437.5
R4         0     0     1    -1   >=    5812.5
R5         1     1     0    -1   >=   69187.5
R6         1     0     1    -1   >=   53812.5
R7         0     1     1    -1   >=     30750
R8         1     1     1     0    =     90000
Kind     Std   Std   Std   Std
Type    Real  Real  Real  Real
Upper    Inf   Inf   Inf   Inf
Lower      0     0     0     0

Model name:  'Nucleolus of a gains game ' - run #1
Objective:  Minimize(R0)

SUBMITTED
Model size: 8 constraints, 4 variables, 19 non-zeros.
Sets:                          0 GUB,       0 SOS.

Using DUAL simplex for phase 1 and PRIMAL simplex for phase 2.
The primal and dual simplex pricing strategy set to 'Devex'.

Found feasibility by dual simplex after   4 iter.

Optimal solution      -6562.5 after       5 iter.

Excellent numeric accuracy ||*|| = 0

MEMO: lp_solve version 5.5.2.0 for 64 bit OS, with 64 bit
   LPSREAL variables.
    In the total iteration count 5, 0 (0.0) were bound flips.
    There were 2 refactorizations, 0 triggered by time and 0 by
    density.
           ... on average 2.5 major pivots per refactorization.
    The largest [LUSOL v2.2.1.0] fact(B) had 18 NZ entries,
    1.0x largest basis.
    The constraint matrix inf-norm is 1, with a dynamic
    range of 1.
    Time to load data was 0.009
    seconds, presolve used 0.000 seconds,
    ... 0.000 seconds in simplex solver, in total 0.009
    seconds.
Using DUAL simplex for phase 1 and PRIMAL simplex for phase 2.
The primal and dual simplex pricing strategy set to 'Devex'.

[...some output omitted...]
```

```
>summary(LEMAIRENUCLEOLUS)
```

```
Nucleolus of a gains game for the given coalitions
        v(S)     x(S)       Ei
   1 46125.0 52687.50 -6562.50
   2 17437.5 24468.75 -7031.25
   3  5812.5 12843.75 -7031.25
```

Next, by analyzing costs instead of gains, we introduce cost allocation problems, usually called airport problems (Littlechild and Thompson, 1977). Consider, for instance, several airlines that are jointly using an airstrip. Obviously, different airlines will have different needs for the airstrip. The larger the planes an airline flies, the longer the airstrip it needs. An airstrip that accommodates a given plane accommodates any smaller airplane at no extra cost. The airstrip is large enough to accommodate the largest plane any airline flies. How should its cost be divided among the airlines?

Note that under this illustration, several situations may be considered. For instance, consider farmers that are distributed along an irrigation drain. The farmer closest to the water gate only needs that the segment to his field would be maintained. Accordingly, the second closest farmer needs that the first two segments be maintained (the segment that goes from the water gate and the first farmer, and that segment from the first farmer to his field), and so on. The cost of maintaining a segment used by several farmers is incurred only once, independently of how many farmers use it. How should the total cost of maintaining the ditch be shared?

In order to illustrate this, consider the following cost airport game,

$$
\begin{aligned}
v(1) &= 26\\
v(2) &= 27\\
v(3) &= 55\\
v(4) &= 57\\
v(12) &= 53\\
v(13) &= 81\\
v(14) &= 83\\
v(23) &= 82\\
v(24) &= 84\\
v(34) &= 110\\
v(123) &= 108\\
v(124) &= 110\\
v(134) &= 110\\
v(234) &= 110\\
v(1234) &= 110
\end{aligned}
$$

After defining the game in *R*, we can see what would be the imputations using the *nucleolus*, taking into the account that the user must set the option "cost" within the nucleolus command,

```
> COALITIONS<-c(26,27,55,57,53,81,83,82,84,110,108,110,110,
110,110)
> AIR<-DefineGame(4,COALITIONS)
> AIRNUCLEOLUS<-Nucleolus(AIR,type="Cost")
```

```
Model name:
               C1    C2    C3    C4    C5    C6
   Minimize     0     0     0     0     1    -1
   Kind       Std   Std   Std   Std   Std   Std
   Type      Real  Real  Real  Real  Real  Real
   Upper      Inf   Inf   Inf   Inf   Inf   Inf
   Lower        0     0     0     0     0     0
   Model name: Nucleolus of a cost game
               C1    C2    C3    C4    C5    C6
   Maximize     0     0     0     0     1    -1
   R1           0     0     0     0     0     1   >=   0
   R2           1     0     0     0     1    -1   <=   26
   R3           0     1     0     0     1    -1   <=   27
   R4           0     0     1     0     1    -1   <=   55
   R5           0     0     0     1     1    -1   <=   57
   R6           1     1     0     0     1    -1   <=   53
   R7           1     0     1     0     1    -1   <=   81
   R8           1     0     0     1     1    -1   <=   83
   R9           0     1     1     0     1    -1   <=   82
   R10          0     1     0     1     1    -1   <=   84
   R11          0     0     1     1     1    -1   <=  110
   R12          1     1     1     0     1    -1   <=  108
   R13          1     1     0     1     1    -1   <=  110
   R14          1     0     1     1     1    -1   <=  110
   R15          0     1     1     1     1    -1   <=  110
   R16          1     1     1     1     0     0    =  110
   Kind       Std   Std   Std   Std   Std   Std
   Type      Real  Real  Real  Real  Real  Real
   Upper      Inf   Inf   Inf   Inf   Inf   Inf
   Lower        0     0     0     0     0     0

   Model name:  'Nucleolus of a cost game ' - run #1
   Objective:  Maximize(R0)

   SUBMITTED
   Model size: 16 constraints,   6 variables,  61 non-zeros.
   Sets:                          0 GUB,       0 SOS.
```

```
Using DUAL simplex for phase 1 and PRIMAL simplex for phase 2.
The primal and dual simplex pricing strategy set to 'Devex'.

Found feasibility by dual simplex after   4 iter.

Optimal solution                  13 after 7 iter.

Excellent numeric accuracy ||*|| = 0

[...some output omitted...]

> summary(AIRNUCLEOLUS)

Nucleolus of a cost game for the given coalitions

 v(S)  x(S)     Ei
1  26 13.00 -13.00
2  27 13.50 -13.50
3  55 40.75 -14.25
4  57 42.75 -14.25
```

2.4 Voting power

Shapley and Shubik (1954) propose the specialization of the *Shapley value* to voting games that measures the real power of a coalition.[a] The Shapley and Shubik index works as follows. There is a group of individuals all willing to vote on a proposal. They vote in order and as soon as a majority has voted for the proposal, it is declared passed and the member who voted last is given credit for having passed it. Let us consider that the members are voting randomly. Then we compute the frequency with which an individual is the one that gets the credit for passing the proposal. That measures the number of times that the action of that individual joining the coalition of their predecessors makes it a winning coalition. Note that if this index reaches the value of 0, then it means that this player is a dummy. When the index reaches the value of 1, the player is a dictator.

During Autumn 2014 Artur Mas (member of the Democratic Party of Catalunya (CiU) and President of Catalunya) said to Oriol Junqueras (leader of the Republican Party of Catalunya (ERC)) that "alternative majorities are possible" after discussing the referendum proposal of *November* 9 (Manchón, 2014). To conclude our paper we analyze these words through Shapley–Shubik power index. As aforementioned, this voting power index often reveals surprising power distribution that is not obvious on the surface. In order to

[a]Voting games are modeled by simple games. Those are cooperative games that can model various voting systems and the characteristic function is $v(S) \in \{0, 1\}$, for all coalitions $S \subseteq N$, where $v(N) = 1$ and $v(S) \leq v(T)$ if $S \subseteq T$.

TABLE 2 Catalan seats distribution after elections of 2003, 2006, and 2012

Year	CiU	PSC	ERC	PP	ICV	C's	CUP
2003	46	42	23	15	9	—	—
2006	48	37	21	14	12	3	—
2012	50	20	21	19	13	9	3

compare the power index of CiU and ERC we use the results of the elections of 2003, 2006, and 2012, whose results are displayed in Table 2.

To perform the Shapley–Shubik power index one simply provides the number of members of each party and the minimum amount of votes needed to pass a vote. For instance, for the 2003 elections, the reader only needs to define an object containing the seats distribution, and another object with the labels of the parties for the analyzed period. Therefore, the Shapley–Shubik power index, with a minimum amount of votes to pass a voting of 68 is

```
> #2003 Elections
> SEATS<-c(46,42,23,15,9)
> PARTIES<-c("CiU","PSC","ERC","PP","ICV")
> E2003<-ShapleyShubik(68,SEATS,PARTIES)
> summary(E2003)
```

```
Distribution of the agents

CiU PSC ERC PP ICV
 46  42  23 15   9

Minimum amount of votes to pass a vote: 68

Shapley-Shubik Power Index

       CiU        PSC        ERC         PP        ICV
0.40000000 0.23333333 0.23333333 0.06666667 0.06666667
```

```
> # 2006 Elections
> SEATS<-c(48,37,21,14,12,3)
> PARTIES<-c("CiU","PSC","ERC","PP","ICV","C's")
> E2006<-ShapleyShubik(68,SEATS,PARTIES)
> summary(E2006)
```

```
Distribution of the agents

CiU PSC ERC PP ICV C's
 48  37  21 14  12   3

Minimum amount of votes to pass a vote: 68

Shapley-Shubik Power Index

       CiU       PSC       ERC        PP       ICV       C's
0.40000000 0.23333333 0.23333333 0.06666667 0.06666667 0.00000000
```

```
> # 2012 Elections
> SEATS<-c(50,20,21,19,13,9,3)
> PARTIES<-c("CiU","PSC","ERC","PP","ICV","C's","CUP")
> E2012<-ShapleyShubik(68,SEATS,PARTIES)
> summary
```

```
Distribution of the agents

CiU  PSC  ERC   PP  ICV C's CUP
 50   20   21   19   13   9   3

Minimum amount of votes to pass a vote: 68

Shapley-Shubik Power Index

  CiU        PSC        ERC        PP         ICV
  0.53333333 0.13333333 0.13333333 0.13333333 0.03333333
  C's        CUP
  0.03333333 0.00000000
```

Having a look to the data of 2003 it might seem that PSC might have much more power than ERC (19 less seats in the camera), and the same should apply to year 2006. After executing `ShapleyShubik()` (results displayed in Table 3) one can see there are no differences in power among ERC and PSC for the chosen years. Another interesting case is the dummy player, both C's (in 2006) and CUP (in 2012) parties, never become pivotal players. Furthermore, one might consider that President Mas was right as there are two more parties with the same Shapley–Shubik power index.

3 Marketing and game theory

Following Cano-Berlanga et al. (2017a) we provide an actual and interesting implementation of game theory in marketing. Specifically, through the definition of coalitions, we determine how the revenues obtained in an online sale

TABLE 3 Shapley–Shubik power index of the catalan parliament

Year	CiU	PSC	ERC	PP	ICV	C's	CUP
2003	0.400	0.233	0.233	0.067	0.067	—	—
2006	0.400	0.233	0.233	0.067	0.067	0.000	—
2012	0.533	0.133	0.133	0.133	0.033	0.033	0.000

should be distributed among the different channels used by the consumer. By doing so, next we introduce some basics about the purchasing procedure from the theoretical point of view.

3.1 The classic consumer theory

von Neumann and Morgenstern (1944) analyze how consumers make purchasing decisions. Specifically, they study the properties of the individuals' preferences that are transferred into a utility function. This function measures the satisfaction or benefit obtained by the consumer from a specific purchasing (i.e., a combination of goods' basket). Consequently, the purchase process is obtained through an optimization problem, where the consumer maximizes his utility function taking into account his budget constraint. Formally,

$$\begin{aligned} &Max \quad u(x_1, x_2, \ldots, x_n) \\ &s.a. \quad \sum_{i=1}^{n} p_i \cdot x_i = m \end{aligned}$$

The solution of this problem leads us to a *demand function* with a negative relationship between the quantity, x_i, and its price, p_i. It is noteworthy that the demand function plays a key-role in the literature, since its proper estimation allows us to know (i) the individuals' reactions when prices change, and, (ii) how the demand of a certain good reacts to its economic context. To illustrate the aforementioned comments, we present a synthetic linear demand function,

$$x_i = A_i(E) - \beta \cdot p_i$$

whose parameters have the following interpretation:

x_i: purchased quantity of good i.
$A_i(E)$: relationship of x_i with the context. This magnitude explains the interaction between the demand of the analyzed good among a large list of factors, such that complementary products, substitute goods and income of the buyer.
β: individuals' reaction to changes in the price. The higher the β, the more sensitive the consumer is to changes in prices.

In this regard, quantitative research of demand functions has provided different developments on how individuals take purchasing decisions in more complex contexts. For instance, Berry et al. (1995) provide a sophisticated study about the demand in automobile sector. Nonetheless, demand models are extremely complicated to estimate: they require a large amount of data, significant computational power and a precise econometric estimation that guarantees a proper statistical behavior. Though quantitative estimation of demand function is such a difficult task, its econometric specification sheds additional light regarding the purchasing process. Therefore, from an empirical perspective, a demand function takes the following expression,

$$x_i = A_i(E) - \beta \cdot p_i + \varepsilon_i,$$

where the error ε_i is introduced to our simple linear demand. The error plays a fundamental role on the consumers' purchasing mechanism, as it provides a random component in the original model. On the one hand, the new term measures the consumers' response to news and different stimulations related to a nondeterministic way of the purchasing dynamics of x_i. On the other hand, the qualitative impact of ε_i is extraordinary, since it explains how exogenous phenomena might alter the purchasing decision. Indeed, the better modeling of the error term has improved the understanding of some economics fields. For instance, Engle (1982) dramatically enhances the comprehension of Financial Markets thanks to his ARCH model, which is a refinement on how to model ε_i in stock returns time series (see Bollerslev, 1987).[b]

In our context, ε_i has two main implications. First, it transforms the initial model to a more realistic approach, since it relaxes the strong rationality hypothesis of the consumer theory. Second, ε_i tell us that applying the right amount of positive pressure, individuals may be exogenously influenced in order to increase the sales of a good (see Scott, 1976, Tybout, 1978, Prabhu and Stewart, 2001, among others). Therefore, even if the demand function remains unknown a marketer can raise his success via publicity, i.e., advertisements.

3.2 Attribution models

In the digital media era, consumers are viewing ads nearly everywhere, through several different marketing channels (organic search, email, display ads, social media, for instance). With a high volume of conversions, a marketer may wonder what channel is more efficient and what channels must be reinforced to improve future sales. Hence, the concept of *Atribution* arises naturally. Attribution concept was originated in psychology and was introduced in marketing during the early 1970s. Within that period of time we find several studies which try to evaluate the success of different marketing techniques (Kannan et al., 2016;

[b]Such sophistication was awarded with the Economics Nobel Prize in 2003.

Li and Kannan, 2014; Mizerski, 1978; Settle and Golden, 1974; Swinyard and Ray, 1977, among others), but the concept of marketing attribution has evolved with the departure from traditional selling strategies. Nowadays, attribution may be defined as the quantification of the influence that each advertising impression has on a consumers' conversions.

Several attribution commonly use single and fractional source. Nonetheless, the problem concerning to these methods is that according to the chosen model, a bias that generates a conflict between the different digital marketing channels may not be avoid. Henceforth, more complex perspectives are available to overcome this issue. At this point, it is noteworthy that digital marketing channels are not isolated, indeed there exists positive feedback between them increasing the likelihood of purchasing. Consequently, *Google Analytics 360* has based its new Data-Driven Attribution model on *cooperative game theory* and the *Shapley value*.

In this context, we propose a fair distribution of the revenues among the considered channels, in order to facilitate the cooperation and to guarantee its stability. By doing so, and due to the features of the analyzed problem, we define the worth of each coalition taking into the account the observed frequencies, i.e., the sequences of touch points (last click, first click, time decay, among others). Then, we use the Shapley value to allocate all the revenues among the different channels.

3.2.1 Sales game

Consider that for a period of time we study the sale success for three channels: Direct, Organic, and CPC (hereinafter, players 1, 2, and 3, respectively). To apply the proposed model, we need not only the independent sales of each channel, but also the sales obtained by the interaction of the channels (see Table 4).

TABLE 4 Data base

Channels	$I(R)$
1	19786
2	20837
3	24008
12	898
13	822
23	822
123	194

Data obtained from each of the channels and their interaction.

Given this information about the frequencies, the associated cooperative game is built as follows. Note that, for the sake of simplifying the implementation of the game and its computation, we apply the matrix format through,

$$B \times \varphi = v(S)$$

where B is a binary squared matrix of dimension $2^n - 1$, containing the coefficients related to $I(R)$ and taking into the account if the players are part of the coalition S; φ is a vector composed by the values $I(R)$; and, $v(S)$ denotes the worth of the coalitions. Applying it for a three-players game, we obtain the following expression,

$$\begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 & 1 & 0 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 \end{bmatrix} \times \begin{bmatrix} I(1) \\ I(2) \\ I(3) \\ I(12) \\ I(13) \\ I(23) \\ I(123) \end{bmatrix} = \begin{bmatrix} v(1) \\ v(2) \\ v(3) \\ v(12) \\ v(13) \\ v(23) \\ v(123) \end{bmatrix}$$

Hence, it is easy to show how the coalitions are built. For instance,

$$v(2) = I(2), v(12) = I(1) + I(2) + I(12), v(23) = I(2) + I(3) + I(23),$$

$$v(123) = I(1) + I(2) + I(3) + I(12) + I(13) + I(23) + I(123),$$

that is,

$$v(2) = 20837, v(12) = 19786 + 20837 + 898, v(23) = 20837 + 24008 + 822,$$

$$v(123) = 19786 + 20837 + 24008 + 898 + 822 + 822 + 194.$$

By using the total number of frequencies obtained in the Table 4, we obtain the value of each coalition (see Fig. 1 and Table 5).

Once the grand coalition N is achieved, in order to cooperate and maximize each agent's gains, how will the profits be distributed among the players? Solving this question, many solutions concepts are proposed in the literature (see Matsumoto and Szidarovszky, 2016, for instance) satisfying two minimal requirements:

- Individual rationality: An allocation x satisfies individual rationality if each player receives a payoff greater or equal to what can be guaranteed on his own, without cooperating with anyone else, i.e., $x_i \geq v(i)$ for all $i \in N$.
- Efficiency: An allocation $x(N)$ is efficient if it distributes the worth of the grand coalition $v(N)$ among all players, i.e., $x(N) = x_1 + \cdots + x_n = v(N)$.

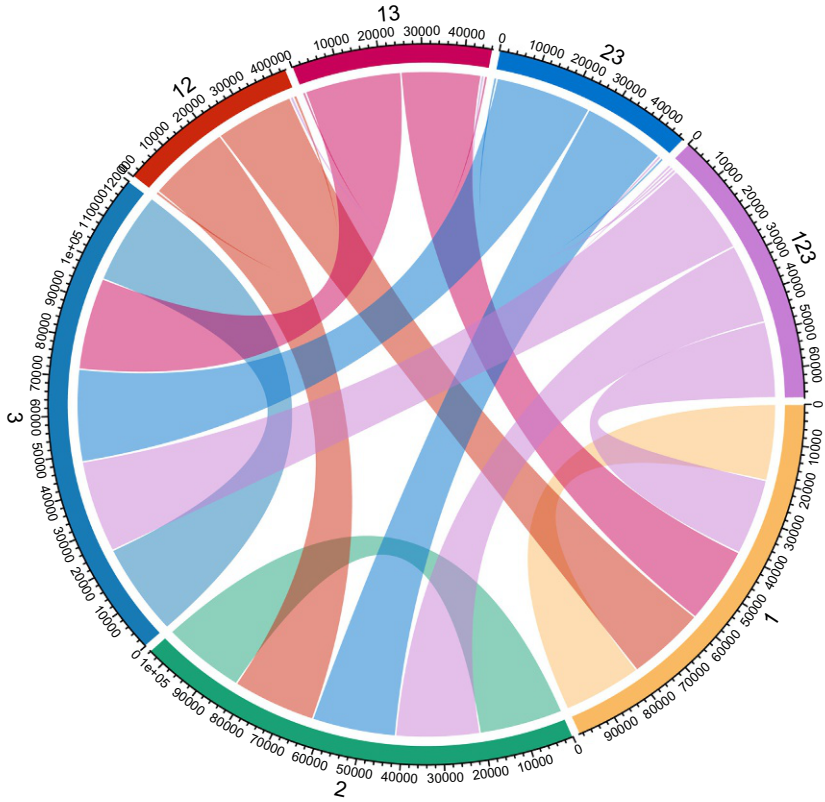

FIG. 1 Graphic representation of a three-players sales channels game. The areas which correspond to the individual coalitions show each player's influence on the total game. The intermediate coalitions areas represent the position and the final composition, approximately.

TABLE 5 Worth of the *sales channels game* characteristic function

Coalition	$v(S)$
1	19786
2	20837
3	24008
12	41521
13	44616
23	45667
123	67367

Data from Table 4.

TABLE 6 The Shapley value applied to the data obtained from Table 5

Player	Shapley value (ϕ_i)
1	20710.67
2	21761.67
3	24894.67

Among all the proposed solutions, we use the Shapley value, due to the fact that it considers the concept of marginality (a key issue in our framework), and it satisfies a set of properties that may be considered as compulsory conditions in our context.

Hence, organizing the information as in Section 2.2, we can check that the Shapley value (ϕ_i) is the average value of each player's marginal contributions, taking into the account all the possible orderings. Specifically, in our example, we obtain Table 6.

Note that the Shapley value proposes an allocation that ensures to each player a larger amount than the worth of the individual coalition (individual rationality), and the sum of the all the payments corresponds with the worth of the grand coalition (efficiency).

The *R* code to reproduce this example is as follows:

```
> Marketing <- c(19786, 20837, 24008, 41521, 44616, 45667, 67367)
> Sales<-DefineGame(3,Marketing)
> Attribution<-ShapleyValue(Sales)
> summary(Attribution)
```

```
Shapley Value for the given game

     Shapley Value
[1,]      20710.67
[2,]      21761.67
[3,]      24894.67
```

4 Claims problems

Finally, we propose the study of conflicting claims problems. A conflicting claims problem is a particular case of the distribution problem, in which the amount to be distributed, the *endowment* E, is not enough to satisfy the agents' claims on it. This model describes the situation faced by a court that has to distribute the net worth of a bankrupt firm among its creditors.

But, it also corresponds with cost-sharing, taxation, or rationing problems. The formal analysis of situations like these, which originates in a seminal paper by O'Neill (1982), shows that a vast number of well-behaved solutions have been defined for solving conflicting claims problems, being the *proportional*, the *constrained equal awards*, the *constrained equal losses*, the *Talmud* and the *random arrival* rules the prominent concepts used.[c]

An illustrative example of conflicting claims problems is the fishing quotas reduction, in which the agent's claim can be understood as the previous captures, and the endowment is the new (lower) level of joint captures (Gallastegui et al., 2003; Iñarra and Skonhof, 2008). A similar example is given by milk quotas among European Union (EU) members.[d] In both examples, *proportionality* is the main principle used. Another example of conflicting claims situations is the September 11 Victim Compensation Fund (VCF), where the income each victim would have earned in a full lifetime was estimated and the individual claim is the legal right to be compensated. Similarly, bankruptcy laws consider the claimants identity to establish a priority rule. Specifically, bankruptcy codes normally list all claims that should be treated identically in various categories and assigns to them lexicographic priorities (Kamiski, 2006). Pulido et al. (2002, 2008) analyze, under the name of bankruptcy problems with references, the real-life case of allocating a given amount of money among the various degree courses that are offered at a (public) Spanish university. The (verifiable) monetary needs of each course constitute their claims. Additionally, there exist reference values for each course, which are set by the government independently, below their claims. Other relevant practical cases also involving more complex rationing situations could be protocols for the reduction of pollution (Giménez-Gómez et al., 2016), water distribution in drought periods, or even some resource allocation procedures in the public health care sector, in which past consumption could be considered as an entitlement, and current needs as a claim (see, for instance, Hougaard et al., 2012, Moreno-Ternero and Roemer, 2012). The formalization of such problems is as follows.

4.1 Claims rules

Consider a set of agents $N=\{1,2,\ldots,n\}$ and amount $E\in\mathbb{R}_+$ of an infinite divisible resource, the *endowment*, that has to be allocated among them. Each agent has a *claim*, $c_i\in\mathbb{R}_+$ on it. Let $c\equiv(c_i)_{i\in N}$ be the claims vector.

A *conflicting claims problem* is a pair (E, c) with $\sum_{i=1}^{n} c_i > E$. Without loss of generality, we will order the agents according to their claims $c_1 \leq c_2 \leq \cdots \leq c_n$ and we will denote by $\mathcal{B}$ the set of all conflicting claims problems.

[c]The reader is referred to Moulin (2002) and Thomson (2003, 2013) for reviews of this literature.
[d]Quotas were introduced in 1984. Each member state was given a reference quantity which was then allocated to individual producers. The initial quotas were not sufficiently restrictive to remedy the surplus situation and so the quotas were cut in the late 1980s and early 1990s. Quotas will end on April 1, 2015.

Given a conflicting claims problem, a rule associates within each problem a distribution of the endowment among the agents. A *rule* is a single-valued function $\varphi : \mathcal{B} \to \mathbb{R}^n_+$ such that $0 \leq \varphi_i(E, c) \leq c_i$, for all $i \in N$ (*nonnegativity* and *claim-boundedness*); and $\sum_{i=1}^{n} \varphi_i(E,c) = E$ (*efficiency*). Next, we present the most used rules.

The *proportional (P)* rule recommends a distribution of the endowment which is proportional to the claims: for each $(E,c) \in \mathcal{B}$ and each $i \in N$, $P_i(E, c) \equiv \lambda c_i$, where $\lambda = \frac{E}{\sum_{i \in N} c_i}$.

The *constrained equal awards (CEA)* rule (Maimonides, twelfth century) proposes equal awards to all agents subject to no one receiving more than his claim: for each $(E, c) \in \mathcal{B}$ and each $i \in N$, $CEA_i(E,c) \equiv \min\{c_i, \mu\}$, where μ is such that $\sum_{i \in N} \min\{c_i, \mu\} = E$.

The *constrained equal losses (CEL)* rule (Maimonides, twelfth century (Aumann and Maschler, 1985) chooses the awards vector at which all agents incur equal losses, subject to no one receiving a negative amount: for each $(E,c) \in \mathcal{B}$ and each $i \in N$, $CEL_i(E,c) \equiv \max\{0, c_i - \mu\}$, where μ is such that $\sum_{i \in N} \max\{0, c_i - \mu\} = E$.

The *Talmud (T)* rule (Aumann and Maschler, 1985) proposes to apply the constrained equal awards rule, if the endowment is not enough to satisfy the half-sum of the claims. Otherwise, each agent receives the half of his claim and the constrained equal losses rule is applied to distribute the remaining endowment: for each $(E,c) \in \mathcal{B}$, and each $i \in N$, $T_i(E, c) \equiv CEA_i(E, c/2)$ if $E \leq \sum_{i \in N} c_i/2$; or $T_i(E,c) \equiv c_i/2 + CEL_i(E - \sum_{i \in N} c_i/2, c/2)$, otherwise.

The *random arrival (RA)* rule (O'Neill, 1982). Consider that each claim is fully honored following an order of the claimants' arrival, until the endowment runs out. In order to remove the unfairness of the first-come first-served scheme associated with any particular order of arrival, the rule proposes to take the average of the awards vectors calculated in this way when all orders are equally probable: for each $(E,c) \in \mathcal{B}$, and each $i \in N$, $RA_i(E,c) \equiv \frac{1}{|N|!} \sum_{\prec \in \mathbb{R}^N} \min\{c_i, \max\{E - \sum_{j \in N, j \prec i} c_j, 0\}\}$.

The *adjusted proportional (AP)* rule (Curiel et al., 1987) is a composition of minimal rights and the proportional rule. First, we attribute to each claimant his minimal right and revise his claim down. Then, the proportional rule is applied to distribute the remaining endowment according to the revised claims: for each $(E,c) \in \mathcal{B}$ and each $i \in N$, $AP_i(E,c) = m_i(E,c) + P(E - \sum_{i \in N} m_i(E,c), c - m(E,c))$.

4.2 Obtaining fishing quotas

As an illustration, we replicate Gallastegui et al. (2003). They analyze the distribution of Northern European Anglerfish Fishery quotas among EU countries in terms of the allocations recommended by different solutions and how this

may affect the sustainable growth of the fishing catches. Specifically, they consider seven countries (France, Spain, UK, Ireland, Belgium, Netherlands, and Germany). Each country has a claim, which depends on its historical fishing catches (13,952; 6256; 4348; 2196; 927; 299; 158, respectively).

To replicate the study of Gallastegui et al. (2003) we can execute the commands one by one or use `Allrules()` to run all of them at once. By doing so, we create objects containing the individual claims and labels of the different countries. After that, running `Allrules()` is straightforward, i.e., `AllRules(13500,CLAIMS,COUNTRIES)`. *R* displays the following magnitudes for this particular case, and also includes the Gini Index of every rule to check inequality among them,

```
> ## replication of Gallastegui et al. (2003), Table 7.
> CLAIMS <- c(158,299,927,2196,4348,6256,13952)
> COUNTRIES <- c("Germany","Netherlands","Belgium",
"Ireland","UK","Spain","France")
> INARRA <- AllRules(13500,CLAIMS,COUNTRIES)
> summary(INARRA)
```

```
Claims of the Agents

   Germany Netherlands     Belgium     Ireland          UK
       158         299         927        2196        4348
     Spain      France
      6256       13952

  Assignments according to the following rules

             Proportional     CEA      CEL   Talmud      RA
Germany            75.81  158.00     0.00    79.0    73.73
Netherlands       143.46  299.00     0.00   149.5   139.53
Belgium           444.79  927.00     0.00   463.5   436.92
Ireland          1053.67 2196.00     0.00  1098.0  1071.42
UK               2086.22 3306.67   662.67  2174.0  2147.42
Spain            3001.71 3306.67  2570.67  3128.0  3101.42
France           6694.34 3306.67 10266.67  6408.0  6529.57
```

Displaying the output of the allocations is undertaken by running `PlotAll()`. Graphical analysis of the inequality among rules is performed by `LorenzRules()` (Figs. 2 and 3).

```
> plot(INARRA,5) ## Display allocations for UK
```

```
> LorenzRules(INARRA) ## Inequality graph
```

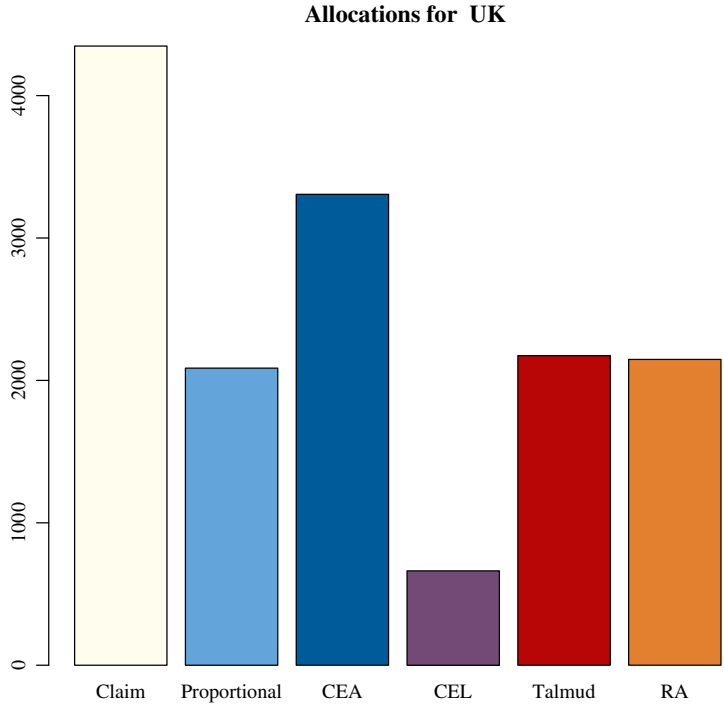

FIG. 2 Fishing captures allocations for UK.

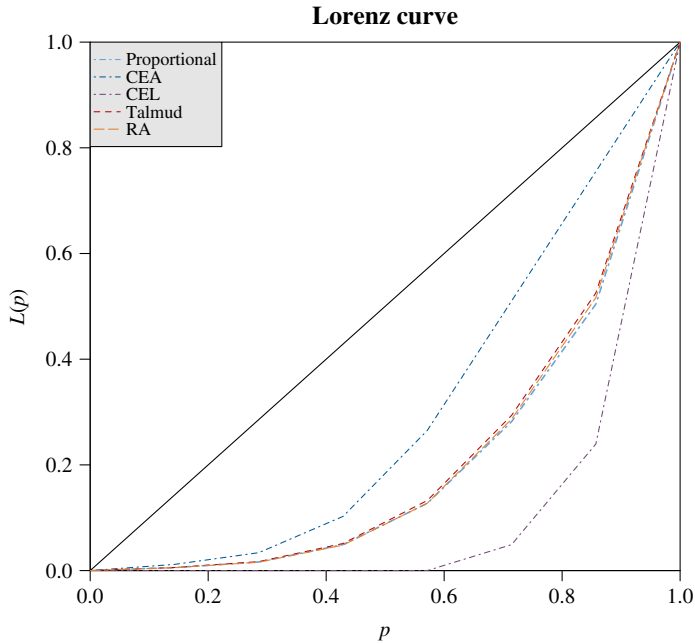

FIG. 3 Inequality analysis for the different rules.

5 Concluding remarks

The present chapter explains an R package for cooperative games. This kind of games, as aforementioned, may be used to many different applications, such as investments and insurance, joint development of projects, production and transportation planning, and environmental economics, among others. In this regard, we provide to the reader a compatible framework with the Google approach. Additionally, we show its application to the attribution context, and we evaluate the impact of a digital campaign on the purchasing process. In doing so, we define a way to transfer the consumers' conversions into a convex cooperative game, and we apply the Shapley value to our data.

Last but not least, we would like to mention that game theory offers a wide range of solutions. Therefore, users have a considerable number of options to calculate different allocations. For instance, in R there are more game theory packages available, such as: *Games* which provides Statistical Estimation of Game-Theoretic Models; *coopProductGame*, which computes cooperative game and allocation rules associated with linear production programming problems; *GameTheoryAllocation* which features new allocations to the framework presented in this chapter and *EvolutionaryGames*, which models situations where strategical behavior appears.

Acknowledgments

Financial support from Generalitat de Catalunya (2014SGR325 and 2014SGR631) and Ministerio de Economía y Competitividad (ECO2016-75410-P (AEI/FEDER, UE) and ECO2017-86481-P (AEI/FEDER, UE)) are acknowledged.

References

Aumann, R.J., 1960. Linearity of unrestrictedly transferable utilities. Naval Res. Logist. Q. 7, 281–284.

Aumann, R.J., Maschler, M., 1985. Game theoretic analysis of a bankruptcy from the Talmud. J. Econ. Theor. 36, 195–213.

Berry, S., Levinsohn, J., Pakes, A., 1995. Automobile prices in market equilibrium. Econometrica: J. Econ. Soc. 841–890.

Bollerslev, T., 1987. A conditionally heteroskedastic time series model for speculative prices and rates of return. Rev. Econ. Stat. 69, 542–547.

Cano-Berlanga, S., Giménez-Gómez, J. M., Vilella, C., 2017a. Attribution Models and the Cooperative Game Theory. CREIP working paper series.

Cano-Berlanga, S., Giménez-Gómez, J.-M., Vilella, C., 2017b. Enjoying cooperative games: the R package GameTheory. Appl. Math. Comput. 305, 381–393.

Curiel, J., Maschler, M., Tijs, S.H., 1987. Bankruptcy games. Z. Oper. Res. 31, A143–A159.

Edgeworth, F.Y., 1881. Mathematical Psychics: An Essay on the Application of Mathematics to the Moral Sciences, vol. 10. Kegan Paul.

Engle, R.F., 1982. Autoregressive conditional heteroscedasticity with estimates of the variance of United Kingdom inflation. Econometrica: J. Econ. Soc. 987–1007.

Gallastegui, M.C., Iñarra, E., Prellezo, R., 2003. Bankruptcy of fishing resources: the Northern European anglerfish fishery. Mar. Resour. Econ. 17, 291–307.

Gillies, D. B., 1953. Some theorems on n-person games (Master thesis). University of Princeton.

Giménez-Gómez, J.-M., Teixidó-Figueras, J., Vilella, C., 2016. The global carbon budget: a conflicting claims problem. Clim. Change 136 (3), 693–703.

Hougaard, J.L., Moreno-Ternero, J., Osterdal, L.P., 2012. A unifying framework for the problem of adjudicating conflicting claims. J. Math. Econ. 48, 107–114.

Iñarra, E., Skonhof, A., 2008. Restoring a fish stock: a dynamic bankruptcy problem. Land Econ. 84 (2), 327–339.

Kamiski, M., 2006. Parametric rationing methods. Games Econ. Behav. 54, 115–133.

Kannan, P.K., Reinartz, W., Verhoef, P.C., 2016. The path to purchase and attribution modeling: introduction to special section. Int. J. Res. Mark. 33 (3), 449–456.

Lemaire, J., 1991. Cooperative game theory and its insurance applications. Astin Bull. 21 (1), 17–40.

Li, H., Kannan, P.K., 2014. Attributing conversions in a multichannel online marketing environment: an empirical model and a field experiment. J. Market. Res. 51 (1), 40–56.

Littlechild, S.C., Thompson, G.F., 1977. Aircraft landing fees: a game theory approach. Bell J. Econ. 186–204.

Manchón, M., 2014. Mas to Junqueras: other majorities are possible in the parliament. Economía Digital, September 16, 2014.

Maschler, M., 1979. Geometric properties of the kernel, nucleolus, and related solution concepts. Math. Oper. Res. 4 (4), 303–338.

Matsumoto, A., Szidarovszky, F., 2016. Game Theory and Its Applications. Springer.

Mizerski, R.W., 1978. Causal complexity: a measure of consumer causal attribution. J. Market. Res. 220–228.

Moreno-Ternero, J.D., Roemer, J.E., 2012. A common ground for resource and welfare egalitarianism. Games Econ. Behav. 75, 832–841.

Moulin, H., 2002. Axiomatic cost and surplus sharing. In: Arrow, K., Sen, A., Suzumura, K. (Eds.), Handbook of Social Choice and Welfare, vol. 1. Elsevier, North Holland, Amsterdam, pp. 289–357.

O'Neill, B., 1982. A problem of rights arbitration from the Talmud. Math. Soc. Sci. 2 (4), 345–371.

Prabhu, J., Stewart, D.W., 2001. Signaling strategies in competitive interaction: building reputations and hiding the truth. J. Market. Res. 38 (1), 62–72.

Pulido, M., Borm, P., Hendrickx, R., Llorca, N., Sánchez-Soriano, J., 2002. Game theory techniques for university management: an extended bankruptcy model. Ann. Oper. Res. 109, 129–142.

Pulido, M., Borm, P., Hendrickx, R., Llorca, N., Sánchez-Soriano, J., 2008. Compromise solutions for bankruptcy situations with references. Ann. Oper. Res. 158 (1), 133–141.

Schmeidler, D., 1969. The nucleolus of a characteristic function game. SIAM J. Appl. Math. 17, 1163–1170.

Scott, C.A., 1976. The effects of trial and incentives on repeat purchase behavior. J. Market. Res. 263–269.

Settle, R.B., Golden, L.L., 1974. Attribution theory and advertiser credibility. J. Market. Res. 181–185.

Shapley, L.S., 1953. A Value for n-Person Games. Princeton University Press, Princeton NJ, pp. 307–317 (Chapter 4).

Shapley, L.S., Shubik, M., 1954. A method for evaluating the distribution of power in a committee system. Am. Polit. Sci. Rev. 48 (3), 787–792.

Swinyard, W.R., Ray, M.L., 1977. Advertising-selling interactions: an attribution theory experiment. J. Market. Res. 14, 509–516.

Thomson, W., 2003. Axiomatic and game-theoretic analysis of bankruptcy and taxation problems: a survey. Math. Soc. Sci. 45 (3), 249–297. ISSN 01654896. https://doi.org/10.1016/S0165-4896(02)00070-7.

Thomson, W., 2013. Axiomatic and Game-Theoretic Analysis of Bankruptcy and Taxation Problems: An Update Mimeo.

Tybout, A.M., 1978. Relative effectiveness of three behavioural influence strategies as supplements to persuasion in a marketing context. J. Market. Res. 15, 229–242.

von Neumann, J., Morgenstern, O., 1944. Theory of Games and Economic Behavior. Princeton University Press, Princeton.

This page intentionally left blank

Index

Note: Page numbers followed by "*f*" indicate figures, "*t*" indicate tables, and "*np*" indicate footnotes.

D

E

F

P

Q

R

S

T

U

V

W

C.R. Rao, Ph.D., Sc.D. (Cantab) is the long-standing Series Editor of the *Handbook of Statistics* since 1988. Professor Rao is a renowned statistician, holding a number of distinguished achievements and awards, including: Developer of Statistics as an Independent Discipline, Padma Vibhushan Awardee (India), Life Fellow of Kings College, Cambridge (UK), Fellow of the Indian National Science Academy, Fellow of the Royal Society (UK), Guy Medal in Gold of Royal Statistical Society (UK), and National Medal of Science, India and USA.

ELSEVIER

North-Holland

An imprint of Elsevier
elsevier.com/books-and-journals

ISBN 978-0-444-64311-7