Maximized Monte Carlo: an R package

Julien Neves *

February 27th, 2017

Abstract

This paper describes the R package **MMC**, which implements the Maximized Monte Carlo (MMC) technique and the simple Monte Carlo technique with tie-breaker for discrete statistics. To illustrate how to use the **MMC** package, we present two examples: a goodness-of-fit test and the Behrens-Fisher problem.

We perform a simulation study, in order to demonstrate the utility of the package, of irregular Wald-type tests and unit root tests for autoregressive models. We compare the MMC procedure implemented in the **MMC** package to the parametric bootstrap and usual asymptotic tests in both cases.

*Graduate student, Department of Economics, McGill University

Contents

1   Introduction

In the past several years, Monte Carlo methods have become progressively more popular. Nowadays, most statistical software will offer some sort of simulation-based techniques for data analysis (e.g. parametric bootstrap, Markov Chain Monte Carlo (MCMC) algorithm, Evolutionary Monte Carlo (EMC) algorithm, etc.). A quick search on the Comprehensive R Archive Network (CRAN) reveals that already many packages implement some Monte Carlo techniques; see **boot** (**canty˙boot˙2016**), **mcmc** (**geyer˙mcmc˙2015**), **MCMCpack** (**martin˙mcmcpack˙2011**), **PAWL** (**bornn˙pawl˙2012**), **ggmcmc** (**fernandez-i-marin˙ggmcmc˙2016**), **exactLoglinTest** (**caffo˙exactloglintest˙2013**).

Monte Carlo techniques were first introduced independently by **dwass˙modified˙1957** and by **barnard˙comment˙1963** for a review, see **dufour˙monte˙2003** The method they proposed details a simple procedure to implement exact tests based on pivotal statistics (i.e. when the distribution does not depend on any nuisance parameters). As a matter of fact, the method only requires that the statistic can be simulated, it does not impose tractability to the distribution of statistic.

The Monte Carlo technique is akin to the parametric bootstrap in that both methods rely on simulating a fully parametric model under the null in order to make any inference. The difference lies in that the parametric bootstrap is only asymptotically justified while the Monte Carlo technique p-value corrects for finite-sample distributions.

Note that the original Monte Carlo technique, as described by **dwass˙modified˙1957** and **barnard˙comment˙1963** is only valid for statistics with continuous distributions. The problem that arises is due to the possible presence of ties in the simulated values for statistics with discrete distributions. **dufour˙monte˙2006** provides an extension of the method to circumvent this problem known as the Monte Carlo with tie-breaker technique (MC with tie-breaker).

Even though the Monte Carlo technique provides a straightforward procedure to build exact tests, one of its main shortfalls is the requirement that the finite-sample distribution of the statistic be free of any nuisance parameters. Sadly, statistics of interest in economics usually depend on some nuisance parameters. Techniques such as the bootstrap and its variants provide asymptotically valid method of inference when the distribution of the statistic is not stable. In order to retrieve exactness in tests where the finite-sample distribution of the statistic depends on some nuisance parameters, **dufour˙monte˙2006** proposed an extension of the original Monte Carlo technique to deal with such complications also known as the Maximized Monte Carlo (MMC).

In this paper, we present an implementation of both the MMC and MC with tie-breaker techniques as described by **dufour˙monte˙2006** for R through the **MMC** package. Readers should note that the main goal of the **MMC** package is not to replace any existing package, but to supplement R users with an accessible and effortless way to apply Monte Carlo tests, and more specifically the MMC procedure which is not yet available anywhere. In order to demonstrate where the use of the **MMC** package could be advantageous, we investigate two different

types of common tests: Wald-type tests and unit root tests.

First, we look at Wald-type tests. The popularity of Wald-type tests can be attributed to the fact that they offer a simple way of testing a wide array of linear and non-linear restrictions using only estimators of the unrestricted model. One issue Wald-type statistics is that, since they are based on asymptotically normal estimators, they need strong regularity assumptions in order to derive any valid inference. Under these regularity conditions, the usual Wald statistic is asymptotically distributed as a chi-squared with degrees of freedom equal to the number of restrictions. As described by **dufour˙wald˙2013** these regularity conditions will be violated whenever the covariance matrix of the set of restrictions does not have full column rank in an open neighborhood of the true value of the parameter vector. Note that this does not preclude the covariance matrix of the set of restrictions to have full column rank in finite samples. Indeed, we could have a covariance matrix where in finite sample the matrix is well-defined and invertible, but, as the sample size goes to infinity, the covariance matrix would converge to a singular matrix. Then, in this case, regularity conditions would be violated. As a matter of fact, situations of this type might be more common than one would first assume. **dufour˙wald˙2013** list a set of common examples where regularity conditions needed by the Wald-type statistics are breached due to an asymptotically singular covariance matrix of the set of restrictions, including, but not limited to, tests of Granger non-causality in VARMA models, see **boudjellaba˙testing˙1992**; **boudjellaba˙simplified˙1992** and test of volatility and co-volatility, see **gourieroux˙size˙2013**

Note that, as stated previously, the Wald statistic can be well-defined in finite sample while converging to a singular matrix as the sample size goes to infinity. Accordingly, Monte Carlo techniques will yield valid inference as they do not rely on the existence of the asymptotic distribution of the statistic, but simply require that we can simulate the finite-sample distribution. As such, Monte Carlo techniques should be robust to this type of violation of regularity conditions, and will allow us to use Wald-type tests even in situations where the usual asymptomatic critical values are invalid or at best misleading.

Second, we turn our attention to test for the order of integration of time series. Since multiple time-series in economics can be characterized as integrated processes of different order, the problem of testing for the presence of integration has been study extensively in the past decades. One of the most important tests for the presence of unit roots in time-series comes from **dickey˙distribution˙1979**; **fuller˙introduction˙1976** Their test based on random walk processes was later extended to processes of order p, also known as augmented Dickey-Fuller tests.

Now, the augmented Dickey-Fuller type tests are based on I(1) processes and do not technically preclude I(2) processes. But, as described in **david˙a.˙dickey˙determining˙1987 haldrup˙robustness˙2002** and **pantula˙testing˙1989** the null-distribution of the augmented Dickey-Fuller statistic is not invariant to the order of integration of the process. In fact, **david˙a.˙dickey˙determining˙1987** showed using Monte Carlo simulations that when testing for a unit root in an I(2) process, the lower-tailed test critical values given in **fuller˙introduction˙1976** will slightly

4

over-reject. Additionally, **haldrup˙robustness˙2002** confirmed that the lower-tail of the distribution (associated with $H_1 : \gamma < 0$) is only slightly affected by the presence of a unit root of multiplicity of 2, but, **haldrup˙robustness˙2002** also showed that the upper-tail of the distribution (associated with $H_1 : \gamma > 0$) is severely distorted. Therefore, in this paper we present a way to correct for this distortion using the MMC technique developed by **dufour˙monte˙2006** and implemented in this package.

This paper is organized as follows. Section 2 provides the framework for both the MC technique with tie-breaker in the case of a discrete statistic and the MMC. Section 3 gives the details for the implementation of the **MMC** package. Also included in section 3 is an example of test of goodness of fit and an example based on testing the equality of means in populations with different variances to demonstrate how the **MMC** package can be directly applied. Section 4 sets the framework for Wald-type tests, describes four example of irregular Wald statistics, and reports their respective simulation study results. Section 5 provides the framework for the augmented Dickey-Fuller test, and the results for the unit root test simulation study. Finally, the source code of the package is provided in the appendix. Note that *.Rd* files are excluded as we are using the package **roxygen2** (**wickham˙roxygen2:˙2015**) to generate them directly from our *.R* files.

2 Monte Carlo tests

Let S be a statistic that can be simulated. Then the distribution of S follows either one of the two following cases

$$S \sim \mathcal{F}(y) \tag{1}$$

$$S \sim \mathcal{F}(y \mid \nu) \tag{2}$$

i.e. where the finite-sample distribution of the statistic S is either pivotal or non-pivotal (i.e. dependent on some the nuisance parameter ν).

2.1 Monte Carlo with tie-breaker

Let H_0 be the null hypothesis we wish to test. Then, we first look at the case where the statistic is pivotal under H_0 (i.e. $S \sim \mathcal{F}(y)$ under H_0).

The usual first step for the Monte Carlo technique is to compute the statistic S_0 using the sample data, and then generate N *i.i.d.* replications S_j for $j = 1, ..., N$ of S under H_0. We do not concern ourselves for the moment with how the S_j are generated, but simply assume that we are indeed able to generate them.

In order to test for H_0, **dufour˙monte˙2003** provide the following p-value function

$$\hat{p}_N(S_0) = \frac{N\hat{G}_n(S_0) + 1}{N + 1} \tag{3}$$

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

$$\hat{G}_N(x) = \frac{1}{N} \sum_{j=1}^{N} I_{[0,\infty]}(S_j - x) \quad \text{where } I_A(z) = \begin{cases} 1 \text{ if } z \in A \\ 0 \text{ if } z \notin A \end{cases} \tag{4}$$

Let α be the desired level for the test. Then, as detailed in **dufour˙monte˙2003** if $\alpha(N+1)$ is an integer, we have

$$P_{H_0}[\hat{p}_N(S_0) \leq \alpha] = \alpha \tag{5}$$

which simply implies that the following critical region

$$\hat{p}_N(S_0) \leq \alpha \tag{6}$$

has exact level equal to α.

It is important to note that $N\hat{G}_N(x)$ can be equivalently described as the number of simulated values of S bigger than x. Therefore, as described in **dufour˙monte˙2003** we have the following relationship

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

where $\hat{R}_N(S_0)$ represents the sample rank of S_0 in $S_0, S_1, ..., S_N$.

We can accordingly replace the following condition

$$\hat{p}_N(S_0) = \frac{N\hat{G}_n(S_0) + 1}{N + 1} \leq \alpha \tag{8}$$

by

$$\hat{p}_N(S_0) = \frac{N + 2 - \hat{R}_N(S_0)}{N + 1} \leq \alpha \tag{9}$$

As one might notice, if the test statistic is discrete, presence of ties in the sample ranks will lead to some difficulty in computing the associated p-value. Accordingly, **dufour˙monte˙2006** provides a way of "breaking" the ties using the uniform distribution.

Let $U_0, U_1, ..., U_N \overset{i.i.d}{\sim} U(0, 1)$, then for every S_i, we can create a pair (S_i, U_i). Using the following lexicographic preference relation

$$(S_i, U_i) \preccurlyeq (S_0, U_0) \iff \{S_i < S_0 \text{ or } (S_i = S_0 \text{ and } U_i \leq U_0)\} \tag{10}$$

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

$$\tilde{R}_N(S_0) = \sum_{i=0}^{N} I[(S_i, U_i) \preccurlyeq (S_0, U_0)] \tag{11}$$

where I is an indicator function for the preference relation.

Therefore, the p-value is now defined as

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

and the test is still exact if $\alpha(N + 1)$ is an integer, i.e.

$$P_{H_0}[\tilde{p}_N(S_0) \leq \alpha] = \alpha \tag{13}$$

for a detailed proof; see **dufour˙monte˙2006**

We can summarize the procedure as the following

Step 1 Compute the statistic S_0 using the sample data

Step 2 Generate N *i.i.d.* replications S_j of the statistic S under H_0

Step 3 Using the replications S_j, compute the following p-value

$$\tilde{p}_N(S_0) = \frac{N + 2 - \tilde{R}_N(S_0)}{N + 1} \tag{14}$$

where $\tilde{R}_N(S_0)$ is the randomized sample rank of S_0

Step 4 Test if $\tilde{p}_N(S_0) \leq \alpha$

This procedure is known as the MC with tie-breaker, and if the statistic is pivotal and $\alpha(N + 1)$ is an integer, the test is exact. Note that the distribution of the statistic is no longer required to be continuous. The MC with tie-breaker is implemented in the **MMC** package under the function name `mc`.

2.2 Maximized Monte Carlo

We now turn our attention to the more interesting case; when the statistic S depends on some nuisance parameter ν.

Let ν_0 be the true value of the nuisance parameter of S. Then, we wish to test

$$H_0 : \nu_0 \in \Omega_0 \tag{15}$$

where Ω_0 is the subset of parameter space Ω consistent with the null hypothesis. To solve this type of problem and retrieve an exact procedure, **dufour˙monte˙2006** proposed to modify the previous Monte Carlo technique 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

$$\hat{p}_N(S_0 \mid \nu) = \frac{N\hat{G}_N(S_0 \mid \nu) + 1}{N + 1} \tag{16}$$

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

$$\hat{G}_N(x \mid \nu) = \frac{1}{N} \sum_{j=1}^{N} I_{[0,\infty]}(S_j(\nu) - x) \quad \text{where } I_A(z) = \begin{cases} 1 \text{ if } z \in A \\ 0 \text{ if } z \notin A \end{cases} \tag{17}$$

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

$$\hat{Q}_N(S_0) = \sup_{\nu \in \Omega_0} \hat{p}_N(S_0 \mid \nu) \tag{18}$$

Step 5 Test if

$$\hat{Q}_N(S_0) \leq \alpha \tag{19}$$

If $\alpha(N+1)$ is an integer and the distribution of S is continuous, then under H_0 we have

$$P[\hat{Q}_n(S_0) \leq \alpha] = \alpha \tag{20}$$

which implies that the MMC procedure has exact level α for the following critical region

$$\hat{Q}_n(S_0) \leq \alpha \tag{21}$$

for a detailed proof; see **dufour˙monte˙2006**

This procedure is defined as the Maximized Monte Carlo (MMC) test and $\hat{Q}_n(S_0)$ as the MMC p-value.

Note that without relying on strong regularity assumptions, the MMC provides a simple method to obtain tests with valid inference even in the presence of nuisance parameters.

As for the MC with tie-breaker, it important to note that we can use ranks instead of directly using the survival function $\hat{G}_n(S_0 \mid \nu)$. Therefore, as previously shown, we can write $\hat{G}_n(S_0 \mid \nu)$ as

$$\hat{G}_n(S_0 \mid \nu) = \frac{N + 1 - \hat{R}_N(S_0 \mid \nu)}{N} \tag{22}$$

where $\hat{R}_N(S_0 \mid \nu)$ 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 $U_0, U_1, ..., U_N \overset{i.i.d}{\sim} U(0,1)$, then we can create the pairs (S_i, U_i) and use the following lexicographic preference relation

$$(S_i(\nu), U_i) \preccurlyeq (S_0, U_0) \iff \{S_i(\nu) < S_0 \text{ or } (S_i(\nu) = S_0 \text{ and } U_i \leq U_0)\} \tag{23}$$

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

$$\tilde{R}_N(S_0 \mid \nu) = \sum_{i=0}^{N} I[(S_i(\nu), U_i) \preccurlyeq (S_0, U_0)] \tag{24}$$

where I is an indicator function for the preference relation.

Thus, if we have that $\alpha(N + 1)$ is an integer and that we can simulate S then for

$$\hat{p}_N(S_0 \mid \nu) = \frac{N + 2 - \hat{R}_N(S_0 \mid \nu)}{N + 1} \tag{25}$$

we have that

$$P[\sup_{\nu \in \Omega_0} \tilde{p}_N(S_0 \mid \nu) \leq \alpha] = \alpha \tag{26}$$

which implies that the test is exact at level α; again for a proof, see **dufour˙monte˙2006**

One issue that arises with the MMC is that, as the size of Ω_0 increases, the method becomes computationally heavier. Thankfully, in the past decades, great improvements have been made to both computer hardware and global optimization algorithms. In fact, section 4 presents some of the global optimization algorithms that have been implemented in the **MMC** package in order to find the supremum of the p-value function. We can hope that future advances in computer science will render the MMC procedure an even more attractive option for any simulation-based test.

Regardless of any past or future improvments in the field of computer science, we can simplify the MMC procedure whenever a consistent point set estimate of ν is available and still maintain the validity of the inference of the test asymptotically. We simply reformulate the previous steps in the following way

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

$$\lim_{T \to \infty} P[\nu_0 \in C_T] = 1 \text{ under } H_0 \tag{27}$$

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 following p-value

$$\hat{p}_N(S_0 \mid \nu) = \frac{N\hat{G}_{TN}(S_0 \mid \nu) + 1}{N + 1} \tag{28}$$

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

$$\hat{G}_{TN}(x \mid \nu) = \frac{1}{N} \sum_{j=1}^{N} I_{[0,\infty]}(S_{Tj}(\nu) - x) \quad \text{where } I_A(z) = \begin{cases} 1 \text{ if } z \in A \\ 0 \text{ if } z \notin A \end{cases} \tag{29}$$

9

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

$$\hat{Q}_{TN}(S_0) = \sup_{\nu \in C_T} \hat{p}_{TN}(S_0 \mid \nu) \tag{30}$$

Step 6 Test if

$$\lim_{T \to \infty} \hat{Q}_{TN}(S_0) \leq \alpha \tag{31}$$

This procedure is referred to as the consistent set estimate MMC (CSEMMC), and again, if $\alpha(N+1)$ is an integer, then we have that

$$\lim_{T \to \infty} P[\hat{Q}_{TN}(S_0) \leq \alpha] = \alpha \tag{32}$$

i.e. that the test has an asymptotic level equal to α.

Finding a consistent estimator $\hat{\nu}$ is usually a simple matter. Methods such as the general method of moments (GMM) or the maximum likelihood estimation (MLE) can provide a consistent estimator. Then, any set of the following form

$$C_T = \{\nu \in \Omega : \| \hat{\nu}_T - \nu \| < d\} \tag{33}$$

where d is any fixed positive constant, will satisfy 27.

The constant d can be made as small as one wants. As one decreases d, the power of the test should increase, but a small d might prevent us from capture discontinuities in the distribution of statistic with respect to the nuisance paramater. We can push this argument even further and reduce the set to

$$C_T = \{\hat{\nu}_T\} \tag{34}$$

where $\hat{\nu}_T$ is a consistent estimator of ν. This is the analog to the parametric bootstrap, and we refer to this case as the Local Monte Carlo (LMC). The difference between the LMC and the simple parametric bootstrap lies in how the p-value is computed. The Monte Carlo technique p-value corrects for the finite-sampleness of the statistic. Note that in order to satisfy 27, we need stronger regularity assumptions on $\hat{\nu}_T$. Therefore, the parametric bootstrap or the LMC are only valid asymptotically under some regularity conditions.

Now, instead of choosing the constant d arbitrarily, for some models, it is possible to derive exact confidence intervals for the nuisance parameters that will satisfy 27. 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.

$$P[\nu_0 \in C_{\nu_0}(\alpha_1)] = 1 - \alpha_1 \text{ under } H_0 \tag{35}$$

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

$$\hat{p}_N(S_0 \mid \nu) = \frac{N\hat{G}_N(S_0 \mid \nu) + 1}{N + 1} \tag{36}$$

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

Step 5 Maximize the p-value function $\hat{p}_N(S_0 \mid \nu)$ over the set $\nu \in C_{\nu_0}(\alpha_1)$ i.e.

$$\hat{Q}_N(S_0) = \sup_{\nu \in C_{\nu_0}(\alpha_1)} \hat{p}_N(S_0 \mid \nu) \tag{37}$$

Step 6 Test if

$$\hat{Q}_N(S_0) \leq \alpha_2 \tag{38}$$

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

$$P[\hat{Q}_N(S_0) \leq \alpha_2] = \alpha_1 + \alpha_2 = \alpha \tag{39}$$

which states that the MMC procedure has exact level $\alpha_1 + \alpha_2 = \alpha$; for more details, see **dufour˙exact˙1996** and **dufour˙monte˙2006**

Thus, if we have a way to derive an exact confidence interval for the nuisance parameter, this two-stage method has both the advantage of reducing the computational load, as the CSEMMC, and conserving the exactness of the test. Usually for a given level α, α_1 and α_2 are chosen such that $\alpha_1 = \alpha_2 = \frac{\alpha}{2}$.

Finally, we turn our attention to the problem of simulating S for a given ν. In a parametric model, it can simulate the statistic by first generating new observations y^* according to some data generating process of the following form

$$y^* = g(\nu, u) \tag{40}$$

where $u = (u_1, ...u_n)$ is a vector of $i.i.d.$ random variables with known distribution and ν is the nuisance parameter vector, and then, computing the statistic on this new vector of observation.

This setup can be extended, as described in **dufour˙monte˙2006** to allow for non-parametric models. First, we assume that the null distribution of S depends on some transformation $T(\cdot)$ of the original observation vector y. In turn, if we assume that $T(y)$ depends on ν, then $S(T(y))$ will depend indirectly on ν. We can extend it further by allowing first $T(y)$ to depend on some $\bar{\nu}$, that in turns depends on some transformation $h(\nu) = \bar{\nu}$. Therefore, we get the following setup

$$y^* = T(y) = g[h(\nu), u] = g[\bar{\nu}, u] \quad \bar{\nu} \in \bar{\Omega} \tag{41}$$

where $\bar{\Omega} = h(\Omega)$. This formulation allows for us to generate S regardless if S is a parametric or a non-parametric statistic. Additionally, the function $h(\cdot)$ allows us to reduce the nuisance parameter set (e.g. through invariance of S) which can help us reduce the cost of the MMC in terms of computational load.

11

3 Implementation

MMC is available on GitHub. In order to install it, simply make sure **devtools** (**wickham˙devtools:˙2016**) is installed, and run the following command lines

```
require(devtools)
install_github("julienneves/MMC")
```

In the next sections, we look more closely at the two functions at the core of **MMC** package: `mc` and `mmc`.

3.1 MMC::mc

The `mc` function implements the Monte Carlo technique with tie-breaker as described in the previous section. The function call is reproduced here for simplicity.

```
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:

y: A vector or data frame.

statistic: A function. Specifies how the statistic is computed.

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

dgp: A function. The function inputs the first argument `y`, and outputs a simulated `y` (i.e. an object of the same type as `y`). It should represent the data generating process under the null. Default value is the function sample(y, replace = TRUE), the bootstrap resampling of `y`.

N: An atomic vector. Number of replications of the test statistic.

type: A character string. Specifies the type of test the p-value function produces. The possible values are `geq`, `leq`, `absolute` and `two-tailed`. Default is `geq`.

We allow for four types of p-value: `leq`, `geq`, `absolute` and `two-tailed`. For one-tailed test, `leq` returns the proportion of simulated values smaller than the statistic while `geq` return the proportion of simulated values greater than the statistic. For two-tailed test, if the statistic is symmetric, one can use the absolute value of the statistic and its simulated value to retrieve a two-tailed test. If the statistic is not symmetric, one can specify the p-value type as

12

`two-tailed` to obtained the p-value computed as twice the minimum of `leq` and `geq` i.e.

$$\tilde{p}_N(S_0 \mid \nu) = 2 \cdot \min\left(\frac{N\tilde{F}_N(S_0 \mid \nu) + 1}{N + 1}, \frac{N\tilde{G}_N(S_0 \mid \nu) + 1}{N + 1}\right) \tag{42}$$

where

$$\tilde{F}_N(S_0 \mid \nu) = 1 - \tilde{G}_N(S_0 \mid \nu) \tag{43}$$

The returned value is an object of class mc", containing the following components:

`S0`: Observed value of `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

3.1.1 Example: Goodness-of-fit test

To demonstrate how the `mc` function can be used, we look at the problem of testing for the equality of two unknown distributions. Let $X_1, ..., X_n$ and $Y_1, ..., Y_m$ be *i.i.d.* observations such that $F(x) = P(X_i \leq x)$ is the cumulative distribution function of X_i and $G(y) = P(Y_j \leq y)$ is the cumulative distribution function of Y_j. Then, we we wish to test

$$H_0 : F = G \tag{44}$$
$$H_1 : F \neq G \tag{45}$$

To solve this problem, we use the Kolmogorov-Smirnov statistic (**smirnoff˙sur˙1939**; **smirnov˙table˙1948**) defined as follows

$$KS = \sup_x \mid \hat{F}_n(x) - \hat{G}_m(x) \mid \tag{46}$$

where

$$\hat{F}_n(x) = \frac{1}{n}\sum_{i=1}^{n} I_{[-\infty, x]}(X_i) \quad \text{where } I_A(z) = \begin{cases} 1 \text{ if } z \in A \\ 0 \text{ if } z \notin A \end{cases} \tag{47}$$

13

and

$$\hat{G}_n(y) = \frac{1}{m} \sum_{j=1}^{m} I_{[-\infty,y]}(Y_j) \quad \text{where } I_A(z) = \begin{cases} 1 \text{ if } z \in A \\ 0 \text{ if } z \notin A \end{cases} \qquad (48)$$

are the empirical distribution functions of X and Y.

It is important to note that the KS is distribution free when the observations are *i.i.d.* and follow a continuous distribution, but this property disappears if the observations follow a discrete distribution. Therefore, in order to retrieve an exact test even when the X_i or Y_i follow a discrete distribution, **dufour˙exact˙2001** demonstrate that the MC procedure with tie-breaker can be fruitful.

In this example, we let $n = m = 8$, $X_i \sim Poisson(4)$, and $Y_j \sim Poisson(4)$.

```r
library(mmc)

# Generate x ~ Poisson(4), y ~ Poisson(4)
x <- rpois(8, lambda = 4)
y <- rpois(8, lambda = 4)
data <- list(x = x, y = y)

# Set data generating process function
dgp <- function(data) {
    x <- rpois(length(data$x), lambda = 4)
    y <- rpois(length(data$y), lambda = 4)
    return(list(x = x, y = y))
}

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

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

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

##
##  Two-sample Kolmogorov-Smirnov test
##
## data:  data$x and data$y
## D = 0.375, p-value = 0.6272
## alternative hypothesis: two-sided
```

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

##
## Monte Carlo with Tie-Breaker
##
## Call:
## mc(y = data, statistic = statistic, dgp = dgp, N = 999, type = "geq")
##
## D = 0.375, N = 999, p-value = 0.656
```

Note that we use the function `suppressWarnings` in order to prevent the generation of warnings messages every time the `mc` function calls the `statistic` function.

As we can see from this quick example, the function `mc` is easily implementable, and permits us to retrieve an exact test, even in the presence of ties, using already available statistics in R.

3.2   `mmc::mmc`

The `mmc` function implements the Maximized Monte Carlo 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. Specifies how the statistic is computed. The function inputs the y, and outputs a scalar numeric.

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

dgp: A function. The function inputs the first argument y and a vector of nuisance parameters v, and outputs a simulated y (i.e. an object of the same type as y). It is important to respect the order of the arguments. It should represent the data generating process under the null. The function can be specified without any nuisance parameters (i.e. without v). Default value is sample(y, replace = TRUE), the bootstrap resampling of y.

est: A vector with length of v. It is the starting point of the algorithm. If `est` is a consistent estimate of v then `mmc` will return both the mmc and LMC. Default is NULL, in which case, default values will be generated automatically.

lower: A vector with length of v. Lower bounds for nuisance parameters under the null

upper: A vector with length of v. Upper bounds for nuisance parameters under the null

N: An atomic vector. Number of replications of test statistic.

type: A character string. Specifies the type of test the p-value function produces. The possible values are `geq`, `leq`, `absolute` and `two-tailed`. Default is `geq`.

method: A character string. Type of algorithm to be used for global optimization. The four available methods are grid search (`gridSearch`), 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`, then the algorithm will stop. This is particularly useful if we are only looking at testing a hypothesis at a particular level. Default is NULL.

Again, the `dgp` function defined by the user is used to generate new observations in order to compute the simulated statistics. The `statistic` and `ran.gen` functions form the building blocks of our procedure. It is thus essential that the functions be written efficiently in order for `mmc` to find the Maximized Monte Carlo p-value rapidly. One possible solution, if the `mmc` function is slow to converge, is to write the user specified functions `statistic` and `ran.gen` in C.

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

S0: Observed value of `statistic`

p.value: Maximized Monte Carlo p-value of `statistic` under null

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 specified in call

16

upper: `upper` vector 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 `est` 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 if the hypothesis was rejected or not at level `alpha`

3.2.1 Global Optimization

Multiple methods are available 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 faster or more efficient algorithms.

Note that one issue we have to face when choosing an algorithm is that the p-value function

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

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

3.2.1.1 gridSearch

The grid search method is the easiest to implement and understand but sadly one of the least efficient. Let Ω^* be the space where we maximize the nuisance parameter $\nu = (\nu_1, \nu_2, ...\nu_m)$ of size m. If we parametrize Ω^* such that $a = (a_1, a_2, ..., a_m)$ be the lower bound and $b = (b_1, b_2, ..., b_m)$ be the upper bound of set where we maximize ν, then the grid search method involves taking n equally spaced points in each interval of the form $[a_i, b_i]$ including a_i and b_i. Combining all these points, we have n^m possible "pairs" where we evaluate the function we which to maximize and find the point that gives the best value.

The problem with this type of method is that as n and m increase, the number of evaluations increase exponentially. Since we cannot really reduce

m, decreasing n 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·numerical·2011**) provides the function `gridSearch` that implements this method. It has the distinctive advantage of providing an easy way to parallelize the problem using the package **parallel** by **r·core·team·r:·2016** This could resolve some of the computing uses associated with the grid search method, but we do not recommend this method in general especially if $m \geq 3$. By default, we set $n = 10$, but we urge the reader to modify this parameter to an appropriate number depending on the problem.

3.2.1.2 GenSA

The simulated annealing is a stochastic, metaheuristic technique that is able to find "good" solution to a global optimization problem even in the presence of multiple local minima. It was originally and independently proposed by **kirkpatrick·optimization·1983** and by **cerny·thermodynamical·1985**

The 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 in x or move to x'. As the algorithm progresses, the probability of moving will converge to 0. This is done through global time-varying parameter T that follows an annealing schedule called the "temperature". Note that usual definition of the algorithm sets the probability of moving equal to 1 when x' is a better than x but it is not necessary for convergence.

The main feature of simulated annealing is that it might accept a worse solution and hence preventing the algorithm from being stuck in a local minimum.

The simulated annealing method is implemented using the **GenSA** package described in **yang·xiang·generalized·2013** The package **stats** available in the base distribution of R also provides the function `optim` which implements the simulated annealing.

As shown in **mullen·continuous·2014** the **GenSA** package outperforms the function `optim` in terms of convergence and speed. Therefore, the choice was easy for our package.

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 **yang·xiang·generalized·2013** and the documentation for **MMC** and **GenSA**.

3.2.1.3 psoptim

Particle swarm optimization was introduced by **eberhart·new·1995** and **shi·modified·1998** As the simulated annealing the method 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 search. The direction and the velocity associated to the particles are guided toward both the best known position for the individual particle and the best known overall position. As the number of

iterations increase, so will the convergence to the best known position for the entire population.

Multiple packages like **pso** (**bendtsen˙pso:˙2012**), **hydroPSO** (**zambrano-bigiarini˙hydropso:˙2014**) and **psoptim** (**ciupke˙psoptim:˙2016**) implement the particle swarm optimization technique in R. For the package **MMC**, we selected the **pso** package and its function `psoptim` (note that both **psotim** and **pso** chose to name their function `psoptim` which might be a bit confusing).

When comparing **pso** and **hydroPSO**, **mullen˙continuous˙2014** found that **pso** perfomed a bit better. At the time **mullen˙continuous˙2014** was published, **psoptim** was not avaiable, but if we compare it to **pso** the controls available are more restricted with **psoptim**.

As for the simulated annealing method, 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 such are set, see **bendtsen˙pso:˙2012** and the documentation for **MMC** and **GenSA**.

3.2.1.4   GA

Genetic algorithm is a subclass of evolutionary algorithm techniques. The technique dates back to the 70s (**holland˙adaptation˙1992**). As the name suggests, the evolutionary algorithms base themselves on the natural selection present in nature, where only the fittest individual survives through the process of mutation, selection, and crossover. For genetic algorithm, each indivual has a set of properties (or genes) that can be mutated or altered until we find the best solution.

The following packages offer some sort of implementation of the genetic algorithm:

- **soma** (**zelinka˙soma:˙2014**)

- **GA** (**scrucca˙ga:˙2013**; **scrucca˙extensions˙2016**)

- **genalg** (**willighagen˙genalg:˙2015**)

- **mcga** (**satman˙machine˙2013**)

- **NMOF** (**gilli˙numerical˙2011**)

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

3.2.2   Example: Behrens-Fisher problem

In order to demonstrate how to apply the `mmc` function and how the different optimization methods compare, we present a simple example where we test for the equality of the means of two normal populations with unknown and not necessarily equal variances. This is known as the Behrens-Fisher (BF)

problem; for more details, see **fisher˙fiducial˙1935 fisher˙asymptotic˙1941** and **behrens˙beitrag˙1929**

Let $X_{11}, ... X_{1n_1} \sim N(\mu_1, \sigma_1^2)$ and $X_{21}, ... X_{2n_2} \sim N(\mu_2, \sigma_2^2)$, we are looking to test the following

$$H_0 : \mu_1 - \mu_2 = 0 \tag{50}$$
$$H_1 : \mu_1 - \mu_2 \neq 0 \tag{51}$$

One solution to this problem is to use the extension of Student's t-test proposed by **welch˙generalization˙1947** for unequal variances with the following form

$$t = \frac{\bar{X}_1 - \bar{X}_2}{\sqrt{\frac{s_1^2}{n_1} + \frac{s_2^2}{n_2}}} \tag{52}$$

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 depends on the following nuisance parameters μ_2, σ_1 and σ_2

welch˙significance˙1938 showed that the distribution of this statistic can be approximated with a Student's t-distribution with the following number of degrees of freedom

$$df \approx \frac{\left(\bar{X}_1 - \bar{X}_2\right)^2}{\frac{s_1^4}{n_1^3 - n_1^2} + \frac{s_2^4}{n_2^3 - n_2^2}} \tag{53}$$

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

Therefore, as previously mentioned, an easy way to retrieve an exact test is to apply the MMC procedure. The following R code excerpt shows how the BF problem can be implemented for $X_{1i} \sim N(0,1)$ with $n_1 = 15$ and $X_{2i} \sim N(0,100)$ with $n_2 = 25$

```
library(mmc)
library(MASS)

# Generate x1 ~ N(0,1) and x2 ~ N(0,100)
x1 <- rnorm(15, mean = 0, sd = 1)
x2 <- rnorm(25, mean = 0, sd = 10)
data <- list(x1 = x1, x2 = x2)

# Fit the sample using maximum likelihood
fit1 <- fitdistr(x1, "normal")
fit2 <- fitdistr(x2, "normal")

# Set the estimator value
est <- c(fit2$estimate, fit1$estimate["sd"])
```

```r
# Set the bounds such that we have a 99% CI
lower <- est - 2.577 * c(fit2$sd, fit1$sd["sd"])
upper <- est + 2.577 * c(fit2$sd, fit1$sd["sd"])

# Set the DGP function under the null (i.e.
# mu_1=mu_2) where v = (mu_2,s_2_s1)
dgp <- function(data, v) {
    x1 <- rnorm(length(data$x1), mean = v[1], sd = v[3])
    x2 <- rnorm(length(data$x2), mean = v[1], sd = v[2])
    return(list(x1 = x1, x2 = x2))
}

# Set the statistic function
statistic <- function(data) {
    test <- t.test(data$x2, data$x1)
    return(test$statistic)
}

# Apply the test statistic
t.test(data$x2, data$x1)

##
##  Welch Two Sample t-test
##
## data:  data$x2 and data$x1
## t = -0.69063, df = 25.615, p-value = 0.496
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -4.001754  1.990067
## sample estimates:
##  mean of x  mean of y
## -0.8528617  0.1529818

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

##
## Maximized Monte Carlo
##
## Call:
## mmc(y = data, statistic = statistic, dgp = dgp, est = est, lower = lower,
##      upper = upper, N = 99, type = "absolute", method = "pso")
##
## t = -0.69063, N = 99
```

```
## Local Monte Carlo: p-value = 0.71
## Maximized Monte Carlo: p-value = 0.82
```

Note that we use the **MASS** package (**venables˙modern˙2002**) to derive the 99% confidence intervals for $\bar{X}_2$, s_1, s_2 using maximum likelihood, but any other technique to derive a consistent set of μ_2, σ_1 and σ_2 would be satisfactory.

The `t.test` from the **stats** (**r˙core˙team˙r:˙2016**) implements Welch's test by default therefore no additional arguments have to be specified.

As one can see, the `mmc` function is fairly straightforward to use and its power comes from the its malleability to any type of problem.

If we return to the topic of global optimization, the choice of the method to use is not always straightforward. For a starting point the BF problem can be replicated for every method and then we can compare how much time it takes each method before reaching its stopping points. In order to do so, we use **microbenchmark** (**mersmann˙microbenchmark:˙2015**) and its plot method that requires **ggplot2** (**wickham˙ggplot2:˙2009**).
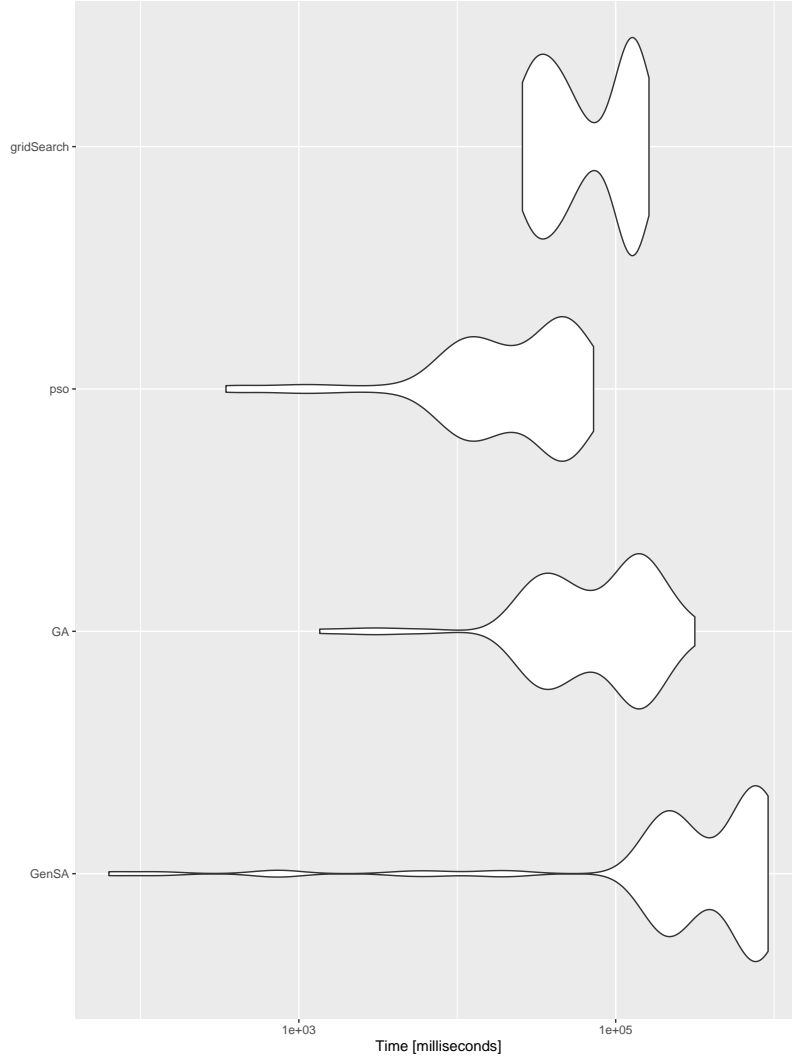
Figure 1: Density plot for evaluation time of mmc(y = data, statistic = statistic, dgp = dgp, est = est, lower = lower, upper = upper, N = 99, type = "absolute") with different global optimization algorithms

As we can see, the `pso` seems to provide the most efficient method in this case. The `GenSA` might take on average the longest time for this example, but we also note that in some cases it finds a solution quite rapidly. Moreover, the grid search method might seem to perform well since it only evaluates the function at 10^3 points. If we double the accuracy, the time needed would increase by a factor of 8 which would make this method ineffective compared to others. Additionally, we did not use any parallelization in the controls of the methods. As such, `GA`

and `gridSearch` performance could have been improved quite significantly.

Finally, since the methods are metaheuristic, this figure is sadly not enough for us to make any comment on how efficient one method is compared to another in finding a "good" solution. In the end, the choice of method might boil down to personal preference. What is important regarding the **MMC** package is that it is able to provide any tools needed to solve any problem thrown at it.

4   Irregular Wald tests

In this section, we look at the problem of Wald-type tests where the regularity conditions are violated. This situation arises when the covariance matrix of the restrictions, as described in **dufour˙wald˙2013; dufour˙rank-robust˙2016** does not have full column rank in an open neighborhood of the true value of the parameter vector.

4.1   Framework

Let $\theta = [\theta_1, ..., \theta_p]'$ and $\psi(\theta) = [\psi_1(\theta), ..., \psi_q(\theta)]'$ where $q \leq p$, then we are looking to test the following set of hypotheses

$$H_0 : \psi(\theta) = \psi_0 \tag{54}$$

$$H_1 : \psi(\theta) \neq \psi_0 \tag{55}$$

Let $\bar{\theta}$ be the true parameter value, we assume that $\psi(\theta) = [\psi_1(\theta), ..., \psi_q(\theta)]'$ is continuously differentiable on Θ where Θ is an open subset of $\mathbb{R}^p$ where $\bar{\theta} \in \Theta$.

Then the usual Wald statistic is defined as follows

$$W_n(\psi_0) = n \left[\psi(\theta) - \psi_0 \right]' \Sigma_\psi^{-1} \left[\psi(\theta) - \psi_0 \right] \tag{56}$$

where Σ_ψ is the covariance matrix of the restriction ψ.

Usually θ and Σ_ψ need to be estimated and the Wald statistic is defined as

$$W_n(\psi_0) = n \left[\psi(\hat{\theta}) - \psi_0 \right]' \hat{\Sigma}_\psi^{-1} \left[\psi(\hat{\theta}) - \psi_0 \right] \tag{57}$$

Note, as described in **dufour˙wald˙2013; dufour˙rank-robust˙2016** we can have two problematic situations:

1. $\hat{\Sigma}_\psi$ has full rank, but convergences to a singular Σ_ψ

2. $\hat{\Sigma}_\psi$ does not have full rank, but $rank[\hat{\Sigma}_\psi] \neq rank[\Sigma_\psi]$

Now, note that if we assume

$$\sqrt{n}(\hat{\theta_n} - \theta) \xrightarrow{n \to \infty} N(0, I) \tag{58}$$

Then, $\hat{\Sigma}_\psi$ has form

$$\hat{\Sigma}_\psi = P(\hat{\theta})\hat{\Sigma}_\theta P(\hat{\theta})' \tag{59}$$

where $P(\theta) = \frac{\partial \psi}{\partial \theta'}$

Therefore, the Wald statistic can be rewritten as follows

$$W_n(\psi_0) = n \left[\psi(\hat{\theta}) - \psi_0 \right]' \left[P(\hat{\theta}) \hat{\Sigma}_\theta P(\hat{\theta})' \right]^{-1} \left[\psi(\hat{\theta}) - \psi_0 \right] \tag{60}$$

4.2 Example 1

This example is given by **dufour`rank-robust`2016** It is a simple example of two restrictions, where we test

$$H_0 : [\theta_1 \theta_2, \theta_1]' = 0 \tag{61}$$
$$H_1 : [\theta_1 \theta_2, \theta_1]' \neq 0 \tag{62}$$

It is easy to see that this is a simple case of redundant restrictions. In fact, both restrictions amount to testing that $\theta_1 = 0$.

In order to simplify the analysis of this example, we let $\hat{\Sigma}_\theta = \Sigma_\theta = I$.

Then, we have the following

$$P(\theta) = \begin{bmatrix} \theta_2 & \theta_1 \\ 1 & 0 \end{bmatrix} \tag{63}$$

and

$$P(\hat{\theta}) \Sigma_\theta P(\hat{\theta})' = \begin{bmatrix} \hat{\theta}_1^2 + \hat{\theta}_2^2 & \hat{\theta}_2 \\ \hat{\theta}_2 & 1 \end{bmatrix} \tag{64}$$

Then, we have

$$P(\hat{\theta}) \Sigma_\theta P(\hat{\theta})' \xrightarrow{n \to \infty} P(\theta) \Sigma_\theta P(\theta)' \tag{65}$$

where $P(\theta) \Sigma_\theta P(\theta)'$ is singular.

Taking the usual Wald statistic we have the following

$$W_n = n\hat{\theta}_1^2 \xrightarrow{n \to \infty} \chi_1^2 \tag{66}$$

Consequently, compared to the usual χ^2 distribution of the W_n, the true asymptotic distribution is then conservative.

4.2.1 Simulation study

4.2.1.1 Level

First, we start by simulating $\hat{\theta}_1 \sim N(\theta_1, \frac{1}{n})$, $\hat{\theta}_2 \sim N(\theta_2, \frac{1}{n})$, where n is the sample size and $\theta_1 = 0$, for different values of θ_2. The empirical levels for the Wald statistic are reported in table 1. Note that since the distribution of the statistic is not dependent on any nuisance parameter, the MMC is not directly applicable. Therefore, we only report the result for the asymptotic test and the LMC.

25

Size	$\theta_2 = 0$		$\theta_2 = \frac{1}{4}$		$\theta_2 = \frac{1}{2}$		$\theta_2 = 1$		$\theta_2 = 5$	
	Asy	LMC	Asy	LMC	Asy	LMC	Asy	LMC	Asy	LMC
15	2.4	6.0	1.2	5.6	1.6	5.6	0.8	4.0	1.6	4.0
25	2.8	6.0	1.2	4.0	1.6	3.6	2.0	6.8	2.8	6.8
50	1.2	4.4	1.2	3.6	1.2	4.8	2.0	2.8	1.6	8.0
100	1.6	4.4	2.4	6.4	0.8	3.6	0.8	5.2	2.4	6.0
150	2.8	4.8	2.4	4.8	1.6	5.2	0.8	4.8	1.2	3.6
200	1.2	5.2	0.8	4.0	3.2	5.2	0.8	4.8	2.4	5.2
250	0.4	6.0	0.8	4.4	1.6	4.4	0.4	2.4	1.6	4.4
300	2.8	5.2	4.0	5.6	0.8	3.6	0.8	4.0	1.6	5.2

Table 1: Empirical levels for 1000 replications of the Wald statistic where $\alpha = 5\%$ and $\theta_1 = 0$, testing $H_0 : [\theta_1\theta_2, \theta_1] = 0$ against $H_1 : [\theta_1\theta_2, \theta_1] \neq 0$

As expected, the empirical level shows that the asymptotic test under-rejects in all cases. The LMC result seems to oscillate around the 5% nominal level, but sometimes over-rejecting (e.g. 8% in the case of $\theta_2 = 5$ and $n = 50$).

4.2.1.2 Power

Now, we test for the power of the Wald statistic based on χ_2^2 critical values and the LMC, by simulating the test statistic under the alternative by picking different values for $\theta_1 \neq 0$ (e.g. $\theta_1 = \frac{1}{4}, \frac{1}{2}, 1$). The empirical levels are reported in table 2, 3 and 4.

Size	$\theta_2 = 0$		$\theta_2 = \frac{1}{4}$		$\theta_2 = \frac{1}{2}$		$\theta_2 = 1$		$\theta_2 = 5$	
	Asy	LMC	Asy	LMC	Asy	LMC	Asy	LMC	Asy	LMC
15	6.8	16.0	10.0	18.0	8.0	16.4	6.8	15.6	6.8	16.4
25	12.8	24.4	10.8	18.0	11.6	24.8	12.4	24.0	11.2	27.2
50	25.2	41.6	20.4	36.4	20.8	42.4	24.0	43.6	27.2	42.8
100	52.0	72.8	52.0	64.8	46.8	66.8	50.4	64.4	50.0	66.8
150	73.2	84.8	78.4	87.2	72.4	85.2	71.6	85.2	77.2	89.2
200	88.0	92.8	88.0	94.8	84.4	93.2	82.0	90.0	87.2	91.2
250	95.2	98.8	92.0	96.8	91.6	96.8	93.2	97.2	93.6	96.4
300	97.2	98.8	98.4	99.2	95.6	98.0	97.6	99.2	96.0	98.8

Table 2: Empirical levels for 1000 replications of the Wald statistic where $\alpha = 5\%$ and $\theta_1 = \frac{1}{4}$, testing $H_0 : [\theta_1\theta_2, \theta_1] = 0$ against $H_1 : [\theta_1\theta_2, \theta_1] \neq 0$

	$\theta_2 = 0$		$\theta_2 = \frac{1}{4}$		$\theta_2 = \frac{1}{2}$		$\theta_2 = 1$		$\theta_2 = 5$	
Size	Asy	LMC	Asy	LMC	Asy	LMC	Asy	LMC	Asy	LMC
15	32.4	44.0	34.4	54.8	32.8	50.8	26.0	44.0	29.6	52.0
25	44.8	62.8	53.2	72.0	53.2	72.0	54.0	70.8	58.4	67.6
50	80.8	90.0	82.0	92.8	90.4	96.8	86.4	92.8	92.0	96.0
100	100.0	100.0	100.0	100.0	99.6	100.0	100.0	100.0	99.6	100.0
150	100.0	100.0	100.0	100.0	100.0	100.0	100.0	100.0	100.0	100.0
200	100.0	100.0	100.0	100.0	100.0	100.0	100.0	100.0	100.0	100.0
250	100.0	100.0	100.0	100.0	100.0	100.0	100.0	100.0	100.0	100.0
300	100.0	100.0	100.0	100.0	100.0	100.0	100.0	100.0	100.0	100.0

Table 3: Empirical levels for 1000 replications of the Wald statistic where $\alpha = 5\%$ and $\theta_1 = \frac{1}{2}$, testing $H_0 : [\theta_1 \theta_2, \theta_1] = 0$ against $H_1 : [\theta_1 \theta_2, \theta_1] \neq 0$

	$\theta_2 = 0$		$\theta_2 = \frac{1}{4}$		$\theta_2 = \frac{1}{2}$		$\theta_2 = 1$		$\theta_2 = 5$	
Size	Asy	LMC	Asy	LMC	Asy	LMC	Asy	LMC	Asy	LMC
15	94	98.4	91.6	98	94.8	97.2	94.0	99.2	91.6	96.4
25	100	100.0	99.6	100	99.6	100.0	99.2	99.6	99.6	99.6
50	100	100.0	100.0	100	100.0	100.0	100.0	100.0	100.0	100.0
100	100	100.0	100.0	100	100.0	100.0	100.0	100.0	100.0	100.0
150	100	100.0	100.0	100	100.0	100.0	100.0	100.0	100.0	100.0
200	100	100.0	100.0	100	100.0	100.0	100.0	100.0	100.0	100.0
250	100	100.0	100.0	100	100.0	100.0	100.0	100.0	100.0	100.0
300	100	100.0	100.0	100	100.0	100.0	100.0	100.0	100.0	100.0

Table 4: Empirical levels for 1000 replications of the Wald statistic where $\alpha = 5\%$ and $\theta_1 = 1$, testing $H_0 : [\theta_1 \theta_2, \theta_1] = 0$ against $H_1 : [\theta_1 \theta_2, \theta_1] \neq 0$

It is clear from tables 2, 3 and 4, that the simulation based method has a consistently higher power. This is as expected, since the asymptotic test based on χ^2_2 critical values should be conservative, therefore reducing both the level and power of the test.

Additionally, we note that as θ_1 diverges from the null hypothesis (i.e. $\theta_1 = 0$), the power of both test increase.

4.3 Example 2

The following example is given by **dufour˙wald˙2013** We are looking to test the following set of hypotheses

$$H_0 : \theta_1^2 + ... + \theta_p^2 = 0 \tag{67}$$

$$H_1 : \theta_1^2 + ... + \theta_p^2 \neq 0 \tag{68}$$

Again, we let $\hat{\Sigma}_\theta = \Sigma_\theta = I$. Then, under H_0, we have $\theta_1 = ... = \theta_p = 0$ and the following derivative matrix

$$P(\theta) = [2\theta_1, ..., 2\theta_p] \tag{69}$$

Hence, the asymptotic covariance matrix of the restriction reduces to

$$P(\hat{\theta})\hat{\Sigma}_\theta P(\hat{\theta})' = 4(\hat{\theta}_1^2 + ... + \hat{\theta}_p^2) \tag{70}$$

where

$$P(\hat{\theta})\hat{\Sigma}_\theta P(\hat{\theta})' \xrightarrow{n\to\infty} 0 \tag{71}$$

Therefore, the regularity condition is violated and we have the following Wald statistic

$$W_n = \frac{n}{4}(\hat{\theta}_1^2 + ... + \hat{\theta}_p^2) = \frac{n}{4}\sum_{i=1}^{p}\hat{\theta}_i^2 \tag{72}$$

It is then easy to see that

$$W_n \xrightarrow{n\to\infty} \frac{1}{4}\chi_p^2 \tag{73}$$

which implies that the test might lead to under-rejection for low p and severe over-rejection for high p.

4.3.1 Simulation study

4.3.1.1 Level

We simulate $\hat{\theta}_i \sim N(0, \frac{1}{n})$ for $i = 1, ...p$, where p is the number of parameters in the restriction. The empirical levels are reported in table 5.

Size	$p=2$ Asy	LMC	$p=5$ Asy	LMC	$p=10$ Asy	LMC	$p=20$ Asy	LMC	$p=30$ Asy	LMC
15	0	1.2	0	2.8	0.4	2.4	24.0	3.2	75.2	1.2
25	0	4.0	0	1.6	0.8	3.6	18.4	1.6	78.4	3.2
50	0	2.4	0	2.4	1.6	3.2	23.6	2.8	79.2	4.8
100	0	2.4	0	3.6	0.4	2.4	25.6	4.8	76.8	2.4
150	0	2.8	0	2.0	1.2	2.8	21.6	2.8	76.4	3.6
200	0	2.0	0	2.8	0.4	2.8	24.8	2.0	72.4	5.6
250	0	2.8	0	1.6	0.0	0.8	18.8	3.2	74.8	5.2
300	0	2.4	0	2.8	0.0	4.4	22.8	4.8	77.2	4.8

Table 5: Empirical levels for 1000 replications of the Wald statistic where $\alpha = 5\%$, $\theta_1 = 0$ and $\theta_2 = ... = \theta_p = 0$, testing $H_0 : \theta_1^2 + ... + \theta_p^2 = 0$ against $H_1 : \theta_1^2 + ... + \theta_p^2 \neq 0$

First, looking at the LMC, we see that we almost always under reject. More precisely, the empirical levels range from 1.2% to 5.6% for the LMC.

The really interesting part of this example is how the asymptotic test starts off by severely under-rejecting for low values of p (e.g. $p = 2, 5, 10$), and then starts to over-reject as p increases. In fact, for $p = 30$, the asymptotic test empirical levels are around 75% for a 5% level test. While in the previous example one might have chosen the asymptotic test without any severe consequence on the level of the test, in this example it is quite the opposite.

4.3.1.2   Power

In this section, we let θ_1 vary from the null hypothesis that $\theta_1 = 0$. Results for the empirical levels when $\hat{\theta}$ is generated under the alternative are reported in tables 6, 7 and 8.

	$p = 2$		$p = 5$		$p = 10$		$p = 20$		$p = 30$	
Size	Asy	LMC	Asy	LMC	Asy	LMC	Asy	LMC	Asy	LMC
15	0.0	7.2	0.0	5.2	2.4	6.0	31.6	5.6	81.2	4.8
25	0.0	12.4	0.0	8.4	2.8	8.4	28.4	5.6	82.4	8.4
50	0.0	16.8	0.8	16.0	7.6	14.8	38.0	6.0	88.0	7.6
100	2.0	42.0	2.8	33.6	11.2	26.8	57.2	18.4	90.8	18.8
150	4.4	61.2	7.6	53.6	23.6	47.2	71.2	33.6	96.0	29.2
200	7.6	80.8	17.2	66.0	39.2	59.2	80.8	45.2	98.4	39.6
250	22.4	90.0	33.2	82.8	59.6	75.2	90.8	66.4	99.2	47.6
300	32.8	96.4	44.8	88.4	70.0	85.2	92.8	63.6	100.0	60.4

Table 6: Empirical levels for 1000 replications of the Wald statistic where $\alpha = 5\%$, $\theta_1 = \frac{1}{4}$ and $\theta_2 = ... = \theta_p = 0$, testing $H_0 : \theta_1^2 + ... + \theta_p^2 = 0$ against $H_1 : \theta_1^2 + ... + \theta_p^2 \neq 0$

	$p = 2$		$p = 5$		$p = 10$		$p = 20$		$p = 30$	
Size	Asy	LMC	Asy	LMC	Asy	LMC	Asy	LMC	Asy	LMC
15	0.0	27.6	1.2	20.8	6.4	16.8	44.0	11.2	88.0	10.4
25	0.4	41.6	1.6	34.4	12.4	25.2	59.6	22.8	91.6	17.2
50	11.2	82.0	18.8	73.2	38.4	60.4	78.8	43.2	98.0	39.2
100	56.0	97.6	68.8	96.0	83.6	92.4	98.8	85.6	100.0	78.8
150	88.0	99.6	93.6	100.0	97.2	99.6	100.0	97.2	100.0	96.8
200	99.2	100.0	98.8	100.0	100.0	100.0	100.0	99.6	100.0	99.6
250	99.6	100.0	100.0	100.0	100.0	100.0	100.0	100.0	100.0	100.0
300	100.0	100.0	100.0	100.0	100.0	100.0	100.0	100.0	100.0	100.0

Table 7: Empirical levels for 1000 replications of the Wald statistic where $\alpha = 5\%$, $\theta_1 = \frac{1}{2}$ and $\theta_2 = ... = \theta_p = 0$, testing $H_0 : \theta_1^2 + ... + \theta_p^2 = 0$ against $H_1 : \theta_1^2 + ... + \theta_p^2 \neq 0$

	p = 2		p = 5		p = 10		p = 20		p = 30	
Size	Asy	LMC	Asy	LMC	Asy	LMC	Asy	LMC	Asy	LMC
15	18.8	90.4	28.4	80.8	52.0	69.2	86.0	58.4	97.6	46.0
25	60.8	98.8	70.4	96.0	86.8	96.0	97.6	84.4	100.0	78.4
50	97.6	100.0	98.8	100.0	100.0	100.0	100.0	99.6	100.0	99.6
100	100.0	100.0	100.0	100.0	100.0	100.0	100.0	100.0	100.0	100.0
150	100.0	100.0	100.0	100.0	100.0	100.0	100.0	100.0	100.0	100.0
200	100.0	100.0	100.0	100.0	100.0	100.0	100.0	100.0	100.0	100.0
250	100.0	100.0	100.0	100.0	100.0	100.0	100.0	100.0	100.0	100.0
300	100.0	100.0	100.0	100.0	100.0	100.0	100.0	100.0	100.0	100.0

Table 8: Empirical levels for 1000 replications of the Wald statistic where $\alpha = 5\%$, $\theta_1 = 1$ and $\theta_2 = ... = \theta_p = 0$, testing $H_0 : \theta_1^2 + ... + \theta_p^2 = 0$ against $H_1 : \theta_1^2 + ... + \theta_p^2 \neq 0$

Looking at tables 6, 7 and 8, we see that, while the power of LMC seems to be constant over different values of p, as n increases, so does the power of the LMC test.

Now for the asymptotic test, when p is low, the power of the test is low compared to the LMC. When p is high, then the power of the test is close to 100% even for low n and θ_1 close to 0. This feature is simply a consequence of the nature of the test statistic, where the χ_p^2 critical values are conservative when p is low and untrustworthy when p is high.

4.4 Example 3

The following example is again given by **dufour˙wald˙2013** It is again a case of redundant restrictions like the example 1, but where

$$H_0 : [\theta_1\theta_2^2, \theta_1^2]' = 0 \tag{74}$$

$$H_1 : [\theta_1\theta_2^2, \theta_1^2]' \neq 0 \tag{75}$$

Therefore, both restrictions can be reduced to the restriction that $\theta_1 = 0$. This in turns leads to the following derivative matrix

$$P(\theta) = \begin{bmatrix} \theta_2^2 & 2\theta_1\theta_2 \\ 2\theta_1 & 0 \end{bmatrix} \tag{76}$$

Then, for $\hat{\Sigma}_\theta = \Sigma_\theta = I$, we have

$$P(\hat{\theta})\hat{\Sigma}_\theta P(\hat{\theta})' = \begin{bmatrix} \hat{\theta}_2^4 + 4\hat{\theta}_1^2\hat{\theta}_2^2 & 2\hat{\theta}_1\hat{\theta}_2^2 \\ 2\hat{\theta}_1\hat{\theta}_2^2 & 4\hat{\theta}_1^2 \end{bmatrix} \tag{77}$$

and it is easy to see that, as $n \to \infty$, the asymptotic covariance matrix converges to a singular one.

Wald-statistic is then

$$W_n = \frac{n}{16}(4\hat{\theta}_1^2 + \hat{\theta}_2^2) \tag{78}$$

This leads the following two cases

Case 1: $\theta_1 = 0$ and $\theta_2 = 0$

$$W_n \xrightarrow{n \to \infty} \frac{1}{4} Z_1^2 + \frac{1}{16} Z_2^2 \leq \frac{1}{4} (Z_1^2 + Z_2^2) \sim \frac{1}{4} \chi_2^2 \tag{79}$$

Case 2: $\theta_1 = 0$ and $\theta_2 \neq 0$

$$W_n \xrightarrow{n \to \infty} +\infty \tag{80}$$

Since the usual Wald-statistic would use χ_2^2 critical values to test H_0, it is easy to see that under case 1, the test would be conservative, while under case 2, it would dangerously over-reject.

4.4.1 Simulation study

4.4.1.1 Level

As for the example 1, we simulate $\hat{\theta}_1 \sim N(\theta_1, \frac{1}{n})$, $\hat{\theta}_2 \sim N(\theta_2, \frac{1}{n})$, where n is the sample size and $\theta_1 = 0$, for different values of θ_2. The empirical levels are reported in table 9. Note that since θ_2 is a nuisance parameter, the MMC is always reported along the asymptotic test and the LMC. Note that for the MMC, we build a 99% confidence interval using maximum likelihood for θ_2 and test the p-value against a level of 4%. Then, this two-stage procedure should yield an exact test at the 5% level.

	$\theta_2 = 0$			$\theta_2 = \frac{1}{4}$			$\theta_2 = \frac{1}{2}$			$\theta_2 = 1$			$\theta_2 = 5$		
Size	Asy	LMC	MMC	Asy	LMC	MMC	Asy	LMC	MMC	Asy	LMC	MMC	Asy	LMC	MMC
15	0	3.8	0.6	0	4.1	0.4	0.0	3.0	0.1	0.0	1.2	0	100	0	0
25	0	3.8	0.6	0	3.5	0.2	0.0	2.7	0.1	0.0	1.0	0	100	0	0
50	0	3.8	0.6	0	3.0	0.1	0.0	1.5	0.0	1.0	0.4	0	100	0	0
100	0	3.8	0.6	0	2.7	0.1	0.0	1.0	0.0	67.0	0.1	0	100	0	0
150	0	3.8	0.6	0	1.9	0.1	0.2	0.5	0.0	99.7	0.1	0	100	0	0
200	0	3.8	0.6	0	1.5	0.0	1.0	0.4	0.0	100.0	0.1	0	100	0	0
250	0	3.8	0.6	0	1.1	0.0	5.9	0.2	0.0	100.0	0.0	0	100	0	0

Table 9: Empirical levels for 1000 replications of the Wald statistic where $\alpha = 5\%$, $\theta_1 = 0$, testing $H_0 : [\theta_1 \theta_2^2, \theta_1^2] = 0$ against $H_1 : [\theta_1 \theta_2^2, \theta_1^2] \neq 0$

In this example, every test under-rejects. What is interesting is that, as θ_2 increases, the LMC and MMC empirical go down even further, but the asymptotic starts to severely over-reject even when $\theta_2 = 1$. This point suggests that the traditional critical values are misleading in this example.

4.4.1.2 Power

Results for empirical levels when $\theta_1 \neq 0$ are reported in tables 10, 11 and 12.

	$\theta_2 = 0$			$\theta_2 = \frac{1}{4}$			$\theta_2 = \frac{1}{2}$			$\theta_2 = 1$			$\theta_2 = 5$		
Size	Asy	LMC	MMC	Asy	LMC	MMC	Asy	LMC	MMC	Asy	LMC	MMC	Asy	LMC	MMC
15	0.0	12.8	2.0	0.0	12.4	2.0	0.0	12.4	2.0	0.4	8.0	1.2	100	0.4	0
25	0.0	20.4	5.2	0.0	22.0	5.6	0.0	16.0	1.2	0.4	6.8	0.4	100	0.4	0
50	0.0	35.2	10.4	0.4	31.6	9.2	0.0	28.0	4.4	6.4	9.6	0.4	100	0.4	0
100	0.8	66.4	34.8	1.2	56.0	21.6	7.2	50.0	6.4	88.0	21.2	0.4	100	0.0	0
150	4.4	82.4	57.2	4.4	82.4	26.8	27.6	62.0	11.2	100.0	30.4	0.8	100	0.0	0
200	7.2	89.6	70.4	18.8	89.2	45.2	54.8	72.0	12.0	100.0	39.2	1.6	100	0.0	0
250	15.2	95.2	82.4	27.6	90.8	54.0	82.4	84.0	16.0	100.0	43.6	0.8	100	0.0	0

Table 10: Empirical levels for 250 replications of the Wald statistic where $\alpha = 5\%$, $\theta_1 = \frac{1}{4}$, testing $H_0 : \left[\theta_1\theta_2^2, \theta_1^2\right] = 0$ against $H_1 : \left[\theta_1\theta_2^2, \theta_1^2\right] \neq 0$

	$\theta_2 = 0$			$\theta_2 = \frac{1}{4}$			$\theta_2 = \frac{1}{2}$			$\theta_2 = 1$			$\theta_2 = 5$		
Size	Asy	LMC	MMC	Asy	LMC	MMC	Asy	LMC	MMC	Asy	LMC	MMC	Asy	LMC	MMC
15	0.0	38.8	17.2	0.0	40.0	11.2	0.0	33.2	8.8	0.8	33.6	3.6	100	1.6	0
25	0.8	66.4	34.8	0.4	62.8	26.0	1.2	56.0	21.6	7.2	50.0	6.4	100	2.4	0
50	7.2	89.6	70.4	10.4	88.4	64.8	18.8	89.2	45.2	54.8	72.0	12.0	100	3.6	0
100	54.0	99.6	97.6	59.2	100.0	94.4	76.8	99.2	77.6	100.0	97.6	42.0	100	7.2	0
150	88.0	100.0	100.0	90.0	100.0	100.0	98.8	100.0	97.2	100.0	100.0	67.6	100	9.2	0
200	98.0	100.0	100.0	98.8	100.0	100.0	100.0	100.0	98.8	100.0	100.0	88.8	100	20.4	0
250	99.6	100.0	100.0	100.0	100.0	100.0	100.0	100.0	99.6	100.0	100.0	94.8	100	38.4	0

Table 11: Empirical levels for 250 replications of the Wald statistic where $\alpha = 5\%$, $\theta_1 = \frac{1}{2}$, testing $H_0 : \left[\theta_1\theta_2^2, \theta_1^2\right] = 0$ against $H_1 : \left[\theta_1\theta_2^2, \theta_1^2\right] \neq 0$

	$\theta_2 = 0$			$\theta_2 = \frac{1}{4}$			$\theta_2 = \frac{1}{2}$			$\theta_2 = 1$			$\theta_2 = 5$		
Size	Asy	LMC	MMC	Asy	LMC	MMC	Asy	LMC	MMC	Asy	LMC	MMC	Asy	LMC	MMC
15	20	97.2	82.4	14.0	92.8	80.0	21.2	95.2	73.2	31.2	91.6	50.8	100	31.6	0.4
25	54	99.6	97.6	56.8	100.0	97.2	59.2	100.0	94.4	76.8	99.2	77.6	100	53.2	0.8
50	98	100.0	100.0	98.0	100.0	100.0	98.8	100.0	100.0	100.0	100.0	98.8	100	94.0	4.8
100	100	100.0	100.0	100.0	100.0	100.0	100.0	100.0	100.0	100.0	100.0	100.0	100	100.0	23.6
150	100	100.0	100.0	100.0	100.0	100.0	100.0	100.0	100.0	100.0	100.0	100.0	100	100.0	68.4
200	100	100.0	100.0	100.0	100.0	100.0	100.0	100.0	100.0	100.0	100.0	100.0	100	100.0	91.2
250	100	100.0	100.0	100.0	100.0	100.0	100.0	100.0	100.0	100.0	100.0	100.0	100	100.0	99.2

Table 12: Empirical levels for 250 replications of the Wald statistic where $\alpha = 5\%$, $\theta_1 = 1$, testing $H_0 : \left[\theta_1\theta_2^2, \theta_1^2\right] = 0$ against $H_1 : \left[\theta_1\theta_2^2, \theta_1^2\right] \neq 0$

For θ_2, the LMC outperfoms both the MMC and the asymptotic test. But, it is interesting to note that the MMC always performs better than the asymptotic test.

Now, difficulties start appearing when θ_2 increases. As θ_2 increases, the distribution of the statistic becomes more and more overpowered by this sole parameter. As such, both the LMC and MMC power goes down. The power of the MMC is more affected by this increase in θ_2 since it maximizes over a confidence interval around $\hat{\theta}_2$. Note that the power of the asymptotic test increases dramatically to 100% as θ_2 increases. Again even if this might seems like a nice property to have, it is simply the result of the critical values becoming invalid for θ_2.

4.5 Example 4

In this example, we relax the assumption that $\hat{\Sigma}_\theta = \Sigma_\theta = I$, and let

$$\Sigma_\theta = \begin{bmatrix} 1 & \rho \\ \rho & 1 \end{bmatrix} \tag{81}$$

i.e. $\text{Var}(\theta_1) = 1$, $\text{Var}(\theta_2) = 1$ and $\text{Cov}(\theta_1, \theta_2) = \text{Corr}(\theta_1, \theta_2) = \rho$.

Then, we keep the same set of restrictions as the previous example i.e.

$$H_0 : [\theta_1\theta_2^2, \theta_1^2]' = 0 \tag{82}$$
$$H_1 : [\theta_1\theta_2^2, \theta_1^2]' \neq 0 \tag{83}$$

As we recall, the Wald statistic has the following form

$$W_n(\psi_0) = n \left[\psi(\hat{\theta}) - \psi_0 \right]' \left[P(\hat{\theta})\Sigma_\theta P(\hat{\theta})' \right]^{-1} \left[\psi(\hat{\theta}) - \psi_0 \right] \tag{84}$$

where in our example

$$P(\hat{\theta}) = \begin{bmatrix} \hat{\theta}_2^2 & 2\hat{\theta}_1\hat{\theta}_2 \\ 2\hat{\theta}_1 & 0 \end{bmatrix} \tag{85}$$

Therefore in this example, it reduces to

$$W_n = n \left[\frac{4\hat{\theta}_1^2 - 4\rho\hat{\theta}_1\hat{\theta}_2 + \hat{\theta}_2^2}{16(1 - \rho^2)} \right] \tag{86}$$

where $\text{Corr}(\theta_1, \theta_2) = \rho$.

It is clear that the asymptotic distribution under this case is a bit more complex than in example 3. We won't go into the details of deriving the asymptotic distribution, but it is easy to see that as $\rho \to 1$, $W_n \to \infty$, which in turns implies that the usual χ_2^2 critical values will over-reject the test statistic.

The problem we face is that if Σ_θ is not given, then we need a consistent estimator for it i.e. we need an estimator $\hat{\Sigma}_\theta$ such that

$$\hat{\Sigma}_\theta \xrightarrow{n \to \infty} \Sigma_\theta \tag{87}$$

If we assume that we have $X_1, ..., X_n$ i.i.d. observations such that $X_i \sim N(\theta, \Sigma_\theta)$, then for

$$\hat{\theta} = \frac{1}{n} \sum_{i=1}^{n} X_i = \bar{X} \tag{88}$$

$$\hat{\Sigma}_\theta = \frac{1}{n} \sum_{i=1}^{n} (X_i - \bar{X})(X_i - \bar{X})' \tag{89}$$

we have the following required convergence

$$\hat{\theta} \xrightarrow{n \to \infty} \theta \tag{90}$$

$$\hat{\Sigma}_\theta \xrightarrow{n \to \infty} \Sigma_\theta \tag{91}$$

Therefore, the Wald statistic in this example takes the following form

$$W_n(\psi_0) = n \left[\psi(\hat{\theta}) - \psi_0 \right]' \left[P(\hat{\theta}) \hat{\Sigma}_\theta P(\hat{\theta})' \right]^{-1} \left[\psi(\hat{\theta}) - \psi_0 \right] \tag{92}$$

where

$$\hat{\theta} = \frac{1}{n} \sum_{i=1}^{n} X_i = \bar{X} \tag{93}$$

$$\hat{\Sigma}_\theta = \frac{1}{n} \sum_{i=1}^{n} (X_i - \bar{X})(X_i - \bar{X})' \tag{94}$$

and

$$P(\hat{\theta}) = \begin{bmatrix} \hat{\theta}_2^2 & 2\hat{\theta}_1\hat{\theta}_2 \\ 2\hat{\theta}_1 & 0 \end{bmatrix} \tag{95}$$

$$\psi(\hat{\theta}) - \psi_0 = \begin{bmatrix} \hat{\theta}_1\hat{\theta}_2^2 \\ \hat{\theta}_1^2 \end{bmatrix} \tag{96}$$

$$\tag{97}$$

4.5.1 Simulation study

4.5.1.1 Level

In this example, θ_2 and ρ are nuisance parameters. Thus, for the MMC, we build a 99% confidence interval for θ_2 and ρ, and we test the p-value against a level of 4%. The empirical levels are reported in table 13.

ρ	Size	$\theta_2 = 0$			$\theta_2 = \frac{1}{4}$			$\theta_2 = \frac{1}{2}$			$\theta_2 = 1$			$\theta_2 = 5$		
		Asy	LMC	MMC	Asy	LMC	MMC	Asy	LMC	MMC	Asy	LMC	MMC	Asy	LMC	MMC
	15	0.3	3.3	0.1	0.2	2.9	0.0	0.5	2.3	0.1	4.7	1.7	0.0	80.2	0.0	0
	25	0.0	2.9	0.1	0.2	2.6	0.0	0.6	2.3	0.0	13.5	1.1	0.0	75.8	0.0	0
	50	0.1	3.9	0.4	0.3	3.4	0.1	1.8	2.5	0.0	69.6	2.3	0.0	64.9	0.0	0
0.25	100	0.1	3.5	0.4	0.5	3.7	0.1	11.0	3.2	0.0	96.0	1.5	0.0	54.0	0.0	0
	150	0.0	2.4	0.2	0.8	3.0	0.2	38.9	2.3	0.0	94.2	1.5	0.0	45.2	0.2	0
	200	0.1	4.6	0.4	0.8	3.1	0.1	71.8	2.5	0.0	94.3	1.2	0.0	37.2	0.2	0
	250	0.1	3.8	0.5	2.5	3.6	0.0	90.8	2.5	0.0	93.5	1.1	0.0	33.5	0.7	0
	15	0.3	2.3	0.0	0.8	3.1	0.0	1.4	4.2	0.0	12.4	4.3	0.0	79.3	0.0	0
	25	0.3	3.1	0.1	0.6	2.9	0.2	1.2	4.1	0.2	26.7	3.3	0.1	73.7	0.0	0
	50	0.1	3.2	0.1	0.6	4.1	0.1	4.9	7.3	0.2	87.5	4.9	0.0	62.7	0.0	0
0.50	100	0.2	4.0	0.1	0.7	6.0	0.4	24.3	5.5	0.4	95.5	4.5	0.0	51.3	0.2	0
	150	0.2	3.3	0.2	1.6	5.3	0.4	61.6	4.8	0.3	94.0	3.4	0.0	42.5	0.4	0
	200	0.1	3.7	0.2	3.3	6.2	0.2	88.2	7.2	0.0	93.3	4.8	0.2	36.1	1.1	0
	250	0.1	3.5	0.2	7.8	6.3	0.3	95.9	6.2	0.0	93.2	4.1	0.1	29.9	1.6	0
	15	0.8	1.3	0.0	2.6	4.5	0.0	6.8	7.1	0.2	31.7	4.2	0.2	75.2	0.0	0
	25	0.7	1.9	0.1	1.2	4.7	0.1	7.2	6.4	0.2	67.5	5.5	0.0	69.8	0.1	0
	50	0.4	2.8	0.1	3.3	8.7	0.4	24.5	11.3	0.6	95.5	6.4	0.2	57.7	0.0	0
0.75	100	0.5	2.6	0.0	6.5	9.6	0.4	71.0	9.5	0.0	95.5	7.8	0.1	45.3	0.5	0
	150	0.3	1.8	0.0	10.7	10.5	0.8	94.5	10.9	0.4	93.7	7.3	0.1	33.8	1.0	0
	200	0.3	2.4	0.0	21.5	12.2	0.5	96.8	10.1	0.0	91.9	8.1	0.0	30.6	2.3	0
	250	0.2	2.5	0.0	30.6	9.4	0.6	97.2	10.8	0.4	91.4	8.1	0.1	23.4	3.9	0
	15	3.0	0.3	0.0	8.3	4.7	0.0	21.3	10.0	0.3	78.6	8.2	0.8	70.3	0.0	0
	25	2.4	0.6	0.0	7.9	6.4	0.0	33.3	10.3	0.7	95.3	7.1	0.3	63.5	0.0	0
	50	2.4	1.0	0.0	16.2	12.8	0.1	67.4	14.1	0.6	95.6	8.5	0.0	49.4	0.1	0
0.90	100	2.1	0.8	0.0	32.9	14.7	0.5	97.2	13.9	0.5	93.5	9.2	0.0	35.8	0.6	0
	150	2.4	0.5	0.0	51.6	15.0	1.2	96.1	15.2	0.8	90.1	9.2	0.2	23.1	2.4	0
	200	2.1	0.6	0.0	73.1	15.0	0.5	96.1	14.4	0.7	90.9	11.1	0.1	19.9	5.8	0
	250	2.2	0.6	0.0	85.9	15.6	0.6	96.0	16.1	0.5	91.0	9.2	0.1	14.6	6.6	0

Table 13: Empirical levels for 1000 replications of the Wald statistic where $\alpha = 5\%$, $\theta_1 = 0$, testing $H_0 : \left[\theta_1\theta_2^2, \theta_1^2\right] = 0$ against $H_1 : \left[\theta_1\theta_2^2, \theta_1^2\right] \neq 0$

Looking at table 13, the results for low values of ρ are consistent with table 10. Again, as θ_2 increases, the asymptotic critical values become invalid.

When ρ increases, the LMC starts to over-reject, while the MMC stays consistent in under-rejecting. In fact, for $\rho = .9$, the LMC empirical level goes up to 16.1%. Therefore, the MMC is the only test that yields valid inferences in every example.

Note that for high ρ, as n increases, the empirical level of the LMC goes up. Additionally, the empirical level of both the LMC and the asymptotic test increases as θ_2, but sharply declines when $\theta_2 = 5$.

4.5.1.2 Power

Empirical levels when simulating under H_1 are reported in tables 14, 15 and 16.

		$\theta_2 = 0$			$\theta_2 = \frac{1}{4}$			$\theta_2 = \frac{1}{2}$			$\theta_2 = 1$			$\theta_2 = 5$		
ρ	Size	Asy	LMC	MMC	Asy	LMC	MMC	Asy	LMC	MMC	Asy	LMC	MMC	Asy	LMC	MMC
	15	0.8	12.0	0.0	0.8	7.2	0.0	0.4	5.6	0.0	4.4	2.4	0	86.8	0.0	0
	25	4.4	19.2	2.4	2.4	12.8	0.8	4.0	10.0	0.8	12.0	2.8	0	87.6	0.0	0
	50	9.2	40.0	12.4	8.8	26.4	6.0	10.8	13.6	0.4	70.8	2.0	0	90.8	0.0	0
0.25	100	23.6	66.8	33.6	21.2	48.8	9.6	39.6	21.6	0.8	100.0	4.0	0	96.4	0.0	0
	150	43.6	82.0	60.4	41.2	67.2	18.4	75.2	31.6	1.2	100.0	2.0	0	99.2	0.0	0
	200	57.6	94.8	72.4	56.4	72.0	22.0	93.2	40.4	1.6	100.0	2.0	0	99.6	0.0	0
	250	70.4	95.2	79.2	77.6	86.4	30.0	99.2	40.0	2.4	100.0	2.8	0	100.0	0.0	0
	15	2.4	12.4	0.0	0.8	5.6	0.0	1.2	2.8	0.0	2.0	0.4	0	86.0	0.0	0
	25	4.0	19.2	0.4	2.8	13.2	0.0	1.6	2.8	0.0	8.8	0.8	0	88.0	0.0	0
	50	10.8	37.6	7.6	6.0	16.4	0.8	4.8	5.2	0.0	66.0	0.0	0	92.0	0.0	0
0.50	100	34.0	77.2	33.6	18.0	33.6	2.0	30.0	4.8	0.4	100.0	0.0	0	96.4	0.0	0
	150	48.4	86.8	46.8	37.6	49.6	5.6	65.2	2.4	0.0	100.0	0.0	0	98.8	0.0	0
	200	74.4	93.6	71.2	56.0	58.8	7.2	84.4	3.2	0.0	100.0	0.0	0	99.2	0.0	0
	250	86.4	98.4	82.8	69.2	65.2	8.8	95.2	4.4	0.4	100.0	0.0	0	100.0	0.0	0
	15	6.4	12.8	0.0	3.2	2.4	0.0	1.6	0.8	0.0	8.0	0.8	0	85.2	0.0	0
	25	10.0	24.4	1.2	4.4	4.8	0.4	2.0	0.8	0.0	16.4	0.0	0	88.0	0.0	0
	50	30.0	49.2	7.6	10.0	9.6	0.4	4.4	0.4	0.0	83.6	0.0	0	90.0	0.0	0
0.75	100	58.8	81.6	24.8	26.4	15.6	0.0	21.2	0.0	0.0	100.0	0.0	0	94.8	0.0	0
	150	89.6	95.2	51.6	46.0	25.2	0.4	54.0	0.0	0.0	100.0	0.0	0	98.8	0.0	0
	200	93.6	98.8	76.0	61.2	30.8	0.4	73.6	0.0	0.0	100.0	0.0	0	99.2	0.0	0
	250	97.6	98.8	88.0	74.8	36.4	0.8	90.0	0.0	0.0	100.0	0.0	0	99.6	0.0	0
	15	21.2	13.2	0.0	8.0	0.8	0.0	5.2	0.4	0.0	24.8	1.2	0	82.8	0.0	0
	25	33.2	30.4	0.4	12.8	1.6	0.0	4.8	0.0	0.0	52.8	0.0	0	82.8	0.0	0
	50	65.6	66.4	4.0	23.6	2.8	0.0	8.8	0.4	0.0	100.0	0.0	0	87.6	0.0	0
0.90	100	94.0	91.2	22.4	52.0	4.0	0.0	20.0	0.0	0.0	99.6	0.0	0	92.4	0.0	0
	150	98.0	97.2	53.2	72.4	4.4	0.0	47.6	0.0	0.0	100.0	0.0	0	97.6	0.0	0
	200	99.6	98.8	75.2	82.8	5.2	0.0	71.2	0.0	0.0	100.0	0.0	0	98.8	0.0	0
	250	100.0	99.6	86.0	90.4	8.8	0.0	90.4	0.0	0.0	100.0	0.0	0	99.6	0.8	0

Table 14: Empirical levels for 250 replications of the Wald statistic where $\alpha = 5\%$, $\theta_1 = \frac{1}{4}$, testing $H_0 : \left[\theta_1\theta_2^2, \theta_1^2\right] = 0$ against $H_1 : \left[\theta_1\theta_2^2, \theta_1^2\right] \neq 0$

		$\theta_2 = 0$			$\theta_2 = \frac{1}{4}$			$\theta_2 = \frac{1}{2}$			$\theta_2 = 1$			$\theta_2 = 5$		
ρ	Size	Asy	LMC	MMC	Asy	LMC	MMC	Asy	LMC	MMC	Asy	LMC	MMC	Asy	LMC	MMC
	15	11.6	38.4	4.8	6.4	30.4	2.8	10.0	26.4	0.8	12.0	10.0	0.0	96.4	0.0	0
	25	21.6	64.4	20.0	24.0	56.8	10.8	22.4	43.2	4.8	42.0	24.4	1.2	99.6	0.4	0
	50	64.8	94.4	62.4	54.0	90.4	39.6	59.6	68.8	16.0	88.4	33.2	0.4	99.6	0.0	0
0.25	100	94.8	100.0	96.0	94.4	99.2	83.2	97.6	96.0	45.6	100.0	54.0	4.4	100.0	0.0	0
	150	100.0	100.0	99.6	100.0	100.0	98.4	99.6	96.8	66.0	100.0	68.0	6.8	100.0	0.4	0
	200	100.0	100.0	100.0	100.0	100.0	99.6	100.0	100.0	83.2	100.0	80.8	10.8	100.0	0.0	0
	250	100.0	100.0	100.0	100.0	100.0	100.0	100.0	100.0	93.2	100.0	89.2	16.4	100.0	0.0	0
	15	19.2	40.4	3.6	14.0	32.4	2.0	10.0	20.0	0.8	9.2	3.6	0.0	97.2	0.0	0
	25	30.8	67.6	11.2	18.8	48.8	4.8	16.0	27.6	0.4	25.2	3.6	0.0	99.2	0.0	0
	50	73.6	96.8	52.8	64.4	87.2	20.8	49.6	48.0	3.6	84.4	5.2	0.0	99.6	0.0	0
0.50	100	99.2	100.0	98.4	94.8	99.2	68.0	92.8	85.2	13.6	100.0	7.2	0.4	100.0	0.0	0
	150	100.0	100.0	99.6	100.0	100.0	93.2	100.0	91.2	24.8	100.0	8.4	0.0	100.0	0.0	0
	200	100.0	100.0	100.0	100.0	100.0	98.8	100.0	99.2	33.2	100.0	4.0	0.0	100.0	0.0	0
	250	100.0	100.0	100.0	100.0	100.0	100.0	100.0	99.6	56.8	100.0	7.6	0.0	100.0	0.0	0
	15	29.6	43.6	4.0	20.8	24.0	0.8	15.6	4.0	0.0	8.0	0.0	0.0	96.4	0.0	0
	25	60.8	81.6	8.8	33.6	50.4	2.0	24.8	11.6	0.0	18.0	0.0	0.0	99.6	0.0	0
	50	94.4	98.4	46.8	80.8	87.6	10.4	61.6	24.8	0.4	72.8	0.0	0.0	99.6	0.0	0
0.75	100	100.0	100.0	92.4	100.0	99.6	44.8	95.2	44.4	0.8	100.0	0.0	0.0	100.0	0.0	0
	150	100.0	100.0	100.0	100.0	100.0	78.0	99.2	61.6	2.4	100.0	0.0	0.0	100.0	0.0	0
	200	100.0	100.0	100.0	100.0	100.0	96.4	100.0	80.0	6.0	100.0	0.0	0.0	100.0	0.0	0
	250	100.0	100.0	100.0	100.0	100.0	99.6	100.0	87.2	6.8	100.0	0.0	0.0	100.0	0.0	0
	15	75.6	69.2	3.2	47.2	20.0	0.4	27.6	0.0	0.0	13.6	0.0	0.0	95.2	0.0	0
	25	93.6	88.8	10.8	76.8	62.0	2.0	50.0	2.0	0.0	18.4	0.0	0.0	99.6	0.0	0
	50	99.6	99.2	48.0	97.2	93.6	6.4	80.8	5.6	0.0	69.2	0.0	0.0	100.0	0.0	0
0.90	100	100.0	100.0	94.8	99.6	100.0	27.6	98.8	10.4	0.0	100.0	0.0	0.0	100.0	0.0	0
	150	100.0	100.0	100.0	100.0	100.0	68.8	100.0	13.2	0.0	100.0	0.0	0.0	100.0	0.0	0
	200	100.0	100.0	100.0	100.0	100.0	92.8	100.0	16.8	0.0	100.0	0.0	0.0	100.0	0.0	0
	250	100.0	100.0	100.0	100.0	100.0	99.2	100.0	26.4	0.0	100.0	0.0	0.0	100.0	0.0	0

Table 15: Empirical levels for 250 replications of the Wald statistic where $\alpha = 5\%$, $\theta_1 = \frac{1}{2}$, testing $H_0 : \left[\theta_1\theta_2^2, \theta_1^2\right] = 0$ against $H_1 : \left[\theta_1\theta_2^2, \theta_1^2\right] \neq 0$

ρ	Size	θ₂ = 0 Asy	LMC	MMC	θ₂ = 1/4 Asy	LMC	MMC	θ₂ = 1/2 Asy	LMC	MMC	θ₂ = 1 Asy	LMC	MMC	θ₂ = 5 Asy	LMC	MMC
	15	70.4	96.0	32.8	73.6	92.4	21.6	76.4	90.8	20.4	73.6	62.0	5.2	100	1.2	0.0
	25	96.4	99.6	87.2	94.8	99.2	72.8	92.8	98.0	43.6	96.8	93.2	13.2	100	4.4	0.0
	50	100.0	100.0	100.0	100.0	100.0	99.6	100.0	100.0	95.2	100.0	98.8	52.4	100	8.4	0.0
0.25	100	100.0	100.0	100.0	100.0	100.0	100.0	100.0	100.0	100.0	100.0	100.0	94.8	100	6.0	0.0
	150	100.0	100.0	100.0	100.0	100.0	100.0	100.0	100.0	100.0	100.0	100.0	99.6	100	10.8	0.4
	200	100.0	100.0	100.0	100.0	100.0	100.0	100.0	100.0	100.0	100.0	100.0	100.0	100	4.8	0.0
	250	100.0	100.0	100.0	100.0	100.0	100.0	100.0	100.0	100.0	100.0	100.0	100.0	100	6.0	0.0
	15	86.4	95.6	22.0	79.2	95.2	20.8	75.6	86.4	10.4	71.2	40.8	2.0	100	0.0	0.0
	25	99.2	100.0	73.6	98.0	99.6	56.8	97.6	98.0	36.8	97.2	67.6	6.4	100	0.0	0.0
	50	100.0	100.0	99.6	100.0	100.0	99.6	100.0	100.0	82.4	100.0	93.6	17.6	100	0.0	0.0
0.50	100	100.0	100.0	100.0	100.0	100.0	100.0	100.0	100.0	100.0	100.0	100.0	50.8	100	0.0	0.0
	150	100.0	100.0	100.0	100.0	100.0	100.0	100.0	100.0	100.0	100.0	100.0	80.0	100	0.0	0.0
	200	100.0	100.0	100.0	100.0	100.0	100.0	100.0	100.0	100.0	100.0	100.0	97.6	100	0.0	0.0
	250	100.0	100.0	100.0	100.0	100.0	100.0	100.0	100.0	100.0	100.0	100.0	98.8	100	0.0	0.0
	15	96.4	98.4	17.6	96.0	97.6	12.4	90.4	88.0	5.2	71.2	11.6	0.8	100	0.0	0.0
	25	100.0	100.0	62.0	100.0	100.0	34.0	99.2	97.6	13.6	96.4	26.8	0.8	100	0.0	0.0
	50	100.0	100.0	100.0	100.0	100.0	98.4	100.0	100.0	60.8	100.0	55.2	1.2	100	0.0	0.0
0.75	100	100.0	100.0	100.0	100.0	100.0	100.0	100.0	100.0	100.0	100.0	91.6	6.0	100	0.0	0.0
	150	100.0	100.0	100.0	100.0	100.0	100.0	100.0	100.0	100.0	100.0	97.2	13.2	100	0.0	0.0
	200	100.0	100.0	100.0	100.0	100.0	100.0	100.0	100.0	100.0	100.0	99.6	21.6	100	0.0	0.0
	250	100.0	100.0	100.0	100.0	100.0	100.0	100.0	100.0	100.0	100.0	100.0	31.6	100	0.0	0.0
	15	100.0	99.6	16.0	99.6	98.8	12.0	98.8	90.8	4.4	87.2	0.4	0.0	100	0.0	0.0
	25	100.0	100.0	63.2	100.0	100.0	25.2	100.0	99.6	10.8	99.6	3.2	0.4	100	0.0	0.0
	50	100.0	100.0	100.0	100.0	100.0	98.8	100.0	100.0	37.6	100.0	8.8	0.4	100	0.0	0.0
0.90	100	100.0	100.0	100.0	100.0	100.0	100.0	100.0	100.0	100.0	100.0	21.2	0.4	100	0.0	0.0
	150	100.0	100.0	100.0	100.0	100.0	100.0	100.0	100.0	100.0	100.0	31.6	0.0	100	0.0	0.0
	200	100.0	100.0	100.0	100.0	100.0	100.0	100.0	100.0	100.0	100.0	56.0	0.4	100	0.0	0.0
	250	100.0	100.0	100.0	100.0	100.0	100.0	100.0	100.0	100.0	100.0	65.2	0.4	100	0.0	0.0

Table 16: Empirical levels for 250 replications of the Wald statistic where $\alpha = 5\%$, $\theta_1 = 1$, testing $H_0 : \left[\theta_1\theta_2^2, \theta_1^2\right] = 0$ against $H_1 : \left[\theta_1\theta_2^2, \theta_1^2\right] \neq 0$

Again by design the power of MMC is always lower than the LMC, but for $\theta_2 = 0$ the MMC is still close to the LMC. But, as θ_2 increases, both the LMC and MMC lose all power. Moreover, as ρ increases both techniques lose power.

As for the asymptotic critical values, again we have great power, but it is still results in critical values that give invalid inferences under the null.

5 Autoregressive models

In this section, we turn our attention to unit root tests in autoregressive models. In particular, we look into the details of one of the most popular sets of unit root tests; the Augmented Dickey-Fuller tests.

5.1 Framework

5.1.1 AR(p) model without drift

Consider the following autoregressive model of order p

$$Y_t = \sum_{j=1}^{p} \phi_j Y_{t-j} + u_t, \quad t = p+1, ..., T \tag{98}$$

where u_t is a sequence of i.i.d. $(0, \sigma^2)$ variables.

We can rewrite the model in the following way

$$\phi(L)Y_t = u_t, \quad t = p+1, ..., T \tag{99}$$

where

$$\phi(L) \equiv 1 - \sum_{j=1}^{p} \phi_j L^j \tag{100}$$

is the characteristic equation of the time process and L is the lag operator (i.e. $LY_t = Y_{t-1}$). Note that, if the characteristic equation has a unit root (i.e. $\phi(1) = 0$) with multiplicity r, then we have an integrated process of order r or simply put an I(r) process.

Tests of unit root can be formulated as

$$H_0 : \phi(1) = 0 \tag{101}$$

or equivalently,

$$H_0 : \sum_{j=1}^{P} \phi_j = 1 \tag{102}$$

In order to test H_0, we can rewrite 98 in the following way

$$\Delta Y_t = \gamma Y_{t-1} + \sum_{j=1}^{p-1} \rho_{j+1} \Delta Y_{t-j} + u_t, \qquad t = p+1, ..., T \tag{103}$$

where

$$\gamma = \sum_{j=1}^{P} \phi_j - 1 \tag{104}$$

$$\rho_j = -\sum_{i=j}^{P} \phi_i \quad j = 2, ..., p \tag{105}$$

and $\Delta Y_t = Y_t - Y_{t-1}$

Then, $H_0 : \phi(1) = 0$ can be written as

$$H_0 : \gamma = 0 \tag{106}$$

Therefore, to test H_0, we can simply use the usual Student's t-test based on least square estimator t_γ. This is referred to as the Augmented Dickey-Fuller test statistic. The distributional property of t_γ has been well documented and **fuller˙introduction˙1976** provides us with the critical points. One should note that the critical points are only valid asymptotically. These critical points are also available in **banerjee˙co-integration˙1993**

5.1.2   AR(p) model with drift

Now, consider the following autoregressive model of order p

$$Y_t = \alpha + \sum_{j=1}^{p} \phi_j Y_{t-j} + u_t, \quad t = p+1, ..., T \tag{107}$$

where α is the drift component, and u_t is a sequence of $i.i.d.$ $(0, \sigma^2)$ variables.

Again, we are looking to test

$$H_0 : \sum_{j=1}^{P} \phi_j = 1 \tag{108}$$

In order to do that, we can rewrite 107 as in the previous case such that

$$\Delta Y_t = \alpha + \gamma Y_{t-1} + \sum_{j=1}^{p-1} \rho_{j+1} \Delta Y_{t-j} + u_t, \qquad t = p+1, ..., T \tag{109}$$

where

$$\gamma = \sum_{j=1}^{P} \phi_j - 1 \tag{110}$$

$$\rho_j = -\sum_{i=j}^{P} \phi_i \quad j = 2, ..., p \tag{111}$$

and $\Delta Y_t = Y_t - Y_{t-1}$

Then, $H_0 : \sum_{j=1}^{P} \phi_j = 1$ can be written as

$$H_0 : \gamma = 0 \tag{112}$$

Like the previous case, asymptotic distributions have been simulated and the critical values for this model are given in **banerjee˙co-integration˙1993** and **fuller˙introduction˙1976**

5.1.3   AR(p) model with time trend

Consider the following autoregressive model of order p

$$Y_t = \alpha + \sum_{j=1}^{p} \phi_j Y_{t-j} + u_t, \quad t = p+1, ..., T \tag{113}$$

where α is the drift component, and u_t is a sequence of $i.i.d.$ $(0, \sigma^2)$ variables.

Again, we can rewrite 113 as

$$\Delta Y_t = \alpha + \gamma Y_{t-1} + \sum_{j=1}^{p-1} \rho_{j+1} \Delta Y_{t-j} + u_t, \qquad t = p+1, ..., T \tag{114}$$

where

$$\gamma = \sum_{j=1}^{P} \phi_j - 1 \tag{115}$$

$$\rho_j = -\sum_{i=j}^{P} \phi_i \quad j = 2, ..., p \tag{116}$$

and $\Delta Y_t = Y_t - Y_{t-1}$

Subsequently, we are looking to test

$$H_0 : \sum_{j=1}^{P} \phi_j = 1 \tag{117}$$

or equivalently,

$$H_0 : \gamma = 0 \tag{118}$$

Again, the critical points tables are available in both **banerjee˙co-integration˙1993** and **fuller˙introduction˙1976**

5.2 Simulation study

In order to test the augmented Dickey-Fuller, we use the **fUnitroots (wuertz˙funitroots:˙2013)** that provides the function `adfTest` that uses the tables provided by **banerjee˙co-integration˙1993** to compute the asymptotic p-value.

Note that the nuisance parameters for the MMC for an AR(p) process are $\phi_2, ... \phi_p$, and depend on the model the time trend term β and/or the drift term α. Estimates and 99% confidence intervals for these parameters are obtained using ordinary least-squares by fitting the AR(p) model using the package **dynlm (zeileis˙dynlm:˙2014)**. Then, for the MMC, we use the two-stage procedure where we maximize over the 99% intervals and test against $\alpha_2 = 4\%$. Then, the MMC test should have a nominal 5% level.

5.2.1 Level

First, we first examine the emperical levels of the asymptotic augmented Dickey-Fuller, the parametric bootstrap, the Local Monte Carlo and the Maximized Monte Carlo. We start by simulating an I(2) process of the following form

$$(1-L)^2 Y_t = u_t \quad t = p+1, ..., T \tag{119}$$
$$Y_t = 2Y_{t-1} - Y_{t-1} + u_t \quad t = p+1, ..., T \tag{120}$$

where the $u_t \sim N(0,1)$ are *i.i.d.*

This model can be rewritten

$$\Delta Y_t = \Delta Y_{t-1} + u_t, \qquad\qquad t = p+1, ..., T \tag{121}$$

where we see that the coefficient of Y_{t-1} disappears (i.e. we are under the null).

The results are reported in tables 17, 18, and 19. Note that the term nc refers to the augmented Dickey-Fuller model without drift, c refers to the augmented Dickey-Fuller model with drift , and ct refers to the augmented Dickey-Fuller model with time-trend.

| | n = 25 | | | | n = 50 | | | | n = 500 | | | |
Size	Asy	Boot	LMC	MMC	Asy	Boot	LMC	MMC	Asy	Boot	LMC	MMC
nc	4.8	7.6	5.2	3.2	7.2	6.8	6.4	2.0	6.8	8.8	6.4	4.8
c	4.4	18.8	16.4	2.0	6.4	18.0	15.2	4.0	3.6	19.6	16.4	1.2
ct	5.2	33.6	29.2	5.2	5.6	38.0	33.2	3.2	6.0	40.8	38.8	3.6

Table 17: Empirical levels for 250 replications of I(2) process where $\alpha = 5\%$, testing $H_0 : \gamma = 0$ against $H_1 : \gamma < 0$

As we can see we get a slight over-rejection, as predicted by **david˙a.˙dickey˙determining˙1987** for the asymptotic version of the test ranging from 5.2% to 7.2%. Looking at the MMC we see that it never over-rejects, and in fact, in some case seems to be conservative compared to the 5% level. On the other hand, the bootstrap and LMC over-reject every time with the LMC slightly outperforming the bootstrap. When the model is well specified (i.e. without drift), the bootstrap and the LMC give results consistent with the asymptotic version of the test, but when the model is misspecified both the bootstrap and LMC empirical rejection levels go well beyond 15%.

This result might seem contradictory as the critical values for the asymptotic tests were obtained using Monte Carlo simulation methods. Therefore, if the asymptotic test slightly over-rejects when the model is misspecified, our intuition would point to the fact that the bootstrap and the LMC should not completely misbehave when the test model is c or ct. To answer this question, we provide figure 2.

In figure 2, we use the 10% asymptotic critical values of the c model as our original statistic S_0, and plot the simulated p-value of a Monte Carlo simulation for different values of α and σ in the following model

$$Y_t = \alpha + \frac{1}{2}Y_{t-1} + \frac{1}{2}Y_{t-2} + u_t \quad t = p+1, ..., T \tag{122}$$

where $u_t \sim N(0, \sigma^2)$ and $n = 25$. We set the Monte Carlo replications to $N = 999$.
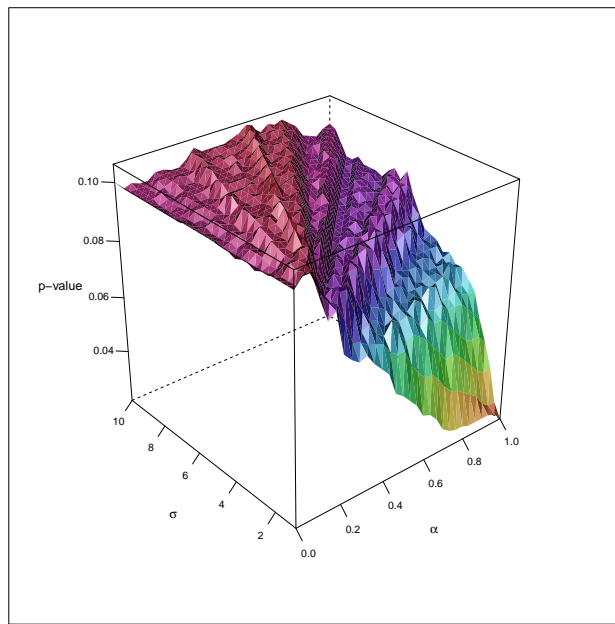
Figure 2:

It is then clear that the ratio of $\frac{\alpha}{\sigma}$ has an influence on the actual distribution of the statistic t_γ. For $\sigma = 1$, as α increases, the simulated p-value decreases. But, as σ increases, the effect disappears and the p-value approaches 10% for any α. This points to the fact that the tables given by **fuller˙introduction˙1976** might not be applicable in case where σ is significantly small.

Since, the MMC is robust to this type of problem we get an appropriate level of rejection.

We can extend this problem to the following model

$$Y_t = \beta t + \frac{1}{2}Y_{t-1} + \frac{1}{2}Y_{t-2} + u_t \quad t = p+1, ..., T \tag{123}$$

where $u_t \sim N(0, \sigma^2)$ and $n = 25$. Then in figure 3, we set the Monte Carlo replications to $N = 999$ and we use 10% asymptotic critical values of the ct model.
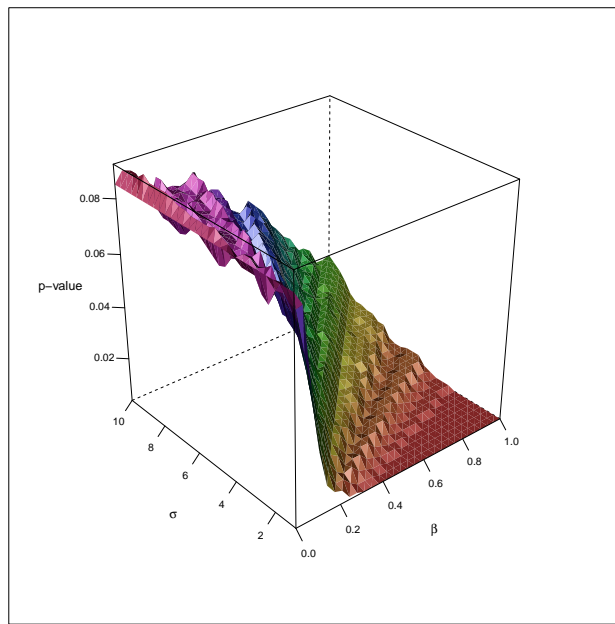
Figure 3:

As we can see in figure 3, we get the same results as in figure 2. In fact, the effect is exacerbated by β explaining why the bootstrap and the LMC over-reject even more when the model used is ct.

Now, in table 18 and table 19, we give the results for $H_1 : \gamma > 0$ and $H_1 : \gamma \neq 0$.

Size	n = 25				n = 50				n = 500			
	Asy	Boot	LMC	MMC	Asy	Boot	LMC	MMC	Asy	Boot	LMC	MMC
nc	12.4	8.8	9.6	2.8	10.8	6.8	5.6	0.8	12.0	6.0	8.4	2.0
c	50.0	11.2	13.2	2.8	40.0	11.2	10.8	1.2	39.6	4.8	5.6	0.8
ct	32.8	5.2	6.0	0.0	28.0	6.8	6.4	0.0	26.0	2.8	3.2	0.0

Table 18: Empirical levels for 250 replications of I(2) process where $\alpha = 5\%$, testing $H_0 : \gamma = 0$ against $H_1 : \gamma > 0$

	n = 25				n = 50				n = 500			
Size	Asy	Boot	LMC	MMC	Asy	Boot	LMC	MMC	Asy	Boot	LMC	MMC
nc	10.0	11.2	7.6	3.2	8.0	7.2	4.8	1.6	11.6	9.6	7.2	4.8
c	44.8	21.2	14.0	1.2	37.6	22.0	16.4	2.0	33.6	18.4	16.4	1.2
ct	27.2	31.2	25.6	4.0	24.8	33.2	26.4	2.4	25.2	35.2	29.6	4.0

Table 19: I(2) process, testing $H_0 : \gamma = 0$ against $H_1 : \gamma \neq 0$

As predicted by **haldrup˙robustness˙2002** the asymptotic test severely over-rejects when $H_1 : \gamma > 0$. The empirical levels range from 10.8% up to 50% for a test with a nominal level of 5%. This seems to indicate that tests of unit roots such as the augmented Dickey-Fuller might incorrectly classify an I(2) process as an explosive process. In turn, the bootstrap and the LMC seem to perform a bit better than the asymptotic test in this case. The MMC again always under-rejects compared to the 5% nominal level.

We now consider the three different I(2) models

$$(1 - L)^2 Y_t = u_t, \quad t = 1, ..., T \tag{124}$$

$$(1 - L)(1 + L)Y_t = u_t, \quad t = 1, ..., T \tag{125}$$

$$(1 - L)^2 (1 - 2L)Y_t = u_t, \quad t = 1, ..., T \tag{126}$$

where the $u_t \sim N(0, 1)$ are *i.i.d.*

The results are reported in tables 20, 21 and 22.

	$(1 - L)^2 y_t = u_t$				$(1 - L)(1 + L)y_t = u_t$				$(1 - L)^2(1 - 2L)y_t = u_t$			
Size	Asy	Boot	LMC	MMC	Asy	Boot	LMC	MMC	Asy	Boot	LMC	MMC
nc	7.2	6.8	6.4	2.0	3.2	5.6	4.8	3.6	5.2	5.2	5.2	1.6
c	6.4	18.0	15.2	4.0	5.2	20.4	15.6	2.8	5.6	17.2	18.0	2.8
ct	5.6	38.0	33.2	3.2	4.0	16.4	12.4	2.4	9.6	36.8	36.0	5.6

Table 20: Empirical levels for 250 replications of I(2) process where $\alpha = 5\%$ and n=50, testing $H_0 : \gamma = 0$ against $H_1 : \gamma < 0$

We can see from table 20 the asymptotic test slightly over-rejects except for the case where the data generating process is $(1 - L)(1 + L)Y_t = u_t$ where the empirical levels are fairly close to the nominal ones. In the case of the bootstrap and LMC, when the model is well specified both tests are consistent with the asymptotic results. In the case where the model is misspecified, the bootstrap and MMC have empirical levels well over 15%. Finally, the MMC is still consistent with the level of 5%. It usually under rejects except in one case where the empirical level is 5.6%.

	$(1 - L)^2 y_t = u_t$				$(1 - L)(1 + L)y_t = u_t$				$(1 - L)^2(1 - 2L)y_t = u_t$			
Size	Asy	Boot	LMC	MMC	Asy	Boot	LMC	MMC	Asy	Boot	LMC	MMC
nc	10.8	6.8	5.6	0.8	4.4	2.8	3.6	3.6	11.6	8.4	9.6	0.0
c	40.0	11.2	10.8	1.2	4.8	1.2	1.2	0.0	43.2	13.2	13.6	0.0
ct	28.0	6.8	6.4	0.0	4.8	2.0	3.2	0.0	34.8	6.0	5.2	0.4

Table 21: Empirical levels for 250 replications of I(2) process where $\alpha = 5\%$ and n=50, testing $H_0 : \gamma = 0$ against $H_1 : \gamma > 0$

	$(1-L)^2 y_t = u_t$				$(1-L)(1+L)y_t = u_t$				$(1-L)^2(1-2L)y_t = u_t$			
Size	Asy	Boot	LMC	MMC	Asy	Boot	LMC	MMC	Asy	Boot	LMC	MMC
nc	10.8	6.8	5.6	0.8	4.4	2.8	3.6	3.6	11.6	8.4	9.6	0.0
c	40.0	11.2	10.8	1.2	4.8	1.2	1.2	0.0	43.2	13.2	13.6	0.0
ct	28.0	6.8	6.4	0.0	4.8	2.0	3.2	0.0	34.8	6.0	5.2	0.4

Table 22: Empirical levels for 250 replications of I(2) process where $\alpha = 5\%$ and n=50, testing $H_0 : \gamma = 0$ against $H_1 : \gamma \neq 0$

Interestingly, the augmented Dickey-Fuller critical values do over-reject for $\gamma > 0$ except for the case $(1-L)(1+L)Y_t = u_t$ where the rejection values are actually consistent with the 5% level. Therefore, the sign of roots might not be trivial in terms of classification of the order of integration of the process.

We now relax the assumption that u_t $N(0,1)$ and let u_t follow the Student's t distribution with different degrees of freedom (e.g. $df = 1, 2, 3$). The results are reported in tables 23, 24 and 25.

	$(1-L)^2 y_t = u_t \sim t_1$				$(1-L)^2 y_t = u_t \sim t_2$				$(1-L)^2 y_t = u_t \sim t_3$			
Size	Asy	Boot	LMC	MMC	Asy	Boot	LMC	MMC	Asy	Boot	LMC	MMC
nc	8.0	7.2	4.8	1.6	3.2	5.2	2.4	2.4	9.2	10.0	6.0	0.4
c	37.6	22.0	16.4	2.0	4.4	15.6	11.6	1.6	39.2	24.0	19.2	1.6
ct	24.8	33.2	26.4	2.4	3.6	11.2	10.0	1.6	35.2	34.4	29.2	4.4

Table 23: Empirical levels for 250 replications of I(2) process where $\alpha = 5\%$ and n=50, testing $H_0 : \gamma = 0$ against $H_1 : \gamma < 0$

	$(1-L)^2 y_t = u_t \sim t_1$				$(1-L)^2 y_t = u_t \sim t_2$				$(1-L)^2 y_t = u_t \sim t_3$			
Size	Asy	Boot	LMC	MMC	Asy	Boot	LMC	MMC	Asy	Boot	LMC	MMC
nc	11.6	7.2	7.6	2.8	12.4	8.4	9.2	2.4	15.2	8.4	9.6	3.6
c	38.0	8.0	7.2	0.8	35.6	12.0	12.8	2.8	35.2	10.0	10.4	2.0
ct	23.6	2.0	3.2	0.4	26.0	4.8	4.8	0.0	24.4	3.2	4.4	0.8

Table 24: I(2) process with n=50, testing $H_0 : \gamma = 0$ against $H_1 : \gamma > 0$

	$(1-L)^2 y_t = u_t \sim t_1$				$(1-L)^2 y_t = u_t \sim t_2$				$(1-L)^2 y_t = u_t \sim t_3$			
Size	Asy	Boot	LMC	MMC	Asy	Boot	LMC	MMC	Asy	Boot	LMC	MMC
nc	7.2	8.0	4.0	1.6	8.8	10.0	5.2	1.2	9.6	8.4	6	1.6
c	34.8	22.8	14.4	2.4	33.6	20.4	16.4	0.8	32.8	21.2	16	2.0
ct	22.8	30.4	25.2	3.2	23.6	34.4	27.2	3.6	24.8	34.8	26	4.8

Table 25: Empirical levels for 250 replications of I(2) process where $\alpha = 5\%$ and n=50, testing $H_0 : \gamma = 0$ against $H_1 : \gamma \neq 0$

The results of tables 23, 24 and 25 are consistent with results when $u_t \sim N(0,1)$. We have over-rejection for every test, except for the MMC.

Finally, we change the order of integration of the process by simulating an I(1) process with increasing order p. The results are reported in tables 26, 27 and 28

	$y_t = \frac{1}{2}\sum_{j=1}^{2} y_{t-j} + u_t$				$y_t = \frac{1}{3}\sum_{j=1}^{3} y_{t-j} + u_t$				$y_t = \frac{1}{4}\sum_{j=1}^{4} y_{t-j} + u_t$			
Size	Asy	Boot	LMC	MMC	Asy	Boot	LMC	MMC	Asy	Boot	LMC	MMC
nc	4.0	5.6	4.8	4.8	4.4	4.0	3.2	2.0	4.8	6.4	4.8	3.2
c	3.6	19.2	15.6	1.6	4.4	20.8	17.2	2.0	2.4	18.8	18.8	0.4
ct	5.2	18.8	17.2	2.4	3.2	19.2	17.2	0.8	2.0	20.0	17.2	2.0

Table 26: Empirical levels for 250 replications of I(1) process where $\alpha = 5\%$ and n=50, testing $H_0 : \gamma = 0$ against $H_1 : \gamma < 0$

	$y_t = \frac{1}{2}\sum_{j=1}^{2} y_{t-j} + u_t$				$y_t = \frac{1}{3}\sum_{j=1}^{3} y_{t-j} + u_t$				$y_t = \frac{1}{4}\sum_{j=1}^{4} y_{t-j} + u_t$			
Size	Asy	Boot	LMC	MMC	Asy	Boot	LMC	MMC	Asy	Boot	LMC	MMC
nc	4.4	5.2	4.4	4	4.8	7.2	6.4	4.8	5.2	5.6	4.4	2.0
c	3.6	1.2	1.2	0	5.6	1.6	1.6	0.4	9.2	4.8	4.4	0.4
ct	6.8	1.6	2.0	0	5.6	1.2	0.8	0.0	5.6	0.8	2.0	0.0

Table 27: Empirical levels for 250 replications of I(1) process where $\alpha = 5\%$ and n=50, testing $H_0 : \gamma = 0$ against $H_1 : \gamma > 0$

	$y_t = \frac{1}{2}\sum_{j=1}^{2} y_{t-j} + u_t$				$y_t = \frac{1}{3}\sum_{j=1}^{3} y_{t-j} + u_t$				$y_t = \frac{1}{4}\sum_{j=1}^{4} y_{t-j} + u_t$			
Size	Asy	Boot	LMC	MMC	Asy	Boot	LMC	MMC	Asy	Boot	LMC	MMC
nc	6.0	6.4	3.2	2.8	4.4	6.0	4.0	2.4	4.0	8.0	5.2	1.6
c	4.4	15.2	9.2	0.8	4.4	14.0	11.2	1.2	7.2	19.2	12.0	0.4
ct	6.8	14.8	12.8	1.6	6.0	14.8	11.2	0.8	4.4	17.2	12.8	0.8

Table 28: Empirical levels for 250 replications of I(1) process where $\alpha = 5\%$ and n=50, testing $H_0 : \gamma = 0$ against $H_1 : \gamma \neq 0$

We see that in this I(1) case, the asymptotic critical values give valid inferences. Additionally, the MMC under-rejects in every case. On the other hand, the bootstrap and the LMC are only consistent when $H_1 : \gamma > 0$. When $H_1 : \gamma < 0$, the bootstrap and the LMC severely over-reject.

Therefore, again, only the MMC is valid in every situation.

5.2.2   Power

Now, we turn our attention to the power of the different tests. We look at three different data generating processes:

1. $y_t = -0.1y_{t-1} - 0.1\Delta y_{t-2} + u_t$

2. $y_t = -0.1y_{t-1} + 0.1\Delta y_{t-2} + u_t$

3. $y_t = 0.1y_{t-1} - 0.1\Delta y_{t-2} + u_t$

The results are described in tables 29, 30 and 31

46

Size	$y_t = -0.1y_{t-1} - 0.1\Delta y_{t-2} + u_t$				$y_t = -0.1y_{t-1} + 0.1\Delta y_{t-2} + u_t$				$y_t = 0.1y_{t-1} - 0.1\Delta y_{t-2} + u_t$			
	Asy	Boot	LMC	MMC	Asy	Boot	LMC	MMC	Asy	Boot	LMC	MMC
nc	21.2	23.2	19.6	14.0	27.6	30.8	28.8	23.6	0.4	0.4	0.0	0
c	10.4	18.0	12.8	2.8	10.8	26.0	20.8	4.0	0.0	0.4	0.4	0
ct	8.0	20.4	16.4	4.4	7.2	17.6	14.8	0.8	0.0	0.0	0.0	0

Table 29: Empirical levels for 250 replications of AR(p) process where $\alpha = 5\%$ and n=50, testing $H_0 : \gamma = 0$ against $H_1 : \gamma < 0$

Size	$y_t = -0.1y_{t-1} - 0.1\Delta y_{t-2} + u_t$				$y_t = -0.1y_{t-1} + 0.1\Delta y_{t-2} + u_t$				$y_t = 0.1y_{t-1} - 0.1\Delta y_{t-2} + u_t$			
	Asy	Boot	LMC	MMC	Asy	Boot	LMC	MMC	Asy	Boot	LMC	MMC
nc	0.0	0.0	0.0	0	0.0	0	0.0	0	96.0	96.0	96.0	96.0
c	0.4	0.0	0.0	0	0.8	0	0.4	0	97.6	96.8	97.2	94.4
ct	2.0	0.8	0.4	0	2.0	0	0.0	0	92.8	88.8	88.0	88.0

Table 30: Empirical levels for 250 replications of AR(p) process where $\alpha = 5\%$ and n=50, testing $H_0 : \gamma = 0$ against $H_1 : \gamma > 0$

Size	$y_t = -0.1y_{t-1} - 0.1\Delta y_{t-2} + u_t$				$y_t = -0.1y_{t-1} + 0.1\Delta y_{t-2} + u_t$				$y_t = 0.1y_{t-1} - 0.1\Delta y_{t-2} + u_t$			
	Asy	Boot	LMC	MMC	Asy	Boot	LMC	MMC	Asy	Boot	LMC	MMC
nc	11.2	14.8	9.2	6.0	14.0	21.2	12.8	8.8	95.6	96.0	95.6	95.6
c	5.2	14.0	8.0	1.2	6.4	14.8	14.0	1.6	97.2	96.8	96.0	93.6
ct	6.0	14.0	9.2	2.0	4.0	10.4	6.8	0.4	92.0	88.8	88.0	84.0

Table 31: Empirical levels for 250 replications of AR(p) process where $\alpha = 5\%$ and n=50, testing $H_0 : \gamma = 0$ against $H_1 : \gamma \neq 0$

Looking at the results it seems that the loss of power of MMC compared to the other tests is minimized when the test is well specified (i.e. when the model is nc). Interestingly, the case where $y_t = 0.1y_{t-1} - 0.1\Delta y_{t-2} + u_t$, all tests have really high power.

6   Conclusion

In this paper, we demonstrated how the **MMC** package provides an easy way to implement Monte Carlo techniques such as the MC technique with tie-breaker and the MMC. These techniques are widely applicable to a wide set of problems. In fact, we demonstrated that with irregular Wald-type tests and unit root tests, only the MMC provides valid inferences compared to the LMC, the bootstrap and the traditional asymptotic distribution.

Future development of the **MMC** package will provide additional tools such as new techniques for global optimization, in order to offer better performance and more versatility to users. Additionally, future versions of **MMC** might include C code. As for the moment, the package is only coded in pure R. This should decrease the computational load required by the **MMC** package.