UNIVERSITÉ PARIS 1
PANTHÉON SORBONNE

Econometrics Master's thesis

UFR02 École d'économie de la Sorbonne

Master 1, Économétrie, Statistiques.

Bayesian information criteria and ARMA processes : Performance analysis by Monte-Carlo methods

Authors:
Mehdi Ferhat
Nicolas Turpin
Alexis Vignard

Supervisor:
Dr. Philippe De Peretti

Abstract

This paper focuses on the study of the best information criteria in the context of an ARMA simulation using Monte Carlo methods. It is shown that information criteria have their own efficiency, and that it can be beneficial to use different criteria depending on the case studied. Two methods of estimating ARMAs are tested: the ordinary least squares method and the maximum likelihood method. We will see that, in general, the two criteria that perform best in the ARMA simulation framework are the BIC and the BICc.

Contents

1 Introduction

How can we be sure to choose the best specification for a predictive model?
From maximum likelihood to Akaike's information criteria, through Bayesian infromation criteria and their development towards convergence in the Scwartz sense, there are so many consistent elements that it always seems arbitrary to make one choice rather than another to obtain an ideal ARMA specification.

To best answer this question, we will break down all the steps necessary for the elaboration of this choice, by promoting a global understanding of the underlying phenomena involved.
It is then appropriate to start with a focus on essential theoretical concepts in time series such as stationarity, how to test it, what is a white noise and an autocovariance function, in order to explain the concepts of ACF and PACF.

Many methods called "information criteria" are then studied. These criteria allow to make an optimal choice on the order of an ARMA(p,q) process that we will develop later, to choose a parsimonious model. Thus, we need to question ourselves whether these different information criteria return the same information with the "true" model. It is then necessary to understand what differences there are between these criteria, defining them one by one while explaining how they are constructed.

This will allow us to better understand the different results returned by the different criteria, especially in terms of the sample size of the ARMA generated, but also its order. After studying in detail how ARMA models are identified (autocorrelation and partial autocorrelation functions), models will be generated to test the different information criteria.

The estimation of ARMAs will be tested in two different ways:

- First, within the framework of maximum likelihood estimation, where tables will be drawn up to quantitatively evaluate the different choices of information criteria models.
- Then by the ordinary least squares method with monte carlo simulation, showing this time the number of occurence where the criteria are wrong for each ARMA generated.

The estimation of the process is actually a way to be able to perform a nearly automatic identification of the orders without looking graphically at their ACFs and PACFs. By estimating the process, we ask the information criteria to choose which model is the most parsimonious, i.e., the one that would provide the best predictive quality for the lowest cost.

This srudy is important because the ability to select the "true" ARMA model is essential in the context of short-term financial forecasts for example. We wish to obtain the best forecast for an ARMA(p,q) with parameters p and q at the minimum, thanks to the information criteria.

2 Basic Concepts in Time Series

Before starting the simulation of the ARMA processes, it is advisable to define the elementary notions of time series, and in particular explain how to make the statistical inferences necessary for an ARMA modeling. This paper is indeed intended to be exhaustive in the explanation of the different steps necessary to generate and identify an ARMA. It is then important to give the definition of the notion of stationarity (and how to test it) but also of the autocovariance and autocorrelation functions. Thus this section will be used as a toolbox in order to fully understand how the identification of an ARMA process works (section 4.2).

2.1 Stationarity

2.1.1 Definition

A time series is a series of data measured at regular time intervals. To study these series, it must verify the property of stationarity[1].

A X_t process is stationary in the weak sense if:

$$\begin{cases} \forall_t \in \mathbb{Z}, \mathbb{E}(X_t) = \mu \\ \forall_t \in \mathbb{Z}, \mathbb{V}(X_t) = \sigma^2 \\ \forall_t \in \mathbb{Z}, \forall_h \in \mathbb{Z}, \mathbb{C}ov(X_t, X_{t,t+h}) = \gamma(h) \end{cases}$$

With $\gamma(h)$ the auto-covariance of h order of X_t.

2.1.2 Stationarity Test

The notion of stationarity is elementary when it is necessary to make any statistical inference on a series (here on ARMAs modeling). Thus, firstly, it is necessary to test its stationarity.

Let a series such that:

$$y_t = \rho y_{t-1} + \varepsilon_t$$

In order to test de stationarity of this time serie, David Dickey and Wayne Fuller developed the Dickey-Fuller test in 1979:

- The Dickey-Fuller test (DF):

  Also called unit root test, it has the particularity of assuming that the residuals follow white noise.

  $$\begin{cases} H_0 : \rho = 1 \iff \text{Non-stationary series, case of random walk} \\ H_1 : \rho < 1 \iff \text{Stationary series} \end{cases}$$

  We can rewrite the series as follows:

  $$\begin{aligned} & y_t = \rho y_{t-1} + \varepsilon_t \\ \Leftrightarrow\ & y_t - y_{t-1} = \rho y_{t-1} - y_{t-1} + \varepsilon_t \\ \Leftrightarrow\ & \Delta y_t = (\rho - 1) y_{t-1} + \varepsilon_t \end{aligned}$$

  its test statistic is:

  $$DF = \frac{\rho - 1}{\sigma_\rho}$$

[1] Time Series Analysis and Its Applications, Shumway, Robert H. and Stoffer, David S., 2000

As said previously, this test is problem because of the very restrictive hypothesis of white residuals. However, they are often autocorrelated or even heteroscedastic: the ADF test will more commonly be used:

- The Augmented Dickey-Fuller test (ADF):

This version of the Dickey-Fuller test consists in the addition of an autoregressive component to bleach the residuals. This autoregressive component makes it possible to take into account the effect of the autocorrelation of errors.

$$\Delta y_t = (\rho - 1) y_{t-1} + \varepsilon_t + \underbrace{\sum_{i=1}^{p} \phi_i \Delta y_{t-i}}_{\text{composante } AR(p)}$$

The test statistic remain the same as the original Dicky-Fuller test.

2.2 The white noise

2.2.1 Definition

The residuals follows weak white noise if:

$$\forall \varepsilon_t \in \mathbb{Z} \begin{cases} \mathbb{E}(\varepsilon_t) = 0 \\ \mathbb{V}(\varepsilon_t) = \sigma^2 \\ \mathbb{C}ov(\varepsilon_t, \varepsilon_{t'}) = 0, \text{si } t \neq t' \end{cases}$$

Then, $\varepsilon_t \sim WN(0, \sigma^2)$, meaning that a white noise is by definition stationary. The notion of white noise is essential for understanding how ARMA models work, especially for the moving average component.

2.2.2 Testing the whiteness of a noise: The Ljung-Box Q-stat

This test is commonly used in the framework of ARMA modeling to check if the residuals of the model are autocorrelated or not. This test is essential because the statistical inference of an ARMA model is problematic if its residuals are correlated. The ljung-box Q-test makes it possible to asymptotically test the autocorrelation at orders greater than 1.

$$\begin{cases} H_0 : \text{Independently distributed residuals} \Leftrightarrow \text{non-autocorrelated residuals} \\ H_1 : \text{Non-independent distributed residuals} \Leftrightarrow \text{autocorrelated residuals} \end{cases}$$

Test's Statistic:

$$Q = T(T+2) \sum_{i=1}^{p} \frac{\rho_i^2}{T-i} \overset{H_0}{\sim} \chi^2(p)$$

Decision rule:

$$Q > \chi^2_{1-\alpha,k}$$

with k degrees of freedom and α quantile of the chi squared distribution.

2.3 The autocovariance and autocorrelation functions

In this study, we will see how to graphically identify the order of an ARMA. This identification is done through two functions, which are called autocorrelation function and partial autocorrelation function. These functions are found from the autocovariance function.

2.3.1 The autocovariance function

The autocovariance function of X_t is defined by:

$$\gamma_h = \mathbb{C}ov(X_t, X_{t+h}) = \mathbb{E}[[X_t - \mathbb{E}(X_t)][X_{t+h} - \mathbb{E}(X_{t+h})]]$$

It measures the covariance between t and h. It makes it possible to characterize linear dependencies within the process X_t.

2.3.2 The autocorrelation function (ACF)

The autocorrelation function of X_t is defined by:

$$\forall h \in \mathbb{Z},\ \rho_h = \mathbb{C}orr(X_t, X_{t+h}) = \frac{\mathbb{C}ov(X_t, X_{t+h})}{\sigma_{X_t}\sigma_{X_{t+h}}} = \frac{\gamma_h}{\gamma_0}$$

With γ_h autocovariance function, and $\rho_h \in [-1, 1],\ \forall h \in \mathbb{Z}$.

It measures the chronological correlation between two periods of the X_t process.

2.3.3 The partial autocorrelation function (PACF)

The partial autocorrelation function of X_t is defined by

$$\rho_h = \mathbb{C}orr(X_t, X_{t+h}/X_{t+1}, ..., X_{t+h-1})$$

It is interpreted as the correlation between the date t and h, by removing every correlations between those two dates. It therefore removes the information between t and h.

It can also be noted as a linear regression

$$EL(X_t/X_{t+1}, ..., X_{t+h-1})$$

The partial autocorrelation function is then denoted by:

$$r_h = \mathbb{C}orr(X_t - EL(X_t/X_{t+1}, ..., X_{t+h-1}), X_{t+h} - EL(X_t/X_{t+1}, ..., X_{t+h-1}))$$

All the information of the intermediate variables is removed.

3 Information criteria

In this section, we present various information criteria. These criteria allow parsimonious choices to be made about the orders of a process (e.g. an $ARMA(p, q)$).
The term "parsimonious" is used to describe a model with the most efficient number of parameters: the model will contain as few parameters as possible. We will see that these information criteria have a different computational approach, and that one criterion may be more suitable than another depending on different cases, different number of observations.

Hannan and Quinn (1979) presented necessary conditions on the penalty functions to determine the order of convergence of an ARMA model. They discovered in their work that the AIC, AICc, FPE and Cp criteria do not converge[2]. On the contrary, they will discover that other criteria such as BIC, SIC and HQ converge. Furthermore, a new approach was revealed in a univariate regression model, suggesting the use of three corrected variables: FPEu, AICu and HQc ("u" for unbiased and "c" for corrected). These three variants were shown to outperform their original criteria. In particular, there is strong competitiveness between AICu and HQc. The reason why these new criteria perform better is their efficiency and asymptotic consistency (i.e. zero overfitting probability).

3.1 Akaike Information Criterion (AIC)

The Akaike information criterion expression is the following:

$$AIC = -2\ln L(\theta) + 2k$$

Where k is the number of parameters in the model. The model that we will choose is the one that minimizes the AIC[3]. Theoretically it is supposed to be the "best" model.
The Akaike criterion comes from the Kullback-Lieibler divergence on information theory.

Let f and g be two probability densities assuming that f is the true unknown law and g an approximation. Consider then that the divergence (i.e. the loss of information) for using g instead of f is defined by:

$$I(f;g) = \int f(t)\ln\frac{f(t)}{g(t)}dt$$

This loss of information can be expressed as the difference between the mathematical expectations and the true law:

$$I(f;g) = \int \ln\big(f(t)\big)f(t)dt - \int \ln\big(g(t)\big)f(t)dt = \mathbb{E}_f\Big(\ln\big(f(t)\big)\Big) - \mathbb{E}_f\Big(\ln\big(g(t)\big)\Big)$$

We wish to compute $\mathbb{E}_f\Big(\ln\big(g(t;\theta)\big)\Big)$ by maximizing θ, because $g(t;\theta)$ is the closest element to f that we do not know. It is for this reason that we wish to estimate θ which we will note $\hat{\theta}$: it is the maximum likelihood estimator. The expression that we wish to calculate becomes $\mathbb{E}_f\Big(\ln\big(g(t;\hat{\theta})\big)\Big)$.
$\hat{\theta}$ will depend on the data used, so the expression mentioned will be by definition random. The data also follow by definition the true law of f, we will then have to take the expectation of our expression to determine this law: $\mathbb{E}_{\hat{\theta}}\mathbb{E}_f\Big(\ln\big(g(t;\hat{\theta})\big)\Big)$.
Since we do not know f, this is not computable. On the other hand, by using a Taylor expansion and making some assumptions, Akaike showed that asymptotically $\mathbb{E}_{\hat{\theta}}\mathbb{E}_f\Big(\ln\big(g(t;\hat{\theta})\big)\Big) \sim \ln L(\hat{\theta}) - k$.
Multiplying by -2, we obtain the AIC: $-2\ln L(\hat{\theta}) + 2k$.

Also, we can find a different writing of this criterion, which can be deduced thanks to the ordinary least squares estimator:

$$AIC = \ln\hat{\sigma}^2_{(k)} + \frac{2k}{T}$$

[2]Regression and time series model selection, Allan D R McQuarrie; Chih-Ling Tsai, 1998
[3]Journal of the Royal Statistical Society. Series B (Methodological): An Exact Maximum Likelihood Estimation Procedure for Regression-ARMA Time Series Models with Possibly Nonconsecutive Data, 1986

With k representing the order of an AR process, or the sum of AR and MA orders in an ARMA model. In both cases, the best choice will be the one where the AIC is minimal.

3.2 Akaike Information Criterion with correction (AICc)

The corrected Akaike information criterion would be more efficient than the AIC when the number of parameters k is large compared to the number of observations. Here is the expression given for this new criterion, by Hurvich and Tsai (1989):

$$AICc = T \ln\left(\frac{RSS}{T}\right) + \frac{2T(k+1)}{T-k-2}$$

$$\Leftrightarrow AICc = T \ln \hat{\sigma}^2_{(k)} + T\frac{1+k/T}{1-(k+2)/T}$$

In theory, if $T/k <\approx 40$, it would be advisable to favor AICc over AIC. We could also rewrite the criterion more generally, in the following form:

$$AICc = AIC + \frac{2k(k+1)}{n-k-1}$$

However, it is important to note that when the number of parameters k is the same whatever the model, we do not find significant differences between the use of an AIC or an AICc. In this particular case, there would be no contraindication to use a classical AIC.

3.3 Akaike Information Criterion Unbiased (AICu)

As we have seen, the AICc information criterion provides better order choices for a time series model than the AIC, depending on the sample size.
As for the AICu[4], it would be an approximate unbiased estimator of the Kullback-Lieibler information.

$$AICu = T \ln\left(\frac{RSS}{T-k}\right) + \frac{2T(k+1)}{T-k-2}$$

This criterion would present a compromise between the efficient (AICc) and consistent (AIC, BIC) information criterion. In other words, by performing simulations, we should find that the AICu performs better for medium to large sample sizes (only when the true model is not of infinite order). In theory, it should also perform better than the BIC that we will present shortly after, unless a true model exists and the sample size is large enough.

3.4 Final Prediction Error (FPE)

Akaike develops another information criterion, the final prediction error (FPE). We can estimate it by:

$$FPE = \hat{\sigma}^2_{(k)}\left(1 + 2\frac{k+1}{T}\right)$$

This criterion aims at minimizing the final prediction error of a time series model, which may present a specification bias. This criterion is interesting because its penalty is similar to that of the AIC, and is intended to be more parsimonious.
Several alternative versions of the FPE criterion exist, notably an unbiased FPE noted FPEu[5].

[4]Statistics Probability Letters, Allan Mc Quarriea Robert Shumwayb Chih-LingTsaic, 1997
[5]The Effectiveness of Information Criteria in Determining Unit Root and Trend Stratus, R. Scott Hacker, 2010

$$FPEu = \frac{n+k}{n-k} s_k^2$$

We can estimate it as follows:

$$FPEu = \hat{\sigma}_k^2 \frac{n+k}{n-k}$$

In terms of signal-to-noise ratio, the FPEu would be more efficient or equal than the FPE, in an overfitting case

3.5 Bayesian Information Criterium (BIC, SIC, SC, SBC, SBIC)

The Bayesian information criterion (BIC) is:

$$BIC = -2 \ln L(\theta) + \ln(n)k$$

That we estimate by:

$$BIC = -2 \ln L(\hat{\theta}) + \ln(n)k$$

We can notice an equation almost similar to the one of the AIC. Here, the difference is that the penalty is stronger, because it depends on the number of observations.
Therefore, in the case of large samples, the BIC would favor models with fewer parameters than the AIC.

Moreover, although similar in their equation form, the AIC and the BIC come from a very different context insofar as the BIC corresponds to a Bayesian choice of models.
To illustrate this, let us take as an example a finite family of m models which we will note M_i depending on θ_i: a vector parameter. We will note $\mathbb{P}(M_i)$ the probabilities *a priori* for each model. By noting θ_i the distribution *a priori* for each of the models $\mathbb{P}(\theta_i/M_i)$, it would be appropriate that the probability *a posteriori* of the model M_i knowing the data x is proportional to $\mathbb{P}(M_i)\mathbb{P}(x/M_i)$.
Considering that the probabilities *a priori* $\mathbb{P}(M_i)$ are equivalent to each other, we would obtain a probability for the model *a posteriori* M_i proportional to $\mathbb{P}(x/M_i) = \int \mathbb{P}(x/M_i;\theta_i)\mathbb{P}(\theta_i/M_i)d\theta_i$ which we call integrated likelihood (or Marginal likelihood).

By generating a limited expansion in the neighborhood of our maximum likelihood estimator, while keeping the idea that the probabilities *a priori* $\mathbb{P}(M_i)$ are uniform (that one model cannot be favored over another), this criterion shows that $\ln\big(\mathbb{P}(x/M_i)\big) \sim \ln\big(\mathbb{P}(x/\hat{\theta}_i, M_i)\big) - \frac{k}{2}\ln(n)$.

With $\ln\big(\mathbb{P}(x/\hat{\theta}_i, M_i)\big)$ the log likelihood for the M_i model. To choose the most likely *a posteriori* M_i model, we must select the minimum BIC. After computing the Bayesian criteria for each model, the probability *a posteriori* would be:

$$\mathbb{P}(M_i/x) = \frac{e^{-0.5BIC_i}}{\sum_{j=1}^{m} e^{-0.5BIC_j}}$$

These probabilities are used to weight the models, so we would obtain a weighted average prediction that we call *model averaging*.

It is also possible to obtain the BIC by using the OLS, its most common form is the following:

$$BIC = -2\sum \ln g(.|., k) + k \ln T$$

Alternatively, there is also a corrected version of the BIC, where n is replaced by the number of subjects *(Raftery, 1995)* : it is called the BICc[6].

$$BICc = -2\ln L(\hat{\theta}) + k\ln(T)$$

As part of the ARMA processes, Schwartz (1978) derives this criterion from exponential family distributions, we then obtain the Scwhartz criterion, which can appear under several names such as SIC, SC, SBC, SBIC:

$$SIC = \ln \hat{\sigma}^2_{(k)} + \frac{k\ln T}{T}$$

$$\Leftrightarrow SIC = T\ln\Big(\frac{RSS}{T}\Big) + \big(\ln T\big)k$$

In the literature, we can find modifications on this criterion: notably Jorma Rissanen (1987, 1988) with his approach of *shortest data description* or *Minimum description length* (MDL) inspired by Shannon's encoding theory. Finally, the criterion obtained will be similar to BIC.

3.6 Hannan Quinn Infomation Criterium (HQ, HQc, HQIC)

In the literature, we can find the Hannan Quinn information criterion under different names. In our case, we will retain the name "HQ"[7]. This criterion was proposed by Hannan and Quinn (1979) especially for ARMA models. The theory is that this criterion is an intermediate between the AIC and the BIC.
The likelihood equation of this criterion is the following:

$$HQ = HQc = HQIC = -2L_{max} + 2k\ln\big(\ln(n)\big)$$

Where L_{max} represents the log likelihood, n the number of observations, and k the number of parameters.

The Hannan Quinn criterion would propose to "soften" the "harshness" that the BIC penalty function can have, relative to the growth of the sample size. This would be achieved while maintaining a strong convergence in the identification of the true order of the ARMA. To do this by OLS estimation, a parameter $c > 1$ must be defined.

$$HQ = \ln \hat{\sigma}^2_{(k)} + 2c\frac{k\ln(\ln T)}{T}$$

A remark to make would be that it is not easy to choose the value of c, and that we have too little information to help in this choice.
Hannan and Quinn themselves use the limit value of $c = 1$ to carry out their tests in simulation processes, in spite of the demonstrations made that $c > 1$ is necessary to obtain a convergence of the HQ criterion.
Nevertheless, in order to do simulations, and in spite of this important information, we will also take the limiting case $c = 1$ in our test phase (in the absence of knowing the true value of c).

By adding a penalty to the HQ equation, by construction we obtain the HQc (the corrected Hannan Quinn criterion):

$$HQc = \ln\big(\hat{\sigma}^2_k\big) + \frac{2\ln\big(\ln(n)\big)k}{n-k-2}$$

Theoretically, this new criterion not only corrects the performance for small samples, it is also an asymptotically consistent criterion (probability of overfitting = 0). This is what makes the HQc criterion particularly interesting in terms of performance.

[6]Comparison of model-based tests and selection strategies to detect genetic polymorphisms influencing pharmacokinetic parameters, Julie Bertrand, Emmanuelle Comets, and France Mentre, 2008

[7]Évaluation de critères d'information pour les modèles de séries chronologiques, John W. Galbraith et Victoria Zinde-Walsh, 2004

3.7 Mallow's C_p

When we encounter a problem of over-fitting (the more variables there are in a model, the smaller the sum of squares of the residuals becomes), we know that the model is not really performing. For example, we could put an infinite number of variables unrelated to the object they explain, yet the error would seem to be minimized. In such a case, we use Mallow's C_p.
The Mallow statistic would allow us to estimate the sum of the squared prediction error, denoted SSE_p, on a sample of data. This defines the target population for Mallow's method:

$$\mathbb{E}\sum_i \Big(\hat{Y}_i - \mathbb{E}(Y_i|X_i)\Big)^2/\sigma^2$$

In this way, the mean squared prediction error will not decrease with the number of variables added to the model.
The model then retains the sample size, the effect sizes of the different predictors, as well as their degree of collinearity.
As soon as $K > P$, the Mallow statistic is of the following form:

$$C_p = \frac{SSE_p}{S^2} - N + 2(P+1)$$

With $SSE_p = \sum_{i=1}^{N} \big(Y_i - Y_{pi}\big)^2$ representing the sum of squared errors for a number P of regressors. We also note Y_{pi}, a term referring to the prediction of the i"th" observation of Y, according to the number of regressors P. For the other terms, we classically find the sample size N and the mean squared error S^2.

There is a different version of Mallow's C_p. For example, in a linear regression of the form:

$$Y = \beta_0 + \beta_1 X_1 + ... + \beta_p X_p + \varepsilon$$

Where our $\beta_0, ...\beta_p$ are the predictor coefficients of the variables $X_1, ..., X_p$ and our ε represent the error terms; we can find the following form for the Mallow C_p expression:

$$C_p = \frac{1}{n}\big(RSS + 2d\hat{\sigma}^2\big)$$

With RSS, the expression for the residual sum of squares on a test model, d representing the number of predictors associated with the model, and $\hat{\sigma}^2$ an estimated variance based on the regression model.
The interest of presenting the Mallow statistic in this different form of calculation is that the minimum value of this C_p will be the same as the previous one, but calculated differently.
In our study, the use of OLS combined with the latter statistic, would be the more appropriate of the two.

Note that this criterion is valid only if the sample size is large, and that it does not allow us to do variable selections.

4 ARMA models

4.1 AR, MA, ARMA and ARIMA models presentation

As seen above, the information criteria allow the best model to be chosen sparingly. This selection of model passes by the identification of the orders of the ARMAs. In this section, we focuses on these models developed by the statisticians George Box and Gwilym Jenkins.

They developped the Box Jenkins approach which consists in the decomposition a time series into two representations:

- An autoregressive component, the AR(p)
- A moving average component, the MA(q)

4.1.1 The autoregressive process AR(p)

The autoregressive component of a time series consists in the modeling of its past (with different time lags).

$$AR(p): \ y_t = \mu + \sum_{i=1}^{p} \phi_i y_{t-i} + \varepsilon_t$$

The moments of an $AR(p)$ process can be calculated using the Yule-Walker equations. Firstly, in order to calculate those moments, the stationarity condition must be verified.

An $AR(p)$ process is stationary when all the roots of its characteristic polynomial have a modulus other than 1 (thus differentiating it from the random walk). It should also be noted that with Wold's results, any causal $AR(p)$ (i.e for which all the characteristic roots of the polynomial are greater than 1 in absolute value, and therefore with residuals following the innovation process) can be decomposed into a $MA(\infty)$ process. This decomposition is particularly useful for demonstrating the stationarity of an autoregressive process, because as we will see later, a $MA(q)$ process is by definition stationary.

The Backward Operator B: In order to obtain its polynomial representation, the backward operator $BX_t = X_{t-1}$ is commonly used. It allows us to highlight the unit root of the process and thus to easily test its stationarity.

$$\phi(B)X_t = \mu + \varepsilon_t \Leftrightarrow \left(1 - \sum_{i=1}^{p} \phi_i B^i\right) X_t = \mu + \varepsilon_t$$

The process will therefore be stationary if and only if the roots of the characteristic polynomial $1 - \sum_{i=1}^{p} \phi_i B^i$ have a modulus different from 1.

Then, the first moment of an $AR(p)$ process is:

$$\mathbb{E}(X_t) = \frac{\mu}{1 - \sum_{i=1}^{p} \phi_i}$$

For the variance and autocovariance, we use the Yule-Walker equations:

$$YW: \gamma_h = \sum_{i=1}^{p} \phi_i \gamma_{h-i} \quad \forall h = 1, ..., p$$

With γ_h autocovariance function of the process of order h (cf 2.3.1).

$$\mathbb{V}(X_t) = \sum_{i=1}^{p} \phi_i \gamma_h + \sigma^2 \quad \forall h = 1, ..., p$$
$$\mathbb{C}ov(X_t, X_{t-h}) = \sum_{i=1}^{p} \phi_i \gamma_{h-i} \quad \forall h = 1, ..., p$$

4.1.2 The Moving Average process MA(q)

The moving average component captures the non-linear trend of the series. We assume that the residuals are white and *i.i.d.* This process indicates whether or not the series depends on a succession of white noise. It smooths over the mistakes of the past.

A $MA(q)$ process is defined as:

$$MA(q): \ y_t = \mu + \sum_{i=1}^{q} \theta_i \varepsilon_{t-i} + \varepsilon_t$$

By adding its backward operator, we get:

$$\theta(B) = \theta_0 + \theta_1 B^1 + ... + \theta_q B^q \Leftrightarrow \theta(B) = \sum_{i=1}^{q} \theta_i B^i$$

As we can see, a $MA(q)$ process is a succession of white noise. They are stationary and not auto-correlated. Indeed, $\varepsilon_t \sim WN(0, \sigma^2)$. Thus, for every q, a $MA(q)$ process is by construction stationnary.

We can therefore proceed to the calculation of its moments:

$$\begin{cases} \mathbb{E}(X_t) = \mu \\ \mathbb{V}(X_t) = \sigma^2(1 + \theta_1^2 + ... + \theta_q^2) = \sigma^2 \left(1 + \sum_{i=1}^{q} \theta_i^2\right) \end{cases}$$

4.1.3 The ARMA(p,q) process

Box and Jenkins combined both of the processes in order to get a more accurate prediction:

$$ARMA(p,q): \ y_t = \mu + \underbrace{\sum_{i=1}^{p} \phi_i y_{t-i}}_{AR(p)} + \underbrace{\sum_{i=1}^{q} \theta_i \varepsilon_{t-i}}_{MA(q)} + \varepsilon_t$$

By applying the backward operator to an ARMA, we obtain (for $\mu = 0$):

$$ARMA(p,q): X_t = \frac{\theta(B)}{\phi(B)} \varepsilon_t$$

There are different cases:

- AR "pure": An AR process is called "pure" if it corresponds to an ARMA process which does not contain a moving average component. The process can also be written $ARMA(p, 0)$.
- MA "pure": Conversely, an MA is called "pure" if it corresponds to an ARMA process which does not contain an autoregressive component. The process can also be written $ARMA(0, q)$
- ARMA (p,q) with p and q greater than 0 which corresponds to the assembly of the two processes seen above.

4.1.4 The ARIMA(p,d,q) process

As mentioned earlier, it is imperative to check whether the ARMA process is stationary if we want to make a statistical inference. However, ARMA processes are not necessarily stationary, this is where the ARIMA process comes in.

The ARIMA process differs from ARMA in that it is a d-order differentiated ARMA. Thus, an ARIMA process is based on integrated series. Differentiating our series allows us to stationary it.

We note a first difference as follows:

$$\Delta^1 y_t = y_t - y_{t-1}$$

An integrated process of order 1, means that the ARMA had to be differentiated once to make it stationary, so we have an $ARIMA(p, 1, q)$. As an illustration we can give the example of the ARIMA(1,1,0)

$$\begin{cases} y_t - y_{t-1} = \phi(y_{t-1} - y_{t-2}) + \varepsilon_t \\ \Leftrightarrow y_t = (1 + \phi)y_{t-1} - \phi y_{t-2} + \varepsilon_t \end{cases}$$

4.2 Identification of the order of the processes

The order of the different processes can be identified by using the autocorrelation and partial autocorrelation functions seen previously (section 2.3). Indeed, depending on the model, the autocorrelations will have very characteristic shapes, allowing the order of the processes to be identified. In this section we will see how the identification of the different processes works, but also their links with the Yule-Walker equations.

4.2.1 Identification of an AR(p) process

The identification of an autoregressive process is based on the autocorrelation and partial autocorrelation functions. It is possible to find these functions using the Yule-Walker equations and the Durbin algorithm.

i) Autocorrelation function and Yule-Walker's equations:

Firstly, we will first look at the link between the autocorrelation function of an autoregressive process, and the Yule-Walker equations[8]:

Let's consider the following autoregressive process:

$$y_t - \phi_1 y_{t-1} - \ldots - \phi_p y_{t-p} = \varepsilon_t$$

[8]Econométrie : Théorie et applications, Valérie Mignon

Firstly, $\forall\ h > 0$, we multiply each member of the equation by y_{t-h}. Then, by taking the expectated value of each variable, we divide by γ_0. We get:

$$(\mathbb{E}[y_t y_{t-h}] - \phi 1 \mathbb{E}[y_{t-1} y_{t-h}] - \ldots - \phi_p \mathbb{E}[y_{t-p} y_{t-h}]) \tfrac{1}{\gamma_0} = \mathbb{E}[\underbrace{\varepsilon_t}_{WN(0,\sigma^2)} y_{t-h}] \tfrac{1}{\gamma_0}$$
$$\Leftrightarrow (y_h - \phi_1 y_{h-1} - \ldots - \phi_h y_{h-p}) \tfrac{1}{\gamma_0} = 0$$

Because $\mathbb{E}[\varepsilon_t y_{t-h}] = 0$.

Moreover, we recall that the autocorrelation function is defined by $\rho_h = \frac{\gamma_h}{\gamma_0}$, then by expanding we find the autocorrelation function of an AR(p) process:

$$\rho_h - \phi_1 \rho_{h-1} - \ldots - \phi_p \rho_{h-p} = 0$$
$$\Leftrightarrow\ \forall\ h > 0,\ \rho_h = \sum_{i=1}^{p} \phi_i \rho_{h-i}$$

And we finally get the Yule-Walker equations:

$$\begin{pmatrix} \rho_1 \\ \vdots \\ \rho_p \end{pmatrix} = \begin{pmatrix} 1 & \rho_1 & \rho_2 & \ldots & \ldots & \rho_{p-1} \\ \rho_1 & 1 & \rho_1 & \rho_2 & \ldots & \rho_{p-2} \\ \rho_2 & \rho_1 & 1 & \rho_1 & \ldots & \vdots \\ \vdots & \vdots & \vdots & \ddots & & \vdots \\ \vdots & \vdots & \vdots & & \ddots & \vdots \\ \rho_{p-1} & \rho_{p-2} & \rho_{p-3} & \ldots & \ldots & 1 \end{pmatrix} \begin{pmatrix} \phi_1 \\ \vdots \\ \phi_p \end{pmatrix}$$

ii) Partial autocorrelation and Durbin's algorithm:

It is possible to find the partial autocorrelations of an autoregressive process from the Durbin algorithm. Indeed, just like the Yule-Walker equations for the autocorrelation function of an AR(p) process, we will demonstrate that we can use the Durbin algorithm to calculate the partial autocorrelation from the Yule-Walker equations:

$$Durbin: \begin{cases} \phi_{11} = \rho_1 \\ \phi_{hh} = \dfrac{\rho_h - \sum_{i=1}^{h-1} \rho_{h-i} \phi_{h-1}}{1 - \sum_{i=1}^{h-1} \rho_h \phi_{h-1}},\ h = 2,\ 3,\ 4\ \ldots \\ \phi_{hi} = \phi_{h-1,i} - \phi_{hh} \phi_{h-1,h-i},\ h = 2,\ 3,\ 4 \ldots ;\ \text{i} = 1,\ 2,\ 3\ \ldots \end{cases}$$

Thus, on an AR(1) process, if we run the Durbin algorithm from the Yule-Walker equations:

$$\Leftrightarrow \begin{cases} \phi_{11} = \rho_1 \\ \phi_{22} = \dfrac{\rho_2 - \phi_{11}\rho_1}{1 - \phi_{11}\rho_1} = \dfrac{\rho_2 - \rho_1^2}{1 - \rho_1^2} = \dfrac{\phi_1^2 - \phi_1^2}{1 - \phi_1^2} = 0 \\ \phi_{hh} = 0\ \forall h \geq 2 \end{cases}$$

We obtain our partial autocorrelation, it becomes zero at rank $h1$.

iii) Graphical detection of an autoregressive process with its ACF and PACF

Now that the autocorrelation and partial autocorrelation functions have been demonstrated, we will see how to graphically identify an AR process from its autocorrelograms.
A "pure" autoregressive process (i.e. an ARMA(p,0)) has specific characteristics: its ACF decreases progressively while its PACF drops sharply as soon as the order p of the AR is reached.

For illustrative purposes, we model a process such as:

$$y_t = -0.8y_{t-1} + \varepsilon_t$$

With T=200.

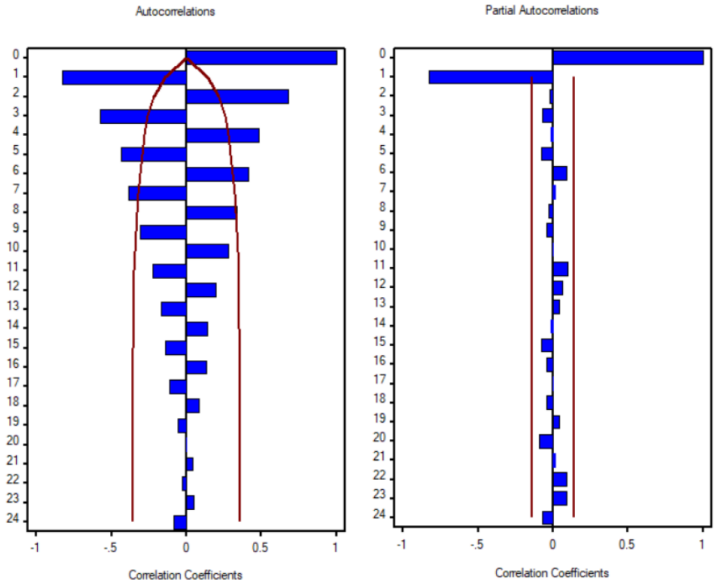

Figure 1: ACF et PACF d'un AR pur, ou ARMA(1,0)

The ACF, on the left, shows us a damped sinusoid, which means that the ACF is slowly decreasing. This is characteristic of the pure autoregressive process. To know its p order, we look at its PACF: after order 1, we observe a clear break, then no other lag is significant: it is a "pure" AR(1).

4.2.2 Identification of an MA(q)

In the same way as for the autoregressive process, we will study here the autocorrelation and partial autocorrelation functions, but this time for the case of the moving average process.

i) Fonction d'autocorrélation d'un MA(q) et fonction d'autocovariance

In the case of the moving average process, the autocorrelation function is found from the autocovariance function:

$$\gamma_h = \mathbb{E}[y_t y_{t-h}] = \mathbb{E}\left[\left(\varepsilon_t - \sum_{i=1}^{q}\theta_i\varepsilon_{t-i}\right)\left(\varepsilon_{t-h} - \sum_{i=1}^{q}\theta_i\varepsilon_{t-h-i}\right)\right]$$

The autocovariance of a moving average process can then take the following values:

$$\gamma_h = \begin{cases} \sigma_\varepsilon^2(-\theta_h + \theta_1\theta_{h+1} + \ldots + \theta_{q-h}\theta_q \;\forall\, h \;=\; 1, \ldots q \\ 0 \;\forall\, h > q \end{cases}$$

From the autocovariance expression above, we find the autocorrelation function of the process:

$$\rho_h = \begin{cases} \dfrac{-\theta_h + \theta_1\theta_{h+1} + \ldots + \theta_{q-h}\theta_q}{1 + \sum_{i=1}^{q} \theta_i^2} & \forall h = 1, 2, \ldots, q \\ 0 \ \ \forall h > q \end{cases}$$

Then, in the case of a moving average process of order q, its autocorrelation is null $\forall h > q$.

ii) Partial autocorrelation function and Durbin's algorithm

As for the AR(p) process, the partial autocorrelation of the MA(q) is calculated with the Durbin algorithm. This can be shown for the case of a MA(1):

$$\Leftrightarrow \begin{cases} \theta_{11} = \rho_1 \\ \theta_{22} = \dfrac{-\rho_1^2}{1 - \rho 1^2} = \dfrac{-\theta_1^2(1-\theta_1^2)}{1-\theta_1^6} \\ \theta_{hh} = \dfrac{-\theta_1^h(1-\theta_1^2)}{(1-\theta_1^{2(h+1)})} \end{cases}$$

It can be deduced that $\theta_{hh} < 0 \ \forall h$ si $\theta_1 > 0$, et θ_{hh} oscille entre le positif et le négatif $\forall h$ si $\theta_1 > 0$.

iii) Graphical detection of the orders of an MA(q) process with its ACF and PACF

In the case of a pure MA(q) (i.e. an ARMA(0,q)), its ACF drops sharply as soon as the order q is reached, while the PACF decreases slowly. As an illustration, we model the following process:

$$y_t = 0.7\varepsilon_{t-1} + \varepsilon_t$$

With T = 200.

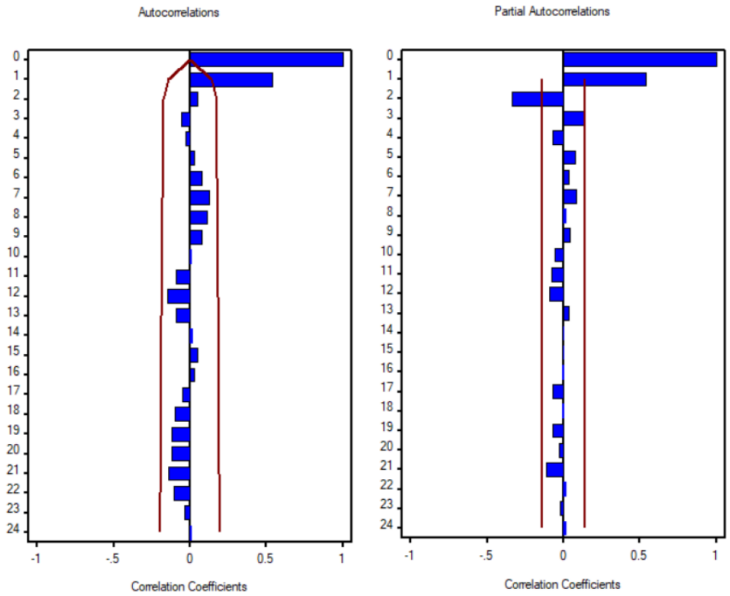

Figure 2: ACF et PACF d'un MA pur, ou ARMA(0,1)

We observe a damped sinusoidal on its PACF that oscillates between positive and negative, indicating that the MA(q) process is pure. The ACF, on the other hand, shows a clear break after lag 1. All other lags are insignificant. We can therefore easily detect an MA(1) process.

The pure AR and MA processes can therefore be easily identified graphically by means of autocorrelograms. The profiles are summarised in the following table:

	ACF	PACF
AR(p)	Slow decay	Clear break after p
MA(q)	Clear break after q	Slow decay

4.2.3 Identification of an ARMA(p,q)

The autocorrelation and partial autocorrelation functions of an ARMA(p,q) are not as easily obtained as for pure ARs and MAs. The same is true for their connection with the Yule-Walker equations and the Durbin algorithm. Thus, in this section, we will only be interested in graphical detection from autocorrelograms for this section.
Indeed, the graphical identification was trivial for the case of pure AR and MA. However, when the orders p and q of an ARMA are simultaneously different from 0, there is no official characteristic form of the autocorrelograms: it depends on the sign of the parameters.

For the example of an ARMA(1,1), there are 4 scenarios:

- $\phi > 0$ et $\phi > \theta$ the autocorrelation is positive
- $\phi > 0$ et $\phi < \theta$ the autocorrelation is negative
- $\phi < 0$ et $\phi > \theta$ the autocorrelation alternates between positive and negative
- $\phi < 0$ et $\phi < \theta$ the autocorrelation alternates between negative and positive

To illustrate this problem, the following ARMA process is modeled:

$$y_t = -0.8y_{t-1} + 0.7\varepsilon_{t-1} + \varepsilon_t$$

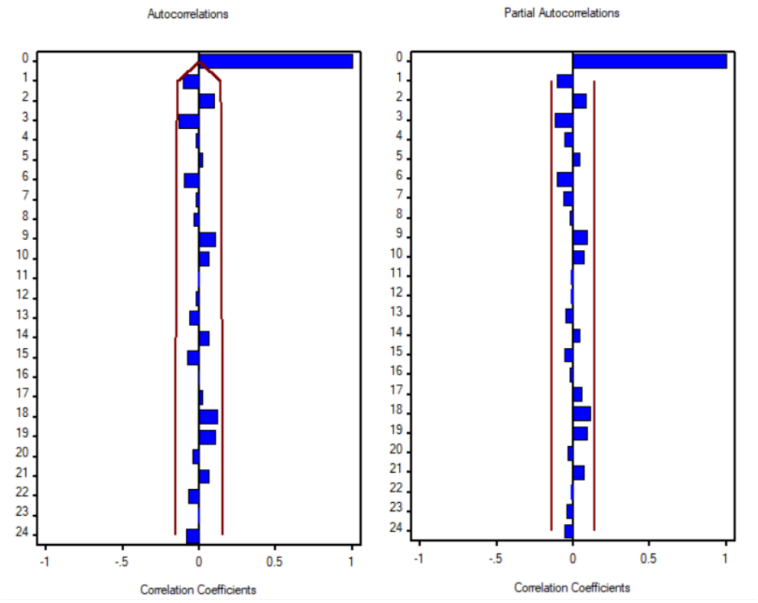

Figure 3: ACF et PACF d'un ARMA(1,1)

Identifying the process seems immediately more delicate. The ACF and PACF show no significant lag above 0. It is impossible to guess the order of our process from its autocorrelograms.

In these cases, we will prefer to directly estimate the processes by scanning the order 0 to p, and 0 to q for each pair (p,q). We will identify the "true" model using the different information criteria seen previously.

4.2.4 Identification d'un ARIMA (p,d,q)

To identify an ARIMA model, we use the same principle as for ARMAs but with differentiated series of order d. In the case of a non-stationary ARMA, differentiating our processes allows us to stationaryize the series (and thus to estimate it).

4.3 Estimation of ARMA and ARIMA processes

Since the graphical identification of ARMAs is very complicated, it is necessary to estimate the processes directly and to use information criteria to proceed with the identification. There are two main methods: The first one is the maximum likelihood method (used by SAS by default), and the second method is the OLS estimation.

4.3.1 Maximum likelihood estimation

This first method is the most common. It consists in maximising the maximum likelihood in order to obtain the best model. This algorithm is used by SAS by default to estimate ARMAs.
It is then necessary to explain how this estimation works in an ARMA process:

Let ψ and X such that:

$$\psi = \begin{pmatrix} \phi_1 \\ \phi_2 \\ \vdots \\ \phi_p \\ \theta_1 \\ \theta_2 \\ \vdots \\ \theta_q \\ \sigma_2 \end{pmatrix}, \; X = \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_T \end{pmatrix} \text{ with T the sample size.}$$

The likelihood function is:

$$\mathcal{L}(\psi|X) = f(x_1, x_2, \ldots, x_T; \psi)$$

And the maximum likelihood estimator is:

$$\hat{\psi}_{ML} = \arg\max_{\psi \in \Psi} \mathcal{L}(\psi|X)$$

As said before, the generated ARMAs must be stationary to be estimated, their residuals of the process must follow a white noise.
Thus, the estimator of the exact Gaussian likelihood of an ARMA process is:

$$\log \mathcal{L}(\psi|X) = -\frac{1}{2}\left(T \log(2\pi) + \log|\Gamma(\psi)| + X'\Gamma(\psi)^{-1}X\right)$$

with $\Gamma(\psi)$ covariance matrix of X on ψ

However, the major problem with this method is that the algorithm can fail.

4.3.2 OLS estimation

To overcome this problem of failure of likelihood maximization, it is also possible to estimate ARMA models using a very simple estimation method: ordinary least squares. This method has the significant advantage of never failing, thus allowing us to obtain reliable results using Monte Carlo simulation (and eventually drawing contourplots).

Thus, in this section, we will study in detail the process of estimating an ARMA by OLS. First, we will see how this estimation is done in theory, and then we will see how it works in SAS using the selection matrix.

i) OLS estimation with ARMA process:

Firstly, the OLS estimaor is given by:

$$\begin{cases} \hat{\beta}_{OLS} = (X'X)^{-1}X'y \\ \hat{\varepsilon_t} = y - X\hat{\beta} \end{cases}$$

However, in ARMA models, the MA processes are composed of lagged residuals that are unknown. It is then necessary to first estimate the AR(p) by OLS to recover its residuals, then to re-inject them into the MA(q) process in order to estimate the ARMA model.

To illustrate it, the following ARMA(1,1) is modeled:

$$ARMA(1,1):\ y_t = \mu + \phi_1 y_{t-1} + \theta_1 \underbrace{\varepsilon_{t-1}}_{Unknown} + \varepsilon_t$$

We filter the data, while taking out the residuals, and we estimate the autoregressive process by OLS:

$$y_t = \mu + \sum_{t=1}^{p} \Phi_1 y_{t-1} + \underbrace{\hat{\varepsilon_t}}_{i.i.d}$$

These estimated residuals are re-injected into the original ARMA(1,1) process on the lagged residuals (which are now known). Finally, we can estimate the initial ARMA(1,1) by OLS. Thus, we will finally estimate the ARMA model by OLS:

$$ARMA(1,1):\ y_t = \mu + \phi_1 y_{t-1} + \theta_1 \underbrace{\hat{\varepsilon_{t-1}}}_{Injected} + \varepsilon_t$$

This method has the advantage of never failing, but we have to keep in mind that we use estimated residuals, and that there is a significant margin of error. We add a random component to our model by estimating the residuals beforehand. Our parameters θ and ϕ will thus have a higher variance, thus distorting all the tstats.

ii) Algorithmic procedure of the OLS estimation: The selection matrix S

In order to estimate our ARMA with OLS, we use the selection matrix in SAS, which allows us to obtain the lags wanted. So, before starting the estimation, it is necessary to calculate the selection matrix by choosing the number of lags.

In our case study, we set a lag of 4 to ensure a good compromise between the fact that the noises are white, but also that there is no unnecessary residual information (with a lag greater than 4 for instance).
The algorithm will repeat itself the number of times necessary to get all the lags:

$$Lag = 4 \begin{cases} S \times y = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} y_1 \\ y_2 \\ y_3 \\ y_4 \\ y_5 \end{pmatrix} = \begin{pmatrix} 0 \\ y_1 \\ y_2 \\ y_3 \\ y_4 \end{pmatrix} \\ S \times y = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix} \begin{pmatrix} y_1 \\ y_2 \\ y_3 \\ y_4 \\ y_5 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ y_1 \\ y_2 \\ y_3 \end{pmatrix} \\ S \times y = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix} \begin{pmatrix} y_1 \\ y_2 \\ y_3 \\ y_4 \\ y_5 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ y_1 \\ y_2 \end{pmatrix} \\ S \times y = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} y_1 \\ y_2 \\ y_3 \\ y_4 \\ y_5 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ y_1 \end{pmatrix} \end{cases}$$

We thus obtain our delay matrix by concatenating all the outputs of $S \times y$.

$$X_1 = \begin{pmatrix} 0 & 0 & 0 & 0 \\ y_1 & 0 & 0 & 0 \\ y_2 & y_1 & 0 & 0 \\ y_3 & y_2 & y_1 & 0 \\ y_4 & y_3 & y_2 & y_1 \end{pmatrix}$$

The first time we use the OLS allowing us to obtain the residuals. Then, once the residuals are known, we repeat the process with the estimated residuals:

$$Lag = 4 \begin{cases} S \times \hat{\varepsilon} = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} \hat{\varepsilon}_1 \\ \hat{\varepsilon}_2 \\ \hat{\varepsilon}_3 \\ \hat{\varepsilon}_4 \\ \hat{\varepsilon}_5 \end{pmatrix} = \begin{pmatrix} 0 \\ \hat{\varepsilon}_1 \\ \hat{\varepsilon}_2 \\ \hat{\varepsilon}_3 \\ \hat{\varepsilon}_4 \end{pmatrix} \\ S \times \hat{\varepsilon} = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix} \begin{pmatrix} \hat{\varepsilon}_1 \\ \hat{\varepsilon}_2 \\ \hat{\varepsilon}_3 \\ \hat{\varepsilon}_4 \\ \hat{\varepsilon}_5 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ \hat{\varepsilon}_1 \\ \hat{\varepsilon}_2 \\ \hat{\varepsilon}_3 \end{pmatrix} \\ S \times \hat{\varepsilon} = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix} \begin{pmatrix} \hat{\varepsilon}_1 \\ \hat{\varepsilon}_2 \\ \hat{\varepsilon}_3 \\ \hat{\varepsilon}_4 \\ \hat{\varepsilon}_5 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ \hat{\varepsilon}_1 \\ \hat{\varepsilon}_2 \end{pmatrix} \\ S \times \hat{\varepsilon} = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} \hat{\varepsilon}_1 \\ \hat{\varepsilon}_2 \\ \hat{\varepsilon}_3 \\ \hat{\varepsilon}_4 \\ \hat{\varepsilon}_5 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ \hat{\varepsilon}_1 \end{pmatrix} \end{cases}$$

We obtain X_2, matrix of lags $S \times \hat{\varepsilon}$

$$X_2 = \begin{pmatrix} 0 & 0 & 0 & 0 \\ \hat{\varepsilon}_1 & 0 & 0 & 0 \\ \hat{\varepsilon}_2 & \hat{\varepsilon}_1 & 0 & 0 \\ \hat{\varepsilon}_3 & \hat{\varepsilon}_2 & \hat{\varepsilon}_1 & 0 \\ \hat{\varepsilon}_4 & \hat{\varepsilon}_3 & \hat{\varepsilon}_2 & \hat{\varepsilon}_1 \end{pmatrix}$$

We can finally apply OLS with X, a concatenated matrix of X_1 and X_2.

$$X = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ y_1 & 0 & 0 & 0 & \hat{\varepsilon}_1 & 0 & 0 & 0 \\ y_2 & y_1 & 0 & 0 & \hat{\varepsilon}_2 & \hat{\varepsilon}_1 & 0 & 0 \\ y_3 & y_2 & y_1 & 0 & \hat{\varepsilon}_3 & \hat{\varepsilon}_2 & \hat{\varepsilon}_1 & 0 \\ y_4 & y_3 & y_2 & y_1 & \hat{\varepsilon}_4 & \hat{\varepsilon}_3 & \hat{\varepsilon}_2 & \hat{\varepsilon}_1 \end{pmatrix}$$

The ARMA process will then be estimated by OLS (with the OLS estimator $\hat{\beta}_{OLS}$ seen above).

5 Process simulation: estimation of ARMAs by maximum likelihood

This part focuses on the simulation of ARMA using the ARIMA procedure under SAS. This procedure, although very practical, uses the maximum likelihood estimation which poses the problem of the impossibility of using the Monte Carlo simulation: conversaly to the OLS procedure, the maximum likelihood algorithm can fail. It is then that the results will be presented in table form, and it will be necessary to approach a critical look at the robustness of the results. This section must be viewed as a first approach to the study of information criteria.

As the results are presented in table form, scanning all the p and q orders would have been indigestible. We propose here a selection of the most interesting orders according to us (which are most often returned by the criteria). Moreover, SAS only return the AIC and the BIC criteria, thus, we will only focus on those two in this section.

5.1 ARMA(1,0): $X_t = 0.6X_{t-1} + \varepsilon_t$

		ARMA(p,q)							
		(0,1)	(1,0)	(1,1)	(2,1)	(1,2)	(2,2)	(3,3)	(4,4)
T = 50	AIC	160.910	157.763	157.700	159.665	159.617	160.938	158.024	166 218
	BIC	162.822	159.675	161.524	165.402	165.353	168.586	169.496	181.514
T = 100	AIC	309.971	301.431	303.387	305.370	305.305	307.227	311.008	311.883
	BIC	312.576	304.036	308.597	313.185	313.121	317.648	326.639	332.724
T = 150	AIC	452.841	440.749	442.391	444.231	443.321	446.277	440.494	439.323
	BIC	455.852	443.760	448.412	453.263	452.353	458.320	458.558	463.408
T = 250	AIC	755.822	704.588	705.440	708.510	704.167	704.734	707.693	711.278
	BIC	759.343	708.109	712.483	719.074	714.732	718.820	728.821	739.450
T = 500	AIC	1466.76	1403.30	1404.99	1404.23	1405.91	1407.35	1401.66	1407.95
	BIC	1470.98	1407.51	1413.41	1416.88	1418.55	1424.21	1426.95	1441.66

5.2 ARMA(0,1): $X_t = \varepsilon_t + 0.8\varepsilon_{t-1}$

		ARMA(p,q)							
		(0,1)	(1,0)	(1,1)	(2,1)	(1,2)	(2,2)	(3,3)	(4,4)
T = 50	AIC	151.837	173.312	179.901	167.476	153.592	168.537	154.171	160.317
	BIC	153.749	175.224	183.725	173.212	159.328	176.185	165.643	175.614
T = 100	AIC	273.129	293.703	274.189	275.282	275.911	278.148	272.612	270.893
	BIC	275.734	296.308	279.399	283.097	283.727	288.569	288.243	291.734
T = 150	AIC	428.606	459.264	430.241	431.385	429.366	431.203	434.814	436.439
	BIC	431.616	462.275	436.262	440.417	438.398	443.246	452.878	460.525
T = 250	AIC	687.997	739.538	689.702	691.627	691.474	694.977	697.817	702.244
	BIC	691.518	743.059	696.745	702.192	702.038	709.063	718.945	730.416
T = 500	AIC	1398.04	1504.13	1400.02	1401.76	1401.91	1403.88	1402.12	1402.36
	BIC	1402.26	1508.34	1408.45	1414.40	1414.55	1420.74	1427.40	1436.08

5.3 ARMA(1,1): $0.6X_{t-1} + \varepsilon_t + 0.8\varepsilon_{t-1}$

		ARMA(p,q)							
		(0,1)	(1,0)	(1,1)	(2,1)	(1,2)	(2,2)	(3,3)	(4,4)
T = 50	AIC	178.305	157.926	144.991	146.794	146.888	148.297	146.990	154.969
	BIC	180.217	159.838	148.815	152.530	152.624	155.945	158.462	170.265
T = 100	AIC	332.553	317.452	286.609	288.441	288.415	290.327	286.377	290.058
	BIC	335.158	320.057	291.820	296.256	296.230	300.748	302.008	310.900
T = 150	AIC	437.871	445.549	405.168	406.581	406.509	408.508	407.866	399.287
	BIC	440.882	448.559	411.190	415.613	415.541	420.551	425.930	423.372
T = 250	AIC	763.875	781.093	683.385	683.789	684.241	685.404	688.250	689.917
	BIC	767.396	784.614	690.428	694.354	694.805	699.490	709.379	718.089
T = 500	AIC	1662.27	1693.85	1502.64	1504.64	1505.88	1506.61	1506.00	1510.73
	BIC	1666.49	1698.07	1511.07	1517.28	1517.28	1523.47	1531.29	1544.45

5.4 ARMA(2,1): $X_t = 0.85X_{t-1}$–$0.65X_{t-2} + \varepsilon_t + 0.5\varepsilon_{t-1}$

		ARMA(p,q)							
		(0,1)	(1,0)	(1,1)	(2,1)	(1,2)	(2,2)	(3,3)	(4,4)
T = 50	AIC	177.222	199.087	168.191	150.332	165.076	152.331	150.771	156.533
	BIC	179.134	200.999	172.015	156.068	170.812	159.980	162.243	171.829
T = 100	AIC	339.724	381.994	322.650	291.838	307.758	293.822	297.548	310.236
	BIC	342.329	384.599	327.860	299.654	315.573	304.243	313.179	331.077
T = 150	AIC	498.123	537.511	466.850	438.501	441.208	434.031	437.252	440.768
	BIC	501.134	540.521	472.871	447.533	450.240	446.073	455.316	464.853
T = 250	AIC	846.924	966.823	815.959	706.259	770.743	705.124	709.011	710.129
	BIC	850.446	970.344	823.002	716.824	781.307	719.210	730.139	738.301
T = 500	AIC	1747.81	1967.79	1661.56	1469.36	1575.70	1470.43	1474.09	1463.14
	BIC	1752.02	1972.01	1669.99	1482.00	1588.34	1487.29	1499.38	1496.86

5.5 ARMA(1,2): $X_t = 0.5X_{t-1} + \varepsilon_t + 0.85\varepsilon_{t-1} + 0.65\varepsilon_{t-2}$

		ARMA(p,q)							
		(0,1)	(1,0)	(1,1)	(2,1)	(1,2)	(2,2)	(3,3)	(4,4)
T = 50	AIC	181.077	165.882	151.935	143.763	132.298	131.641	134.753	137.355
	BIC	182.989	167.794	155.759	149.499	138.034	139.289	146.225	152.651
T = 100	AIC	358.825	331.760	312.224	297.381	281.304	282.820	284.276	287.271
	BIC	361.430	334.365	317.435	305.196	289.119	293.241	299.907	308.113
T = 150	AIC	565.991	472.065	443.360	433.234	408.961	409.962	412.438	416.696
	BIC	569.002	475.076	449.381	442.265	417.992	422.005	430.502	440.781
T = 250	AIC	889.443	780.514	739.313	723.110	695.200	696.472	699.127	700.217
	BIC	892.965	784.035	746.356	733.675	705.765	710.558	720.256	728.389
T = 500	AIC	1818.70	1648.76	1559.70	1522.35	1424.27	1425.59	1426.00	1426.33
	BIC	1822.91	1652.98	1568.13	1535.00	1436.91	1442.44	1451.29	1460.05

5.6 ARMA(2,2): $X_t = 0.8X_{t-1}–0.7X_{t-2} + \varepsilon_t + 0.6\varepsilon_{t-1} + 0.5\varepsilon_{t-2}$

		ARMA(p,q)							
		(0,1)	(1,0)	(1,1)	(2,1)	(1,2)	(2,2)	(3,3)	(4,4)
T = 50	AIC	177.782	188.563	167.999	151.086	156.999	150.004	154.002	154.561
	BIC	179.694	190.475	171.823	156.822	162.736	157.652	165.474	169.857
T = 100	AIC	368.518	397.638	346.344	305.881	311.222	291.033	294.727	297.621
	BIC	371.123	400.243	351.554	313.696	319.037	301.454	310.358	318.462
T = 150	AIC	530.431	581.294	503.672	434.685	465.133	419.488	425.655	439.722
	BIC	533.442	584.305	509.694	443.716	474.165	431.531	443.719	463.808
T = 250	AIC	1004.45	1106.88	959.552	750.662	850.719	710.217	773.887	740.025
	BIC	1007.97	1110.40	966.595	761.227	861.284	724.303	795.016	768.197
T = 500	AIC	1858.28	2055.80	1756.39	1464.45	1596.22	1393.84	1392.94	1395.64
	BIC	1862.50	2060.01	1764.82	1477.09	1608.86	1410.70	1418.23	1429.36

5.7 General conclusions

We can conclude that in general the BIC seems to be more accurate than the AIC. We noticed during the generation of the ARMAs that the value we attributed to the coefficients played a great role in the ability of the criteria to find the "true model" (i.e. the data generated process). Indeed, if the coefficients are too low, the ability of the criteria to find the true model is very reduced. Moreover, the BIC is wrong twice in the end, whereas until then, it was rather the AIC that was wrong.

We should be careful with these results because when the criteria fail to find the "true" data generated process, the error is very small. For instance, is 286.377 very different from 286.609? Thanks to this approach, we can see how much a criterion is wrong (the extent of its error). Thus, it seems that the higher the ARMA orders, the more difficult it is for the criteria to choose the "true" model.

6 Process simulation: estimations of ARMAs by OLS

In this section, we look at a second way of estimating ARMAs: the ordinary least squares method. Coupled with the Monte Carlo method, we will be able to see how often the criteria are wrong, and thus generalise their effectiveness.

Monte Carlo simulations are methods that revolutionised the study of physical, mathematical and economic phenomena. These simulations represent a way of estimating a quantity by using random numbers. These simulations can be seen as a statistical estimation method to study the behaviour of a random variable. Since ARMA processes are randomly distributed, we will use these simulations to study the effectiveness of the criteria for different sample sizes.

For a matter of clarity, only the most important results will be presented in this section (as contourplots). Every contourplots are included in the appendix and are available to the reader.

6.1 ARMA(1,0): $X_t = 0.6X_{t-1} + \varepsilon_t$

Annexe ($\downarrow_1$)

When a pure AR of order 1 is generated, we notice that almost all the information criteria return the correct model. However, the BICc seems at T=50 to be wrong by returning many times an ARMA(0,0), and thus a white noise.

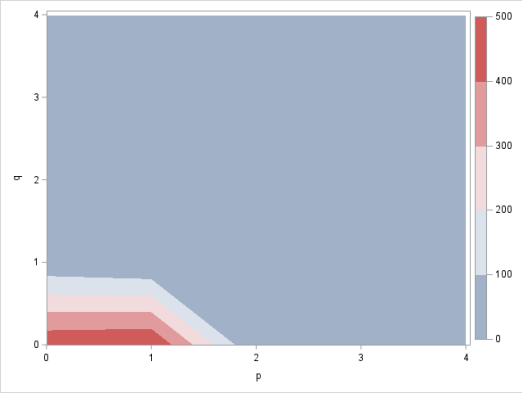

Figure 4: BICc, T = 50.

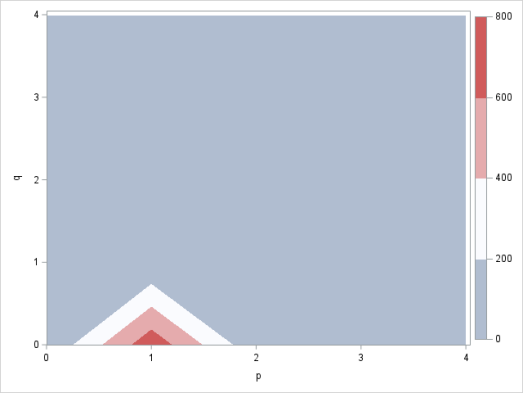

Figure 5: AIC, T = 50.

This error disappears as the sample size increases. Thus, for a pure AR, all criteria seem to perform well regardless of the sample size (with the exception of BICc for T=50).

6.2 ARMA(0,1): $X_t = \varepsilon_t + 0.8\varepsilon_{t-1}$

Annexe ($\downarrow_2$)

When a pure MA is generated, the information criteria are not as unanimous as for the pure AR. For T=50, we notice that the AIC, AICu, FPE and FPEu detect some pure AR(2). However, these errors remain marginal, and they return the correct model in the vast majority of cases. We can also note that the BICc has a contourplot with a less concentrated pattern, which shows a slightly lower accuracy (although it returns globally the right model). For T=100, all criteria seem to return the correct model, and their accuracy is more or less identical. However, when moving to T=150 and T=250, we notice that the AIC, AICu, FPE and FPEu sometimes get it wrong by returning ARMA(1,2).

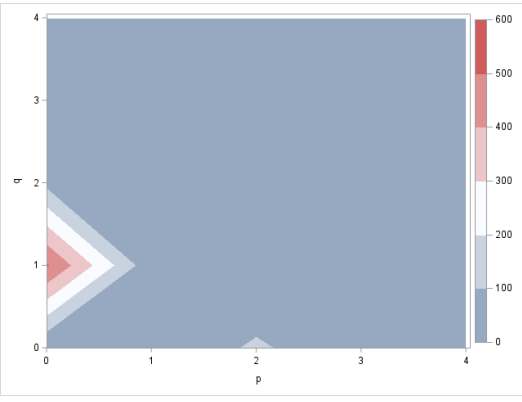

Figure 6: AIC, T = 50.

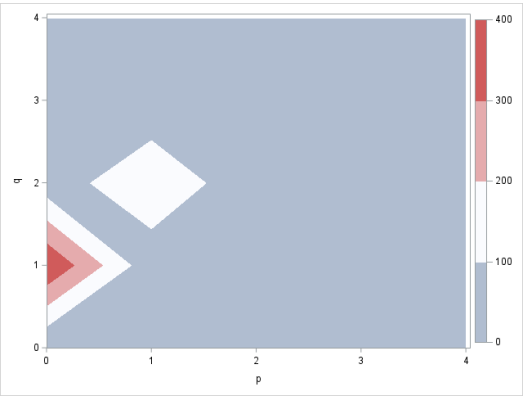

Figure 7: AIC, T = 250.

Finally, when we increase to T=500, the same criteria that were already wrong before (AIC, AICu, FPE and FPEu) are totally wrong and no longer find the true pattern. They do return some pure MA, but also a lot of

ARMA(1,2), ARMA(2,3), and even ARMA(3,4). We also notice that the HQ and HQc criteria lose their precision when the sample is too large, because they also start to return ARMA(1,2).

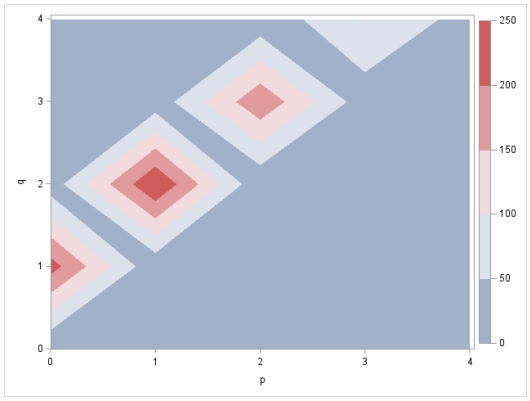

Figure 8: FPE, T = 250.

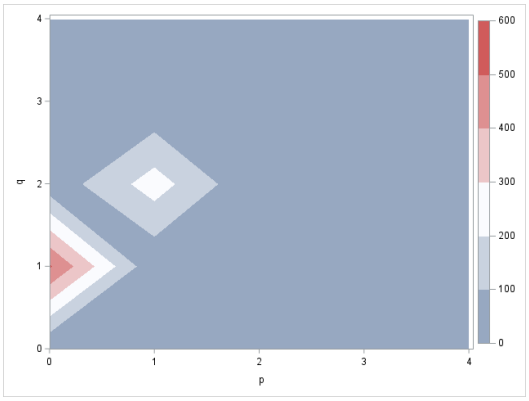

Figure 9: HQ, T = 250.

Thus, the most accurate for this section are the BIC and the BICc, which seem robust to the sample size effect.

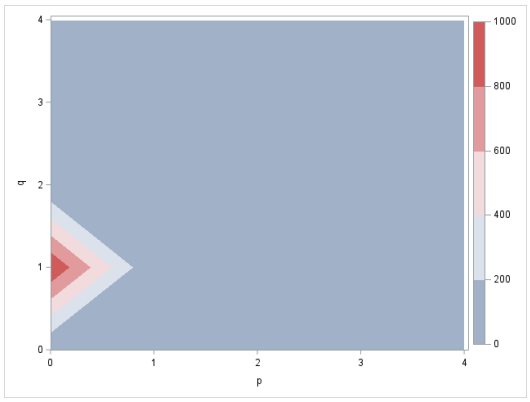

Figure 10: BIC, T = 500.

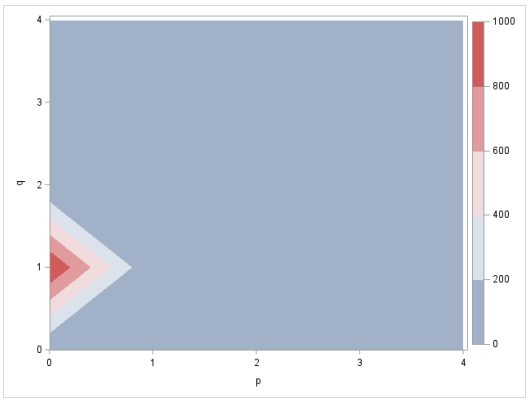

Figure 11: BICc, T = 500.

6.3 ARMA(1,1): $0.6X_{t-1} + \varepsilon_t + 0.8\varepsilon_{t-1}$

Annexe ($\downarrow_3$)

When the sample is small (T=50), the information criteria seem to make some small errors, although they all find ARMA(1,1) in the vast majority of cases. Indeed, with the exception of the FPE which is very accurate, they all detect some AR(2) in small quantities, The only one to really make a mistake is, to our great surprise, the BICc, which chooses an AR(1) for the most part. This seems surprising in view of the previous results, as it was the one that was least wrong until then. The BIC is slightly more wrong than the other criteria, although it still chooses the right ARMA most of the time.

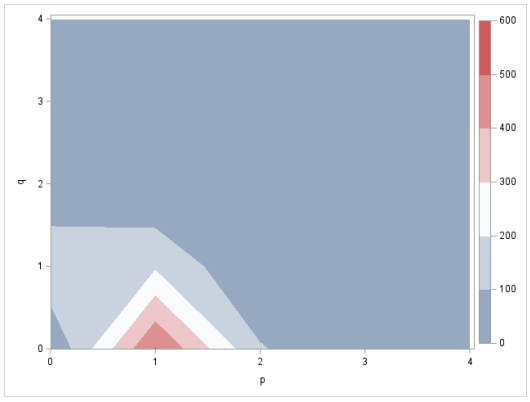

Figure 12: BICc, T = 50.

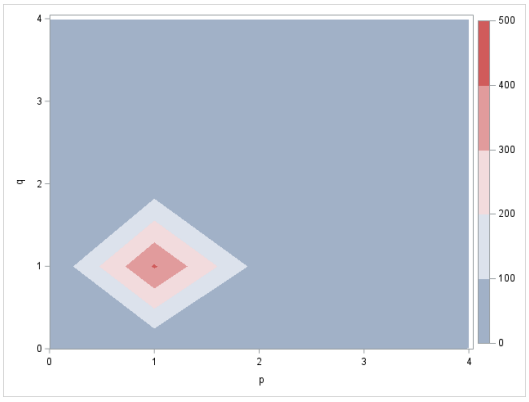

Figure 13: FPE, T = 50.

When T=100 and T=150, the BIC and BICc correct themselves completely, and no longer make any errors. The same is true for the AICc, HQ, and HQc. The other criteria keep the same conclusions as with T=50.

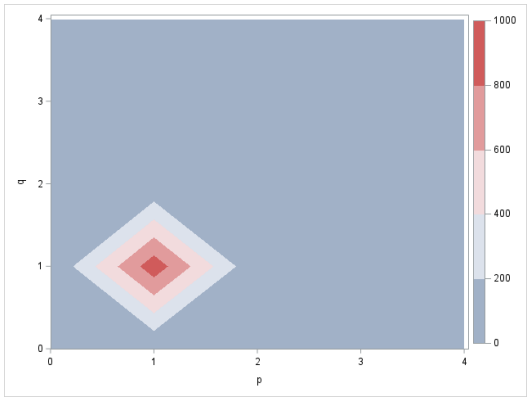

Figure 14: BIC, T = 150.

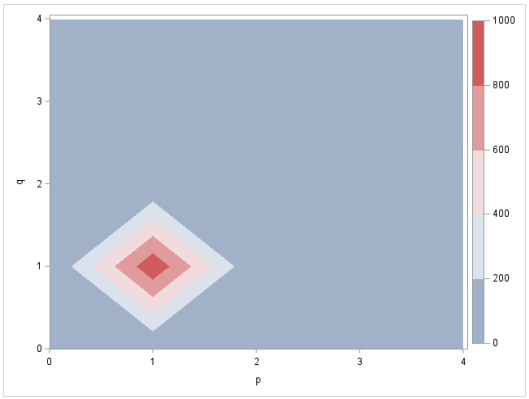

Figure 15: BICc, T = 150.

When T=250, the only ones that do not go wrong are the AICc, BIC and BICc (The HQ and HQc don't seem able to bear the sample size effect).

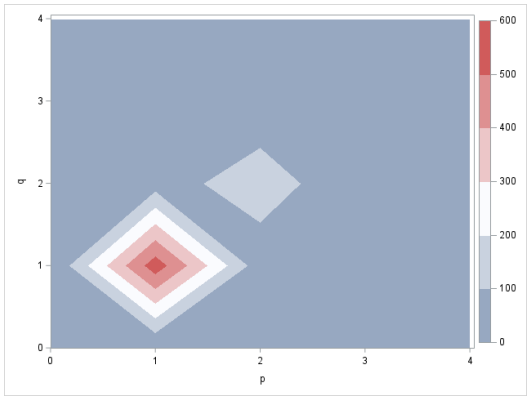

Figure 16: HQ, T = 250.

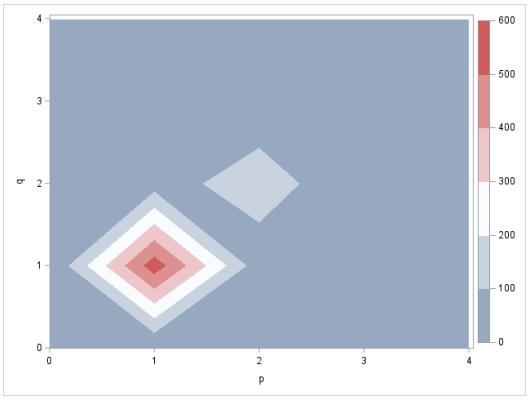

Figure 17: HQc, T = 250.

Finally, when the sample is very large, i.e. when T=500, the criteria that were not wrong (AICc, BIC and BICc) remain robust, whereas those that were slightly wrong are now very wrong.

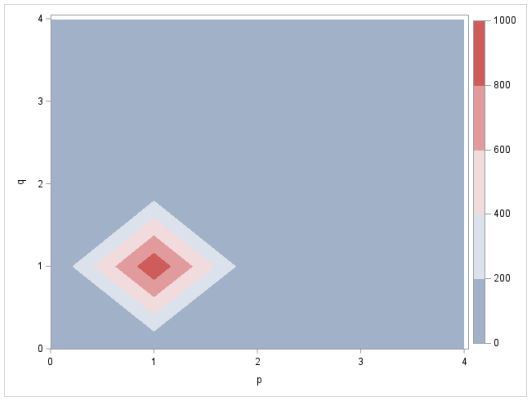

Figure 18: BIC, T = 500.

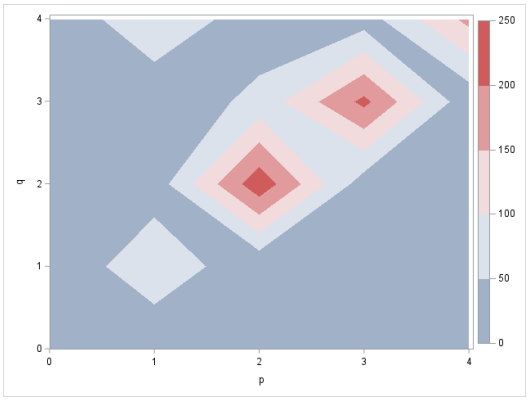

Figure 19: FPE, T = 500.

Thus, once again, we note that the asymptotic properties of the BIC and BICc are very interesting: although they are not accurate at small sample sizes, they correct themselves as soon as the sample size increase (unlike the others which confirm their error with a larger sample size).

6.4 ARMA(2,1): $X_t = 0.85X_{t-1}$–$0.65X_{t-2} + \varepsilon_t + 0.5\varepsilon_{t-1}$

Annexe ($\downarrow_4$)

When the sample is small (T=50), we notice something quite interesting: no criterion is really accurate. Some criteria return globally the right model (AIC, AICu, FPE, FPEu, HQ and HQc) but there are many errors. We notice that the AICc, BIC and BICc are completely wrong when choosing an AR(2).

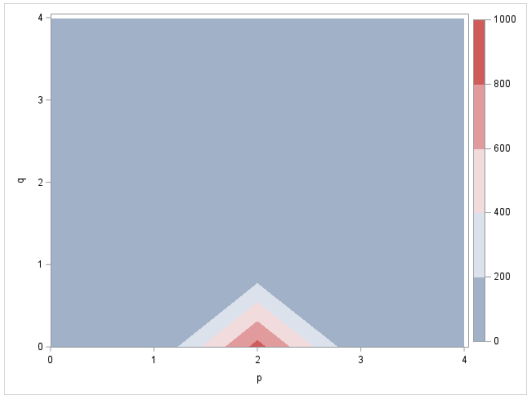

Figure 20: BIC, T = 50.

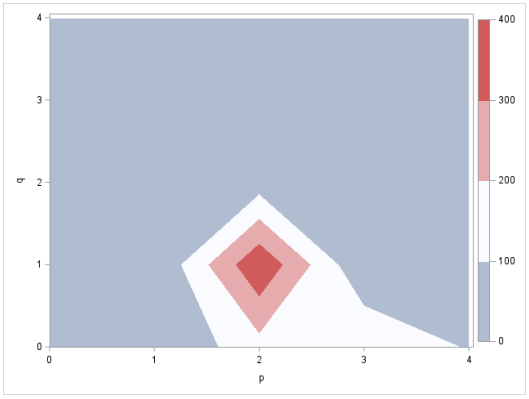

Figure 21: FPE, T = 50.

The errors that the criteria made globally with T=50 are repeated for T=100, with a few exceptions. The case of the BIC is very interesting because we can see that it gradually converges towards the correct model. It used to return an AR(2), whereas we see that it is gradually moving towards the ARMA(2,1). We also notice that the HQ and the HQc are very accurate, indeed they are the only ones not to be wrong at all.

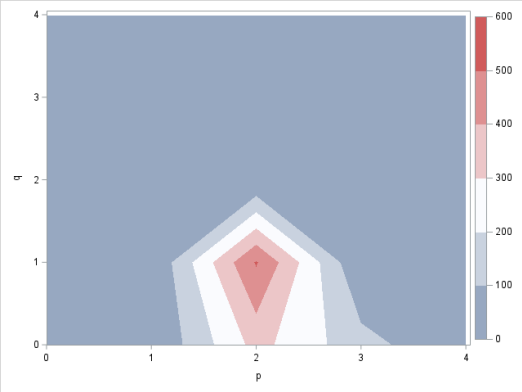

Figure 22: BIC, T = 100.

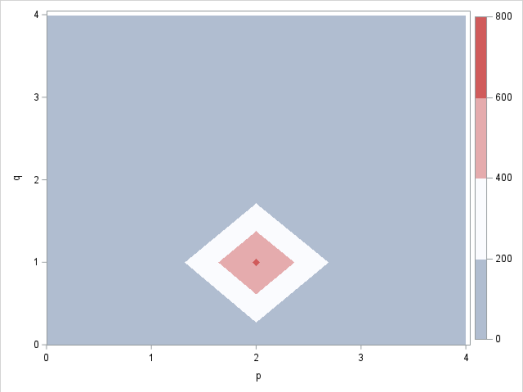

Figure 23: HQ, T = 100.

For T=150 the criteria are already much more precise, the BICc is the only one not to return the correct model. However, we can see that it goes up towards the ARMA(2,1), correcting itself. We can already guess that with larger samples it will return the correct model.

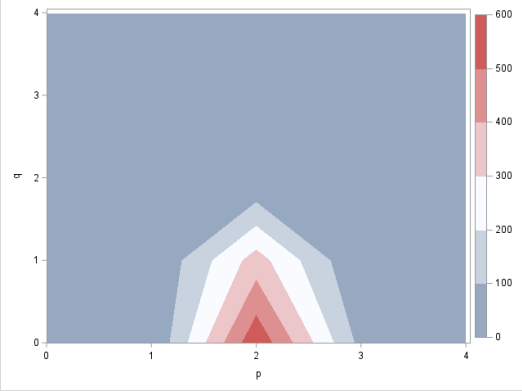

Figure 24: BICc, T = 150.

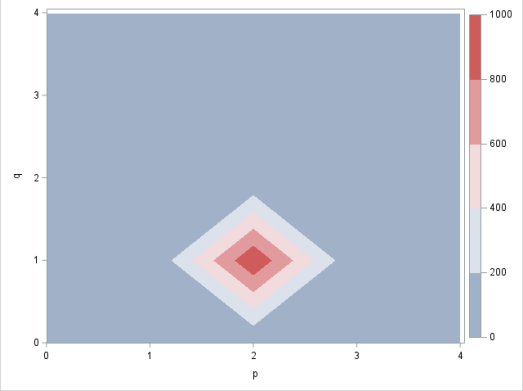

Figure 25: BICc, T = 500.

The intuition seen before is confirmed, all the criteria choose the ARMA(2,1) for T=250 and 500. This model is particularly interesting because as the sample increases, we can notice that the BIC and the BICc progressively converge towards the good model.

We note that this is the first time that all criteria return all models perfectly with an asymptotical sample.

6.5 ARMA(1,2): $X_t = 0.5X_{t-1} + \varepsilon_t + 0.85\varepsilon_{t-1} + 0.65\varepsilon_{t-2}$

Annexe ($\downarrow_5$)

The criteria are all wrong when the sample is small. However, we note that the AIC and AICu, HQ, HQc, FPE and FPu sometimes return the correct model (but this is marginal, they are wrong in the vast majority of cases).

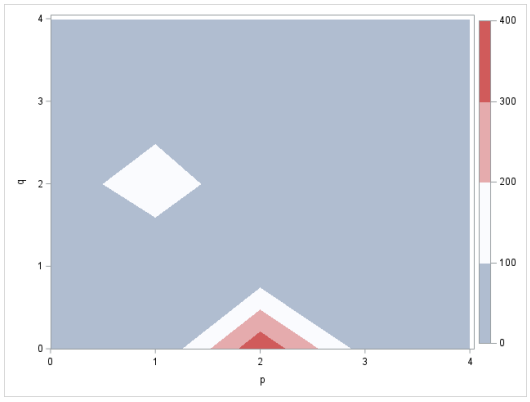

Figure 26: AIC, T = 50.

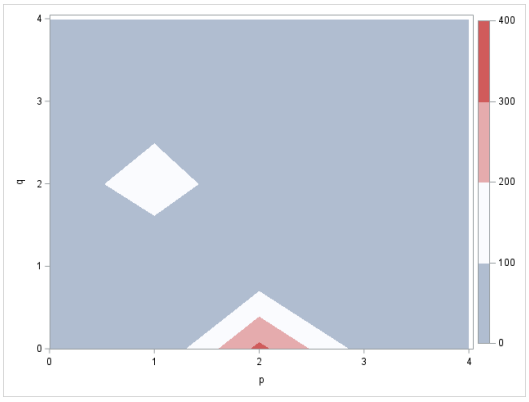

Figure 27: FPE, T = 50.

When T=100, the FPE, FPEu, HQ and HQc start to find the true model although many errors remain.

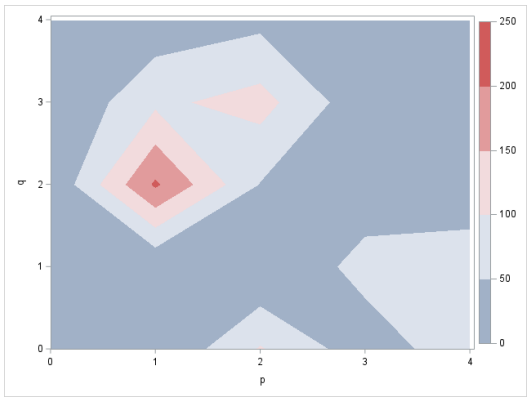

Figure 28: FPEu, T = 100.

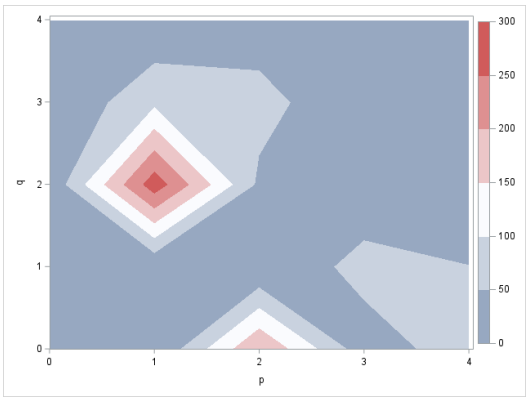

Figure 29: HQc, T = 100.

When the sample size is further increased they do not support the size effect and start to return more and more ARMA(2,3). When T=250 and 500, the criteria all make errors except... the BIC and BICc which are the most accurate. It should be noted that the AICc is also very accurate.

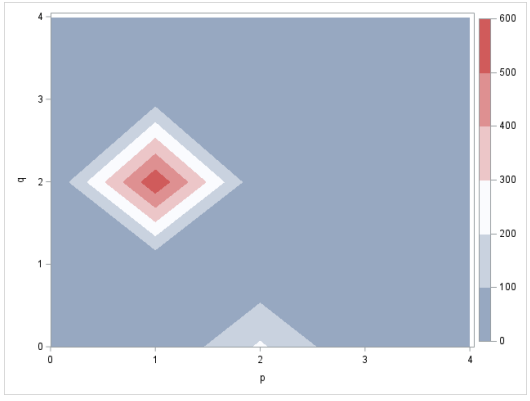

Figure 30: BIC, T = 250.

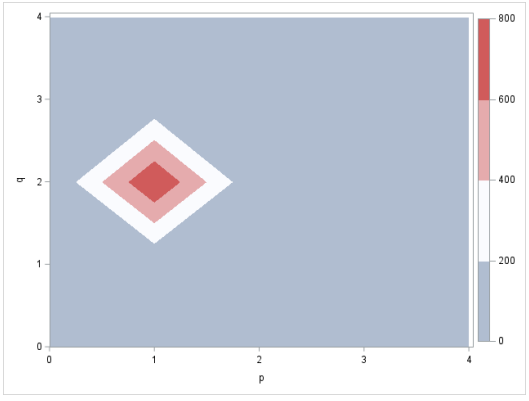

Figure 31: BICc, T = 500.

The BIC and BICc were completely wrong at low sample sizes, and eventually converge to the correct model again at high sample sizes.

6.6 ARMA(2,2): $X_t = 0.8X_{t-1}–0.7X_{t-2} + \varepsilon_t + 0.6\varepsilon_{t-1} + 0.5\varepsilon_{t-2}$

Annexe ($\downarrow_6$)

For ARMA(2,2), everything that was concluded earlier seems to be repeated. The BIC and BICc are totally wrong at low sample sizes, while the other criteria occasionally return the correct pattern. When we increase the sample slightly towards 100-150, the AIC, AICc, AICu, HQ, HQc, FPE and FPEu find the correct model overall, while the BIC and BICc are always wrong: they do not have enough observations to converge.

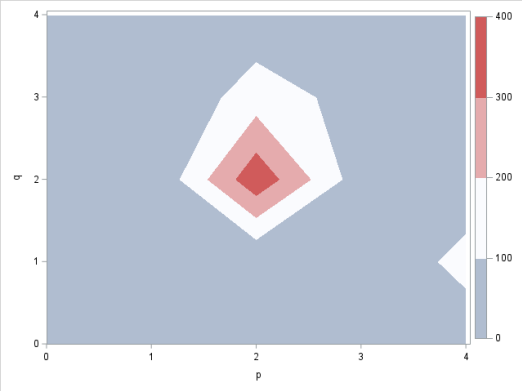

Figure 32: AIC, T = 150.

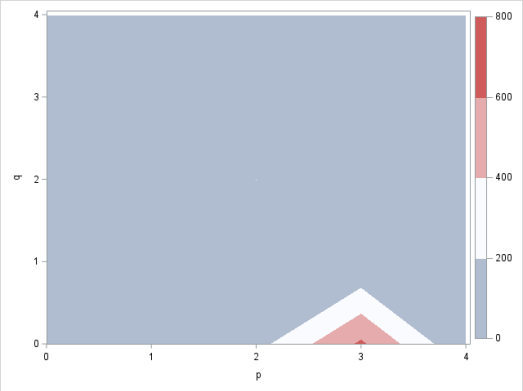

Figure 33: BIC, T = 150.

Finally, when T=250 and then 500, the same pattern occurs and the trends are reversed: the BIC and the AICc are the most accurate, and the others lose accuracy as the sample increases. This time, however, the BICc failed to converge, and remain wrong because it returns a large amounts of AR(3).

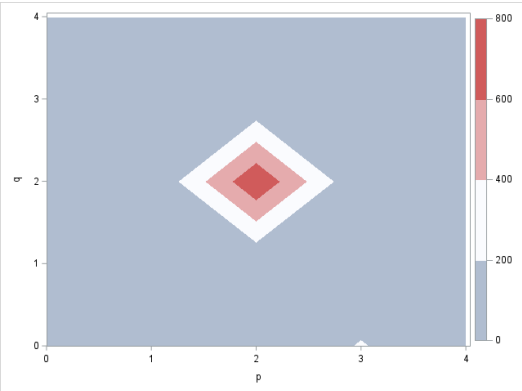

Figure 34: AICc, T = 500.

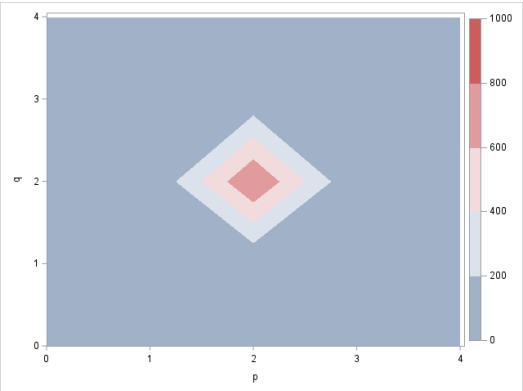

Figure 35: BIC, T = 500.

6.7 General conclusions

Overall the HQ, HQc, FPE, FPEu, AIC, AICc and AICu criteria are relevant for some models and sample types, but do not support large samples. The best information criteria are the BIC and BICc which almost always converge to the true model, although at low sample sizes they are less accurate than the others. Although the BICc is supposed to be an improvement on the BIC, but for the case of ARMA model generation it does not

really seem to perform better than the BIC. It is also noted that AICc and AICu sometimes do better than AIC, but that there is no real decision rule here.

The convergence properties of the BIC and BICc initially stated by Hannan Quinn in 1979 are well observed in our ARMA modeling case, whereas indeed the AIC and FPE do not seem to converge.
Nevertheless, based on theory alone, the AICc, BIC and BICc should have been effective even at low sample sizes (which did not turn out to be the case in practice).
The AICu and FPEu were mainly supposed to perform well in the case of overfiting, which is probably why they do not stand out in our case study.

The biggest surprise of our study case is that the HQ and HQc do not seem to converge as stated by Hannan Quinn in 1979. They should normally converge and withstand a large sample, however this is not the case at all. The most likely explanation for all these distortions between theory and practice can be explained by several things:

- Firstly, the coefficient of the generated ARMAs plays a determining role in the ability of the criteria to detect the DGP. Indeed, during the generation of the models, we noticed that too weak coefficients complicated the detection of the "true" models by the criteria.
- Secondly, it can be seen that the higher the order of the generated ARMA, the more likely the criteria will be wrong
- Finally, we have taken the particular case of ARMA modeling. It may be that Hannan's conclusions work very well for other models, but that this is slightly less true for ARMAs.

7 Conclusion

The numerical approach seen with maximum likelihood method allows us to see how much the criteria are wrong, and therefore to know the magnitude of a single error (unlike the contourplot which simply reports whether or not the correct model is found). The contourplot approach with the OLS method, on the other hand, allows us to study how wrong the models are if the experiment is repeated many times. These different approaches have allowed us to observe that the best criterion in the framework of an ARMA simulation remains the BIC and the BICc. Some other criteria can be used, especially in small samples where some are very interesting (HQ, HQc, FPe and FPEu). As for the AIC, AICu and AICc, these criteria seem to be very unpredictable: depending on the model generated or the number of obersvation, they can be very effective (especially the AICc and AICu).

8 Annexes

Back to ARMA (1,0) part (↑[1])

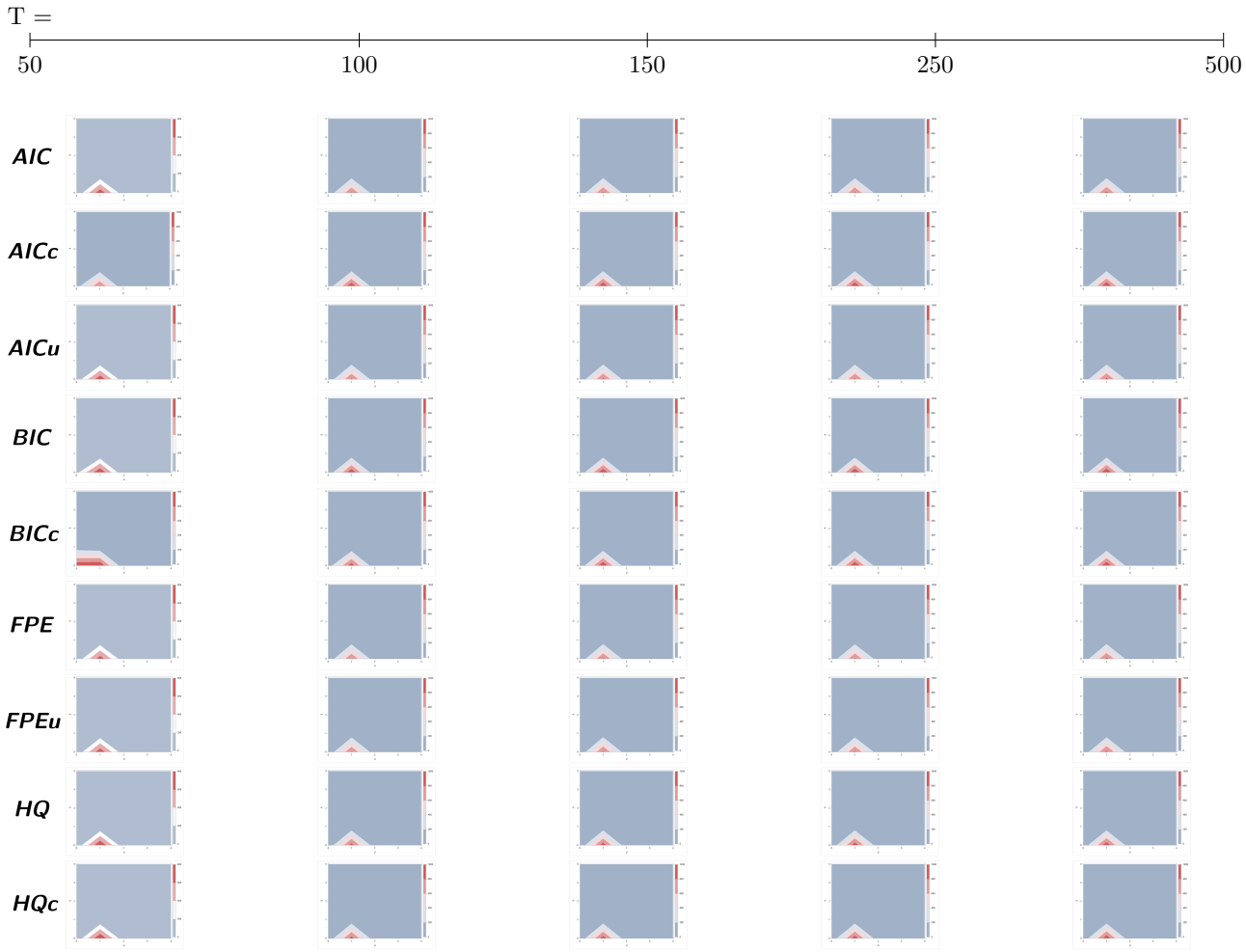

Figure 36: Information criteria returned contour lines for a DGP ARMA: (1,0)

Back to ARMA (0,1) part($\uparrow^2$)

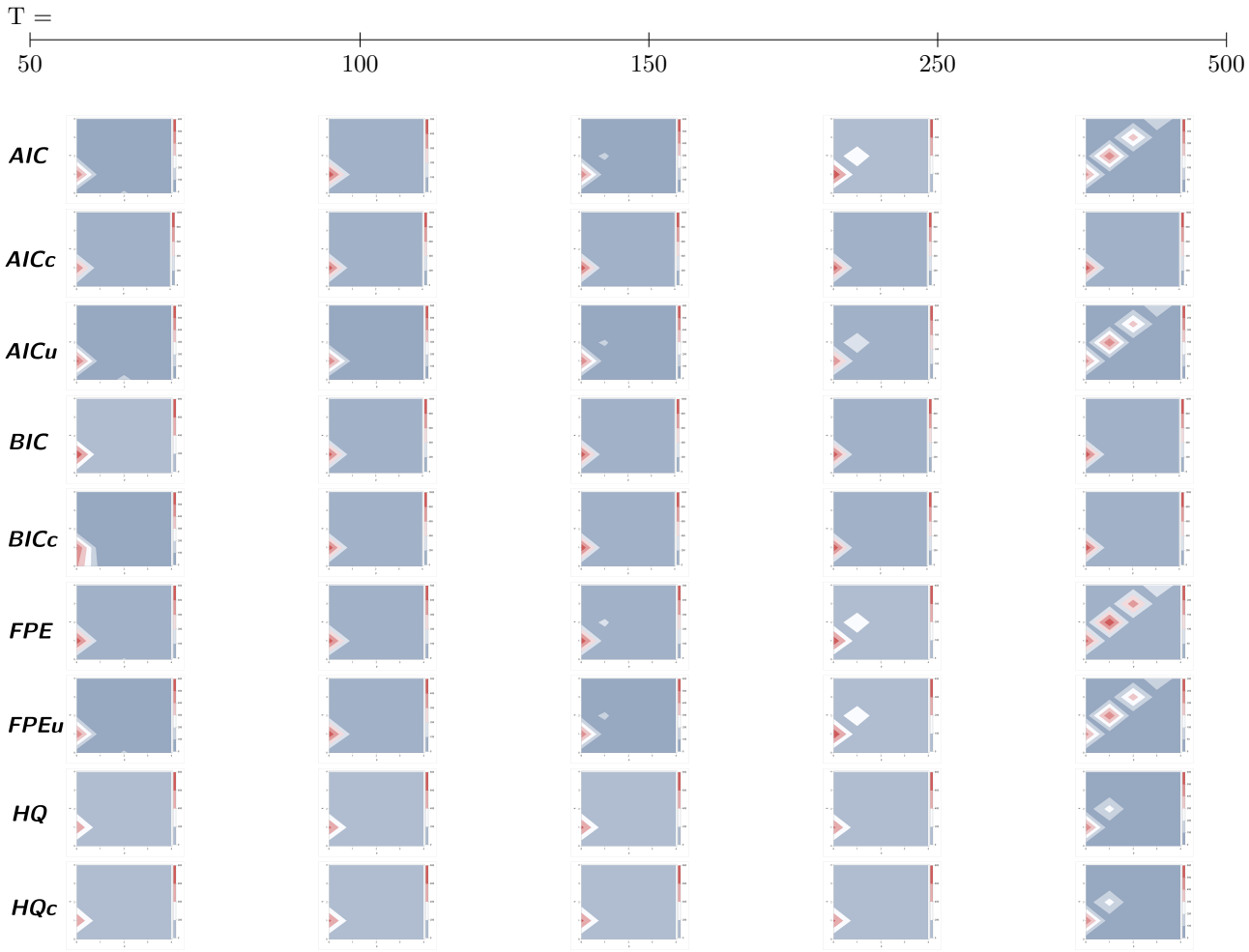

Figure 37: Information criteria returned contour lines for a DGP ARMA: (0,1)

Back to ARMA (1,1) part (↑[3])

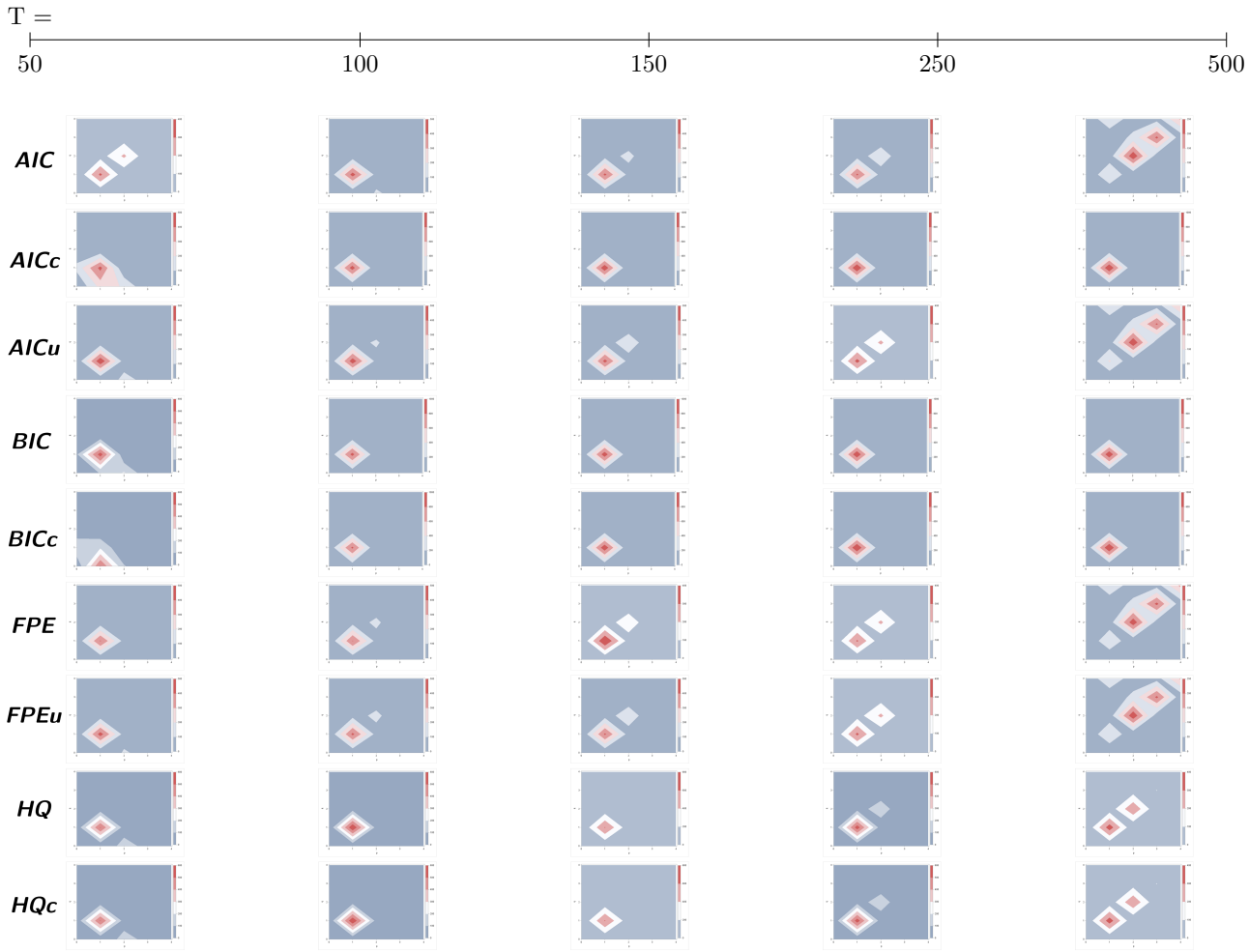

Figure 38: Information criteria returned contour lines for a DGP ARMA: (1,1)

Back to ARMA (2,1) part (↑[4])

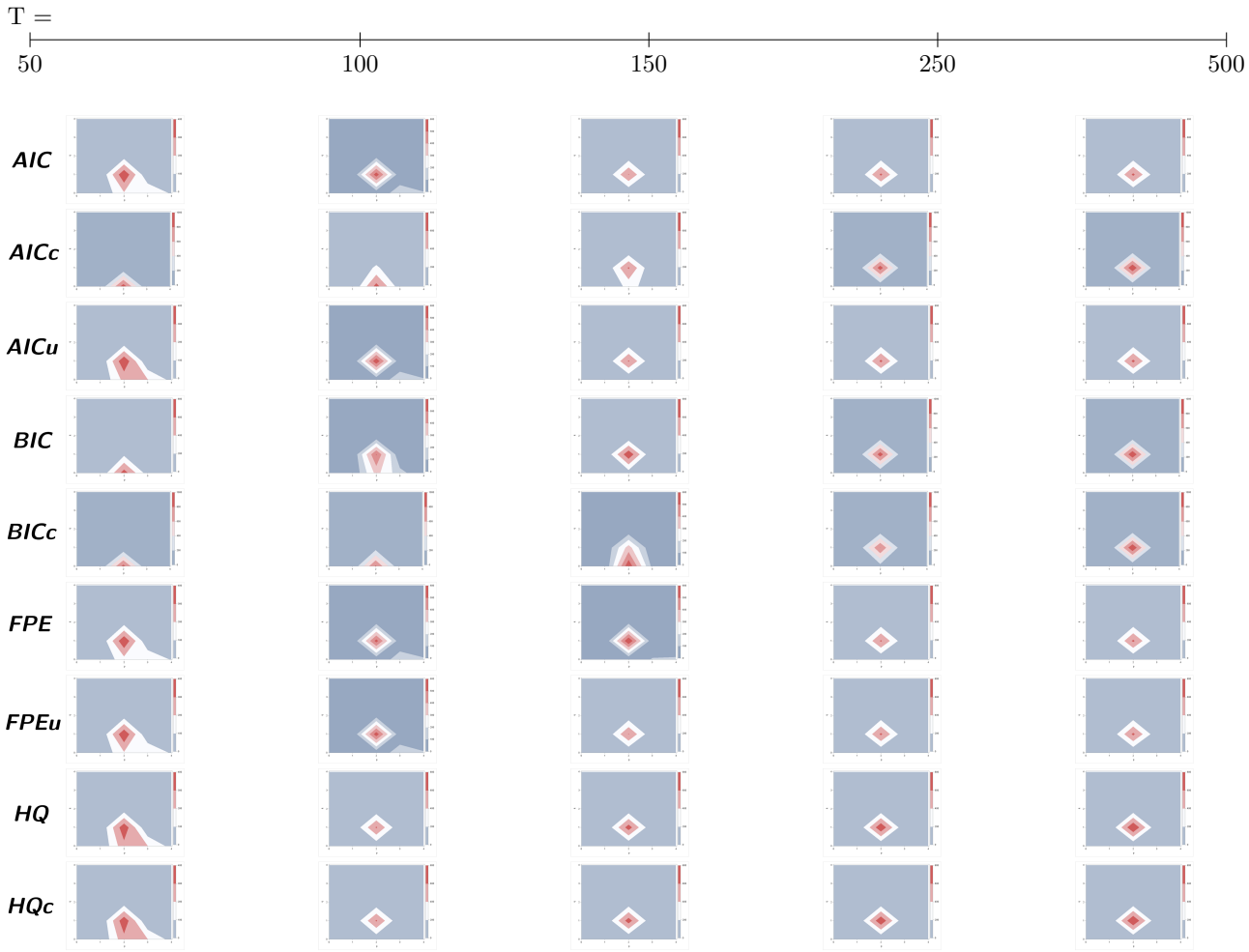

Figure 39: Information criteria returned contour lines for a DGP: ARMA (2,1)

Back to ARMA (1,2) part (↑[5])

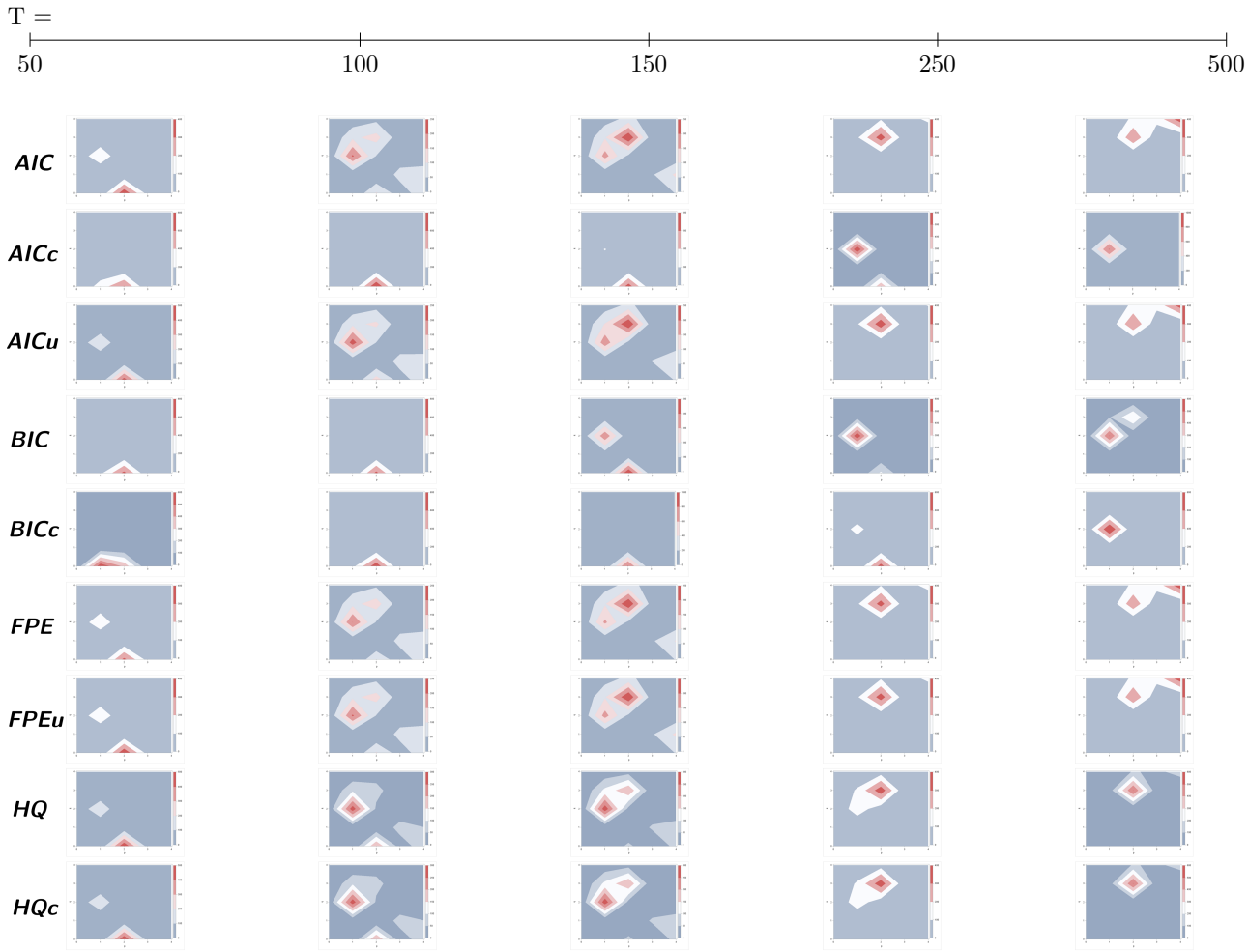

Figure 40: Information criteria returned contour lines for a DGP ARMA: (1,2)

Back to ARMA (2,2) part (↑[6])

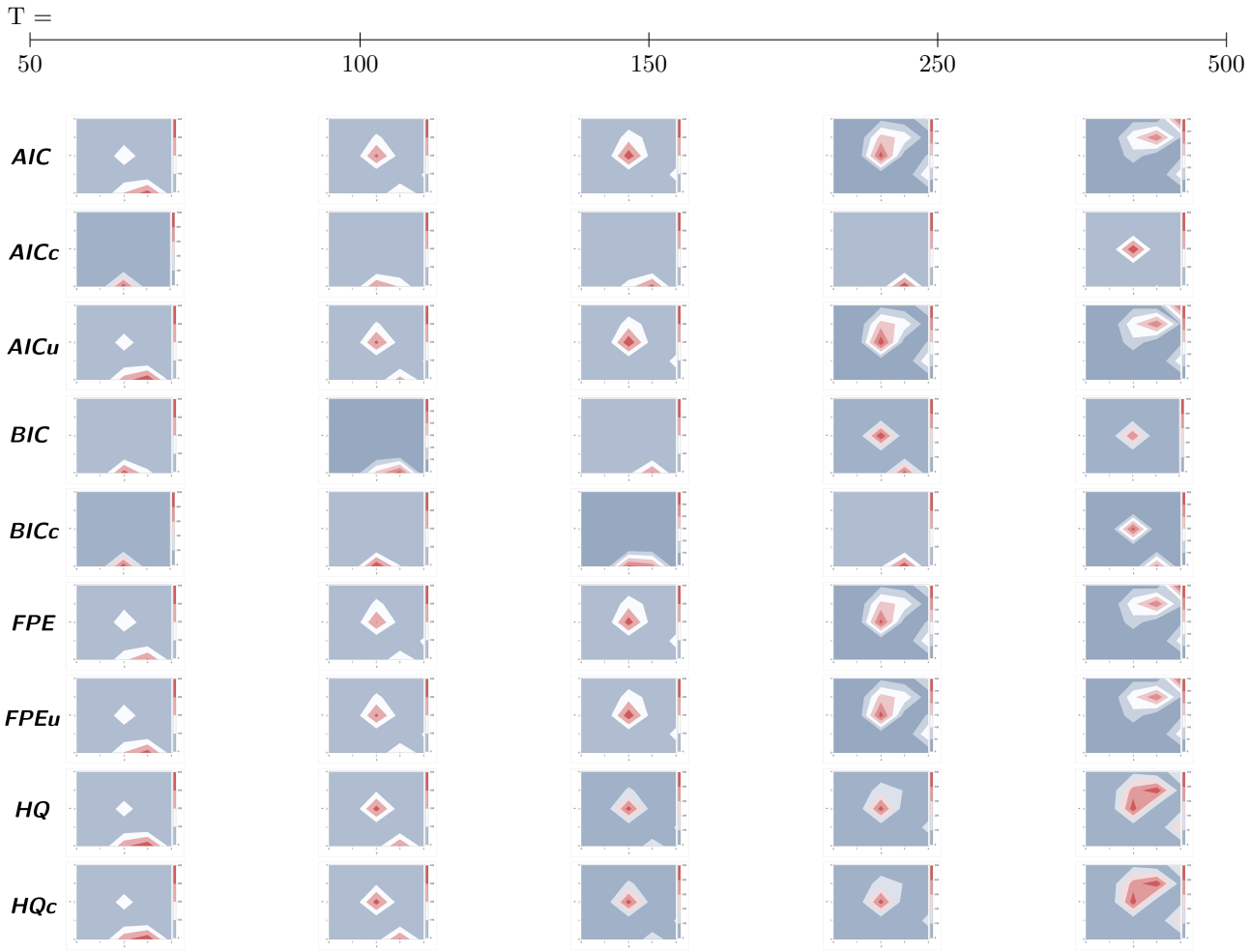

Figure 41: Information criteria returned contour lines for a DGP: ARMA (2,2)

References

[1] DR Anderson, KP Burnham, and GC White. Comparison of akaike information criterion and consistent akaike information criterion for model selection and statistical inference from capture-recapture studies. *Journal of Applied Statistics*, 25(2):263–282, 1998.

[2] Julie Bertrand, Emmanuelle Comets, and France Mentre. Comparison of model-based tests and selection strategies to detect genetic polymorphisms influencing pharmacokinetic parameters. *Journal of Biopharmaceutical Statistics*, 18(6):1084–102, 2008.

[3] Régis Bourbonnais et al. *Econométrie*. Dunod Paris, France, 2011.

[4] John W. Galbraith and Victoria Zinde-Walsh. Évaluation de critères d'information pour les modèles de séries chronologiques. *L'Actualité Economique*, 80(2):207–227, Juin-Sept 2004.

[5] R Scott Hacker. The effectiveness of information criteria in determining unit root and trend status. Working Paper Series in Economics and Institutions of Innovation 213, Royal Institute of Technology, CESIS - Centre of Excellence for Science and Innovation Studies, 2010.

[6] Jean-Pierre Lecoutre. *Statistique et probabilités*. Dunod, 2002.

[7] Chih/ling MCQUARRIE, Allan D.R./TSAI. Regression and time series model selection. 1998.

[8] Valérie Mignon. *Econométrie : Théorie et applications*. 2008.

[9] A. Raftery. Bayesian model selection in social research. *Sociological Methodology*, 25:111, 1995.

[10] Robert H. Shumway and David S. Stoffer. *Time Series Analysis and Its Applications*. Springer, 2000.

[11] M. A. Wincek and G. Reinsel. An exact maximum likelihood estimation procedure for regression- arma time series models with possibly nonconsecutive data. *Journal of the royal statistical society series b-methodological*, 48:303–313, 1986.