Goodness of Fit and Lasso Variable Selection in Time Series Analysis

Sohail Chand

Thesis submitted to The University of Nottingham

for the degree of Doctor of Philosophy

January, 2011

Dedicated to my mother and the memory of my father*!*

I love my father as the stars - he's a bright shining example and a happy twinkling in my heart. Terri Guillemet

Abstract

This thesis examines various aspects of time series and their applications. In the first part, we study numerical and asymptotic properties of Box-Pierce family of portmanteau tests. We compare size and power properties of time series model diagnostic tests using their asymptotic χ^2 distribution and bootstrap distribution (dynamic and fixed design) against various linear and non-linear alternatives. In general, our results show that dynamic bootstrapping provides a better approximation of the distribution underlying these statistics. Moreover, we find that Box-Pierce type tests are powerful against linear alternatives while the CvM due to Escanciano (2006b) test performs better against non linear alternative models.

The most challenging scenario for these portmanteau tests is when the process is close to the stationary boundary and value of m, the maximum lag considered in the portmanteau test, is very small. In these situations, the χ^2 distribution is a poor approximation of the null asymptotic distribution. Katayama (2008) suggested a bias correction term to improve the approximation in these situations. We numerically study Katayama's bias correction in Ljung and Box (1978) test. Our results show that Katayama's correction works well and confirms the results as shown in Katayama (2008). We also provide a number of algorithms for performing the necessary calculations efficiently.

We notice that the bootstrap automatically does bias correction in Ljung-Box statistic. It motivates us to look at theoretical properties of the dynamic bootstrap in this context. Moreover, noticing the good performance of Katayama's correction, we suggest a bias correction term for the Monti (1994) test on the lines of Katayama's correction. We show that our suggestion improves Monti's statistic in a similar way to what

Katayama's suggestion does for Ljung-Box test. We also make a novel suggestion of using the pivotal portmanteau test. Our suggestion is to use two separate values of m, one a large value for the calculation of the information matrix and a smaller choice for diagnostic purposes. This results in a pivotal statistic which automatically corrects the bias correction in Ljung-Box test. Our suggested novel algorithm efficiently computes this novel portmanteau test.

In the second part, we implement lasso-type shrinkage methods to linear regression and time series models. We look through simulations in various examples to study the oracle properties of these methods via the adaptive lasso due to Zou (2006). We study consistent variable selection by the lasso and adaptive lasso and consider a result in the literature which states that the lasso cannot be consistent in variable selection if a necessary condition does not hold for the model. We notice that lasso methods have nice theoretical properties but it is not very easy to achieve them in practice.

The choice of tuning parameter is crucial for these methods. So far there is not any fully explicit way of choosing the appropriate value of tuning parameter, so it is hard to achieve the oracle properties in practice. In our numerical study, we compare the performance of k-fold cross-validation with the BIC method of Wang et al. (2007) for selecting the appropriate value of the tuning parameter. We show that k-fold cross-validation is not a reliable method for choosing the value of the tuning parameter for consistent variable selection.

We also look at ways to implement lasso-type methods time series models. In our numerical results we show that the oracle properties of lasso-type methods can also be achieved for time series models. We derive the necessary condition for consistent variable selection by lasso-type methods in the time series context. We also prove the oracle properties of the adaptive lasso for stationary time series.

Acknowledgements

I am very grateful to my research supervisors, Professor Andrew Wood and Dr. Chris Brignell. I really learnt a lot under their supervision. Through out my studies, I found them very helpful and they were always available whenever I needed help. This work could not be a reality without their encouragement and expert guidance.

It had been a pleasure to have an office in the Pope building, with many friends around. I will not forget those chats and debates, we had during the lunch breaks.

I am indebted to thank all my relatives, especially my mother and brothers, who missed me back at home and counted down every single day throughout this period. I would like to thank my wife, Faiza, for being with me and for her continuing love and support throughout the long period of this project. I cannot forget all the delicious meals, especially the packed lunches, she cooked for me. I also thank Dawood, my son, and my daughters, Maryam and Ayesha, for all the lovely moments I spent with them in my leisure time. They always made me happy and put a smile on my face.

The work in this thesis was funded by the University of the Punjab, Pakistan, under the Faculty Development Program of the Higher Education Commission, Pakistan, whom I gratefully acknowledge.

Contents

List of Figures

List of Tables

List of Algorithms

CHAPTER 1

Introduction

1.1 Introduction

A time series is a set of observations y_t, with each observation being recorded at specified time t (Brockwell and Davis, 1991). Time series models have wide applications in science and technology. Examples of time series can be found in almost every field of life including, for example, economics, astronomy, physics, agriculture, genetic engineering and commerce.

Mathematical models play an important role in the statistical analysis of data. These models can be deterministic or stochastic. In time series analysis the first and most important step is to identify the appropriate class of mathematical models for the data. As in regression problems, model criticism is an important stage in time series model building, where the fitted model is under scrutiny. To improve the model, we need to go through an iterative procedure of identification, estimation and diagnostic checking. The diagnostic checking not only examines the model for possible shortcomings but it can also suggest ways to improve the model in the next iterative stage (Box and Jenkins, 1994, Chapter 8).

In this thesis, we are interested in goodness-of-fit tests used for diagnostic checking of linear time series models so we will only consider the linear time series with finite second order moment. We will mainly look at overall goodness-of-fit tests suggested in literature e.g. Box and Pierce (1970), Ljung (1986), Monti (1994), Escanciano (2007), Katayama (2008), Katayama (2009). The goodness-of-fit tests used to test the signifi-

cance of a group of first m, say, autocorrelations are called portmanteau tests. A review of literature on goodness-of-fit tests is briefly given in Section 1.4 and discussed with more detail in Section 2.2.1.

Variable selection, especially in high dimensional settings, is important to have the optimal subset of predictors. In regression we have methods, e.g. the lasso (Tibshirani, 1996), which can do variable selection and parameter estimation simultaneously. Variable selection sometimes lead to greater prediction accuracy (Hastie et al., 2001, p.57). These methods have not been widely discussed for time series models. In this thesis we have developed a novel approach to the use of lasso-type methods for multivariate time series analysis including a study of the oracle properties of our proposals. Thus we have mainly focused on two aspects of time series model building, namely (i) goodness of fit tests for diagnostic checking of time series models and (ii) applications of shrinkage methods to time series models.

The first part of this thesis includes numerical and theoretical results on time series goodness-of-fit testing, which is an important part of model building. We study goodness-of-fit tests under their distribution based on first-order asymptotic theory (Ljung and Box, 1978, McLeod, 1978, Katayama, 2008) and distribution approximated by a variety of bootstrap methods including dynamic (MacKinnon, 2006) and fixed design bootstrap methods (Escanciano, 2007). We present some numerical results for the bootstrap distributions of these tests and also provide some theoretical justification of dynamic bootstrap methods. For details see Section 2.3.1.

In the second part of the thesis, we investigate oracle properties (Fan and Li, 2001) of lasso-type methods for regression and time series models. Firstly, we look at the implementation of the lasso (Tibshirani, 1996) and adaptive lasso (Zou, 2006) to linear regression models. We discuss the scenarios where the lasso does not achieve the oracle properties while the adaptive lasso does. We find the necessary and almost sufficient condition discussed by Zou (2006) and Zhao and Yu (2006) is an important condition for consistent variable selection for these lasso-type methods. We also notice for the adaptive lasso that normalisation of the predictors after rescaling with the adaptive weights results in the adaptive lasso with uniform weights i.e. the standard lasso.

The properties of lasso-type methods are well studied for regression models, see e.g. Tibshirani (1997) discussed the application of lasso to Cox proportional hazard models, Van De Geer (2008) studied the application of the lasso to high-dimensional generalized linear models. But application of lasso-type methods to time series models is still in its early stages. Some discussion can be found on the ways to implement lasso-type methods to time series data, see e.g. Haufe et al. (2008), Gustafsson et al. (2005), Hsu et al. (2008), Nardi and Rinaldo (2008) but we cannot find any theoretical results in the time series setting.

Haufe et al. (2008) studied the sparse causal discovery of multivariate time series using simulation study. They compared the performance of group lasso (Yuan and Lin, 2006) and ridge regression (Hoerl and Kennard, 1970) with multiple testing (Hothorn et al., 2008). Gustafsson et al. (2005) applied lasso to time series data of gene-to-gene network. Hsu et al. (2008) has shown good performance of the lasso for multivariate time series models in comparison to the conventional information-based AIC (Akaike, 1974) and BIC (Schwarz, 1978) methods. He also proved the asymptotic distribution of lasso estimates under vector autoregressive models. Nardi and Rinaldo (2008) has derived set of conditions when the lasso estimation is consistent in model selection, estimation and prediction but these results are proved for univariate autoregressive process.

We present the implementation of lasso-type methods to vector autoregressive models. We prove the necessary condition for the consistent variable selection property of these methods. We also give theoretical proofs for the oracle properties of the adaptive lasso for time series models on the lines of Zou (2006). Our results show, as for regression models, the oracle properties of the adaptive lasso also hold for stationary time series models.

The rest of this chapter is organised as follows: Section 1.2 gives some important definitions used in time series analysis. Important types of stochastic processes are defined in Section 1.3. We will give a literature review of time series model diagnostic checking in Section 1.4. Bootstrap methods are briefly defined in Section 1.5. Finally, in Section 1.6, we will define the importance of variable selection with a brief survey of popular methods used to do variable selection.

1.2 Some Definitions

Time series data have unique characteristics and importance as the dependence among the observations can be used to forecast the phenomenon for some future time. Time series analysis provides the tools for the analysis of this dependence. The use of time series stochastic and dynamic models play a vital role in this analysis. Assuming that, in time series each observation y_t is a realization of a certain random variable Y_t, we can consider the time series $\{y_t\}_{t\geq 1}$ as a realization of the family of random variables $\{Y_t\}_{t\geq 1}$. Now we define briefly some of the important types of time series models. Now we give some important definitions, for details see e.g. Box and Jenkins (1994).

Mean and Variance

If Y_t is a stationary process, defined later in this section, with probability distribution $p(y_t)$ then the mean and variance are defined as

$$\mu = E\left(Y_t\right) = \int_{-\infty}^{\infty} y_t p(y_t) dy_t,$$

and

$$\sigma^2 = E\left(Y_t - \mu\right)^2 = \int_{-\infty}^{\infty} \left(y_t - \mu\right)^2 p(y_t) dy_t.$$

For the stationary time series $\{y_t : t = 1, \ldots, n\}$, the sample mean and variance can be defined as

$$\bar{y} = \frac{1}{n} \sum_{t=1}^{n} y_t,$$

and

$$s^2 = \frac{1}{n} \sum_{t=1}^{n} \left(y_t - \bar{y}\right)^2.$$

Strict Stationarity

Let $\{y_t\}_{t\geq 1}$ be an observed series of the stochastic process $\{Y_t\}_{t\geq 1}$ then

$$F_{Y_{t_1},\ldots,Y_{t_n}}(y_1,\ldots,y_n) = P\left(Y_{t_1} \leq y_1,\ldots,Y_{t_n} \leq y_n\right)$$

denote the joint distribution function of $Y_{t_1},\ldots,Y_{t_n}$ for any t_1, t_2, …, $t_n \in \mathbb{Z}$. Then a time series $\{Y_t\}$ is said to be stationary if for any $k \in \mathbb{Z}$, and $n = 1, 2, \ldots$

$$F_{Y_{t_1},Y_{t_2},\ldots,Y_{t_n}}(y_1,\ldots,y_n) = F_{Y_{t_1+k},Y_{t_2+k},\ldots,Y_{t_n+k}}(y_1,\ldots,y_n).$$

Thus, shifting the times of the observations backward or forward by an integer amount k does not affect the joint distribution. This definition is often termed strict stationarity, see e.g. Tong (1990).

Stationarity processes are considered to be in a state of equilibrium. Given the normality assumption, stationarity is the primary assumption in time series analysis as a stationary process can be described by its mean, variance and spectral density function (Box and Jenkins, 1994, p.43). In practical situations, stationarity may or may not hold, but there are various ways of transforming time series data to approximate stationarity, see e.g. Box and Jenkins (1994).

Weak Stationarity

A weaker form of stationarity is that the mean and variance of the process Y_t are constant and their autocovariance function, defined later in this section, does not depend on time t but only on lag k. This is also called second order stationarity as it requires conditions only up to the second order moment. Since a Gaussian process is fully characterised by its first and second order moments, for such processes weak stationarity implies strict stationarity, see e.g. Box and Jenkins (1994).

Autocovariance and Autocorrelation Function

The autocovariance at lag k, denoted by c_k, is the covariance between Y_t and Y_{t+k}.

If Y_t is a stationary process then c_k does not depend on t and is defined as

$$c_k = cov\left(Y_t, Y_{t+k}\right) = E\left(Y_t - \mu\right)\left(Y_{t+k} - \mu\right), \quad k = 0, \pm 1, \pm 2, \ldots.$$

The Y_t process is said to be white noise if $c_k = 0$, when $|k| \geq 1$. The autocorrelation at lag k is

$$r_k = \frac{c_k}{c_0}, \quad k = 0, \pm 1, \pm 2, \ldots. \tag{1.2.1}$$

For the stationary time series $\{y_t : t = 1, \ldots, n\}$, the sample autocovariance function, $\hat{c}_k$ at lag k, can be defined as

$$\hat{c}_k = \frac{1}{n}\sum_{t=1}^{n-k}\left(y_t - \bar{y}\right)\left(y_{t+k} - \bar{y}\right), \quad k = 0, \pm 1, \pm 2, \ldots.$$

Note that divisor is used as n instead of $n - k$ to ensure that the matrix $\hat{C} = [\hat{c}_{i-j}]_{i,j=1}^{n}$ is non-negative definite (Brockwell and Davis, 1991, p.29) and thus for a stationary time series with finite second order moment, we can define autocorrelation function $\hat{r}_k$, at lag k,

$$\hat{r}_k = \frac{\hat{c}_k}{\hat{c}_0} \quad k = 0, \pm 1, \pm 2, \ldots. \tag{1.2.2}$$

The estimated autocorrelation coefficients $\hat{r}_k$ are approximately independently and identically distributed (i.i.d.) with zero mean and

$$var(\hat{r}_k) = \frac{1}{n}.$$

Note that $c_0 = \sigma^2$ and $r_0 = 1$ and for the sample version $\hat{c}_0 = s^2$ and $\hat{r}_0 = 1$, where σ^2 is the variance of the process $\{Y_t : t \in \mathbb{N}\}$ and s^2 is the sample variance.

Partial Autocorrelation Function

The partial autocorrelation function of a stationary process Y_t with finite second order moment, ω_k, can be defined as the correlation between Y_t and Y_{t+k} after removing the effect of intervening observations $Y_{t+1}, \ldots, Y_{t+k-1}$. We can define $\omega_k = \phi_{kk}$ as the

kth coefficient in the autoregressive representation of order k of the jth autocorrelation coefficient

$$r_j = \phi_{k1} r_{j-1} + \ldots + \phi_{kk} r_{j-k} \quad j = 1, \ldots, k. \tag{1.2.3}$$

The sample partial autocorrelation can be defined in parallel to (1.2.3) as

$$\hat{r}_j = \hat{\phi}_{k1} \hat{r}_{j-1} + \ldots + \hat{\phi}_{kk} \hat{r}_{j-k} \quad j = 1, \ldots, k,$$

thus $\hat{\omega}_k = \hat{\phi}_{kk}$ (Brockwell and Davis, 1991, p.102).

Partial autocorrelation plays a vital role in determining the order of the autoregressive model underlying a time series, details given in definition of autoregressive models in Section 1.3. Under the hypothesis that the underlying process is autoregressive of order p, the estimated partial autocorrelation coefficients $\hat{\omega}_k$ of order greater than p are approximately i.i.d. with zero mean and

$$var(\hat{\omega}_k) \approx \frac{1}{n}, \quad k \geq p + 1, \tag{1.2.4}$$

see e.g. Box and Jenkins (1994, p.68).

1.3 Some Important Types of Time Series

In this section we give definitions of some important time series models.

Moving Average model

A process

$$y_t = \beta(L)\varepsilon_t, \tag{1.3.1}$$

is called a moving average process of order q, denoted as MA(q) process, where

$$\beta(L) = 1 + \beta_1 L + \beta_2 L^2 + \ldots + \beta_q L^q, \tag{1.3.2}$$

and ε_t is a white noise sequence (see e.g. Box and Jenkins, 1994). The operator L is called the lag operator such that $L^j Y_t = Y_{t-j}$. For finite q, moving average processes are always stationary and their autocorrelation function r_k, defined in (1.2.1), cuts off to zero for $k \geq q+1$. This is an important property of moving average processes and plays an important role in model identification underlying an observed sample time series.

Autoregressive model

A process

$$\alpha(L)y_t = \varepsilon_t \tag{1.3.3}$$

is called an autoregressive process of order p, denoted as $AR(p)$ process, where

$$\alpha(L) = 1 - \alpha_1 L - \alpha_2 L^2 - \ldots - \alpha_p L^p. \tag{1.3.4}$$

An $AR(p)$ process is said to be stationary when roots of $\alpha(L) = 0$ lie outside the unit circle or roots of $\alpha(L^{-1}) = 0$ lie inside the unit circle, where

$$\alpha(L^{-1}) = 1 - \alpha_1 L^{-1} - \alpha_2 L^{-2} - \ldots - \alpha_p L^{-p}.$$

The autocorrelation function of an AR(p) process is infinite in extent e.g. it can be a damped sine wave or an exponentially decaying curve. For example, for an AR(1) process $y_t = \phi y_{t-1} + \varepsilon_t$, autocorrelation function shows an exponential decay if the autoregressive parameter is positive i.e. $0 < \phi < 1$ while it makes a damped sine wave if autoregressive parameter is negative i.e. $-1 < \phi < 0$ (Box and Jenkins, 1994, p.58) But the partial autocorrelation function of AR(p) process is non-zero only for first p lags i.e. $\omega_k = 0$ for $k \geq p+1$ (Brockwell and Davis, 1991, p.100).

Autoregressive Moving Average model

A process

$$\alpha(L)y_t = \beta(L)\varepsilon_t \tag{1.3.5}$$

is called an autoregressive moving average process, denoted as ARMA(p,q) (see e.g. Box and Jenkins, 1994), where $\beta(L)$ and $\alpha(L)$ are defined in (1.3.2) and (1.3.4) respectively. It is important to note that typically a stationary time series can be represented simultaneously by an autoregressive, moving average or mixed autoregressive moving average process of adequate order. The ARMA(p,q) model results in a more parsimonious model representation.

An ARMA(p,q) model can be represented in AR(∞) form as

$$\pi(L)y_t = \varepsilon_t, \tag{1.3.6}$$

where $\pi(L) = \alpha(L)\beta(L)^{-1} = \sum_{i=0}^{\infty}\pi_i L^i$. We can also write ARMA$(p,q)$ model in MA(∞) form as

$$y_t = \psi(L)\varepsilon_t, \tag{1.3.7}$$

where $\psi(L) = \beta(L)\alpha(L)^{-1} = \sum_{i=0}^{\infty}\psi_i L^i$. See e.g. Wei (2006, Chapter 3), Brockwell and Davis (2002, Chapter 6) for detailed discussion including the applications of linear models.

Autoregressive Integrated Moving Average model

Suppose we have a non stationary ARMA$(p+d,q)$ process of the form $\alpha'(L)y_t = \beta(L)\varepsilon_t$, such that d roots of $\alpha'(L) = 0$ lie on the unit circle. In such situations we can write it as a stationary process w_t such that $\alpha(L)w_t = \beta(L)\varepsilon_t$ where $w_t = \nabla^d y_t$. We can say y_t is an ARIMA(p,d,q) and w_t is an ARMA(p,q). The $\alpha(L)$ and $\beta(L)$ are defined in (1.3.4) and (1.3.2) respectively.

Non Linear Time Series Models

Many time series especially occurring in the natural sciences and engineering cannot be modeled by linear processes. These kinds of time series can have trends which can be best modeled by nonlinear processes. The model building process for nonlinear time series is much more complicated than for linear time series. The important types of nonlinear time series includes bilinear, threshold autoregressive, exponential autoregressive, autoregressive conditional heteroscedastic (ARCH), generalized autoregres-

sive heteroscedastic (GARCH) and stochastic and random coefficient models see e.g. Fan and Yao (2003, Chapter 1), Li (2004, Chapter 5), Brockwell and Davis (1991, Chapter 13) and Chatfield (2004, Chapter 11). Some of these models are defined in Section 2.4.

As we have discussed earlier, finite order moving average processes are always stationary, so in the analysis of these processes for uniqueness purposes we need some conditions on the parameters of the process. Here we give the definition of invertibility, an important condition on moving average processes.

Invertibility

A moving average process $\{Y_t\}$ is said to be invertible, if the roots of $\beta(L) = 0$ lie outside the unit-circle where $\beta(L)$ is defined in (1.3.2). The invertibility condition is independent of the stationarity condition and can also be applied to non-stationary linear time series. Invertibility is required for uniqueness purposes as two normal stationary processes can have same autocorrelation function see e.g. Chatfield (2004, p.37).

1.4 Diagnostic Checking

As mentioned earlier, time series model building is a three stage iterative process consisting of identification, estimation and diagnostic checking. Once the model is identified and fitted to an observed series, the next stage is to check the model for possible discrepancies.

One approach is to assume that the fitted model is under-fitted and so suggest a new model with some additional parameters. This method is called overfitting but the practical problem is to know the directions in which the model should be augmented. An analysis of the identified and overfitted model leads to the conclusion if the additional parameters are needed. Information criteria like Akaike information criterion (AIC) (Akaike, 1974) and Bayesian information criterion (BIC) (Schwarz, 1978) can be used for the final model selection. See McQuarrie and Tsai (1998, Chapter 3) Box and Jenkins (1994, Section 8.1.2) for details.

Residuals obtained from the fitted model are important for investigating the pos-

sible discrepancies in the model and also to further suggest some modifications to the model. Residuals are analysed and checked if they satisfy the model assumptions. Any significant differences from the model assumptions mean we fail to prove that our fitted model is correct.

Residuals plots may be the first step to look at the patterns and behaviour of residuals. Residuals plots along with plots of residual autocorrelations and partial autocorrelations provide an important set of diagnostic tools. Any non random pattern on the residuals plot or any significant residual autocorrelation suggest modification in the fitted model. Figure 1.1 shows that autocorrelations and partial autocorrelations for series z is lying within the confidence limits therefore we can consider series z as a purely random series. For series x, autocorrelation plot is cutting off after lag 1 and partial autocorrelation function is showing a damped sine wave pattern, a behaviour of moving average process of order 1. Similarly for series y, partial autocorrelation is dying off after lag 1 with autocorrelation showing a pattern of damped sine wave, so series y can be identified an autoregressive process of order 1. The identified moving average and autoregressive models should be considered as possible candidate models which can be further tested in the iterative procedure of model building.

The autocorrelation (partial autocorrelation) plot of the residuals is the graph where residuals autocorrelations up to some finite lag, say m, are plotted along with large sample confidence limits. Any autocorrelation (partial autocorrelation) lying outside these limits indicates some non randomness in the residuals. Instead of testing the significance of individual autocorrelations, Box and Pierce (1970) suggested a portmanteau test for the first m autocorrelations. Later, several modifications of Box-Pierce were suggested in the literature. A survey of these tests is given in Section 2.2.1.

In the following section, we give a brief review of some bootstrap methods which are commonly used for time series models, but before that we give a description of the Monte Carlo method.

Since the paper by Meteopolis and Ulam (1949) and the advent of high speed computers, the Monte Carlo method has been applied in almost every field of science e.g. physics, biological sciences, finance etc. Monte Carlo methods use repeated sampling

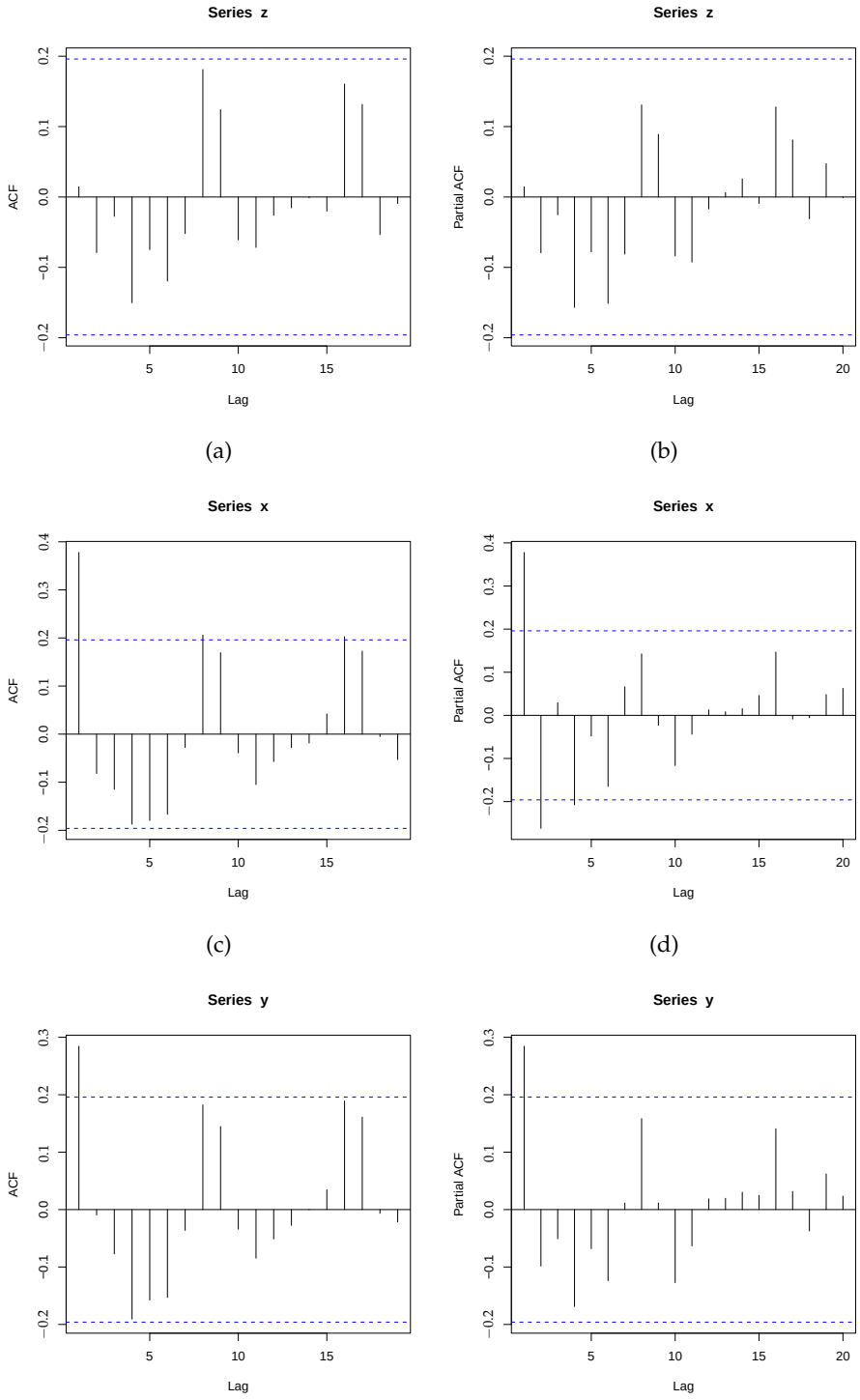


Figure 1.1: Examples of autocorrelation function and partial autocorrelation functions.

and provide an efficient numerical method to solve a statistical problem for example, we can obtain the first few moments of a distribution even without having any priori knowledge of this distribution. For details see Robert and Casella (2004).

1.5 Bootstrap Methods

In practice, we come to situations when it is very hard or sometimes impossible to work out the asymptotic distribution of an estimator. In these situations, an approximation of the asymptotic distribution can be obtained by a resampling method. Though the concept of bootstrap methods goes back to the 1930s, Efron (1979) first introduced it in a unified way.

Bootstrap methods are based on a simple idea that the relation between population and sample can be recreated by resampling from the sample. Bootstrap methods provide mechanisms to generate bootstrap samples. The concept of bootstrap methods is quite simple in the case of i.i.d. random variables but the situation becomes complicated for time series data (Lahiri, 2003).

One way to resample from time series data is the block bootstrap method where the sample is divided into blocks, overlapping (Künsch, 1989) or non-overlapping (Politis and Romano, 1992), of a certain length. Block length is an important issue and is chosen such that the dependence structure in the original sample can still be explained by the bootstrap sample. Under the stationarity condition each block should have the same joint probability distribution. Block bootstrapping is a non parametric bootstrap method. There are some other parametric and semi parametric bootstrap methods for time series data. Assuming that we have some knowledge of the underlying distribution, say Gaussian, the parametric bootstrap is sampling from the estimated distribution.

Semi parametric bootstrap methods use the model structure to resample the residuals. The residuals are obtained by fitting the model to time series data. The residuals can be considered approximately i.i.d.. Having the resamples of these residuals, we can use the fitted model to obtain the bootstrap samples of the time series. Sometimes, residuals may require some transformation for centering and scale adjustment, see e.g. Stute et al. (1998).

See Section 2.3.1 for more discussion on bootstrap methods.

1.6 Variable Selection

In the second part of our thesis we look at the implementation of shrinkage methods to regression and time series. Here we briefly introduce these methods, mainly in the context of regression analysis. Later in Chapter 5, we will implement some of these methods in the regression setting while application to time series data will be discussed in Chapter 6.

Discovering the relationships between the response variable $\{y_i : i = 1, \ldots, n\}$ and the set of predictors $\{\boldsymbol{x}_j : j = 1, \ldots, p\}$ is one of the objectives of regression analysis. This relationship is later used for statistical inference and prediction. The linear regression model is usually defined as:

$$y_i = \beta_0 + \beta_1 x_{i1} + \ldots + \beta_p x_{ip} + \varepsilon_i, \quad i = 1, \ldots, n.$$

In vector form, we can write the model as

$$y_i = \beta_0 + \boldsymbol{x}_i^T \boldsymbol{\beta} + \varepsilon_i \tag{1.6.1}$$

such that $y_i \in \mathbb{R}$ is the response variable, $\boldsymbol{x}_i = (x_{i1}, \ldots, x_{ip})^T \in \mathbb{R}^p$ is the p-dimensional set of predictors, $\varepsilon_i \sim N(0, \sigma^2)$ and $\boldsymbol{\beta} = (\beta_1, \ldots, \beta_p)$ is the set of parameters and β_0 is a constant.

The ultimate question is to estimate the β_j's using a set of training data $(\boldsymbol{x}_1^T, y_1)$, $\ldots$, $(\boldsymbol{x}_n^T, y_n)$, where $\boldsymbol{x}_i = (x_{i1}, \ldots, x_{ip})^T \in \mathbb{R}^p$. The method of least squares, based on minimising the residual sum of squares, is the most popular method to estimate the model. The least squares estimates always provide non-zero estimates even if true model is sparse i.e. some of the model parameters are exactly zero. This is the reason that least squares estimates generally have low bias but may suffer from large prediction variance especially when true model is sparse. The large prediction variance is due to the fact that least squares estimation always ends up with the full model see e.g. Hastie et al. (2001, p.57).

1.6.1 Subset Selection

In model building, we often have a large set of predictors. As all the variables are not equally important for the model, so we seek for a parsimonious model. The parsimonious models are very important for prediction purposes as overfitted models sometimes have the higher prediction variance see e.g. McQuarrie and Tsai (1998, Section 1.2), Hastie et al. (2001, p.57).

Variable selection in regression is so important that Bradley Efron, the inventor of bootstrap methods, has named it as one of the most important problems in statistics (Hesterberg et al., 2008). All the predictors, in general, are not worth to include in the model especially when p is very large. We look for a subset of $\{\beta_j : j = 1, \ldots, p\}$ which optimizes a criterion, see Hocking and Leslie (1967). This criterion can be based on certain model goodness measures like prediction error, goodness of fit measures or on estimating some measures of distance between the model based on the subset and the true model, see e.g. Seber and Lee (2003).

Searching through all possible subset models is computationally intensive. Best subset selection produces a model that is interpretable and has possibly lower prediction error than the full model. It is one way to fit a simple model but, as mentioned by (Fan and Peng, 2004), is not feasible with a large set of predictors. Methods such as forward stepwise selection and backward elimination, called greedy algorithm, provide a good path through them (Hastie et al., 2001, p.58). More recently, there are suggestions e.g. Hall et al. (2009b) and Cho and Fryzlewicz (2010) based on tilting for variable selection in high-dimensional setting.

Shrinkage methods are another choice which lead to a simpler model in terms of number of variables in the model. In the following section, we give a review of some of the important shrinkage methods.

1.6.2 Shrinkage Methods

Subset selection is a discrete process i.e. either a candidate variable is included or excluded from the model. Thus a model, though simpler, can have a relatively high prediction variance. Shrinkage methods answer this problem in an opposite way i.e. retain

all the predictors but use a penalised least squares method instead of standard least squares estimation. The concept of shrinkage was first introduced by James and Stein (1961). Shrinkage is desired when a simpler model is desired at a cost of increased prediction error but this increase is relatively lesser than result for a discrete process like subset selection. Here we give some brief description of some of the shrinkage methods. For more detailed discussion see Section 5.2.

Ridge Regression

Ridge regression (Hoerl and Kennard, 1970) is a form of shrinkage method, which shrinks the coefficients by imposing a penalty on the sum of squares of the parameters. Ridge regression was primarily suggested for improving the estimation of regression coefficients when the predictors are highly correlated. We can also define ridge regression as a mean or mode of a posterior distribution of response variable with a suitable chosen of prior distribution for regression parameter, in which case we can see that optimal performance of ridge regression much depends on the distribution of regression coefficients , see e.g. Hastie et al. (2001, p.64). One drawback of ridge regression is that it fails to produce a simple model as it retains all the variables, see e.g. Seber and Lee (2003). Ridge regression is preferred to variable subset selection when objective is to minimize prediction error (Frank and Friedman, 1993).

Garrote

Shrinkage and simple models are desired simultaneous for an interpretable model with low prediction variance (Hastie et al., 2001, Section 3.4). Subset selection provides the simpler model but fails to shrink while ridge regression shrinks the regression coefficients but retains all the variables in the model. Garrote (Breiman, 1995) does shrinkage while setting some the coefficients exactly to zero at the same time. Garrote puts a penalty on each individual least squares estimate of β_j's. As the penalty term increases the shrinkage coefficients get smaller and some are even forced to zero. Due to the condition on the shrinkage coefficient to be nonnegative this version of garrote is also called nonnegative garrote, this condition was further relaxed by Breiman (1996).

Lasso

The lasso (Tibshirani, 1996), least absolute shrinkage and selection operators, is an L_1 penalised least squares regression. Like garrote it shrinks some of the coefficients while setting the rest of them exactly to zero. This property of the lasso makes it a method which enjoys good properties of best subset regression and ridge regression (Hastie et al., 2001, p.82) The lasso estimator of β for model (1.6.1) is defined by

$$\hat{\beta}_j^* = \text{argmin} \left\{ \sum_{i=1}^{n} \left(y_i - \beta_0 - \sum_{j=1}^{p} \beta_j x_{ij} \right)^2 + \lambda \sum_{j=1}^{p} |\beta_j| \right\}, \quad j = 1, \ldots, p,$$

where $\lambda > 0$ is a user-defined tuning parameter. The choice $\lambda = 0$ corresponds to the least squares estimate and larger values of λ result in a higher amount of shrinkage i.e. relatively more variables will shrink to zero. The theoretical properties of the lasso method are very appealing but it had been computationally expensive until Efron et al. (2004) suggested an efficient least angle regression (LARS) algorithm for finding the solution path of the lasso method. The LARS correctly organizes the calculations thus the computational cost of the entire p steps is of the same order as that required for the usual least squares solution for the full model.

Variable selection is an important property of shrinkage methods. Zou (2006) and Zhao and Yu (2006) has discussed a necessary condition for the lasso methods to achieve consistency in variable selection. Zou (2006) has also suggested the use of adaptive weights and showed that this results in putting a different penalty on each parameter, which leads to consistent variable selection. The same LARS algorithm (Efron et al., 2004) can be used to obtain the adaptive lasso estimates, see Section 5.2.4 for detailed discussion on the adaptive lasso.

The rest of this thesis is organised as follows:

Goodness of fit testing is an important stage of time series model building. In Chapter 2, we study the properties of time series goodness of fit tests. Though these properties are well studied but there is not much literature available studying these tests especially for semi-parametric bootstrap methods. We give some numerical results to compare the performance of various resampling residuals approaches in providing an

approximation of finite sample distributions underlying these tests. We also compared the power of these tests against various linear and non-linear alternative models.

Katayama (2008) derived a bias term in Ljung-Box test. This motivated us to compare Katayama's bias corrected Ljung-Box test with other goodness-of-fit tests using asymptotic and dynamic bootstrap method. In Chapter 3, we numerically study the effect of the Katayama (2008) bias correction term in the Ljung and Box (1978) portmanteau test. We also suggest a set of algorithms to estimate this bias correction term. In this same Chapter 3, we present a novel suggestion for a bias correction term in Monti (1994) test on the lines of Katayama (2008). Chapter 3 also includes numerical results on Katayama's suggested multiple test (Katayama, 2009). We suggest a hybrid bootstrap approach to estimate the joint significance levels of this multiple portmanteau test. We also suggest the use of pivotal portmanteau test and an algorithm for its efficient computation.

The results in Chapter 2 lead to the conclusion that dynamic bootstrap sampling provide an approximation of the finite sample distribution better than first order asymptotic theory. This motivated us to provide a theoretical justification of this finding. In Chapter 4, where we have provided theoretical insight of good performance of dynamic bootstrap methods in estimating the distribution of the portmanteau tests especially when m is small and the process is close to stationarity boundary. We provide a set of lemmas to prove the asymptotic normality of the least squares estimates. We have proved the bounds on the cumulants of the residuals which are used to derive the normality of bootstrap least squares estimates. We discuss an approach to use these results as a justification of good performance of dynamic bootstrap method for portmanteau tests. We, along the lines of Katayama (2008), derive and suggest a bias correction term in Monti's(1994) test.

Issues like selection of tuning parameter for these shrinkage methods and conditions required to achieve oracle performance by these methods are still areas of interest. In the second part of our thesis, we look at the oracle properties of lasso-type methods. In particular, we study the property of consistent variable selection for these methods. In Chapter 5, numerical results on variable selection of the lasso (Tibshirani, 1996) and adaptive lasso (Zou, 2006) are given and discussed. We present some interesting nu-

merical results about the selection of the tuning parameter. Lasso-type methods are originally suggested for linear regression models and their theoretical properties are proved in the regression context (see e.g. Knight and Fu, 2000).

Shrinkage methods are now widely used in regression setting but it is less explored for time series setting. Though time series models have some similarities with regression models, the results are not trivial (see e.g. Anderson, 1971). In Chapter 6, we apply lasso-type methods to the multivariate time series models. We also give some novel results about the application of these methods to linear time series models. Like regression, we derive a necessary (but not sufficient) condition for consistent variable selection by lasso-type methods. We prove the asymptotic normality of the adaptive lasso and show that the adaptive lasso is always consistent in variable selection.

Finally, in Chapter 7, we give a summary and conclusion of our results. Future directions of our work are also discussed in this same chapter.

CHAPTER 2

Bootstrap Goodness of Fit Tests for Time Series Models

2.1 Introduction

In this chapter, we mainly look at the properties of goodness-of-fit tests for linear time series models under semi-parametric bootstrap methods. Model criticism is an important stage of model building and thus goodness of fit tests provide a set of tools for diagnostic checking of the fitted model. Box and Pierce (1970) test and its several other versions are perhaps the most commonly used type of portmanteau test (Mainassara et al., 2009). The portmanteau tests are used as overall tests for an entire set of, say, the first m autocorrelations assuming that the true model is correct.

The asymptotic distribution of these tests is well studied in the literature and many researchers have questioned the appropriateness of the χ^2_{m-p-q} distribution as its approximation under ARMA(p,q) as a true model, see e.g. Katayama (2008) and references therein. Moreover, the choice of m is very important in the χ^2_{m-p-q} approximation and power of these tests.

In this chapter, we numerically study the size and power of some of the popular time series goodness of fit tests. Escanciano (2006b) has studied power of various goodness of fit tests under the fixed design wild bootstrap. Horowitz et al. (2006) has compared performance of Box and Pierce (1970) test with some other tests under block-of-block bootstrapping. See Section 2.3.1 for more detailed discussion.

Most of the literature in time series bootstrap goodness-of-fit tests is related to non-parametric bootstrap methods. Results in Escanciano (2007) and Katayama (2008) motivated us to look at size and power properties of these goodness-of-fit tests for bootstrap methods using resampling residuals. The novelty of our study is that we study the size of the tests under various semi-parametric bootstrap designs described in Section 2.3.1. Moreover, we also compare the power of these tests with the Cramer von Mises (CvM) (Escanciano, 2007) statistic against various linear and non-linear alternative models. We also study size and power of various versions of Box-Pierce test, Monti (1994) test and CvM test. To the best of our knowledge, these tests have not been compared in these scenarios in the literature.

Our results show that Box-Pierce type tests do well against the linear alternatives but fail to perform against the non-linear alternatives, while the situation reverses for the CvM statistic due to Escanciano (2007), i.e, the CvM statistic does well against non linear alternatives but much less well against linear alternatives. Moreover, dynamic bootstrap methods show better performance than the fixed design bootstrap in our example. We have not found any advantage of using wild residuals in our simulations.

The remainder of the chapter is organized as follows. In the next section a review of the literature on available diagnostic tests is given. Section 2.3 describes the different bootstrap methods in a time series context. Section 2.4 gives the estimation procedure and algorithms for Monte Carlo simulations for computing the size and power of the tests. Finally, Section 2.5 presents the results of simulations and discussion of the results.

2.2 Literature Review

In practice, there are many possible linear and non-linear models for a problem under study e.g. autoregressive, moving average, mixed ARMA models, threshold autoregressive etc. Box and Jenkins (1994) have described time series model building as a three-stage iterative procedure that consists of identification, estimation and validation.

Identification of the model is partly science and partly art. There are no exact ways

of identifying the underlying model though there are some tools, for example, the autocorrelation and partial autocorrelation plots to identify the general class of underlying model, see Box and Jenkins (1994, p.196). See Section 1.3 for the definitions of autocorrelation and partial autocorrelation. Importantly, it should be noted that at the identification stage, especially dealing with complex situations, we identify a class of models that will later be efficiently fitted and then go through the diagnostic checking phase (Box and Jenkins, 1994). Identification of a single model makes the practitioner assume that the data are generated under this particular identified model. To overcome this problem, model averaging methods such as Bayesian model averaging can be used see e.g. Hoeting et al. (1999) and references therein.

There are rigorous ways to estimate the parameters of autoregressive models such as the methods of maximum likelihood estimation, least squares estimation and Yule-Walker estimation. Moving average models can be estimated through the innovations method, see e.g. Brockwell and Davis (1991, Chapter 8). The estimates of moving average models and the mixed models can also be obtained graphically or through iterative estimation procedures such as non-linear minimization (see e.g. Box and Jenkins, 1994, Chapter 7).

Time series models should be able to describe the dependence among the observations, see e.g. Li (2004). It is a well-discussed issue that in time series model criticism, the residuals obtained from fitting a potential model to the observed time series play a vital role and can be used to detect departures from the underlying assumptions, (Box and Jenkins, 1994; Li, 2004).

In particular, if the model is a good fit to the observed series then the residuals should behave somewhat like a white noise process. So, taking into account of the effect of estimation, the residuals obtained from a good fit should be approximately uncorrelated. While looking at the significance of residual autocorrelations, one approach is to test the significance of each individual residual autocorrelation which seems to be quite cumbersome. Another approach is to have some portmanteau test capable of testing the significance of the first, say m, residual autocorrelations (Box and Jenkins, 1994; Li, 2004), an approach we now describe.

2.2.1 Diagnostic Tests

Since Box and Pierce (1970) paper, the portmanteau test has become the vital part of time series diagnostic checking. Several modifications and versions of Box and Pierce (1970) has been suggested in the literature, see e.g. Ljung and Box (1978), McLeod and Li (1983), Monti (1994), Katayama (2008), Katayama (2009). These tests are capable of testing the significance of the autocorrelations (partial autocorrelations) up to a finite number of lags.

The third stage of diagnostic checking process (Box and Jenkins, 1994) provides a practitioner an opportunity to test the model before using it for forecasting. This stage not only checks the fitted model for inadequacies but can also suggest improvements in the fitted model in the next iteration of this model building procedure. In this section we will do a literature review of the available diagnostic tests for fitted time series models.

The residuals are very commonly used as a diagnostic tool to test the goodness of fit of models. In a time series context, if the fitted model is good then it should be able to explain the dependence pattern among successive observations. In other words, all the dependence in terms of autocorrelations and partial autocorrelations of the data generating process (DGP) should be explained by the fitted model so there should be no significant autocorrelation and partial autocorrelation in successive terms of the residuals.

In practice the most popular way for diagnostic checking a time series model is the portmanteau test, which tests whether any of a group of the first m autocorrelations $(\hat{r}_1, \ldots, \hat{r}_m)$ of a time series are significantly different from zero. This type of test was first suggested by Box and Pierce (1970), in which they studied the distribution of residual autocorrelations in ARIMA processes defined in Section 1.3. Based on the autocorrelations of the residuals obtained by fitting an ARMA(p, q) model defined in Section 1.3.5 to $\{y_t\}$, they suggested the following portmanteau test

$$Q_m = n \sum_{k=1}^{m} \hat{r}_k^2, \tag{2.2.1}$$

where $\hat{r}_k$ is the residual autocorrelation at lag k defined in (1.2.2). In practice, the op-

timal choice of m is difficult as the use of the χ^2_{m-p-q} approximation and diagnostic checking require large values of m which results in less power and unstable size of test, as noticed by Ljung (1986), Katayama (2008). Katayama (2009) suggested a multiple portmanteau test to overcome this problem, for details see Section 3.4.

Box and Pierce (1970) suggested that $Q_m \sim \chi^2_{m-p-q}$, for moderate values of m and the fitted model is adequate, under the following conditions:

1. $\psi_j \leq O\left(n^{-1/2}\right)$ for $j \geq m - p$, and
2. $\frac{m}{n} = O\left(n^{-1/2}\right)$,

where ψ_j are the weights in the MA(∞) representation as defined in (1.3.7). This approximation requires substitution of residuals, $\hat{\varepsilon}_t$, for the error term, ε_t, in the model but erroneous use of such kind of substitution can lead to a serious underestimation of significance level in diagnostic checking, see (Pierce, 1972) and references therin. Many other researchers have also questioned the distribution of Q_m, (see e.g. McLeod, 1978 and references therein). The choice of m is an important issue.

In the discussion of Prothero and Wallis (1976), Chatfield has mentioned the poor power properties of Q_m and has recommended focusing on residual autocorrelations at the first few lags and seasonal lags. Similar suggestions are also made by Davies et al. (1977). In the same discussion on the Prothero and Wallis paper, Chatfield and Newbold also pointed out the poor approximation of the finite-sample distribution of Q_m. Prothero and Wallis (1976), in their reply to this discussion, suggested the use of the correction factor $(n+2)\,/\,(n-k)$ to Q_m. However, this correction factor may inflate the variance of the resulting statistic relative to that of the asymptotic χ^2_{m-p-q} distribution (see e.g. Davies et al., 1977, Ansley and Newbold, 1979).

An important point to note is that the statistic Q_m has been developed assuming the normality of the white noise process ε_t. As the results of Anderson and Walker (1964) suggest the asymptotic normality of the autocorrelation of a stochastic process is independent of the normality of the stochastic process and only depends on the assumption of finite variance, so the portmanteau test is expected to be insensitive to the normality assumption.

Ljung and Box (1978) suggested the use of the modified statistic

$$Q_m^* = n(n+2)\sum_{k=1}^{p}\frac{\hat{r}_k^2}{n-k}. \tag{2.2.2}$$

They have shown that the modified portmanteau statistic Q_m^* has a finite sample distribution which is much closer to χ^2_{m-p-q}. Their results also show that Q_m^* is insensitive to the normality assumption of ε_t. As pointed out by many researchers e.g. Davies et al. (1977), Ansley and Newbold (1979), the true significance levels of Q_m tends to be much lower than predicted by the asymptotic theory and though the mean of Q_m^* is much closer to the asymptotic distribution, this corrected version of the portmanteau test has an inflated variance. But Ljung and Box (1978) pointed out that approximate expression of variance given by Davies et al. (1977) overestimates the variance of Q_m^*.

Frequently in the literature larger values of m have been used in Q_m and Q_m^*, and the most commonly suggested value is $m = 20$ (see e.g. Davies et al., 1977, Ljung and Box, 1978). Ljung (1986) suggests the use of smaller values of m and has shown that for small values of m, Q_m^* has an approximate $a\chi_b^2$ distribution, where a and b are constants to be determined.

Ljung and Box (1978) also studied the empirical significance levels and empirical powers of Q_m^* for various choices of m and showed that the empirical significance levels for an AR(1) process are close to the nominal level for small choices of m ($m = 10$ or 20) in all the cases except when the AR parameter is close to the boundary of non-stationarity region. This is a very challenging scenario for the χ^2 approximation. We will look at this issue in Chapter 3. Ljung and Box (1978) also showed that approximating asymptotic distribution of $Q_m \sim \chi^2_\nu$, where $\nu = E(Q_m)$ results in performance of Q_m similar to that of $Q_m^* \sim \chi^2_{m-p-q}$.

We have already mentioned that the partial autocorrelation function is an important tool in determining the order of an autoregressive process (Quenouille, 1947). The portmanteau tests Q_m and Q_m^* are based on the autocorrelations. Monti (1994) suggested a portmanteau test

$$Q_m^*(\hat{\omega}) = n(n+2)\sum_{k=1}^{m}\frac{\hat{\omega}_k^2}{n-k}, \tag{2.2.3}$$

where $\hat{\omega}_k$ is the residual partial autocorrelation at lag k. She showed that $Q_m^*(\hat{\omega})$, analogously to Q_m^*, has an asymptotic null distribution χ^2_{m-p-q} and that $Q_m^*(\hat{\omega})$ is more powerful than Q_m^* especially when the order of the moving average component is understated.

As we have discussed earlier, the asymptotic distribution of Q_m and Q_m^* is questioned by several authors in the literature. Though small values of m solve this problem in some situations, it does not work in all cases, for example when the process is nearly stationary, see Ljung (1986). In a very recent paper, Katayama (2008) has suggested a bias correction term, in the Ljung and Box (1978) statistic Q_m^*, defined as

$$B_{m,n}^* = \hat{\boldsymbol{r}}^T \boldsymbol{V}\boldsymbol{D}\boldsymbol{V}\hat{\boldsymbol{r}},$$

where $\hat{\boldsymbol{r}} = (\hat{r}_1, \ldots, \hat{r}_m)^T$, $\boldsymbol{V} = \text{diag}\left(\sqrt{n\,(n+2)\,/(n-1)}, \ldots, \sqrt{n\,(n+2)\,/(n-m)}\right)$, $\boldsymbol{D} = \boldsymbol{X}\left(\boldsymbol{X}^T\boldsymbol{X}\right)^{-1}\boldsymbol{X}^T$ and $\boldsymbol{X}$ is an $(m \times (p+q))$ matrix partitioned into p and q columns, such that each (i,j) element of the partitioned matrix of $\boldsymbol{X}$ is given

$$\boldsymbol{X} = (-\alpha_{i-j}^* \vdots -\beta_{i-j}^*)$$

where α_i^* and β_i^* are defined by

$$\frac{1}{\alpha(L)} = \sum_{i=0}^{\infty} \alpha_i^* L^i$$

and

$$\frac{1}{\beta(L)} = \sum_{i=0}^{\infty} \beta_i^* L^i$$

and $\alpha_i^* = \beta_i^* = 0$ for $i < 0$. Katayama (2008) showed the importance of this correction term especially for small values of m and when the roots of the ARMA(p,q) process lie near the boundary of non-stationarity region. So the bias corrected portmanteau test is given by

$$Q_m^{**} = Q_m^* - B_{m,n}^*.$$

For more discussion on Katayama (2008), see Chapter 3.

McLeod (1978, Theorem 1) has showed that $\hat{\boldsymbol{r}}$ is approximately normal with mean $\mathbf{0}$ and $Var(\hat{\boldsymbol{r}}) = (\boldsymbol{I} - \boldsymbol{C})/n$, where $\boldsymbol{C} = \boldsymbol{X}\mathbb{J}^{-1}\boldsymbol{X}$, $\boldsymbol{I}$ is the identity matrix and $\mathbb{J}$ is the Fisher information matrix defined in (3.3.1). We noticed that approximation of $\boldsymbol{C}$ by $\boldsymbol{D} = \boldsymbol{X}(\boldsymbol{X}^T\boldsymbol{X})^{-1}\boldsymbol{X}$, especially when m is small, is a source of bias in approximating the asymptotic distribution of portmanteau tests. We found the use of pivotal statistic automatically corrects for the bias mentioned in Katayama (2008). Pivotal statistics are useful as their asymptotic distribution does not depend on unknown parameters, for details see e.g. Hall (1992, Ch.3). For more details see Section 3.3.

Katayama (2008) suggested a multiple portmanteau test is based on several portmanteau test for a range of small to medium values of m. He also discussed the linkage between his suggested multiple test and the test due to Pena and Rodriguez (2002). He suggested a method based on some iterative procedure to approximate joint distribution of the multiple test as the computation of the distribution is very hard due to correlated elements. See Section 3.4 for details.

For the past few decades the interest of researchers, especially working in the field of financial time series, has been focused on nonlinear models. It has been pointed out by several researchers that the Box-Pierce type tests fail to show good power against nonlinear models (see e.g. Escanciano, 2006b; Pena and Rodriguez, 2002). One important difference between nonlinear and linear models is that former do not inherit properties of innovations e.g. a GARCH model with Gaussian innovations does not have to have a finite order fourth moment, for more detailed discussion see e.g. Fan and Yao (2003, Chapter 4). McLeod and Li (1983) used the sample autocorrelation of the squared residuals to test for linearity against the nonlinearity and showed its good power against departures from linearity.

Escanciano (2007) proposed diagnostic tests based on the CvM test using the weights suggested by Bierens (1982), given by

$$CvM_{exp,P} = \frac{1}{n\hat{\sigma}^2}\sum_{t=1}^{n}\sum_{s=1}^{n}\hat{\varepsilon}_t\hat{\varepsilon}_s \exp\left(-\frac{1}{2}|\mathbf{I}_{t-1,P} - \mathbf{I}_{s-1,P}|^2\right), \tag{2.2.4}$$

where $\hat{\sigma}^2 = \sum_{t=1}^{n} \hat{\varepsilon}_t^2 / n - 1$ is the variance of residuals and

$$\boldsymbol{I}_{t-1,P} = (y_{t-1}, y_{t-2}, \ldots, y_{t-P}) \tag{2.2.5}$$

is the information set at time $t-1$ and dimension P. It can be noticed that the distance $|\mathbf{I}_{t-1,P} - \mathbf{I}_{s-1,P}|^2$ increases very fast with P which results in weights being near 0 when P is relatively large. We have considered the CvM statistic with this weight scheme in our study as it has shown good power properties reported in Escanciano (2006b).

2.3 Methodology

We now consider various versions of the statistics defined in (2.2.1), (2.2.2), (2.2.3) and (2.2.4). We compare empirical size and power of these tests against various linear and non-linear classes of models. Mainly we compare the dynamic and fixed design bootstrap methods but we also look at the usefulness of transformed residuals in bootstrap methods.

2.3.1 Bootstrap Methods

Bootstrap methods are used to estimate the distribution of a test statistic or an estimator. The bootstrap is usually implemented using resampling. Under conditions that hold in wide variety of applications, the bootstrap provides approximations to distributions of statistics that are at least as accurate as, and sometimes are more accurate than, the approximations of first-order asymptotic distribution theory (see Hardle et al., 2003). The reliability of a bootstrap method depends upon the extent to which the bootstrap data generating process (DGP) mimics the true DGP (see MacKinnon, 2006).

The bootstrap method was first suggested by Efron (1979) as a more general method than jackknife. For a detailed discussion on jackknife methods see e.g. Shao and Tu (1995). The idea of bootstrapping residuals was described in Efron (1988) in the context of regression. Much of the earlier work in bootstrap methods was done on i.i.d. random variables data.

For time series data, the dependence structure of the DGP makes it difficult to ap-

ply the bootstrap methods. In general, there are two main bootstrap methods that are used in time series i.e. model-based bootstrap methods and block-resampling bootstrap methods. Generally, the model-based bootstrap methods are called resampling-residuals bootstrap methods.

In block bootstrapping, we divide the sample into overlapping or non-overlapping blocks of a certain length (Hall et al., 1995). The performance of block bootstrap methods much depend on block length. Under the stationarity condition each block should have the same joint probability distribution. In our study we consider only the model-based bootstrapping, as model-based bootstrap methods tend to be more accurate than block bootstrap methods (Lahiri, 2003) and also as our objective is to compare two model-based bootstrap methods, namely dynamic bootstrap and fixed design bootstrap. Lahiri (1999) provides a good comparison of block bootstrap methods with non-random and random block lengths.

Suppose we have a sample time series $\{y_t\}_{t=1}^{n}$ generated by a DGP defined by

$$y_t = f\left(\boldsymbol{I}_{t-1,P}, \boldsymbol{\theta}\right) + \epsilon_t, \tag{2.3.1}$$

where $\boldsymbol{I}_{t-1,P}$ is the information set defined earlier in (2.2.5) and $\boldsymbol{\theta}$ is the vector of model parameters. Suppose the fitted model is

$$\hat{y}_t = f\left(\boldsymbol{I}_{t-1,P}, \hat{\boldsymbol{\theta}}\right), \quad t = P, P+1, \ldots$$

where $\hat{\boldsymbol{\theta}}$ is the estimate of $\boldsymbol{\theta}$. Thus the residuals are

$$\hat{\varepsilon}_t = y_t - \hat{y}_t, \tag{2.3.2}$$

We assume that initial data $y_{t-P}, \ldots, y_0$ are available.

Fully parametric bootstrap method

If the distribution of the error term, ε_t, is assumed to be known up to unknown parameters, then we can use the knowledge of the distribution to select a bootstrap sample. Suppose, for example, that $\varepsilon_t \sim N(0, \sigma^2)$ then the bootstrap DGP will be given

as,

$$y_t^\dagger = f\left(\boldsymbol{I}_{t-1,P}^\dagger, \hat{\boldsymbol{\theta}}\right) + \varepsilon_t^\dagger, \quad t = 1, 2, \ldots$$

where $\varepsilon_t^\dagger \sim N(0, \sigma^2)$, σ^2 is known and $\boldsymbol{I}_{t-1,P}^\dagger = \left(y_{t-1}^\dagger, \ldots, y_{t-P}^\dagger\right)$ is the parametric bootstrap of $\boldsymbol{I}_{t-1,P}$ defined in (2.2.5). If the true parameters are unknown then respective maximum likelihood estimates are used for these unknown parameters (Chernick, 1999, p.124)

Semi-parametric time series bootstrap methods

Under the assumption that the DGP given in (2.3.1) is the true model for the given sample time series, the residuals given in (2.3.2) will serve the purpose of an i.i.d. sample. The following approaches are used in semi-parametric time series bootstrap methods.

Dynamic bootstrap If the error terms, ε_t's, in our DGP are i.i.d., with common variance σ^2, then we can generally make very accurate inferences by using the the dynamic bootstrap (DB) (MacKinnon, 2006). This method requires the i.i.d. assumption of the error term and only mild conditions on its distribution. The DB is defined as:

$$y_t^* = f\left(\boldsymbol{I}_{t-1,P}^*, \hat{\boldsymbol{\theta}}\right) + \varepsilon_t^* \quad \text{for } t = 1, 2, \ldots, n, \tag{2.3.3}$$

where $\boldsymbol{I}_{t-1,P}^* = \left(y_{t-1}^*, \ldots, y_{t-P}^*\right)$ is the dynamic bootstrap of the information set defined in (2.2.5) and ε_t^* is selected at random with replacement from the vector of the residuals $\left(\hat{\varepsilon}_1, \hat{\varepsilon}_2, \ldots, \hat{\varepsilon}_n\right)$.

Dynamic wild bootstrap The dynamic wild bootstrap (DWB) is a simple modification of the dynamic bootstrap. The only difference is to resample the rescaled residuals instead of residuals. These rescaled residuals are usually named as wild bootstrap. Various rescaling schemes have been suggested in the literature, see e.g. Liu (1988) or Stute et al. (1998). The DWB is defined as:

$$y_t^o = f\left(\boldsymbol{I}_{t-1,P}^o, \hat{\boldsymbol{\theta}}\right) + \varepsilon_t^o \quad \text{for } t = 1, 2, \ldots, n, \tag{2.3.4}$$

where $\boldsymbol{I}^o_{t-1,P} = \left(y^o_{t-1}, \ldots, y^o_{t-P}\right)$ is the DWB of the information set defined in (2.2.5) and $\varepsilon^o_t = \hat{\varepsilon}^*_t.v_t$, such that the sequence v_t is i.i.d. with zero mean, unit variance and finite fourth moment. We can define a sequence $\{v_t\}$ of i.i.d. Bernoulli variates for transforming the i.i.d. residuals to wild residuals e.g using as in Liu (1988)

$$v_t = \begin{cases} -1 & \text{with } p = \frac{1}{2} \\ +1 & \text{with } p = \frac{1}{2}. \end{cases} \tag{2.3.5}$$

or in Stute et al. (1998)

$$v_t = \begin{cases} \frac{1-\sqrt{5}}{2} & \text{with } p = \frac{1+\sqrt{5}}{2\sqrt{5}} \\ \frac{1+\sqrt{5}}{2} & \text{with } p = 1 - \frac{1+\sqrt{5}}{2\sqrt{5}}. \end{cases} \tag{2.3.6}$$

The fixed design wild bootstrap In fixed design wild bootstrap (FWB), the bootstrap sample is generated from the fixed design $\boldsymbol{I}_{t-1,P}$. This method is called fixed design as, unlike dynamic bootstrap, the information set for observed series is used. Moreover, the residuals are transformed to wild residuals using the suggested transformations (see Liu, 1988 and Stute et al., 1998). The FWB is defined as:

$$y^\diamond_t = f\left(\boldsymbol{I}_{t-1,P}, \hat{\boldsymbol{\theta}}\right) + \varepsilon^o_t \quad \text{for } t = 1, 2, \ldots, n, \tag{2.3.7}$$

where ε^o_t is as defined above.

Fully non-parametric Bootstrap methods

The model based (i.e. parametric or semi-parametric) bootstrap methods are based upon the assumption of i.i.d. of error terms, ε_t. When this assumption is violated, we cannot resample the residuals. The sieve and block bootstrap are the most popular bootstrap methods for non-i.i.d. error terms.

The sieve bootstrap Suppose that the error term ε_t follows an unknown stationary process with homoscedastic innovations. This method is implemented in three steps.

- The model is estimated, imposing the null hypothesis if there is any, and the residuals $\hat{\varepsilon}_t$'s are obtained.

- For several values of p, an AR(p) model is fitted to $\hat{\varepsilon}_t$'s as

$$\hat{\varepsilon}_t = \sum_{i=1}^{p} \pi_i \hat{\varepsilon}_i + u_t.$$

  Maximum-Likelihood method or Yule-Walker equations are preferred to estimate this model. After p has been chosen as the order for the best model, the model-based approach can be used to obtain $u_t^{\ddagger}$, resamples of u_t or rescaled u_t.

- The final step is to generate bootstrap data using the equation

$$y_t^{\ddagger} = f\left(\boldsymbol{I}_{t-1,P}, \hat{\boldsymbol{\theta}}\right) + u_t^{\ddagger}.$$

The sieve bootstrap assumes that u_t are i.i.d., so it cannot be applied to heteroscedastic models. The other limitation is the ability of the AR(p) process to provide a good approximation to every stationary, stochastic process.

The block bootstrap The main idea of the block bootstrap is to divide the quantities that are to be resampled, which might be residuals, rescaled residuals or $[y_t \ \boldsymbol{I}_{t-1,P}]$ pairs, into blocks of b consecutive observations. There are several suggestions to form the blocks, these blocks may be either overlapping or non-overlapping and their length may be either fixed or variable. There are two main methods in the block bootstrap.

- Moving-block bootstrap

  The best approach is considered as to form the overlapping blocks of fixed length (see e.g. Lahiri, 1999), called moving-block bootstrap. For this method, there are $n - b + 1$ blocks, constructed such as the first block consists of the first b observations i.e. $Z_1 = \{y_t : t = 1, \ldots, b\}$, the second block consists of observations 2 through $b + 1$ i.e. $Z_2 = \{y_t : t = 2, \ldots, b + 1\}$, and the last contains observations $n - b + 1$ through n i.e. $Z_{n-b+1} = \{y_t : t = n - b + 1, \ldots, n\}$. Then a bootstrap

sample is a sample selected from these blocks. The choice of b is somewhat subjective and must be chosen carefully. Ideally, the block size b should not be too small or large because if the block size is too small the dependence will be broken and for too large block size, there will be lack of randomness in the bootstrap samples.

- Block-of-blocks bootstrap
  We define the block as $Z_t = [y_t \ \boldsymbol{I}_{t-1,p}]$ and then block of blocks are constructed as $[Z_1, \ldots, Z_b]$, $[Z_2, \ldots, Z_{b+1}]$, $\ldots$, $[Z_{n-b+1}, \ldots, Z_n]$. Bootstrap samples are resampled from these block-of-blocks. It has the capability to mimic any kind of dynamic model. Moreover, it can handle heteroscedasticity and serial correlation.

For more detailed discussion on bootstrapping time series see e.g. Lahiri (1999), Lahiri (2003).

In this study we use semi-parametric bootstrap methods. Now we provide a simple example to illustrate semi-parametric bootstrap methods. We consider a time series of only five observations say $y_1, \ldots, y_5$, generated by an AR(1) process $y_t = \phi y_{t-1} + \varepsilon_t$. We assume initial value y_0 is available. There are suggestions on choosing the initial data see e.g. Box and Jenkins (1994, Chapter 8). We can describe semi-parametric bootstrap methods as follows:

1. Obtain $\hat{\phi}$, an estimate of ϕ so estimated model as $\hat{y}_t = \hat{\phi} y_{t-1}$. We use this model to obtain fitted values $\hat{y}_1, \ldots, \hat{y}_5$.

2. Obtain residuals $\hat{\varepsilon}_1, \ldots, \hat{\varepsilon}_5$ using $\hat{\varepsilon}_t = y_t - \hat{y}_t$.

3. Resample the residuals $\hat{\varepsilon}_1, \ldots, \hat{\varepsilon}_5$ by drawing random samples of size $n = 5$ with replacement. Suppose the first random sample is $\hat{\varepsilon}_2, \hat{\varepsilon}_5, \hat{\varepsilon}_1, \hat{\varepsilon}_2, \hat{\varepsilon}_3$. So we can say $\varepsilon_1^* = \hat{\varepsilon}_2$, $\varepsilon_2^* = \hat{\varepsilon}_5$, $\varepsilon_3^* = \hat{\varepsilon}_1$, $\varepsilon_4^* = \hat{\varepsilon}_2$ and $\varepsilon_5^* = \hat{\varepsilon}_3$.

4. For wild bootstrap, we use rescaled residuals, say $\hat{\varepsilon}_t^o$ using a transformation say as in (2.3.5). Suppose we have $v_t = (+1, -1, -1, +1, -1)$. Then $\varepsilon_1^o = \hat{\varepsilon}_2$, $\varepsilon_2^o = -\hat{\varepsilon}_5$, $\varepsilon_3^o = -\hat{\varepsilon}_1$, $\varepsilon_4^o = \hat{\varepsilon}_2$ and $\varepsilon_5^o = -\hat{\varepsilon}_3$.

5. Dynamic bootstrap samples can be obtained as $y_t^* = \hat{\phi} y_{t-1}^* + \varepsilon_t^*$, $t = 1, \ldots, 5$,

assuming $y_0^* = y_0$. Note that in dynamic bootstrap sample, observation at time t, y_t^* is obtained recursively from previous bootstrap observation at time $t-1$, y_{t-1}^*.

6. Dynamic wild bootstrap samples can be obtained as $y_t^o = \hat{\phi} y_{t-1}^o + \varepsilon_t^o$, $t = 1, \ldots, 5$, assuming $y_0^o = y_0$. The only difference between dynamic bootstrap sampling and dynamic wild bootstrap sampling that in later we are using the transformed bootstrap residuals as described above.

7. Fixed design wild bootstrap samples can be obtained as $y_t^\dagger = \hat{\phi} y_{t-1} + \varepsilon_t^o$, $t = 1, \ldots, 5$, assuming $y_0^\dagger = y_0$. Note that in fixed design bootstrap, unlike dynamic wild bootstrap, observation at time t, $y_t^\dagger$ is obtained recursively from previous sample observation at time $t-1$, y_{t-1}.

We use these stated bootstrap procedures for obtaining the empirical size and power of goodness of fit tests. The algorithms based on these bootstrap procedures are given in Section 2.4.1. In the next section, we describe the estimation methods used in our study.

2.4 Parameter Estimation

In our numerical results showed in Section 2.5, we estimate the AR(p) model (1.3.3) under various bootstrap designs discussed earlier in Section 2.3.1. The least squares (OLS) estimates (Gonçalves and Kilian, 2004) of $\boldsymbol{\alpha} = (\alpha_1, \ldots, \alpha_p)$ are obtained as below:

$$\hat{\boldsymbol{\alpha}}^* = \left(n^{-1} \sum_{t=1}^{n} \boldsymbol{I}_{t-1,P}^* \boldsymbol{I}_{t-1,P}^{*\mathrm{T}} \right)^{-1} n^{-1} \sum_{t=1}^{n} \boldsymbol{I}_{t-1,P}^* y_t^*,$$

$$\hat{\boldsymbol{\alpha}}^o = \left(n^{-1} \sum_{t=1}^{n} \boldsymbol{I}_{t-1,P}^o \boldsymbol{I}_{t-1,P}^{o\mathrm{T}} \right)^{-1} n^{-1} \sum_{t=1}^{n} \boldsymbol{I}_{t-1,P}^o y_t^o,$$

$$\hat{\boldsymbol{\alpha}}^\diamond = \left(n^{-1} \sum_{t=1}^{n} \boldsymbol{I}_{t-1,P} \boldsymbol{I}_{t-1,P}^{\mathrm{T}} \right)^{-1} n^{-1} \sum_{t=1}^{n} \boldsymbol{I}_{t-1,P} y_t^\diamond,$$

where $\boldsymbol{I}_{t-1,P}$ is the information set defined in (2.2.5) while $\boldsymbol{I}^*_{t-1,P}$ and $\boldsymbol{I}^o_{t-1,P}$ are the information sets for DB and DWB respectively defined in Section 2.3.1. The y^*_t, y^o_t and $y^\diamond_t$ are defined in (2.3.3), (2.3.4) and (2.3.7).

In this chapter, we mainly look at the size and power of the diagnostic tests. We use the bootstrap distributions under the semi-parametric bootstrap designs discussed in Section 2.3.1. We compute the empirical size of the tests from the bootstrap distribution under null with a specified nominal significance level. We also look at empirical power of test against various alternative models. For the sake of convenience, we denote the statistic of interest as T, e.g. Q_m, Q^*_m, $Q^*_m(\hat{\omega})$ and $CvM_{exp,P}$.

2.4.1 Algorithms

In this section, we give the algorithms for the Monte Carlo method used to compute the empirical size and power of the diagnostic tests defined in Section 2.2.1. For each Monte Carlo run, a sample time series $\{y_t\}_{t=1}^n$ is simulated under the model $\mathcal{M}$. For empirical size, $\mathcal{M}$ is the true model while for the computation of power it is the alternative model. In both of the situations, we estimate the true model for the simulated sample time series and T is calculated from the residuals, $\hat{\varepsilon}_t = y_t - \hat{y}_t$, where $\{y_t\}_{t=1}^n$ are the fitted values assuming the initial data are known.

In the following algorithms, we describe the procedure for dynamic bootstrap sampling but the same methods can be applied to other semi-parametric methods i.e. dynamic wild bootstrap and fixed design wild bootstrap methods. Algorithm 1 gives the

Algorithm 1: Bootstrap sampling procedure

Step 1 Generate bootstrap sample y^*_t using true model and resamples of $\hat{\varepsilon}_t$, say ε^*_t.
Step 2 Fit the true model to the bootstrap sample y^*_t and obtain residuals as $\hat{\varepsilon}^*_t = y^*_t - \hat{y}^*_t$, where $\hat{y}^*_t$ is the fitted series.
Step 3 Using the residuals, $\hat{\varepsilon}^*_t$, calculate test-statistic T, say, T^*.
Step 4 Repeat Step 1-3 for each of the B bootstrap samples.

bootstrap procedure used in our numerical study. From this algorithm, we obtain the bootstrap approximation of the distribution of the test. We will use this algorithm to compute empirical size and power in the following algorithms of our simulation study consisting of N Monte Carlo runs.

Algorithm 2: Computation of empirical size.

Step 1 Obtain T^* using Algorithm 1, for each of the B bootstrap samples, reject true model if $T^* \geq T$, otherwise accept it.
Step 2 Determine the proportion of B bootstrap samples, say $\hat{p}_c$, for which the null hypothesis is rejected.
Step 3 Repeat Step 1-2 for each of the N Monte Carlo runs.
Step 4 Empirical size, $\hat{\alpha}$, is determined as the proportion of Monte Carlo runs for which the $\hat{p}_c \leq \alpha$, where α is the level of significance,

$$\hat{\alpha} = \frac{\#(\hat{p}_c \leq \alpha)}{N}.$$

The size of a test is helpful in assessing how reasonable is our assumption of the null distribution. We can compute the size when the sample is simulated under the true model. This is the probability of rejecting the true model when the true model is true model.

The power of a test is the probability of rejecting a false null hypothesis. For empirical power, as mentioned earlier, the sample is generated under the alternative model. Algorithms 2 and 3 state the Monte Carlo procedure we use to determine the empirical size and power of test.

Algorithm 3: Computation of empirical power.

Step 1 Calculate $100(1-\alpha)th$ percentile, say $T^*_{1-\alpha}$, of the bootstrap distribution of T^* obtained using Algorithm 1.
Step 2 Reject true model if $T \geq T^*_{1-\alpha}$ otherwise accept it.
Step 3 Repeat Step 1-2 for each of the N Monte Carlo runs.
Step 4 Empirical power, $1-\hat{\beta}$, is determined as below,

$$1-\hat{\beta} = \frac{\#(T \geq T^*_{1-\alpha})}{N}.$$

In the next section, we look at different examples and compute the empirical size and power of the diagnostic tests. In Section 1.3, we have defined some important linear time series models, now we give definitions of some non-linear models which we will study as alternative models in empirical power study of portmanteau tests. As we will compare our results with Escanciano (2007) so we consider the following nonlinear models.

Exponential Autoregressive model

An exponential autoregressive model, EXPAR(p), is defined as

$$y_t = \sum_{i=1}^{p} \left[\alpha_i + \pi_i \exp(-\gamma y_{t-1}^2)\right] + \varepsilon_t$$

For detailed discussion see e.g. Tong (1990, p.108).

Threshold Autoregressive model

The threshold autoregressive, TAR(p), model is defined as

$$y_t = \begin{cases} \sum_{i=1}^{p} \alpha_i^{(1)} y_{t-i} + \varepsilon_t & \text{if } y_{t-i} < r \\ \sum_{i=1}^{p} \alpha_i^{(2)} y_{t-i} + \varepsilon_t & \text{if } y_{t-i} \geq r \end{cases}$$

where r is called the threshold, below r the AR parameters are $\alpha_i^{(1)}$ and above r these are $\alpha_i^{(2)}$ (see e.g. Chatfield, 2004, p.200). Threshold models were developed and introduced by Tong and Lim (1980) which are basically piecewise linear AR models. For more discussion on bilinear models see also Tang and Mohler (1988) and references therein.

2.5 Results and Discussion

In this section, first we numerically study how well asymptotic results hold for the portmanteau tests. For this we look at the means and variances of the portmanteau tests and compare with their asymptotic counterparts. Secondly, we compute and compare the empirical size of the diagnostic tests under various semi-parametric bootstrap designs discussed in Section 2.3.1. Finally, we compare the empirical power of these tests under same bootstrap designs against a variety of linear and non linear alternative models.

We study the following AR(p) processes,

$$y_t = 1.05 + 1.41y_{t-1} - 0.77y_{t-2} + \varepsilon_t, \tag{2.5.1}$$

$$y_t = 1.05 + 1.41y_{t-1} - 0.77y_{t-2} + 0.2y_{t-3} + \varepsilon_t, \tag{2.5.2}$$

$$y_t = 1.05 + 1.41y_{t-1} - 0.77y_{t-2} + 0.2y_{t-3} - 0.1y_{t-4} + \varepsilon_t. \tag{2.5.3}$$

	p	2			3			4		
	m	5	10	25	5	10	25	5	10	25
Asymp.	Mean	3	8	23	2	7	22	1	6	21
	SD	2.45	4.00	6.78	2.00	3.74	6.63	1.41	3.46	6.48
Q_m	Mean	3.28	7.37	19.46	2.36	6.63	18.51	1.73	5.65	17.17
	SD	2.42	3.71	6.19	1.87	3.33	5.92	1.63	3.12	5.61
Q_m^*	Mean	3.46	8.00	22.96	2.50	7.21	22.02	1.84	6.19	20.51
	SD	2.55	4.03	7.30	1.98	3.64	7.05	1.74	3.42	6.68
$Q_m^*(\hat{\omega})$	Mean	3.46	8.32	23.44	2.55	7.51	22.80	1.88	6.49	21.65
	SD	2.48	4.07	6.52	2.03	3.77	6.66	1.81	3.68	6.76

Table 2.1: Mean and standard deviation of portmanteau tests, based on 1000 Monte Carlo runs of samples of size 100 for AR(p) processes given at the start of Section 2.5.

All the above models are examples of stationary AR processes. The above AR(2) and AR(3) models are also studied by Escanciano (2007) and for comparison purposes we are considering the same models.

2.5.1 Mean and Variance

The asymptotic distribution of Q_m, with AR(p) as the true true model, is χ^2_{m-p} derived by Box and Pierce (1970). The same asymptotic distribution is also proved for Q_m^* by Ljung and Box (1978) and for $Q_m^*(\hat{\omega})$ by Monti (1994). These asymptotic results are a good approximation when n is large relative to m.

It has been reported in the literature that Q_m suffers from location bias (see, e.g., Davies et al., 1977; Ljung and Box, 1978; Kheoh and McLeod, 1992). These papers have looked at various sample sizes ranging from 50 to 500 but one thing is common in these results that they have considered only $m \geq 10$ and we could not find any single reference looking at empirical mean and variance of these portmanteau tests for small choices of m. Moreover, we could not find any literature looking at empirical mean and variance of $Q_m(\hat{\omega})$.

Table 2.1 gives the empirical means of the portmanteau tests using 1000 Monte Carlo runs. The asymptotic mean of these portmanteau tests is $m - p$. It can be noticed that Q_m is overestimating the asymptotic mean for small values of m while the pattern reverses for larger choices of m, where it is underestimating the asymptotic mean. The modified version of Q_m, i.e. Q_m^*, shows a positive location bias greater than

the bias for Q_m for small value of m but performing better for large values of m where its empirical mean is approximating well the asymptotic mean. Again, the direction of the location bias for Q_m^* is not the same for various choices of m. Monti's test, $Q_m^*(\hat{\omega})$, shows a positive location bias for all choices of m. This bias is relatively lower than for Q_m but greater than for Q_m^*.

The asymptotic variance of these portmanteau tests is $2(m-p)$. In general empirical variance of Q_m is lower while for Ljung-Box test, Q_m^*, it is higher than the asymptotic variance. This confirms that Ljung-Box test Q_m^* corrects the location bias but it also increases the variance, see e.g. Kwan and Sim (1996). Monti's test, $Q_m^*(\hat{\omega})$, show some inflated variances but in general the results are seen to be quite accurate.

We conclude that Q_m suffers from bias, it generally underestimates the mean. This underestimation of mean becomes serious for large values of m. The Ljung-Box test, Q_m^*, corrects the bias in location but in some cases we noticed an increased variance e.g. when $m = 25$. The empirical means for Monti's test, $Q_m^*(\hat{\omega})$, though, are not as accurate as for Q_m^*.

2.5.2 Empirical Size

In this section, we study the empirical size of the diagnostic tests using the semi-parametric bootstrap methods; dynamic bootsrap (DB), dynamic wild bootstrap (DWB) and fixed design wild bootstrap (FWB). These bootstrap methods are defined earlier in Section 2.3.1. The Monte Carlo experiment consists of 1000 runs of 200 bootstrap samples. Each bootstrap sample is of length 100. Various versions of these goodness-of-fit tests are considered by looking at various choices of P for $CvM_{exp,P}$ and of m for the portmanteau tests. All the empirical size results are obtained using Algorithm 2. We also test the significance of difference between Monte Carlo size estimate and the nominal size using the Monte Carlo confidence limits. We compute approximate 95% confidence limits as

$$\hat{\alpha} \pm 2\sqrt{\frac{\hat{\alpha}(1-\hat{\alpha})}{N}}, \tag{2.5.4}$$

where $\hat{\alpha}$ is the empirical size estimate and N is the number of Monte Carlo runs.

	$\alpha = 1\%$			$\alpha = 5\%$			$\alpha = 10\%$		
	DB	DWB	FWB	DB	DWB	FWB	DB	DWB	FWB
$CvM_{exp,3}$	1.4	1.5	1.5	6.1	6.6*	6.4	12.2*	12.9*	11.6
$CvM_{exp,5}$	1.5	1.6	1.9	5.5	5.2	5.9	10.9	10.8	10.8
$CvM_{exp,7}$	1.9*	2.4*	1.5	5.9	6.5	4.9	12.0	11.0	11.5
Q_5	1.3	1.3	0.2*	4.9	4.5	1.2*	9.8	10.1	2.9*
Q_{10}	1.0	1.1	0.3*	4.6	4.8	1.8*	9.4	9.1	3.7*
Q_{25}	1.1	0.7*	0.4*	4.7	4.3	1.8*	8.5	9.0	4.7*
Q_5^*	1.3	1.2	0.2*	5.0	4.4	1.2*	9.6	9.9	2.8*
Q_{10}^*	1.0	1.1	0.2*	4.7	4.9	1.7*	9.2	9.1	3.7*
Q_{25}^*	1.1	0.9*	0.4*	4.8	4.5	1.9*	8.9	9.2	4.7*
$Q_5^*(\hat{\omega})$	1.2	1.4	0.3*	4.6	4.8	0.9*	9.2	9.1	2.5*
$Q_{10}^*(\hat{\omega})$	1.5	1.5	0.4*	4.9	5.3	2.0*	8.3	9.2	4.3*
$Q_{25}^*(\hat{\omega})$	1.1	1.0	0.5*	4.9	4.6	2.0*	9.3	9.8	5.4*

Table 2.2: Bootstrap empirical size ($\hat{\alpha}$ in %), based on $N = 1000$ Monte Carlo runs of 200 bootstrap samples of size 100 for AR(2) process, $y_t = 1.05 + 1.41y_{t-1} - 0.77y_{t-2} + \varepsilon_t$. An asterisk (*) indicates that the approximate 95% confidence interval $\hat{\alpha} \pm \sqrt{\hat{\alpha}(1-\hat{\alpha}/N)}$ does not contain the nominal α.

Two main objectives in this size study are (1) to look at how different choices of P and m effect the size of these tests and (2) to compare the various semi-parametric bootstrap methods. Moreover, we also make comparison between $CvM_{exp,P}$ and the portmanteau tests considered in this study. We consider $P = 3, 5$ and 7 as in Escanciano (2007) and choices of as $m = 5, 10$ and 25 as discussed in literature see e.g. Ljung and Box (1978) and Ljung (1986). The ordinary least squares estimates of models are obtained using the rules stated in Section 2.4.

The results given in Tables 2.2-2.3 show the empirical size of the statistics under study for the three bootstrap methods. It is difficult to conclude exclusively which bootstrap method is better in terms of estimating the size of test. In general, the dynamic bootstrap comes out to be the best bootstrap method among the considered choices under the scenarios studied.

Table 2.2 gives the results for empirical size for an AR(2) process given in (2.5.1). For $CvM_{exp,P}$ statistic, the choice $P = 5$ comes out to be the best among the considered choices of P. In the case of AR(2), we are unable to find a clear advantage of one bootstrap method over the other bootstrap methods but DB may be considered performing well in most of the cases. Escanciano (2007) and Escanciano (2006a) has suggested the use of FWB but our results do not show any advantage for fixed design or wild residu-

	$\alpha = 1\%$			$\alpha = 5\%$			$\alpha = 10\%$		
	DB	DWB	FWB	DB	DWB	FWB	DB	DWB	FWB
$CvM_{exp,3}$	0.4*	0.5*	0.6*	3.6	4.0	4.3	8.2	8.0	7.8
$CvM_{exp,5}$	0.9*	0.7*	0.8*	3.7	4.1	4.8	8.7	8.5	8.8
$CvM_{exp,7}$	1.2	0.3*	0.8*	4.4	3.9	3.9	9.3	8.9	8.7
Q_5	0.8*	1.1	0.0*	4.9	5.3	0.2*	9.9	9.5	0.4*
Q_{10}	1.2	1.4	0.0*	4.9	4.9	0.3*	8.8	9.1	1.2*
Q_{25}	1.0	1.0	0.0*	4.2	4.3	0.4*	9.0	9.1	1.6*
Q_5^*	0.8*	1.1	0.0*	4.9	5.3	0.2*	9.9	9.6	0.4*
Q_{10}^*	1.2	1.4	0.0*	5.1	5.1	0.2*	8.6	8.7	1.4*
Q_{25}^*	1.3	1.0	0.0*	4.3	4.3	0.6*	9.0	9.2	2.0*
$Q_5^*(\hat{\omega})$	0.9*	1.0	0.0*	5.1	5.3	0.1*	9.8	9.3	0.6*
$Q_{10}^*(\hat{\omega})$	1.0	1.6	0.1*	5.4	5.6	0.7*	10.1	10.0	1.5*
$Q_{25}^*(\hat{\omega})$	1.5	1.1	0.3*	4.4	4.8	1.2*	10.1	9.8	3.4*

Table 2.3: Bootstrap empirical size ($\hat{\alpha}$ in %), based on $N = 1000$ Monte Carlo runs of 200 bootstrap samples of size 100 for AR(4) process, $y_t = 1.05 + 1.41y_{t-1} - 0.77y_{t-2} + 0.2y_{t-3} - 0.1y_{t-4} + \varepsilon_t$. An asterisk (*) indicates that the approximate 95% confidence interval $\hat{\alpha} \pm \sqrt{\hat{\alpha}(1-\hat{\alpha}/N)}$ does not contain the nominal α.

als therefore a study looking at some more examples is required to further explore this issue.

For the portmanteau tests, Q_m, Q_m^* and $Q_m^*(\hat{\omega})$, we do not find the results clearly advocating for a particular choice of m but $m = 5$ can be considered as the most appropriate choice working for all the portmanteau tests in this study. These results clearly indicate that dynamic bootstrapping is outperforming the fixed design bootstrap. There are clear indications that for the portmanteau tests, FWB underestimates the size. In general, dynamic bootstrap design is the best bootstrap method among the considered methods to approximate the finite sample distribution of the portmanteau tests considered.

The results for the AR(3) process are similar to those for the AR(4) process, so we omit the results for AR(3).

Table 2.3 shows the empirical size for our AR(4) process given in (2.5.3). The results do not lead to any obvious choice of P for $CvM_{exp,P}$ test but $P = 7$ can be considered a better choice as other choices of P lead to overestimation of size. Again we cannot see any clear advantage of using FWB, for which we are underestimating the size in all cases and for all the goodness-of-fit tests. For the portmanteau tests, we reach the same conclusions as for the AR(2) process.

From the above discussion, we can conclude that dynamic bootstrap methods provide a better approximation of the distribution of these goodness-of-fit tests. The fixed design bootstrap method shows poor performance, in general, and fails, specifically, for the portmanteau tests. We can say on the basis of our numerical findings that for $CvM_{exp,P}$ test, a larger value of P is required for a higher order autoregressive process to capture the dependence on the terms with larger lag. For the portmanteau tests, the smaller choice of m comes out to be the best, in general, but one limitation should be kept in mind that the examples we study in this section are of a stationary process with roots well inside the stationarity region. Conclusions may dramatically change for a non-stationary or near-stationary process. We will look at this issue in some more detail in Chapter 3.

With this we conclude our discussion on the size of the goodness-of-fit tests. In the next section, we look at the empirical power of these goodness-of-fit tests.

2.5.3 Empirical Power

In this section, we look at some numerical examples to compare the empirical power of the goodness of fit tests. We present and compare the power against linear and non-linear alternative class of models under a linear true model. Empirical power results are obtained using Algorithm 3 consisting of 1000 Monte Carlo runs of 200 bootstrap samples. Each bootstrap sample is of size $n = 100$.

Linear Alternatives

Mixed ARMA models are the most commonly used models in applications. In this section we compare the power of the tests against several versions of ARMA$(2,2)$ process. In this example, we simulate the series for the alternative model, ARMA$(2,2)$ process, given below:

$$y_t = 1.05 + 1.41y_{t-1} - 0.77y_{t-2} + 0.33k\varepsilon_{t-1} + 0.21k\varepsilon_{t-2} + \varepsilon_t,$$

where $\varepsilon_t \sim N(0,1)$. We fit an AR$(2)$ model to this sample and the power results in the following table of the percentage of Monte Carlo runs we rejected the true model.

	$k=0$			$k=0.5$			$k=1.0$			$k=2.0$		
	DB	DWB	FWB	DB	DWB	FWB	DB	DWB	FWB	DB	DWB	FWB
$CvM_{exp,3}$	6.1	6.2	5.4	7.0	7.3	7.9	10.7	10.1	10.6	20.0	20.5	20.6
$CvM_{exp,5}$	4.8	4.4	4.7	8.3	8.1	8.0	7.5	7.7	8.5	15.1	15.0	16.7
$CvM_{exp,7}$	4.9	4.9	5.3	8.0	7.9	7.8	10.0	9.9	10.6	11.7	13.0	14.9
Q_5	5.2	5.1	2.0	8.4	8.8	3.8	42.7	43.3	26.4	99.2	99.3	97.8
Q_{10}	5.6	5.8	2.3	8.7	8.4	3.9	33.1	34.1	21.2	96.1	96.7	93.1
Q_{25}	5.2	4.9	1.5	10.0	9.8	5.3	29.4	27.9	20.4	90.9	91.7	85.9
Q_5^*	5.3	5.3	2.0	8.2	8.6	3.6	42.0	42.2	25.9	99.1	99.1	97.8
Q_{10}^*	5.9	6.0	2.4	8.4	8.0	3.9	32.3	32.9	20.7	95.6	96.1	92.5
Q_{25}^*	5.7	5.0	2.3	9.6	8.9	5.3	28.0	26.9	18.9	88.9	88.1	82.4
$Q_5^*(\hat{\omega})$	4.8	5.0	1.2	10.3	10.9	5.0	47.7	47.1	30.2	99.5	99.6	98.4
$Q_{10}^*(\hat{\omega})$	5.6	6.0	2.2	80	8.0	3.4	33.6	33.7	21.3	97.9	98.3	95.6
$Q_{25}^*(\hat{\omega})$	4.4	4.7	2.7	9.4	9.9	6.2	26.3	26.2	19.1	91.7	91.1	87.1

Table 2.4: Power (in %) , based on 1000 Monte Carlo runs of 200 bootstrap samples of size 100 for AR(2), against ARMA$(2,2)$, $y_t = 1.05 + 1.41y_{t-1} - 0.77y_{t-2} + 0.33k\varepsilon_{t-1} + 0.21k\varepsilon_{t-2} + \varepsilon_t$.

Importantly, note that we consider various values of k ranging from 0 to 2. It can be noticed that choice $k = 0$ corresponds to our AR(2) process (2.5.1) so we expect very low power in this case, actually as low as the level of significance. On the other hand, as the value of k increases, the MA component in an ARMA process increases in absolute value and this should result in a higher power, reaching a maximum of 100%, for some value of k.

Table 2.4 gives the results for empirical power of the goodness-of-fit tests. It can be very clearly noticed that $CvM_{exp,P}$ has less power while portmanteau tests, Q_m, Q_m^* and $Q_m^*(\hat{\omega})$, have better power against this linear class of alternatives. Our results confirm the results reported in the literature, see e.g Hong and Lee (2003),Escanciano (2006b). Though we have provided the power results for both of dynamic and fixed design bootstrap methods, we discuss the results for dynamic bootstrap method only, as we found and discussed in the previous section that dynamic bootstrapping provides the best approximation to the finite sample distribution especially for the portmanteau tests.

We can see from these results as we increase the value of k, in general, the power for each of the goodness-of-fit tests increases but the increase that for $CvM_{exp,P}$ is not exponential and it attains a maximum power around 20% even for $k = 2$. In contrast to this, the portmanteau tests show an exponential increase in power with an increase in

k and reaches nearly to maximum power of 100%.

Moreover, it can also be seen that as the value of m increases for the portmanteau tests, these tests become generally less powerful. This result is well known and reported in the literature, see e.g. Hong and Lee (2003), Katayama (2009). The same kind of behaviour can be seen for $CvM_{exp,P}$ test and it also shows a decrease in power for larger values of P, this is also reported in Escanciano (2006b).

Non Linear Alternatives

In this section, we look at the empirical power of the goodness-of-fit tests against some popular non-linear alternatives. We consider several versions of non linear EXPAR(2) and TAR(2) models. It has been reported in the literature that the portmanteau tests, we are studying, have poor power against non-linear alternatives especially for TAR models (Escanciano, 2006b). We will use the same choices of P and m as we have used in previous section of power against linear alternatives i.e. $P = 3, 5, 7$ for $CvM_{exp,P}$ test and $m = 5, 10, 25$ for residual autocorrelations based portmanteau tests.

First, we take an EXPAR(2) model, defined as

$$\begin{aligned} y_t &= (0.138 + k(0.316 + 0.982y_{t-1})e^{(-3.89y_{t-1}^2)})y_{t-1} - (0.437 \\ &\quad + k(0.659 + 1.260y_{t-1})e^{(-3.89y_{t-1}^2)})y_{t-2} + 0.2\varepsilon_t, \end{aligned}$$

where $\varepsilon_t \sim N(0, 1)$. The empirical power of diagnostic tests is computed using Algorithm 3.

Table 2.5 reports the empirical power of the diagnostic tests. The situation looks quite opposite to the linear case in the previous section. As we can see, $k = 0$ will correspond to an AR(2) process and with an increase in value of k, the non-linear component in the model will become dominant.

The results in Table 2.5 suggest that residual autocorrelations based portmanteau tests have low power against this class of non-linear alternatives while $CvM_{exp,P}$ is showing good power in this case. As it can be seen that $CvM_{exp,P}$ power increases exponentially with an increase in k and attains the maximum power 100% at $k = 2$ while power for the portmanteau tests can reach around 43%. These results confirm

	$k = 0.2$			$k = 0.8$			$k = 1.0$			$k = 2.0$		
	DB	DWB	FWB	DB	DWB	FWB	DB	DWB	FWB	DB	DWB	FWB
$CvM_{exp,3}$	5.1	5.4	3.1	29.6	30.0	21.2	69.5	70.2	61.0	100	100	100
$CvM_{exp,5}$	6.8	7.3	4.2	26.8	27.9	21.8	67.1	67.2	61.2	100	100	100
$CvM_{exp,7}$	5.8	6.1	3.7	21.8	22.3	18.3	62.2	63.1	56.0	100	100	100
Q_5	3.8	4.3	0.6	8.7	8.5	3.5	11.2	11.7	5.2	42.3	42.2	23.1
Q_{10}	6.5	6.7	1.4	5.7	6.2	2.3	8.6	7.5	4.4	29.5	29.0	18.0
Q_{25}	7.2	6.4	2.9	5.8	4.9	2.7	8.0	7.5	4.3	31.1	30.7	22.3
Q_5^*	4.1	4.4	0.7	8.8	8.7	3.5	11.1	11.9	5.3	43.0	43.2	24.0
Q_{10}^*	6.3	6.6	1.4	5.8	5.9	2.1	8.4	7.5	4.4	29.6	28.7	18.0
Q_{25}^*	7.1	6.5	3.0	6.1	5.2	3.0	8.0	7.7	4.1	29.7	30.0	21.4
$Q_5^*(\hat{\omega})$	4.7	4.8	1.0	8.6	8.3	2.8	11.9	11.9	5.6	40.6	39.4	22.6
$Q_{10}^*(\hat{\omega})$	6.6	5.6	2.1	5.6	6.1	2.6	8.6	7.8	3.3	28.7	27.9	16.9
$Q_{25}^*(\hat{\omega})$	4.9	5.7	2.8	4.9	5.0	3.0	7.7	7.3	4.5	28.5	28.9	22.6

Table 2.5: Power (in %), based on 1000 Monte Carlo runs of 200 bootstrap samples of size 100 for AR(2) against EXPAR(2).

our earlier findings that power decreases for larger values of P and m.

Now, we move to threshold autoregressive model, another class of non-linear models. Theory suggests that TAR models are more challenging than EXPAR models for the diagnostic tests. We consider the following TAR(2) model

$$y_t = \begin{cases} (1.435 - 0.815k) + (1.385 - 0.135k)y_{t-1} \\ \qquad + (-0.835 + 0.405k)y_{t-2} + \varepsilon_t \qquad \text{for } y_{t-2} \leq 3.25 \\ (1.435 + 0.815k) + (1.385 + 0.135k)y_{t-1} \\ \qquad + (-0.835 - 0.405k)y_{t-2} + \varepsilon_t \qquad \text{for } y_{t-2} > 3.25 \end{cases}$$

where $\varepsilon_t \sim N(0,1)$. We can see by controlling the value of k, we can control the amount of nonlinearity in the model. The lower values of k corresponds to low levels of nonlinearity while larger values of k will result in a highly nonlinear model. We use a range of values of k where the model does not blow up. We use the same Algorithm 3 to compute the empirical power.

Table 2.6 reports the empirical power of the diagnostic tests for AR(2) against TAR(2) models. These results generally confirm the known fact that threshold models are challenging for the goodness-of-fit tests. The residual autocorrelations based portmanteau tests show very low power against the TAR model. Though $CvM_{exp,P}$ is showing better power results especially for smaller choice of P, i.e. $P = 3$, it still cannot achieve the

	$k=0.2$			$k=0.8$			$k=1.0$			$k=1.5$		
	DB	DWB	FWB	DB	DWB	FWB	DB	DWB	FWB	DB	DWB	FWB
$CvM_{exp,3}$	7.1	7.2	6.2	43.8	42.7	40.2	49.1	49.0	45.6	48.0	49.1	43.5
$CvM_{exp,5}$	5.2	4.9	4.5	24.7	23.9	22.2	29.9	29.5	26.0	32.4	32.0	29.0
$CvM_{exp,7}$	6.7	6.6	5.9	14.0	13.7	12.1	14.9	14.1	12.5	18.5	18.9	16.7
Q_5	6.0	5.6	1.7	5.8	5.4	1.6	4.8	4.8	1.3	6.0	5.2	1.6
Q_{10}	5.8	5.4	1.8	5.3	6.6	2.5	5.4	5.6	1.5	4.5	4.4	1.5
Q_{25}	6.9	6.5	2.8	6.8	6.5	3.4	6.8	6.8	3.0	5.0	4.5	2.0
Q_5^*	5.9	5.5	1.7	5.7	5.4	1.6	4.9	4.7	1.3	6.0	5.1	1.5
Q_{10}^*	5.4	5.3	2.1	5.4	6.5	2.5	5.2	5.6	1.7	4.4	4.4	1.6
Q_{25}^*	6.6	6.3	2.8	6.7	6.8	3.4	6.3	7.0	7.2	4.8	4.8	1.9
$Q_5^*(\hat{\omega})$	6.2	5.8	1.9	6.1	5.9	2.3	6.0	6.3	1.1	5.8	5.3	1.9
$Q_{10}^*(\hat{\omega})$	5.4	5.4	1.6	5.7	5.9	1.7	5.9	5.3	2.2	5.2	5.9	1.0
$Q_{25}^*(\hat{\omega})$	6.7	5.8	3.8	7.2	6.5	3.6	6.8	7.2	4.1	5.7	5.5	3.2

Table 2.6: Power (in %), based on 1000 Monte Carlo runs of 200 bootstrap samples of size 100 for AR(2), against TAR(2).

same high power as it did against EXPAR(2) models.

Importantly, it should be noted that choice of P and m is very crucial and the power results may improve for some smaller values of P and m. Noting the result reported in Escanciano (2006b), where $CvM_{exp,P}$ has achieved power of 81% against TAR(1) model, we tried smaller values of P, i.e. $P = 1, 2$. For $P = 1$, power for $CvM_{exp,P}$ even further decreases to around 20% while for $P = 2$, it shows an improvement and power rises to 60%.

As dynamic bootstrap has shown good approximation of finite sample distribution of goodness of fit test considered in this study. We will provide a theoretical insight of this finding in Chapter 4.

In the next section, we implement these goodness-of-fit tests to a real dataset.

2.5.4 Real Data Example

We implement the goodness-of-fit tests Q_m, Q_m^*, $Q_m^*(\hat{\omega})$ and $CvM_{exp,P}$, defined earlier in Section 2.2.1, to the Canadian lynx data set. This data set consists of the annual figures of the Canadian lynx trapped in the Mckenzie River district of northwest Canada for the period $1821 - 1934$ inclusive, thus in total 114 observations. Moran (1953) fitted an AR(2) model to the logarithm of lynx data. We also consider the same specification for our study. We report the empirical p-values for the above mentioned goodness-

	DB	DWB	FWB
$CvM_{exp,2}$	0.000*	0.000*	0.000*
$CvM_{exp,4}$	0.000*	0.000*	0.000*
$CvM_{exp,6}$	0.015*	0.015*	0.010*
$CvM_{exp,10}$	0.000*	0.000*	0.015*
Q_3	0.015*	0.005*	0.090
Q_5	0.065	0.125	0.240
Q_{10}	0.035*	0.040*	0.090
Q_{20}	0.030*	0.045*	0.075
Q_3^*	0.015*	0.005*	0.085
Q_5^*	0.065	0.125	0.240
Q_{10}^*	0.025*	0.040*	0.085
Q_{20}^*	0.030*	0.045*	0.075
$Q_3^*(\hat{\omega})$	0.020*	0.025*	0.120
$Q_5^*(\hat{\omega})$	0.115	0.135	0.275
$Q_{10}^*(\hat{\omega})$	0.010*	0.045*	0.070
$Q_{20}^*(\hat{\omega})$	0.000*	0.000*	0.015*

Table 2.7: p-values for Canadian lynx data, based on 200 bootstrap samples under AR(2) as true model. An asterisk (*) indicates that the p-value < 0.05.

of-fit tests in Table 2.7. These p-values are obtained as stated in Step-2 of Algorithm 2.

We found this AR(2) specification is rejected by $CvM_{exp,P}$ at $\alpha = 5\%$ for all considered choices of P and under all bootstrap design. Also for $CvM_{exp,P}$ results are quite similar for each bootstrap designs. Again, we cannot find any difference in the results for $CvM_{exp,P}$ under various bootstrap methods.

The results for portmanteau tests seem to be much dependent on choice of m. For the considered choices of m, the only choice which, for all bootstrap designs, gives insignificant results is $m = 5$. It is also noticed that for the fixed design wild bootstrapping, contrary to dynamic bootstrapping, the results for all portmanteau tests are insignificant at $\alpha = 5\%$ except for $Q_{20}^*(\hat{\omega})$.

The results suggest an AR(2) specification is not satisfactory for this Canadian lynx data set. Several other authors have also reported this fact, see e.g. Moran (1953), Tong (1990).

2.6 Conclusion

We look at finite sample properties of the portmanteau tests and found that Monti's test more closely approximates its finite sample distribution as compared to the Box-Pierce and Ljung-Box tests. The Box-Pierce test suffers from location bias and Ljung-Box test can correct this bias only for large values of m. Moreover, this bias correction in mean can result in an increased variance.

Dynamic bootstrap methods come out superior to fixed design bootstrap methods. Though, for $CvM_{exp,P}$ statistic, fixed design bootstrap method in some situations have performed well but, in general, we cannot see any obvious advantage of it over dynamic bootstrap.

Portmanteau tests are powerful against the linear alternatives while the CvM statistic has shown more power against non-linear alternatives. The choice of m for portmanteau tests and P for $CvM_{exp,P}$ test is important. Our results suggest that approximation of the finite sample distribution and power of these goodness-of-fit tests highly depends on the choice of these parameters, P and m.

CHAPTER 3

Improved Portmanteau Tests

3.1 Introduction

We noticed in Chapter 2 that the dynamic bootstrap correctly estimates the finite sample distribution of the autocorrelations based portmanteau tests. The asymptotic distribution of these tests for an ARMA(p,q) model is considered a χ^2_{m-p-q} distribution. There is a vast literature questioning the appropriateness of χ^2_{m-p-q} as an underlying distribution, see e.g. Davies et al. (1977), Katayama (2009). We have seen in Section 2.5 that, with an increase in m the empirical significance level also increases and the empirical power decreases and the same sort of results have been reported by several other researchers (see e.g. Ljung and Box, 1978; Katayama, 2008).

The estimation of the true distribution underlying the portmanteau tests and the choice of m are questions that need addressing, as these tests are widely used in practice as diagnostic checks of fitted time series models. Katayama (2008) has derived a bias term in Q^*_m for χ^2 approximation and using this bias term he also suggested a bias corrected Ljung-Box test, Q^{**}_m.

We discussed in Section 2.2.1 and also noticed in Section 2.5 the importance of the choice of m. There are some suggestions to choose the value of m, say $m = 15$ or 20, but none of them are very precise, see e.g. Davies et al. (1977) and Katayama (2009). In practice, it is quite difficult to suggest an optimum value of m. Noticing that for these portmanteau tests, an approximation of the asymptotic distribution and sufficient power cannot be achieved for a single value of m, Katayama (2009) suggested a

multiple test. This test is based on a set of values of m ranging from small to medium.

Section 3.2 gives a novel suggestion for a bias correction term in Monti (1994) test on the lines of Katayama (2008). In Section 3.2.1, we suggest a novel algorithm for the efficient computation of the correction term in general ARMA(p,q) models. In this section, we also look at the effect of these bias correction terms on Q_m^* and $Q_m^*(\hat{\omega})$. The theoretical results of this bias correction on Monti (1994) have been given in Chapter 4. In Section 3.3, we give a novel suggestion for the use of pivotal portmanteau test and compared its empirical distribution with other portmanteau tests and the relevant asymptotic χ^2 distribution. Finally, in Section 3.4, we give some numerical results on the multiple test suggested by Katayama (2008).

3.2 Portmanteau Tests Bias Correction

Portmanteau tests are an important part of the diagnostic testing stage of time series model building. The paper by Box and Pierce (1970) is considered as a breakthrough in diagnostic checking of time series models. They derived the normal distribution of residual autocorrelations in ARIMA(p,d,q) models. They showed that if the model is fitted using the true parameter values then the residuals will be uncorrelated random deviates such that $n\sum_{k=1}^{m} r_k^2 \sim \chi_m^2$ and $Var(r_k) = (n-k)/n(n+2) \approx 1/n$. Using these results they showed that the statistic $n(n+2)\sum_{k=1}^{m}(n-k)^{-1}r_k^2$ asymptotically follows χ_m^2 distribution. The following statistic, a further approximation for large m, is suggested to use as a diagnostic test for the residuals of an ARMA(p,q) process

$$Q_m = n\sum_{k=1}^{m}\hat{r}_k^2 \sim \chi_{m-p-q}^2,$$

where $\hat{r}_k$ is the kth order residual autocorrelation defined in (1.2.2). Ljung and Box (1978) mentioned that Q_m suffers from location bias and thus suggested the use of modified statistic

$$Q_m^* = n(n+2)\sum_{k=1}^{m}\frac{\hat{r}_k^2}{n-k}.$$

Many authors, see e.g. McLeod (1978), Katayama (2008), have mentioned the poor approximation of Q_m and Q_m^* especially when m is small and the process is near the stationary boundary. For diagnostic purposes, small values of $m > p+q$ are desired. In this section, we study the size of the improved statistic Q_m^{**} suggested by Katayama (2008). We also give a novel suggestion to correct the bias in Monti (1994) test, $Q_m^*(\hat{\omega})$.

The computation of bias correction term especially for higher order processes is not very simple. In Section 3.2.1, we suggest novel algorithms, to efficiently compute the bias correction terms.

Bias Correction in Ljung-Box test

Katayama (2008) has derived in Box-Pierce test a positive extra random variable given as

$$B_{m,n}^* = \hat{\boldsymbol{r}}^T \boldsymbol{V}\boldsymbol{D}\boldsymbol{V}\hat{\boldsymbol{r}}, \tag{3.2.1}$$

where $\hat{\boldsymbol{r}} = (\hat{r}_1, \ldots, \hat{r}_m)$ is the vector of first m residual autocorrelations,

$$\boldsymbol{V} = \text{diag}\left(\sqrt{\frac{n(n+2)}{n-1}}, \ldots, \sqrt{\frac{n(n+2)}{n-m}}\right),$$

and $\boldsymbol{D} = \boldsymbol{X}(\boldsymbol{X}^T\boldsymbol{X})^{-1}\boldsymbol{X}^T$. Each (i,j)th element of $\boldsymbol{X}$, an $(m \times (p+q))$ matrix, as defined in McLeod (1978) and Katayama (2008), is given by

$$\boldsymbol{X} = \left(-\alpha_{i-j}^* \vdots -\beta_{i-j}^*\right). \tag{3.2.2}$$

Elements of blocks matrices $\left[\alpha_{i-j}^* : i = 1, \ldots, m; j = 1, \ldots, p\right]$ and $\left[\beta_{i-j}^* : i = 1, \ldots, m; j = 1, \ldots, q\right]$ are defined as

$$\alpha^*(L) = \frac{1}{\alpha(L)} = \sum_{i=0}^{\infty} \alpha_i^* L^i \tag{3.2.3}$$

and

$$\beta^*(L) = \frac{1}{\beta(L)} = \sum_{i=0}^{\infty} \beta_i^* L^i. \tag{3.2.4}$$

Moreover, $\alpha_i^* = \beta_i^* = 0$ for $i < 0$. The calculation of α_i^* and β_i^* is quite challenging for higher order processes, we suggest novel Algorithm 4 for their computation.

Katayama suggested a new bias corrected Ljung-Box test given as

$$Q_m^{**} = Q_m^* - B_{m,n}^*.$$

where $B_{m,n}^*$ is as defined in (3.2.1).

Bias Correction in Monti's test

Monti's test, like Ljung-Box test, given by

$$Q_m^*(\hat{\omega}) = n(n+2)\sum_{k=1}^{m}\frac{\hat{\omega}_k^2}{n-k}$$

has a location bias. Moreover, we also noticed that a bias correction term on the lines of Katayama (2008) is also workable for the Monti's test. Working along the lines of Katayama (2008) we suggest an improvement in Monti's test in Section 4.4. Thus the new bias corrected Monti's test is given by

$$Q_m^{**}(\hat{\omega}) = Q_m^*(\hat{\omega}) - B_{m,n}^*(\hat{\omega}), \tag{3.2.5}$$

where

$$B_{m,n}^*(\hat{\omega}) = \hat{\boldsymbol{\omega}}^T \boldsymbol{V}\boldsymbol{D}\boldsymbol{V}\hat{\boldsymbol{\omega}}$$

and $\hat{\boldsymbol{\omega}} = (\hat{\omega}_1, \ldots, \hat{\omega}_m)$ is the vector of first m residual partial autocorrelations defined in (1.2.3).

3.2.1 Algorithms

The main component of the bias correction terms is matrix $\boldsymbol{X}$ defined in (3.2.2) above. The computation of $\boldsymbol{X}$ in $B_{m,n}^*$ and $B_{m,n}^*(\hat{\omega})$ is challenging especially for higher order processes. In this section we suggest an efficient novel algorithm for the computation of the $\boldsymbol{X}$. Given an AR(p) or MA(q) polynomial, the following algorithm is for the

computation of α_i^* and β_i^*, the components of $\boldsymbol{X}$.

Algorithm 4: Coefficients of reciprocal of AR and MA polynomials

Step 1 Calculate the p roots, say $\gamma_1, \ldots, \gamma_p$, of $\alpha(L^{-1}) = 0$.
Step 2 Define matrix $A_{(p\times p)}$ such that

$$\boldsymbol{A}(i,j) = \begin{cases} \gamma_{p-j+1} & i \leq j \\ 0 & i > j \end{cases}.$$

Step 3 Compute the components in the infinite polynomial of $\alpha^*(L)$ using the recursive rule

$$\boldsymbol{S}_{r+1} = \boldsymbol{A}\boldsymbol{S}_r$$

for $r \in \mathbb{Z}^+$, where $\boldsymbol{S}_r$ is a vector of length p, such that $\boldsymbol{S}_0 = \mathbf{1}_{p\times 1}$.
Step 4 Thus α_i^* is the first element of $\boldsymbol{S}_i$.

Algorithm 4 provides an efficient way to compute the coefficients of reciprocal polynomials i.e. $\alpha(L)^{-1}$ and $\beta(L)^{-1}$.

Justification of Algorithm 4

Algorithm 4 is key in obtaining the Katayama's correction term and obtaining non linear least squares estimates of mixed ARMA process. In the following lemma, we will prove the main result used in this algorithm.

Lemma 3.2.1. *If $\gamma_1, \ldots, \gamma_p < 1$ are the roots of $\alpha(L^{-1}) = 0$, then the coefficient of L^j in the expansion of $\alpha(L)^{-1}$ is given by*

$$\delta_j = S_{j1},$$

such that S_{j1} is the first element of $\boldsymbol{S}_j = \boldsymbol{A}\boldsymbol{S}_{j-1}$ for $j = 1, 2, \ldots$, where

$$\boldsymbol{A} = \begin{pmatrix} \gamma_p & \gamma_{p-1} & \gamma_{p-2} & \cdots & \gamma_1 \\ 0 & \gamma_{p-1} & \gamma_{p-2} & \cdots & \gamma_1 \\ 0 & 0 & \gamma_{p-2} & \cdots & \gamma_1 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \ldots & \gamma_1 \end{pmatrix},$$

and $\boldsymbol{S}_0 = \mathbf{1}_p$, where $\mathbf{1}_p$ is the p-vector of ones.

Proof. As $\gamma_1, \ldots, \gamma_p < 1$ are the roots of $\alpha(L^{-1}) = 0$, where $\alpha(L) = 1 - \alpha_1 L - \ldots - \alpha_p L^p$ is the polynomial as defined in Section 1.3, we can write

$$\begin{aligned}
\phi(z)^{-1} &= \left[(1-\gamma_1 z).(1-\gamma_2 z)\ldots..(1-\gamma_p z)\right]^{-1} \\
&= \prod_{i=1}^{p}(1-\gamma_i z)^{-1} \\
&= \sum_{j=0}^{\infty}\delta_j z^j,
\end{aligned}$$

where $\delta_0 = 1$. In general, for an $AR(p)$ process,

$$\delta_j(\gamma_1, \ldots, \gamma_p) = \sum_{\substack{u_i \geq 0 \\ \sum u_i = j}} \gamma_1^{u_1}\gamma_2^{u_2}\ldots\gamma_p^{u_p};$$

where the summation above is over all $(u_1, \ldots, u_p)$ such that the u_i are non-negative integers with $\sum_{i=1}^{p} u_i = j$. Now we derive a recursive rule for the determination of δ_j, which is easy to implement on a computer.

Now considering u_1 as it ranges 0 to j,

$$\begin{aligned}
\delta_j\left(\gamma_1, \ldots, \gamma_p\right) &= \sum_{i=0}^{j}\gamma_1^i\delta_{j-i}\left(\gamma_2, \ldots, \gamma_p\right) \\
&= \delta_j\left(\gamma_2, \ldots, \gamma_p\right) + \sum_{i=1}^{j}\gamma_1^i\delta_{j-i}\left(\gamma_2, \ldots, \gamma_p\right),
\end{aligned}$$

which can be further simplified as

$$\delta_j\left(\gamma_1, \ldots, \gamma_p\right) = \delta_j\left(\gamma_2, \ldots, \gamma_p\right) + \gamma_1\sum_{i=1}^{j}\gamma_1^{i-1}\delta_{j-i}\left(\gamma_2, \ldots, \gamma_p\right).$$

Putting $h = i - 1$,

$$\begin{aligned}
\delta_j\left(\gamma_1, \ldots, \gamma_p\right) &= \delta_j\left(\gamma_2, \ldots, \gamma_p\right) + \gamma_1\sum_{h=0}^{j-1}\gamma_1^h\delta_{j-h-1}\left(\gamma_2, \ldots, \gamma_p\right) \\
&= \delta_j\left(\gamma_2, \ldots, \gamma_p\right) + \gamma_1\delta_{j-1}\left(\gamma_1, \ldots, \gamma_p\right).
\end{aligned}$$

Using the same recursive rule we can write,

$$\delta_j\left(\gamma_1,\ldots,\gamma_p\right) \;=\; \sum_{i=1}^{p}\gamma_i\delta_{j-1}\left(\gamma_i,\ldots,\gamma_p\right).$$

Using the matrix representation, $\delta_j(\gamma_1,\ldots,\gamma_p) = S_{j1}$ where S_{j1} is the first element of the vector $\boldsymbol{S}_j$ defined recursively as

$$\boldsymbol{S}_{j+1} = \boldsymbol{A}\boldsymbol{S}_j, \quad \text{for } j = 0,1,\ldots,$$

where $\boldsymbol{S}_0 = \mathbf{1}_p$. Hence the lemma is proved. □

Note that calculating $\boldsymbol{S}_t$ given $\boldsymbol{S}_{t-1}$ requires $O(p^2)$ floating point operations (flops). Consequently the amount of computation required to calculate $\boldsymbol{S}_1,\ldots,\boldsymbol{S}_T$ is $O(p^2T)$ flops.

There exists a duality between AR and MA processes. Moreover, an ARMA(p,q) process can also be represented by an AR(∞) and MA(∞) process. Some problems like least squares estimation of ARMA(p,q) models require its representation in the form of an AR process. Algorithm 5 suggests a way to achieve this representation.

Algorithm 5: Computation of weights of AR representation of an ARMA(p,q) process

Step 1 Given $\alpha(L)$ and $\beta(L)$, obtain the $\{\beta_i^* : i = 1,\ldots,N\}$ using Algorithm 4. Due to the invertibility condition, the series $\beta(L)^{-1}$ is convergent. So for practical purposes we can consider first N coefficients of this infinite polynomial.
Step 2 Let $\pi(L) = \beta(L)^{-1}\alpha(L)$ be the AR polynomial of infinite order. Now π_r, the coefficient of L^r in the expansion of $\pi(L)$, can be obtained as

$$\pi_r = \beta_r^* - \sum_{i=1}^{p}\alpha_i\beta_{r-i}^*$$

Computation of Katayama's bias correction term requires the components of the reciprocal polynomials of AR and MA polynomials. Algorithm 6 in connection with Algorithm 4 can be used for computation of the correction term.

Estimates of AR(p) processes can be obtained by equating the sample and theoretical autocovariances at lags $0,1,\ldots,p$, but this approach is neither simple nor efficient for MA(q) processes. As showed by Box and Jenkins (1994), the ε_t's are always linear

Algorithm 6: Computation of Katayama (2008) correction term for an ARMA(p,q) process

Step 1 Using Algorithm 4, obtain $\{(\alpha_i^*, \beta_i^*) : i = 1, \ldots, m-1\}$, where m is the maximum value of lag used in diagnostic checks (see Section 2.2.1).
Step 2 Form the matrix $\boldsymbol{X}_{m\times(p+q)}$ given as

$$\boldsymbol{X} = \left(-\alpha_{i-j}^* \vdots -\beta_{i-j}^*\right),$$

and as defined in (3.2.2).
Step 3 Calculate the bias correction term

$$B_{m,n}^* = \hat{\boldsymbol{r}}^T \boldsymbol{V}\boldsymbol{D}\boldsymbol{V}\hat{\boldsymbol{r}},$$

where $\hat{\boldsymbol{r}} = (\hat{r}_1, \ldots, \hat{r}_m)^T$ is the vector of first m residual autocorrelations,
$\boldsymbol{D} = \boldsymbol{X}\left(\boldsymbol{X}^T\boldsymbol{X}\right)^{-1}\boldsymbol{X}^T$ and
$\boldsymbol{V} = \text{diag}\left(\sqrt{n(n+2)/(n-1)}, \sqrt{n(n+2)/(n-2)}, \ldots, \sqrt{n(n+2)/(n-m)}\right)$.

functions of AR parameters α_i's but nonlinear functions of MA parameters β_j's. Methods such as innovations algorithm (see e.g. Brockwell and Davis, 1991) can be used to obtain the preliminary estimates of MA coefficients. These preliminary estimates can be further refined using nonlinear optimization procedures.

We suggest the following algorithm, which represents an ARMA(p,q) process as a finite order AR process and thus non-linear least squares estimates can be obtained.

Algorithm 7: Nonlinear least squares estimation of ARMA(p,q) process

Step 1 Obtain the π weights i.e. the AR representation of the ARMA(p,q) process using Algorithm 5.
Step 2 Obtain the residuals using the rule $\hat{\varepsilon}_t = \sum_{i=0}^{t-1} \hat{\pi}_i y_{t-i}$ for $t = 1, \ldots, n$, where y_t is the observed series and $\hat{\pi}_i$'s are the estimates of π_i's.
Step 3 Define the function $T = \sum_{t=1}^{n} \hat{\varepsilon}_t^2$ for sum of squares of residuals. Minimize T using an algorithm based on numerical derivatives, such as Gauss-Newton algorithm (Bjorck, 1996), to obtain the non-linear least square estimates of coefficients in the polynomials $\alpha(L)$ and $\beta(L)$.

Now we implement estimation using Algorithm 7 for some real data sets. We compare the performance of our suggested algorithms to the other standard methods of estimation. We fit ARMA(p,q) models to real data sets. The non-linear least squares estimates are obtained as explained in Algorithm 7. For comparison purposes, maximum likelihood estimates are also obtained using the *arima* function in R. We notice the importance of starting values as for some of the choices of starting values we end with a

local maximum and not the global one. Originally we randomly select the starting values for the non-linear least squares estimates from $N(0,1)$. For comparison purposes we also use the maximum likelihood estimates as the starting values.

Here we define some notations used:

$\hat{\boldsymbol{\theta}}^*$: Non-linear least squares estimates obtained using Algorithm 7 with starting values randomly selected from $N(0,1)$.

$\hat{\boldsymbol{\theta}}^\dagger$: Maximum likelihood estimates obtained using *arima* function in R with starting values using conditional sum of squares.

$\hat{\boldsymbol{\theta}}^\diamond$: Non-linear least squares estimates obtained using Algorithm 7 when maximum likelihood estimates are used as starting values.

In the following examples we give estimates obtained for above three methods.

Example 1 (Level of lake Huron 1875-1972)

An ARMA$(1,1)$ model is fitted to the mean corrected series.

$$\hat{\boldsymbol{\theta}}^* = (0.73729, 0.35448),$$
$$\hat{\boldsymbol{\theta}}^\dagger = (0.74457, 0.32128),$$
$$\hat{\boldsymbol{\theta}}^\diamond = (0.73729, 0.35448).$$

Estimates using the innovation algorithm are reported in Brockwell and Davis (2002) as $\hat{\boldsymbol{\theta}} = (0.7234, 0.3596)$. As we can see, $\hat{\boldsymbol{\theta}}^*$ are similar to $\hat{\boldsymbol{\theta}}^\diamond$.

Example 2 (Annual minimum level of Nile river 622-871)

An ARMA(5,2) model is fitted and we obtain the following estimates

$$\hat{\boldsymbol{\theta}}^* = (-0.30052, -0.03304, 0.64926, 0.05235, 0.23546, 0.67679, 0.30676, -0.44153),$$
$$\hat{\boldsymbol{\theta}}^\dagger = (-0.32446, -0.06114, 0.63305, 0.06926, 0.24816, 0.70305, 0.35138, -0.41786),$$
$$\hat{\boldsymbol{\theta}}^\diamond = (-0.30052, -0.03304, 0.64926, 0.05235, 0.23546, 0.67679, 0.30676, -0.44153).$$

Estimates using *Autofit* option in *ITSM* are reported in Brockwell and Davis (2002) as $\hat{\boldsymbol{\theta}} = (-0.323, -0.060, 0.633, 0.069, 0.248, 0.702, 0.350, -0.419)$.

Example 3 (Australian monthly electricity production Jan 1956 - Aug 1995)
An ARMA(2,3) model is fitted.

$$\hat{\boldsymbol{\theta}}^{*} = (-0.62629, -0.93884, -0.02729, 0.47441, -0.63674),$$
$$\hat{\boldsymbol{\theta}}^{\dagger} = (-0.26344, -0.93566, -0.45651, 0.81668, -0.62687),$$
$$\hat{\boldsymbol{\theta}}^{\diamond} = (-0.26376, -0.93145, -0.45327, 0.80878, -0.61842).$$

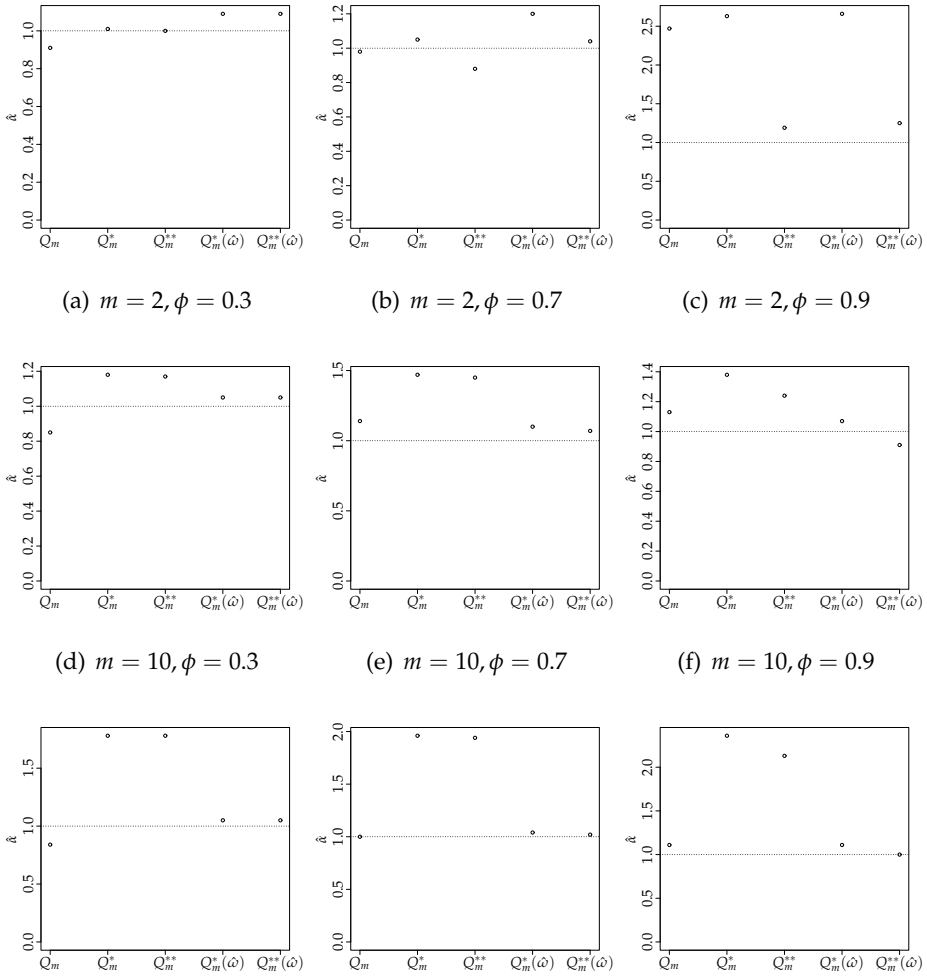
These results show that the idea of estimating ARMA(p, q) model by transforming it into an AR process works and the results are comparable with those obtained by the innovations algorithm and maximum likelihood estimates.

3.2.2 Numerical Results

In this section, we look at the effect of bias correction terms considering various numerical examples. We give the Monte Carlo estimates of size for the Box-Pierce test, Ljung-Box test and its bias corrected version. We also look at Monti's test and our novel suggestion (3.2.5) to correct the bias in it. We obtain the size estimates from asymptotic and dynamic bootstrap distributions.

Consider an AR(1) process, $y_t = \phi y_{t-1} + \varepsilon_t$ such that $|\phi| < 1$. We simulate a sample time series of 200 observations for an AR(1) process. In order to look at different levels of stationarity, we consider three different values of ϕ, i.e $\phi = 0.3, 0.7$ and 0.9. Remember that as $|\phi| < 1$ is the stationarity condition for an AR(1) process and as $|\phi|$ approaches 1 we move near to the stationary boundary. We also assume $\varepsilon_t \sim N(0, 1)$. The choices of $m = 2, 3, 5, 10$ and 25 are considered but the results are given only for $m = 2, 10$ and 25 as the results for $m = 3$ and $m = 5$ are not very different from as for $m = 2$.

Figure 3.1 gives the empirical size of five tests viz. Box-Pierce, Ljung-Box, Monti's test, bias corrected Ljung-Box test and bias corrected Monti's test. Bias correction terms are computed using Algorithm 6. It can be seen that when the process is well inside the stationary boundary, i.e. for $\phi = 0.3$, as m increases, bias in Q_m^* increases and so too for its bias corrected version, Q_m^{**}. We can see that as m increases, the role of the bias



(a) $m = 2, \phi = 0.3$ (b) $m = 2, \phi = 0.7$ (c) $m = 2, \phi = 0.9$

(d) $m = 10, \phi = 0.3$ (e) $m = 10, \phi = 0.7$ (f) $m = 10, \phi = 0.9$

(g) $m = 25, \phi = 0.3$ (h) $m = 25, \phi = 0.7$ (i) $m = 25, \phi = 0.9$

Figure 3.1: Empirical size (Nominal level 1%) for AR(1) process, $y_t = \phi y_{t-1} + \varepsilon_t$, based on 10,000 replications of sample size $n = 200$ for the test statistics using asymptotic distribution χ^2_{m-1}.

correction term becomes minimal. The reason that Katayama's bias correction does not work in this situation is as mentioned before that it is suggested for small values of m and for a near-stationary process.

Importantly, note that Monti's test, $Q_m^*(\hat{\omega})$ does not suffer from bias in this case of a process well inside the stationarity region and our suggested bias corrected Monti's test, $Q_m^{**}(\hat{\omega})$ is working equally well in this situation.

Now, as we move away to the stationarity boundary, i.e. $\phi = 0.9$, for small value of m, i.e. $m = 2$, both Q_m^* and $Q_m^*(\hat{\omega})$ suffer from the bias and this is the only case when Katayama's type correction comes into play and we can see that bias corrected versions

are showing the size close to the nominal level.

Bias in Q_m^* is greater than that of $Q_m^*(\hat{\omega})$ for large values of m. Moreover, it can be seen that Katayma's bias correction term does not work in this scenario and no improvement can be seen for $m = 25$. However for large values of m bias in Monti's test is relatively smaller than in the Ljung-Box test.

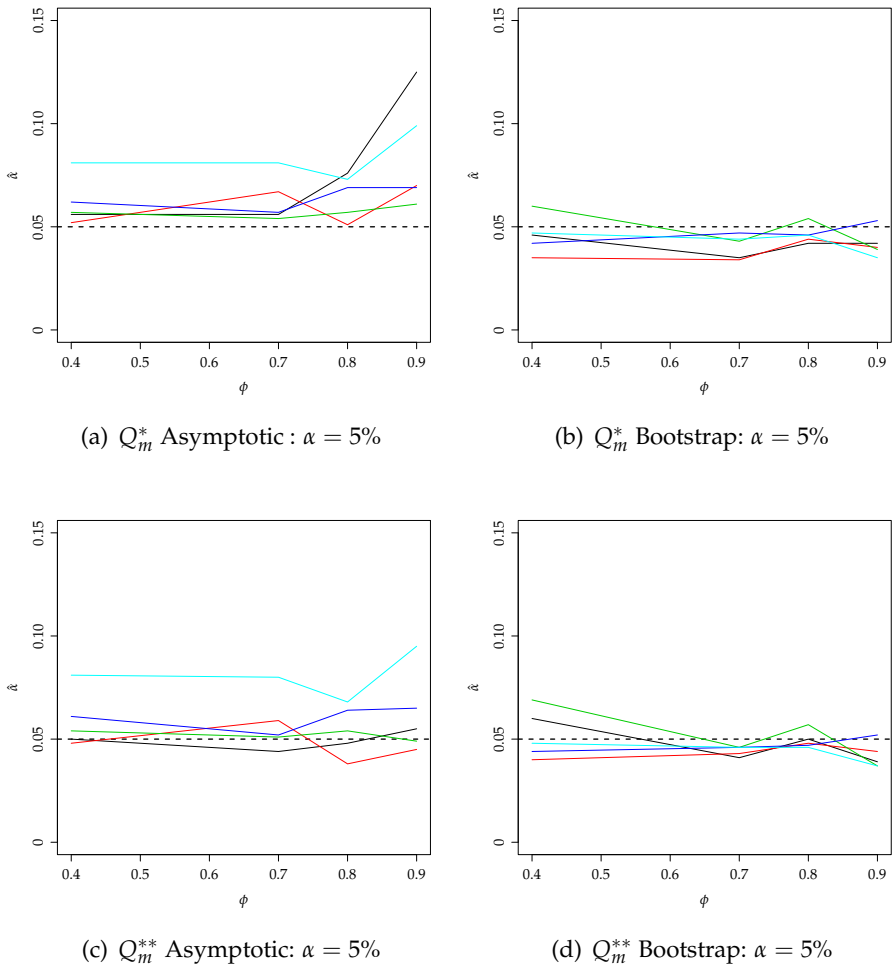
In Chapter 2, we have seen that dynamic bootstrap methods provide a good approximation of the asymptotic distribution. Now we look at the size of the bootstrap distributions of these tests and compare them with the results obtained for the asymptotic distribution. The results for the asymptotic distribution are based on $10,000$ Monte Carlo runs. The bootstrap estimates are obtained from 1000 Monte Carlo runs of 200 bootstrap samples. Sample size is 100 in these examples.

Figure 3.2 gives the empirical size for the Ljung-Box test and its bias corrected version at nominal level $\alpha = 5\%$. The results confirm the earlier findings but some more interesting facts can be noticed.

In the case of the asymptotic distribution, it can be seen that for smaller choices of m bias is introduced when the process is near to the stationarity boundary and Katayama's suggestion is correcting this bias but for larger values of m e.g. for $m = 25$, the Ljung-Box test shows a consistent amount of bias which Katayama's suggestion is unable to correct. Note that for large values of m bias occurs even for the process which is well inside the stationarity region.

For asymptotic distributions, Ljung-Box test show bias in estimating size for very small and large values of m, while this bias is low for moderate choices of m. In contrast, Monti's test show larger bias for smaller values of m and bias reduces for larger values of m.

Now in the case of dynamic bootstrap distribution, we can see that the Ljung-Box test and its bias corrected version do not show any pattern over the values of ϕ. Though it shows a general tendency of underestimating the size, the amount of bias is relatively negligible especially when the process is near the stationary boundary. Moreover, it shows the bootstrap approximation to the asymptotic distribution is robust to the choice of m.



(a) Q_m^* Asymptotic : $\alpha = 5\%$

(b) Q_m^* Bootstrap: $\alpha = 5\%$

(c) Q_m^{**} Asymptotic: $\alpha = 5\%$

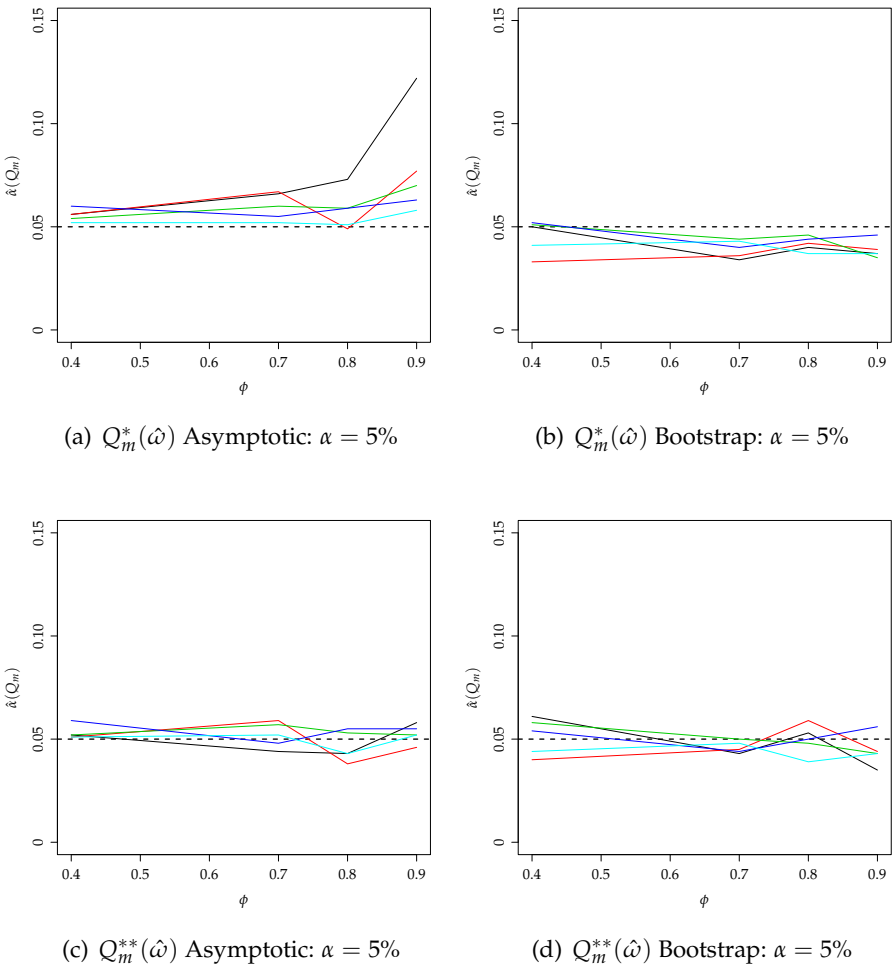
(d) Q_m^{**} Bootstrap: $\alpha = 5\%$

Figure 3.2: Ljung-Box test: Asymptotic empirical size for AR(1) process, $y_t = \phi y_{t-1} + \varepsilon_t$, based on 10000 runs of sample size 100 and Bootstrap empirical size, based on 1000 Monte Carlo runs of 200 bootstrap samples of size 100. Key: $m = 2$ $m = 3$; $m = 5$; $m = 10$; $m = 25$

Figure 3.3 gives the plots of empirical size for the asymptotic distributions and bootstrap distributions of Monti's test and its bias corrected version.

For the asymptotic distribution, like the Ljung-Box test, Monti's test, for small values of m, also shows a large amount of bias for near stationary process. We can see that our suggested bias correction term in Monti's test works successfully. Again it can be concluded that Monti's test has a better approximation of the asymptotic distribution for large values of m and it can be seen that bias for Monti's test reduces with an increase in m.

The bootstrap distribution again shows a similar bias correction for Monti's test as for the Ljung-Box test. The numerical results for the bootstrap distribution lead



(a) $Q_m^*(\hat{\omega})$ Asymptotic: $\alpha = 5\%$

(b) $Q_m^*(\hat{\omega})$ Bootstrap: $\alpha = 5\%$

(c) $Q_m^{**}(\hat{\omega})$ Asymptotic: $\alpha = 5\%$

(d) $Q_m^{**}(\hat{\omega})$ Bootstrap: $\alpha = 5\%$

Figure 3.3: Monti's test: Asymptotic empirical size for AR(1) process, $y_t = \phi y_{t-1} + \varepsilon_t$,, based on 10000 runs of sample size 100 and Bootstrap empirical size, based on 1000 Monte Carlo runs of 200 bootstrap samples of size 100. Key: $m = 2$ $m = 3$; $m = 5$; $m = 10$; $m = 25$

us to look into the theory to see if the dynamic bootstrap automatically does the bias correction, these results are given in Chapter 4.

3.3 Novel Pivotal Portmanteau Test

The choice of an optimal value of m is a critical issue. For small values of m, we have bias in approximating the asymptotic distribution of the portmanteau tests while for large values of m the empirical significance level increases and the empirical power decreases. As showed by McLeod (1978), the large sample distribution of autocorrelation

$\hat{\boldsymbol{r}} = (\hat{r}_1, \ldots, \hat{r}_m)$ is normal with mean $\mathbf{0}$ and covariance matrix

$$Var(\hat{\boldsymbol{r}}) = \frac{1}{n}\left(\boldsymbol{I} - \boldsymbol{C}\right),$$

where $\boldsymbol{I}$ is the identity matrix of appropriate order and $\boldsymbol{C} = \boldsymbol{X}\mathbb{J}^{-1}\boldsymbol{X}^T$, $\boldsymbol{X}$ is as defined in (3.2.2). The Fisher information matrix $\mathbb{J}$ for an ARMA(p,q) model defined in (1.3.5) is given by

$$\mathbb{J} = \sum_{i=1}^{\infty} \boldsymbol{d}_i \boldsymbol{d}_i^T, \tag{3.3.1}$$

where $\boldsymbol{d}_i$ is the ith row of $\boldsymbol{X}$, so that

$$\boldsymbol{X}\boldsymbol{X}^T = \sum_{i=1}^{m} \boldsymbol{d}_i \boldsymbol{d}_i^T.$$

In deriving the bias in the asymptotic distribution of $\hat{\boldsymbol{r}}$, Katayama (2008) has used the first order approximation of $\boldsymbol{C} = \boldsymbol{X}\mathbb{J}^{-1}\boldsymbol{X}^T$ as

$$\boldsymbol{D} = \boldsymbol{X}(\boldsymbol{X}\boldsymbol{X}^T)^{-1}\boldsymbol{X}^T.$$

As m increases

$$\boldsymbol{X}\boldsymbol{X}^T = \mathbb{J} + o(1).$$

Now the problem is that for diagnostic purposes we need small values of m while approximation of the Fisher information matrix requires some large values of m. For these portmanteau tests the information matrix $\mathbb{J}$ is approximated using the same value of m as used for diagnostic purposes. This results in a bias in approximating χ^2_{m-p-q} as its correct asymptotic distribution.

We make a novel suggestion here for computation of $\mathbb{J}$, which corrects the bias. We suggest calculating $\boldsymbol{X}_0$, an approximation of $\boldsymbol{X}$, for some large value of m, say $m_0 \to \infty$, so that $\boldsymbol{X}_0\boldsymbol{X}_0^T = \sum_{i=1}^{m_0} \boldsymbol{d}_i \boldsymbol{d}_i^T$, is as close to $\mathbb{J} = \sum_{i=1}^{\infty} \boldsymbol{d}_i \boldsymbol{d}_i^T$ while for computation of the portmanteau test we still use the small value of m. Accurate calculation of $\mathbb{J}$ is feasible due to the efficiency of the algorithms in Section 3.2. The use of two different values of

m leads to bias correction without sacrificing the power. So our new statistic is

$$Q_m^{\dagger} = n\hat{\boldsymbol{r}}^T \left(\boldsymbol{I} - \boldsymbol{C}_0\right)^{-1} \hat{\boldsymbol{r}}, \tag{3.3.2}$$

where

$$\boldsymbol{C}_0 = \hat{\boldsymbol{X}} \left(\hat{\boldsymbol{X}}_0^T \hat{\boldsymbol{X}}_0\right)^{-1} \hat{\boldsymbol{X}}^T,$$

where $\hat{\boldsymbol{X}}$ is obtained using the same value of m as used in the portmanteau test while $\hat{\boldsymbol{X}}_0$ is calculated for the considered larger choice of m, i.e. $m_0 >> m$.

In the following examples we will show how our new proposed statistic works especially in challenging scenarios e.g. small m while the process is near the stationarity boundary.

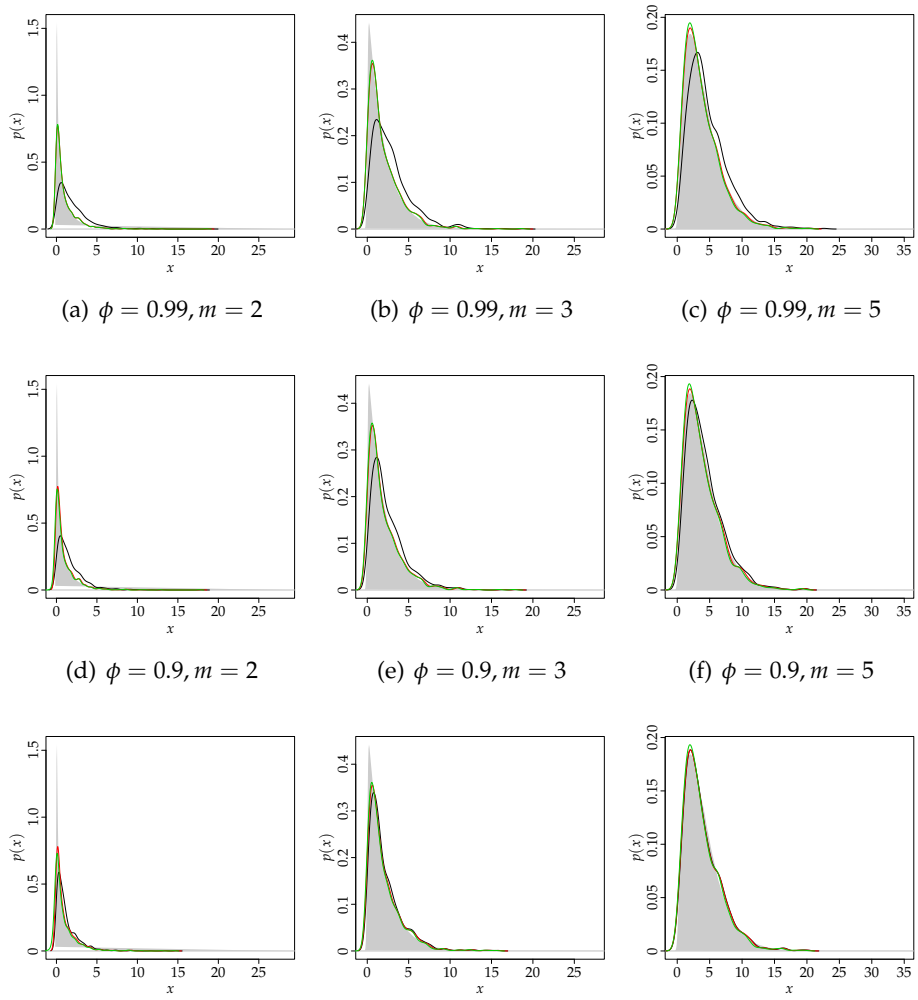
3.3.1 Examples

Consider an AR(1) process

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

where $|\phi| < 1$ and $\varepsilon_t \sim N(0,1)$. To study the stationary and near stationary processes, we simulate a time series of $n = 200$ observations for $\phi = 0.7, 0.9$, and 0.99. As mentioned by Ljung (1986), Katayama (2008), to make the situation challenging we consider small values of m as $m = 2, 3$, and 5. While for the computation of the novel pivotal statistic $Q_m^{\dagger}$ we consider $m_0 = 150$, which is quite large relative to the choices of m in these examples.

Here we look at three statistics, the Ljung-Box statistic Q_m^*, Katayama's corrected Ljung-Box statistic Q_m^{**} and our new suggested statistic $Q_m^{\dagger}$. Figure 3.4 shows a shaded curve for the relevant asymptotic χ^2 distribution while coloured lines show the density curves for the three tests based on their empirical distributions. All the calculations are for 1000 Monte Carlo runs. In all the situations, the novel portmanteau test $Q_m^{\dagger}$ performs as well as Katayama's corrected Q_m^{**}. It can be noticed that $\phi = 0.99$ and $m = 2$ is the most challenging situation, where the distribution of Q_m^* is not approximating



(a) $\phi = 0.99, m = 2$ (b) $\phi = 0.99, m = 3$ (c) $\phi = 0.99, m = 5$

(d) $\phi = 0.9, m = 2$ (e) $\phi = 0.9, m = 3$ (f) $\phi = 0.9, m = 5$

(g) $\phi = 0.7, m = 2$ (h) $\phi = 0.7, m = 3$ (i) $\phi = 0.7, m = 5$

Figure 3.4: Density plots based on 1000 Monte Carlo runs of 200 observation of AR(1) process $y_t = \phi y_{t-1} + \varepsilon_t$. Shaded curve represents the density plot of relevant χ^2_{m-1} distribution. Key: ■ Ljung-Box Q^*_m, ■ Katayama Q^{**}_m, ■ Novel pivotal portmanteau test $Q^{\dagger}_m$ based on $m_0 = 150$.

the asymptotic distribution χ^2_1, while $Q^{\dagger}_m$ and Q^{**}_m show a good approximation to the asymptotic distribution. The only concern is the bias in estimating the peak of the distribution.

The novel pivotal portmanteau test and Katayama's bias corrected test show almost similar good performance in approximating the tails of the asymptotic χ^2 distribution. These results again confirm the need of bias correction when m is small and the process is near stationary. It can be noticed that as m increases, e.g. $m = 5$, the uncorrected statistics are also getting a good approximation of asymptotic distribution especially when the process is not very close to stationarity boundary.

α	1%				5%				10%			
ϕ	0.99	0.9	0.7	0.3	0.99	0.9	0.7	0.3	0.99	0.9	0.7	0.3
Q_2^*	1.7	1.3	1.7	2.3	5.6	5.0	6.0	7.4	9.9	10.5	10.7	11.2
Q_2^{**}	1.5	1.0	1.9	2.4	5.1	4.4	5.8	7.5	10.2	9.3	10.6	11.3
$Q_2^\dagger$	1.6	1.0	2.0	2.4	5.2	4.3	6.0	7.6	10.4	9.4	10.8	11.9
Q_3^*	2.0	2.5	2.2	2.3	5.4	6.0	6.2	5.3	11.8	11.0	10.4	11.0
Q_3^{**}	1.8	1.8	2.0	2.2	5.4	5.6	6.4	5.3	10.5	10.2	10.3	10.9
$Q_3^\dagger$	1.8	1.8	2.0	2.1	5.4	5.6	6.3	5.4	10.6	10.5	10.3	10.9
Q_4^*	1.2	1.6	1.5	0.9	5.4	4.8	5.5	6.3	11.2	9.2	9.8	9.6
Q_4^{**}	1.1	1.7	1.6	0.9	5.7	5.0	5.3	6.3	9.9	8.9	10.0	9.7
$Q_4^\dagger$	1.1	1.7	1.6	0.9	5.6	5.1	5.5	6.4	10.0	8.8	9.9	9.9
Q_5^*	1.1	1.2	1.9	1.7	5.3	5.2	5.8	5.7	11.0	10.4	11.1	11.1
Q_5^{**}	1.5	1.3	1.9	1.7	4.0	4.9	5.8	5.7	9.9	10.4	11.0	11.2
$Q_5^\dagger$	1.5	1.3	1.8	1.8	4.0	4.6	5.8	5.9	10.1	10.4	10.9	11.1

Table 3.1: Bootstrap empirical size (in %) of the Ljung-Box Q_m^*, Katayama Q_m^{**}, New test $Q_m^\dagger$, based on 1000 Monte Carlo runs of 200 bootstrap samples of size 200 for AR(1) process $y_t = \phi y_{t-1} + \varepsilon_t$.

Now we look at the size of these tests using the bootstrap distribution. We use the dynamic bootstrapping defined in Section 2.3.1. Results for the bootstrap distributions in Table 3.1 confirm our earlier results shown in Figure 3.2. Bootstrap size results do not show any specific pattern for different choices of ϕ and m. Interestingly, contrary to asymptotic results, for bootstrap we notice some positive bias for stationary processes and small value of m e.g. $\phi = 0.3$ and $m = 2$.

In the next section, we will look at multiple portmanteau test suggested by Katayama (2009).

3.4 Multiple Portmanteau Test

We have noticed and discussed in earlier sections that the asymptotic distribution and performance of the Ljung and Box (1978) portmanteau test is highly dependent on the choice of m. For diagnostic checking, and to use the chi-square as an asymptotic distribution, m should be moderately large (see e.g. McLeod and Li, 1983). On the other hand, unnecessarily large choices of m lead to unstable test size and decrease the power of test, see Sections 2.5.2 and 2.5.3. Also see e.g. Ljung (1986), Katayama (2009).

Katayama (2009) suggested another way to deal with the problem of the choice of m. He suggested the use of a multiple portmanteau test which can be considered a

collection of standard portmanteau tests with different degrees of freedom. He also suggested an algorithm for numerical computation of the joint distribution of this test. As estimation of the multiple portmanteau test is quite complicated, Katayama (2009) has made suggestion to estimate probability of type-I error, α, for some specified probability of type-II error, β. In the next section, we suggest and show that bootstrap method can be used to approximate the asymptotic distribution of multiple portmanteau test.

3.4.1 Examples

In this example we simulate $n = 200$ observations for $AR(1)$ process $y_t = 1.05 + \phi y_{t-1} + \varepsilon_t$. We consider two choices of $AR(1)$ parameter viz. $\phi = 0.3$ and 0.9. As the results do not differ much for these choices of ϕ, so we give here only the results for $\phi = 0.9$. This simulation study consists of 1000 Monte carlo runs of 500 bootstrap samples.

	DF(2,6,12)			DF(4,8,12)			DF(12,18)			DF(18,24)		
		β			β			β			β	
	1%	5%	10%	1%	5%	10%	1%	5%	10%	1%	5%	10%
Monte Carlo estimate												
$\hat{\alpha}$	2.6	9.6	17.4	2.9	10.7	18.7	2	7.4	14.2	2.5	7.6	14.1
Hybrid bootstrap average												
$\hat{\alpha}$	2.6	10.7	19.6	2.6	10.1	18.3	2.4	8.4	15.0	2.6	8.6	14.9
Katayama (2009) estimate												
$\hat{\alpha}$	2.3	10.3	19.4	2.1	9.6	18.0	1.5	7.2	13.9	1.5	6.9	13.4
MSE												
Bootstrap	0.51	3.25	8.09	0.58	2.27	3.50	0.65	2.79	3.15	0.53	2.64	3.28
Katayama	0.12	0.49	3.88	0.63	1.28	0.49	0.22	0.03	0.09	1.06	0.46	0.56

Table 3.2: Empirical size of multiple test for specified β. A time series of length $n = 200$ is simulated for $AR(1)$ process $y_t = 1.05 + 0.9y_{t-1} + \varepsilon_t$, where $\varepsilon_t \sim N(0,1)$. Estimates obtained from 1000 Monte Carlo runs of 500 bootstrap samples of size 200 selected using the dynamic bootstrapping by resampling residuals.

We compute the joint significance level α for specified marginal significance level β. We also compare the bootstrap estimation and Katayama (2009) estimation with the Monte Carlo estimate. A hybrid bootstrap approach is used for the estimation of the significance level. This approach can be implemented using Algorithm 2 with a modification of replacing the T by the critical value of the relevant asymptotic distribution. The only condition to use Katayama (2008) estimates is the condition of even degrees

of freedom. The Monte Carlo method and bootstrap method have the advantage that they don't require this condition.

Table 3.2 gives the empirical size estimates obtained using the hybrid bootstrap, Monte Carlo and Katayama (2009) suggested methods. Our results confirm the results reported in Katayama (2008). The estimates obtained by Katayama's method and the mean of bootstrap estimates are approximately equal when $\beta = 1\%$. While for larger choices of β, in general, Katayama's estimates are closer to the Monte Carlo estimates. Moreover, we also observe that Katayama's suggestion underestimates the size except when a very small value of m is used e.g. $m = 2$ while in contrast to this, the bootstrap estimates average is greater than the Monte Carlo estimates.

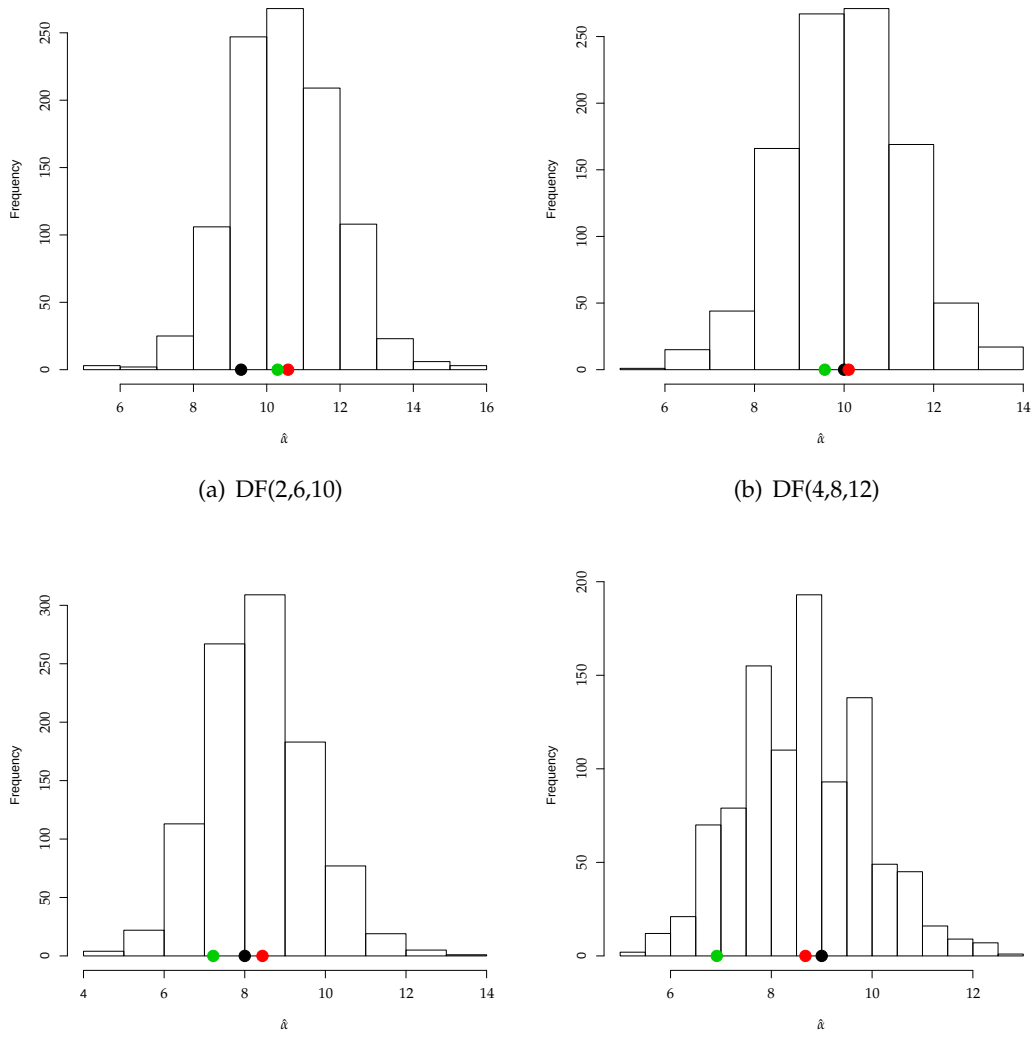
Now we give some plots to look at the empirical distribution of hybrid bootstrap estimates of α.

Figure 3.5 shows plots of the bootstrap distribution of estimated size of the multiple test for specified $\beta = 5\%$. The results do not differ much for other choices of β. We can see bootstrap estimates show a general tendency of overestimating the size as Monte Carlo estimates, in all the cases, lie at the left tail of the bootstrap distribution. It can also be noticed that Katayama's estimates are underestimating the size as they are generally less than the Monte Carlo estimate except for DF$= (2, 6, 10)$. This result may be due to the small value of m i.e. $m = 2$ in this case.

We have also studied some other choices of AR parameter and an important point to note is that estimates are not very sensitive to the value of process parameters.

3.5 Conclusion

Bias in portmanteau tests and choice of m are two important issues which are dealt within this chapter. As we have seen that Katayama's suggested bias correction for the Ljung-Box test works for a near stationary process with small values of m. This is the case where Monti's test also shows larger amount of bias otherwise our results suggest that Monti's test has generally lower bias than the Ljung-Box test. Our novel suggestion, along the lines of Katayama (2008), show an improvement in Monti's test



(a) DF(2,6,10)

(b) DF(4,8,12)

(c) DF(12,18)

(d) DF(18,24)

Figure 3.5: Estimated significance level for $\beta = 5\%$. A time series of length $n = 200$ has been simulated for $AR(1)$ process $X_t = 1.05 + 0.9X_{t-1} + \varepsilon_t$, where $\varepsilon_t \sim N(0,1)$. Estimates obtained from 1000 Monte Carlo runs of 500 bootstrap samples of size 200. Key: • Mean of hybrid bootstrap estimate, • Monte Carlo estimate, • Katayama (2009) estimate

and corrects the bias. Moreover, we gave a novel result that dynamic bootstrapping does an automatic bias correction in these portmanteau tests.

The computation of the bias correction term, especially for higher order processes, is not very simple. We suggested a novel algorithm able to efficiently compute the bias correction term.

We also made a novel suggestion to use pivotal portmanteau test using two different values of m, a relatively large value of m for the estimation of the information

matrix to remove the bias. This can be efficiently computed our suggested novel algorithms in Section 3.2.1 and then using a small value of m, for diagnostic test purposes. Our numerical results showed that this novel suggestion of using pivotal portmanteau test corrects the bias as good as Katayama's suggestion does.

Finally, we studied Katayama's multiple test. It is hard to derive a joint asymptotic distribution of this test and Katayama (2009) suggested an iterative method to obtain the estimates of significance level under some conditions. We, in our examples, suggest that a hybrid bootstrap method is easy to implement and performs, in some cases, better than Katayama's method.

CHAPTER 4

Theoretical Results

4.1 Introduction

In the previous chapter, numerical results have shown that the dynamic bootstrap gives a good approximation of the distribution of portmanteau tests. Now, we give theoretical results to support these numerical findings. We prove a central limit theorem for the asymptotic distribution of the dynamic bootstrap. An important point is that we have proved these results for AR(p) processes without making any assumptions specific to AR structure. Therefore, the results hold true for a wider class of stationary models and can be proved at the cost of greater technical complexity.

In this chapter, the main goal is to derive the asymptotic distribution of the least squares estimator of the autoregressive coefficients under the dynamic bootstrap. However, we also derive the bias term in Monti (1994) test. McLeod (1978) has given the asymptotic distribution of the residual autocorrelations under the true linear model. Mann and Wald (1943) have proved the asymptotic normality of the maximum likelihood estimates of a linear difference equation which is also true for the least squares estimates but the use of martingale limit theory helps in providing a simpler proof of this result which can also be easily justified for dynamic bootstrap method. The rest of this chapter is as follows.

Section 4.2 gives results on the asymptotic distribution of the least squares estimator of the AR coefficients. Theorem 4.2.1 gives the asymptotic distribution of the least squares estimator in the AR(p) setting, while Theorem 4.2.2 proves the corresponding

result for the dynamic bootstrap least square estimates and shows that the limiting distributions of least squares estimates and dynamic bootstrap estimates are same with probability 1.

In the process of proving these theorems we prove a number of technical lemmas which may be of independent interest. Lemma 4.2.3 gives a bound on the coefficients of an MA representation of a finite order AR(p) process. In Lemma 4.2.4 we prove an upper bound for the kth order cumulant of $\{y_t\}_{t\geq 1}$. Lemma 4.2.5, proved under the conditions of Theorem 4.2.1, shows that the variance of components of the sample covariance matrix of y_t decrease at rate n^{-1}, which leads to the convergence result proved in Corollary 4.2.6. Lemma 4.2.5 is used to establish that the conditions for the martingale central limit theorem due to Brown (1971) stated in Theorem 4.2.7 hold in our case. Section 4.3 explains the challenges in extending the theoretical development to include higher-order properties of the dynamic bootstrap in the time series context.

In the previous chapter, we have also looked at some numerical results for bias correction in Monti's test along the lines of Katayama (2008). We derive the bias term in Monti's test in Section 4.4 and suggest an improved test to correct this bias.

4.2 Asymptotic Distribution of Dynamic Bootstrap Estimator

We shall focus on the AR(p) model. Throughout we assume that initial data $y_{-p+1}, \ldots, y_0$ are available, and

$$\begin{aligned} y_t &= \alpha_0 + \sum_{j=1}^{p} \alpha_j y_{t-j} + \varepsilon_t \\ &= \alpha_0 + \alpha_1 y_{t-1} + \ldots + \alpha_p y_{t-p} + \varepsilon_t \\ &= \boldsymbol{x}_t^T \boldsymbol{\alpha} + \varepsilon_t, \end{aligned} \tag{4.2.1}$$

for $1 \leq t \leq n$, $\boldsymbol{x}_t = \left(1, y_{t-1}, \ldots, y_{t-p}\right)^T$ and ε_t are i.i.d. with zero mean and finite variance σ^2. Let $\hat{\boldsymbol{\alpha}} = \left(\hat{\alpha}_0, \ldots, \hat{\alpha}_p\right)^T$ denote the least square estimator of $\boldsymbol{\alpha}$, given by

$$\hat{\boldsymbol{\alpha}} = \left(\sum_{t=1}^{n} \boldsymbol{x}_t \boldsymbol{x}_t^T \right)^{-1} \sum_{t=1}^{n} \boldsymbol{x}_t y_t. \tag{4.2.2}$$

The residuals are defined as

$$\hat{\varepsilon}_t = y_t - \boldsymbol{x}_t^T \hat{\boldsymbol{\alpha}}, \quad 1 \leq t \leq n.$$

We now consider the dynamic bootstrap. Let $\varepsilon_1^*, \ldots, \varepsilon_n^*$ denote a sample drawn randomly with replacement from the set of residuals $\{\hat{\varepsilon}_1, \ldots, \hat{\varepsilon}_n\}$. Recursively we define

$$y_t^* = \begin{cases} y_t, & t = -p+1, \ldots, 0 \\ \boldsymbol{x}_t^{*T} \hat{\boldsymbol{\alpha}} + \varepsilon_t^*, & t = 1, \ldots, n, \end{cases}$$

where $\boldsymbol{x}_t^* = \left(1, y_{t-1}^*, \ldots, y_{t-p}^*\right)^T$, $1 \leq t \leq n$. The bootstrap least squares estimator is,

$$\hat{\boldsymbol{\alpha}}^* = \left(\sum_{t=1}^n \boldsymbol{x}_t^* \boldsymbol{x}_t^{*T} \right)^{-1} \sum_{t=1}^n \boldsymbol{x}_t^* y_t^*.$$

Substituting from (4.2.1) into (4.2.2), assuming $\boldsymbol{\alpha}_0 = \left(\alpha_{00}, \alpha_{01}, \ldots, \alpha_{0p}\right)^T$ is the true $\boldsymbol{\alpha}$, we obtain

$$\begin{aligned} \hat{\boldsymbol{\alpha}} &= \left(\sum_{t=1}^n \boldsymbol{x}_t \boldsymbol{x}_t^T \right)^{-1} \sum_{t=1}^n \boldsymbol{x}_t \left(\boldsymbol{x}_t^T \boldsymbol{\alpha}_0 + \varepsilon_t \right) \\ &= \boldsymbol{\alpha}_0 + \left(\sum_{t=1}^n \boldsymbol{x}_t \boldsymbol{x}_t^T \right)^{-1} \sum_{t=1}^n \boldsymbol{x}_t \varepsilon_t, \end{aligned}$$

from which it follows that

$$\sqrt{n}\left(\hat{\boldsymbol{\alpha}} - \boldsymbol{\alpha}_0\right) = \left(\frac{1}{n} \sum_{t=1}^n \boldsymbol{x}_t \boldsymbol{x}_t^T \right)^{-1} \frac{1}{\sqrt{n}} \sum_{t=1}^n \boldsymbol{x}_t \varepsilon_t. \tag{4.2.3}$$

Similarly,

$$\sqrt{n}\left(\hat{\boldsymbol{\alpha}}^* - \hat{\boldsymbol{\alpha}}\right) = \left(\frac{1}{n} \sum_{t=1}^n \boldsymbol{x}_t^* \boldsymbol{x}_t^{*T} \right)^{-1} \frac{1}{\sqrt{n}} \sum_{t=1}^n \boldsymbol{x}_t^* \varepsilon_t^*. \tag{4.2.4}$$

The following theorems tell us that the distribution of $\sqrt{n}\left(\hat{\boldsymbol{\alpha}}^* - \hat{\boldsymbol{\alpha}}\right)$ converges to that of $\sqrt{n}\left(\hat{\boldsymbol{\alpha}} - \boldsymbol{\alpha}_0\right)$. Let $\triangle_n$ denote the distribution of $\sqrt{n}\left(\hat{\boldsymbol{\alpha}} - \boldsymbol{\alpha}_0\right)$ and let $\hat{\triangle}_n$ denote the distribution of $\sqrt{n}\left(\hat{\boldsymbol{\alpha}}^* - \hat{\boldsymbol{\alpha}}\right)$ conditional on the set of residuals $\{\hat{\varepsilon}_1, \ldots, \hat{\varepsilon}_n\}$.

Theorem 4.2.1 (Asymptotic distribution of the least squares estimator)**.** *Suppose that the time series* $\{y_t\}$ *in (4.2.1) is a stationary* $AR(p)$ *process, and that* $\{\varepsilon_t\}_{t\geq 1}$ *is an i.i.d. process with zero mean and* $E|\varepsilon_t|^4 < \infty$*. Let* $\triangle_n$ *denote the distribution of least squares estimates specified in (4.2.3). Then*

$$\triangle_n \xrightarrow{d} N_{p+1}\left(\mathbf{0}, \sigma^2 \boldsymbol{A}^{-1}\right) \text{ as } n \to \infty,$$

where $\boldsymbol{A}$ *is defined in (4.2.5) and more explicitly in (4.2.16).*

Condition of finite fourth moment is required to prove Lemma 4.2.4. The corresponding theorem for the bootstrap estimator is as follows.

Theorem 4.2.2 (Asymptotic distribution of the bootstrap least squares estimator)**.** *Suppose that the assumptions of Theorem 4.2.1 hold. Let* $\hat{\triangle}_n$ *denote the distribution of the bootstrap least squares estimator specified in (4.2.4). Then,*

$$\hat{\triangle}_n \xrightarrow{d} N_{p+1}\left(\mathbf{0}, \sigma^2 \boldsymbol{A}^{-1}\right) \text{ as } n \to \infty,$$

where $\boldsymbol{A}$ *is the same as mentioned in Theorem 4.2.1.*

Thus $\triangle_n$ and $\hat{\triangle}_n$ have the same limiting distribution under an $\mathrm{AR}(p)$ model.

4.2.1 Outline of Proofs of Theorems 4.2.1 and 4.2.2

The proofs of Theorems 4.2.1 and 4.2.2 are provided in Section 4.2.2. In this section, we provide an outline of the way we prove these theorems.

Step 1 Show that

$$E\left(\frac{1}{n}\sum_{t=1}^{n} \boldsymbol{x}_t \boldsymbol{x}_t^T\right) \to \boldsymbol{A}. \tag{4.2.5}$$

This result is proved in Corollary 4.2.6.

Step 2 Apply a martingale central limit theorem to $n^{-1/2}\sum_{t=1}^{n}\boldsymbol{x}_t\varepsilon_t$, to establish that

$$\frac{1}{\sqrt{n}}\sum_{t=1}^{n}\boldsymbol{x}_t\varepsilon_t \xrightarrow{d} N_{p+1}\left(\mathbf{0},\sigma^2\boldsymbol{A}\right), \tag{4.2.6}$$

where $\boldsymbol{A}$ is as defined above. This result is proved in Lemma 4.2.8.

In Step 3, we shall make use of Slutsky's theorem, which states that if $\{Y_n\}_{n\geq 1}$ is a sequence of random variables such that $Y_n \xrightarrow{d} Y$, and X_n is a sequence of random variables such that $X_n \xrightarrow{p} 0$, then $X_n + Y_n \xrightarrow{d} Y$; see e.g. Taniguchi and Kakizawa (2000) for further details.

Step 3 Noting (4.2.3), we may combine (4.2.5) and (4.2.6) using Slutsky's theorem stated above. Thus

$$\sqrt{n}\left(\hat{\boldsymbol{\alpha}}-\boldsymbol{\alpha}_0\right) \xrightarrow{d} N_{p+1}\left(\mathbf{0},\sigma^2\boldsymbol{A}^{-1}\right).$$

The details of the proof of Theorem 4.2.2 are similar to those of Theorem 4.2.1. In particular (because $\hat{\triangle}_n$ is a random distribution as it depends on the sample) the same method shows that

$$\hat{\triangle}_n = \sqrt{n}\left(\hat{\boldsymbol{\alpha}}^*-\hat{\boldsymbol{\alpha}}\right) \xrightarrow{d} N_{p+1}\left(\mathbf{0},\hat{\sigma}^2\hat{\boldsymbol{A}}^{-1}\right),$$

with probability one, where $\hat{\sigma}^2$ is the sample estimate of the error variance, σ^2 and $\hat{\boldsymbol{A}}$ is the sample analogue of $\boldsymbol{A}$. Moreover under the moment condition $E|\varepsilon_t|^4 < \infty$, $\hat{\sigma}^2 \xrightarrow{p} \sigma^2$, $\hat{\boldsymbol{A}} \xrightarrow{p} \boldsymbol{A}$, so $\hat{\sigma}^2\hat{\boldsymbol{A}}^{-1} \xrightarrow{p} \sigma^2\boldsymbol{A}^{-1}$, i.e. the limit distribution in Theorem 4.2.2 is the same as that in Theorem 4.2.1. See Section 4.2.4 for further details.

4.2.2 Auxiliary Results

The result proved in the following lemma is probably well known but we have not managed to find a reference for it. It shows that the coefficients ψ_r decay to 0 exponen-

tially fast, where ψ_r is the coefficient of L^r in the expansion of

$$\begin{aligned}\alpha(L)^{-1} &= \sum_{j=0}^{\infty} \psi_j L^j \\ &= \psi(L).\end{aligned}$$

In the following lemma, we will establish an explicit bound for the ψ_j's, the coefficients in an infinite MA representation of a finite order AR(p) process.

Lemma 4.2.3. *Suppose $\{a_i^{-1} : i = 1, \ldots, p\}$ are the roots of $\alpha(L) = 1 - \alpha_1 L - \alpha_2 L^2 - \ldots - \alpha_p L^p = 0$, and are such that $|a_i| < 1$ for all $i = 1, \ldots, p$, as is the case under the assumptions of Theorem 4.2.1. Define $\tilde{a}_0 = \max\left(|a_1|, \ldots, |a_p|\right)$. Then for any $\delta \in]\tilde{a}_0, 1[$ there exist a constant $\nu = \nu(\delta)$ independent of r such that*

$$|\psi_r| \leq \nu a_0^r, \quad r \geq 1, \tag{4.2.7}$$

where $a_0 = \tilde{a}_0/\delta$.

Proof. As $\{a_i^{-1} : i = 1, \ldots, p\}$ are the roots of $\alpha(L) = 1 - \alpha_1 L - \alpha_2 L^2 - \ldots - \alpha_p L^p = 0$, we can write

$$\alpha(L) = -\alpha_p \prod_{i=1}^{p} \left(L - a_i^{-1}\right).$$

But the constant coefficient of $\alpha(L)$ is 1 which implies $\alpha_p = (-1)^{p+1} \prod_{i=1}^{p} a_i$. Therefore

$$\begin{aligned}\alpha(L) &= (-1)^p \prod_{i=1}^{p} a_i \prod_{i=1}^{p} \left(L - a_i^{-1}\right) \\ &= \prod_{i=1}^{p} (1 - a_i L).\end{aligned}$$

Thus

$$\begin{aligned}\alpha(L)^{-1} &= \prod_{i=1}^{p} (1 - a_i L)^{-1} \\ &= \prod_{i=1}^{p} \sum_{j=0}^{\infty} a_i^j L^j \\ &= 1 + \psi_1 L + \psi_2 L^2 + \ldots,\end{aligned}$$

and it follows that

$$\begin{aligned} |\psi_r| &= \left| \sum_{k_1+\ldots+k_p=r} \prod_{i=1}^{p} a_i^{k_i} \right| \\ &\leq \sum_{k_1+\ldots+k_p=r} \prod_{i=1}^{p} \left| a_i^{k_i} \right| \\ &\leq \sum_{k_1+\ldots+k_p=r} \prod_{i=1}^{p} \max_{1\leq i\leq p} |a_i|^{k_i}, \end{aligned}$$

which can be further written as

$$|\psi_r| \leq \sum_{k_1+\ldots+k_p=r} \tilde{a}_0^r,$$

where $\tilde{a}_0 = \max\left(|a_1|, \ldots, |a_p|\right)$ and $\sum_{k_1+\ldots+k_p=r}$ means sum over all non-negative integers $k_1, \ldots, k_p$, such that $k_1 + \ldots + k_p = r$. Therefore

$$|\psi_r| \leq \tilde{a}_0^r \sum_{k_1+\ldots+k_p=r} 1. \tag{4.2.8}$$

The coefficient $\sum_{k_1+\ldots+k_p=r} 1$ can be calculated explicitly, since the RHS of the (4.2.8) is the coefficient of L^r in the expansion of $(1-\tilde{a}_0 L)^{-p}$. Taking the $(p-1)$th order derivative with respect to z of both sides of the identity $(1-z)^{-1} = \sum_{j=0}^{\infty} z^j$, we obtain

$$\begin{aligned} \frac{(p-1)!}{(1-z)^p} &= \sum_{j=p-1}^{\infty} \frac{j!}{(j-p+1)!} z^{j-p+1} \\ &= \sum_{r=0}^{\infty} \frac{(p+r-1)!}{r!} z^r. \end{aligned}$$

Therefore,

$$\frac{1}{(1-z)^p} = \sum_{r=0}^{\infty} \frac{(p+r-1)!}{(p-1)!r!} z^r. \tag{4.2.9}$$

Thus, from (4.2.8) and (4.2.9), it follows that

$$\begin{aligned} |\psi_r| &\leq \frac{(p+r-1)!}{(p-1)!r!}\tilde{a}_0^r \\ &= \frac{(p+r-1)!}{(p-1)!r!}\delta^r\left(\frac{\tilde{a}_0}{\delta}\right)^r \\ &= \nu_r(\delta)a_0^r, \end{aligned}$$

where

$$\nu_r(\delta) = \frac{(p+r-1)!}{(p-1)!r!}\delta^r,$$

and $a_0 = \tilde{a}_0/\delta$ are such that $\delta \in]\tilde{a}_0, 1[$. Since $\delta \in (0,1)$, and ν_r is maximum at $r = (1-p)/\log\delta$, therefore

$$\sup_{r=0,1,\ldots} \nu_r(\delta) = \nu = \nu(\delta) < \infty.$$

Therefore, (4.2.7) holds as required. □

In the following lemma, we use Lemma 4.2.3 to prove a bound for the corresponding cumulant of an AR(p) process in terms of the joint cumulant of the error term ε_t.

Lemma 4.2.4. *Under the assumptions of Theorem 4.2.1,*

$$|Cum_k(y_{\tau_1},\ldots,y_{\tau_k})| \leq |\rho_k(\varepsilon)|\,\nu^k a_0^{\sum_{j=1}^k(\tau_j-\tau_0)}\frac{1-a_0^{k\tau_0}}{1-a_0^k},$$

where ν and a_0 are the bounds in Lemma 4.2.3, $\rho_k(\varepsilon)$ is the kth cumulant of ε_t, and $\tau_0 = \min(\tau_1,\ldots,\tau_k)$.

Proof. Consider the AR(p) process defined in (4.2.1),

$$y_t = \alpha_0 + \sum_{j=1}^p \alpha_j y_{t-j} + \varepsilon_t. \tag{4.2.10}$$

First of all, note that if we define

$$\tilde{y}_t = y_t - \frac{\alpha_0}{1-\sum_{k=1}^p \alpha_k}, \quad t = 1,2,\ldots,$$

then, substituting into (4.2.10) we obtain

$$\tilde{y}_t + \frac{\alpha_0}{1-\sum_{k=1}^{p}\alpha_k} = \alpha_0 + \sum_{j=1}^{p}\alpha_j\left(\tilde{y}_{t-j} + \frac{\alpha_0}{1-\sum_{k=1}^{p}\alpha_k}\right) + \varepsilon_t,$$

which implies

$$\tilde{y}_t = \alpha_0 - \frac{\alpha_0}{1-\sum_{k=1}^{p}\alpha_k} + \frac{\alpha_0\sum_{j=1}^{p}\alpha_j}{1-\sum_{k=1}^{p}\alpha_k} + \sum_{j=1}^{p}\alpha_j\tilde{y}_{t-j} + \varepsilon_t,$$

which leads to

$$\tilde{y}_t = \sum_{j=1}^{p}\alpha_j\tilde{y}_{t-j} + \varepsilon_t.$$

So to simplify, we assume $\alpha_0 = 0$ without any loss of generality.

We now wish to express each y_t as a linear combination of the errors $\varepsilon_1, \ldots, \varepsilon_t$ and the initial data $y_0, \ldots, y_{-j}$. We can do this by performing $(t-1)$ successive substitutions of $y_{t-k} = \sum_{j=1}^{p}\alpha_j y_{t-k-j} + \varepsilon_{t-k}$ but replacing $\varepsilon_0, \varepsilon_{-1}, \ldots, \varepsilon_{-j}$ by $y_0, y_{-1}, \ldots, y_{-j}$. An equivalent way to do this is to define

$$\tilde{\varepsilon}_t = \begin{cases} \varepsilon_t & t \geq 1 \\ y_t & -p+1 \leq t \leq 0 \\ 0 & t < -p+1. \end{cases}$$

Then

$$\begin{aligned} y_t &= \sum_{j=0}^{\infty}\psi_j\tilde{\varepsilon}_{t-j} \\ &= \sum_{j=0}^{t-1}\psi_j\varepsilon_{t-j} + \sum_{k=0}^{p-1}\psi_{t+k}y_{-k}. \end{aligned}$$

Since we are conditioning on the initial data, the second term on RHS is non-random. Moreover, by Lemma 4.2.3, it is exponentially small. Hence, to simplify calculations but without loss of generality, we shall assume that

$$y_t = \sum_{j=0}^{t-1}\psi_j\varepsilon_{t-j}, \quad t = 1, 2, \ldots. \tag{4.2.11}$$

Using the mulitilinearity property of joint cumulants, and the assumed i.i.d. property of the $\{\varepsilon_t\}$ sequence,

$$\begin{aligned} Cum_k(y_{\tau_1},\ldots,y_{\tau_k}) &= Cum_k\left(\sum_{j_1=0}^{\tau_1-1}\psi_{j_1}\varepsilon_{\tau_1-j_1},\ldots,\sum_{j_k=0}^{\tau_k-1}\psi_{j_k}\varepsilon_{\tau_k-j_k}\right) \\ &= \sum_{j_1=0}^{\tau_1-1}\cdots\sum_{j_k=0}^{\tau_k-1}\psi_{j_1}\ldots\psi_{j_k}Cum_k\left(\varepsilon_{\tau_1-j_1},\ldots,\varepsilon_{\tau_k-j_k}\right) \end{aligned}$$

and

$$Cum_k\left(\varepsilon_{\tau_1-j_1},\ldots,\varepsilon_{\tau_k-j_k}\right) = \begin{cases} \rho_k(\varepsilon) & \tau_1-j_1=\ldots=\tau_k-j_k \\ 0 & \text{otherwise.} \end{cases}$$

Thus writing $\tau_0 = \min(\tau_1,\ldots,\tau_k)$,

$$\begin{aligned} |Cum_k(y_{\tau_1},\ldots,y_{\tau_k})| &= \left|\rho_k(\varepsilon)\sum_{j=0}^{\tau_0-1}\psi_{j+\tau_1-\tau_0}\ldots\psi_{j+\tau_k-\tau_0}\right| \\ &\leq |\rho_k(\varepsilon)|\sum_{j=0}^{\tau_0-1}\prod_{q=1}^{k}\nu a_0^{j+\tau_q-\tau_0} \\ &= |\rho_k(\varepsilon)|\sum_{j=0}^{\tau_0-1}\nu^k a_0^{kj+\sum_{q=1}^{k}(\tau_q-\tau_0)} \\ &= |\rho_k(\varepsilon)|\,\nu^k a_0^{\sum_{q=1}^{k}(\tau_q-\tau_0)}\frac{1-a_0^{k\tau_0}}{1-a_0^k}, \end{aligned}$$

because $\left|\psi_{j+\tau_q-\tau_0}\right| \leq \nu a_0^{j+\tau_q-\tau_0}$, by Lemma 4.2.3. □

Our next lemma shows that relevant sums of products of the y_t process are $O(n^{-1})$.

Lemma 4.2.5. *Suppose that the assumptions of Theorem 4.2.1 hold. Then for each $r,s = 1,\ldots,p$,*

$$Var\left(\frac{1}{n}\sum_{t=1}^{n}y_{t-r}y_{t-s}\right) = O(n^{-1}). \tag{4.2.12}$$

Proof. Using the result in (4.2.11) and writing $a = t_1 - r$, $b = t_1 - s$, $c = t_2 - r$, and

$d = t_2 - s$, we have

$$\begin{aligned} Var\left(\frac{1}{n}\sum_{t=1}^{n} y_{t-r}y_{t-s}\right) &= \frac{1}{n^2}\sum_{t_1=1}^{n}\sum_{t_2=1}^{n} Cov\left(y_{t_1-r}y_{t_1-s}, y_{t_2-r}y_{t_2-s}\right) \\ &= \frac{1}{n^2}\sum_{t_1=1}^{n}\sum_{t_2=1}^{n} \kappa^{ab,cd}, \end{aligned}$$

where $\kappa^{ab,cd}$ is a generalised cumulant. Using the rule for expressing generalised cumulants in terms of ordinary cumulants (see McCullagh, 1987, p.31),

$$\begin{aligned} \kappa^{ab,cd} &= \kappa^{a,b,c,d} + \kappa^{a}\kappa^{b,c,d} + \kappa^{b}\kappa^{a,c,d} + \kappa^{c}\kappa^{a,b,d} + \kappa^{d}\kappa^{a,b,c} \\ &\quad + \kappa^{a,c}\kappa^{b,d} + \kappa^{a,d}\kappa^{b,c} + \kappa^{a}\kappa^{b}\kappa^{c,d} + \kappa^{a}\kappa^{c}\kappa^{b,d} \\ &\quad + \kappa^{a}\kappa^{d}\kappa^{b,c} + \kappa^{b}\kappa^{c}\kappa^{a,d} + \kappa^{b}\kappa^{d}\kappa^{a,c}. \end{aligned}$$

As all first order cumulants are zero i.e. $\kappa^{a} = \kappa^{b} = \kappa^{c} = \kappa^{d} = 0$, the above expression can be further simplified as

$$\kappa^{ab,cd} = \kappa^{a,b,c,d} + \kappa^{a,c}\kappa^{b,d} + \kappa^{a,d}\kappa^{b,c}.$$

Using Lemma 4.2.4 with Lemma 4.2.3, we can write

$$\begin{aligned} \left|\kappa^{a,b,c,d}\right| &= \left|cum_4\left(y_{t_1-r}, y_{t_1-s}, y_{t_2-r}, y_{t_2-s}\right)\right| \\ &\leq \left|\rho_4(\varepsilon)\right| \nu^4 a_0^{\sum_{j=1}^{4}(\tau_j-\tau_0)} \frac{1-a_0^{4\tau_0}}{1-a_0^4}, \end{aligned} \tag{4.2.13}$$

where $\tau_1 = t_1 - r$, $\tau_2 = t_1 - s$, $\tau_3 = t_2 - r$, $\tau_4 = t_2 - s$ and $\tau_0 = \min\left(\tau_1, \tau_2, \tau_3, \tau_4\right)$. From elementary considerations,

$$\sum_{j=1}^{4}\left(\tau_j - \tau_0\right) \leq 2|t_1 - t_2| + 2|r - s|,$$

and so continuing from (4.2.13),

$$\begin{aligned} \left|\kappa^{a,b,c,d}\right| &\leq \left|\rho_4(\varepsilon)\right| \nu^4 a_0^{2|t_1-t_2|+2|r-s|} \frac{1-a_0^{4\tau_0}}{1-a_0^4} \\ &= C_{r,s}\, a_0^{2|t_1-t_2|}, \end{aligned} \tag{4.2.14}$$

where

$$C_{r,s} = |\rho_4(\varepsilon)|\, \nu^4 a_0^{2|r-s|} \frac{1 - a_0^{4\tau_0}}{1 - a_0^4}.$$

Therefore using the result in (4.2.14),

$$\left| \frac{1}{n^2} \sum_{t_1=1}^{n} \sum_{t_2=1}^{n} \kappa^{t_1-r,t_1-s,t_2-r,t_2-s} \right| \leq \frac{1}{n^2} \sum_{t_1=1}^{n} \sum_{t_2=1}^{n} C_{r,s}\, a_0^{2|t_1-t_2|}.$$

Substituting $m = t_1 - t_2$, we can write

$$\begin{aligned} \frac{1}{n^2} \sum_{t_1=1}^{n} \sum_{t_2=1}^{n} C_{r,s}\, a_0^{2|t_1-t_2|} &= \frac{C_{r,s}}{n^2} \sum_{m=-n+1}^{n-1} (n - |m|)\, a_0^{2|m|}, \\ &= \frac{C_{r,s}}{n} \sum_{m=-n+1}^{n-1} \left(1 - \frac{|m|}{n}\right) a_0^{2|m|} \\ &\leq \frac{C_{r,s}}{n} \sum_{m=-\infty}^{\infty} a_0^{2|m|}. \end{aligned}$$

The remaining sum can be further expressed as a sum of two infinite geometric series, and therefore

$$\begin{aligned} \left| \frac{1}{n^2} \sum_{t_1=1}^{n} \sum_{t_2=1}^{n} \kappa^{t_1-r,t_1-s,t_2-r,t_2-s} \right| &\leq \frac{C_{r,s}}{n} \frac{2}{1 - a_0^2} \\ &= O(n^{-1}). \end{aligned}$$

Similar calculations, using Lemma 4.2.4 again, show that

$$\frac{1}{n^2} \sum_{t_1=1}^{n} \sum_{t_2=1}^{n} \left\{ \kappa^{a,c} \kappa^{b,d} + \kappa^{a,d} \kappa^{b,c} \right\} = O(n^{-1}),$$

and thus (4.2.12) is proved. □

Corollary 4.2.6. *Under the assumptions of Theorem 4.2.1,*

$$E\left(\frac{1}{n} \sum_{t=1}^{n} \boldsymbol{x}_t \boldsymbol{x}_t^T \right) \to \boldsymbol{A} \text{ as } n \to \infty, \tag{4.2.15}$$

where $\boldsymbol{A} = [a_{ij}]_{i,j=1}^{p+1}$ *and*

$$a_{ij} = \begin{cases} 1 & i = j = 1 \\ \mu & i = 1, j > 1 \text{ or } i > 1, j = 1 \\ \sigma^2 \sum_{j=0}^{\infty} \psi_j \psi_{j+k} & i, j > 1, \ |i-j| = k \\ 0 & \text{elsewhere.} \end{cases} \tag{4.2.16}$$

where $\mu = E(y_t) = \alpha_0 / \left(1 - \sum_{j=1}^{p} \alpha_j\right)$ *and* $\sigma^2 = Var(\varepsilon_t)$.

Proof. Consider the model (4.2.11). In this case,

$$\begin{aligned} E\left(\frac{1}{n}\sum_{t=1}^{n} y_{t-r}y_{t-s}\right) &= \frac{1}{n}\sum_{t=1}^{n} E\left(\sum_{j_1=0}^{t-r-1} \psi_{j_1}\varepsilon_{t-r-j_1} \sum_{j_2=0}^{t-s-1} \psi_{j_2}\varepsilon_{t-s-j_2}\right) \\ &= \frac{\sigma^2}{n}\sum_{t=1}^{n} \sum_{j=0}^{t-\max(r,s)-1} \psi_j \psi_{j+|r-s|}. \end{aligned}$$

But

$$\frac{\sigma^2}{n}\sum_{t=1}^{n} \sum_{j=0}^{t-\max(r,s)-1} \psi_j \psi_{j+|r-s|} \ \rightarrow \ \sigma^2 \sum_{j=0}^{\infty} \psi_j \psi_{j+|r-s|} \tag{4.2.17}$$

as $n \rightarrow \infty$ and so Corollary 4.2.6 follows, since each term in the expectation on the LHS of (4.2.15) is of the form of (4.2.17), apart from the entries with $i = 1$ or $j = 1$. The cases with $i = 1$ or $j = 1$ follow from the fact that the first component of $\boldsymbol{x}_t$ is 1. □

4.2.3 Martingale Central Limit Theorem

To establish (4.2.6) in Step 2, we make use of the following result due to Brown (1971). Using the terminology defined in Section 2 of Brown's paper, let

$$\{\xi_{nt}, \mathcal{F}_{nt}, t = 1, 2, \ldots, n; n = 1, 2, \ldots\},$$

be a martingale difference sequence. Note that $\mathcal{F}_{n1} \subseteq \mathcal{F}_{n2} \subseteq \ldots$, where $\mathcal{F}_{nt} = \sigma(\varepsilon_1, \ldots, \varepsilon_t)$ is a sigma algebra, for definition see e.g. Williams (1991, p.15). A martingale, say X_t, is defined as a stochastic process such that its conditional expectation at time t given that all the previous observations up to some earlier time s is equal to the observation at the

earlier time i.e. $E(X_t|X_1,\ldots,X_s) = X_s$. While we will say a process Y_t is a martingale difference sequence relative to X_t if and only if $E(Y_{t+1}|X_s, -\infty < s \leq t) = 0$ for all t, for definition see e.g. Brockwell and Davis (1991, p.546).

The characteristic function for a martingale difference sequence ξ_{nt} conditioned on $\mathcal{F}_{n,t-1}$ can be defined as

$$\phi_{nt}(v) = E\left(e^{iv\xi_{nt}}|\mathcal{F}_{n,t-1}\right), \tag{4.2.18}$$

and let

$$\begin{aligned}
\sigma_{nt}^2 &= E\left(\xi_{nt}^2|\mathcal{F}_{n,t-1}\right) \\
V_n^2 &= \sum_{t=1}^{n}\sigma_{nt}^2 \\
s_n^2 &= E(V_n^2) \\
f_n(v) &= \prod_{t=1}^{n}\phi_{nt}(v/s_n) \\
b_n &= s_n^{-2}\max_{1\leq t\leq n}\sigma_{nt}^2
\end{aligned}$$

for $n = 1, 2, \ldots$. A key condition is that

$$s_n^{-2}V_n^2 \xrightarrow{p} 1 \quad \text{as } n \to \infty. \tag{4.2.19}$$

For the class of martingales satisfying the above condition (4.2.19), the Lindeberg condition (see e.g. Billingsley, 1979, p.310) is said to hold if

$$s_n^{-2}\sum_{k=1}^{n}E\left[\xi_{nk}^2 I\left(|\xi_{nk}| > \eta\right)\right] \xrightarrow{p} 0. \tag{4.2.20}$$

for each fixed $\eta > 0$, where $I(.)$ is an indicator function.

Theorem 4.2.7. *(Brown, 1971) Assume that (4.2.19) holds. Then*

$$\begin{aligned}
f_n(v) &\xrightarrow{p} e^{-\frac{1}{2}v^2} \\
b_n &\xrightarrow{p} 0 \\
P\left[\sum_{t=1}^{n}\xi_{nt}/s_n \leq x\right] &= \mathbf{\Phi}(x),
\end{aligned}$$

as $n \to \infty$ if and only if the Lindeberg condition (4.2.20) holds, where $\mathbf{\Phi}(.)$ is the cdf of $N(0,1)$.

We make use of Theorem 4.2.7 in the proof of the following lemma.

Lemma 4.2.8. *Under the assumptions of Theorem 4.2.1,*

$$\frac{1}{\sqrt{n}} \sum_{t=1}^{n} \boldsymbol{x}_t \varepsilon_t \to N_{p+1}\left(\mathbf{0}, \sigma^2 \boldsymbol{A}\right),$$

where $\boldsymbol{A} = [a_{jk}]_{j,k=1}^{p+1}$ is defined in (4.2.16) and $\sigma^2 = Var(\varepsilon_t)$.

Proof. Consider for fixed $\boldsymbol{c}$

$$\begin{aligned} T_n &= \boldsymbol{c}^T \left(\frac{1}{\sqrt{n}} \sum_{t=1}^{n} \boldsymbol{x}_t \varepsilon_t \right) \\ &= \frac{1}{\sqrt{n}} \sum_{t=1}^{n} (\boldsymbol{c}\boldsymbol{x}_t) \varepsilon_t, \end{aligned}$$

where $\varepsilon_1, \varepsilon_2, \ldots$ are i.i.d. and $\boldsymbol{x}_t = \left(1, y_{t-1}, \ldots, y_{t-p}\right)^T$. We shall first prove that asymptotic normality holds for each fixed $\boldsymbol{c}$ and then use the Cramér-Wold device to deduce that $n^{-1/2} \sum_{t=1}^{n} \boldsymbol{x}_t \varepsilon_t$ is asymptotically normal. [The Cramér-Wold device states that if $\{(\boldsymbol{X}_n)\}_{n \geq 1}$ is a sequence of random vectors and $\boldsymbol{X}$ is another random vector and for each fixed $\boldsymbol{c}$, $\boldsymbol{X}_n^T \boldsymbol{c} \xrightarrow{d} \boldsymbol{X}^T \boldsymbol{c}$, then we may conclude $\boldsymbol{X}_n \xrightarrow{d} \boldsymbol{X}$; see, for example, Van der Vaart (2000, p.16)]. Under the assumptions of Theorem 4.2.1, ε_k is independent of $\boldsymbol{x}_k$. Define

$$\xi_{nk}(\boldsymbol{c}) = \frac{1}{\sqrt{n}} (\boldsymbol{c}^T \boldsymbol{x}_k) \varepsilon_k, \quad k = 1, \ldots, n.$$

To simplify notation write $\xi_{nk} = \xi_{nk}(\boldsymbol{c})$. Then

$$T_n = \sum_{t=1}^{n} \xi_{nt}.$$

We now check the conditions (4.2.19) and (4.2.20).

Proof that Condition (4.2.19) is satisfied.

Consider the sigma field $\mathcal{F}_{nt} = \sigma(\varepsilon_1, \ldots, \varepsilon_t)$, then

$$\begin{aligned} \sigma_{nt}^2 &= E\left[\xi_{nt}^2 | \mathcal{F}_{n,t-1}\right] \\ &= \frac{1}{n}(\boldsymbol{c}^T \boldsymbol{x}_t)^2 E[\varepsilon_t^2] \\ &= \frac{\sigma^2}{n}(\boldsymbol{c}^T \boldsymbol{x}_t)^2, \end{aligned}$$

since $\boldsymbol{x}_t$ is known when we condition on $\mathcal{F}_{n,t-1}$.

Thus

$$\begin{aligned} V_n^2 &= \sum_{t=1}^{n} \sigma_{nt}^2 \\ &= \frac{\sigma^2}{n} \sum_{t=1}^{n} (\boldsymbol{c}^T \boldsymbol{x}_t)^2 \\ &= \sigma^2 \boldsymbol{c}^T \left(\frac{1}{n} \sum_{t=1}^{n} \boldsymbol{x}_t \boldsymbol{x}_t^T \right) \boldsymbol{c}, \\ &\xrightarrow{p} \sigma^2 \boldsymbol{c}^T \boldsymbol{A} \boldsymbol{c}, \text{ as } n \to \infty. \end{aligned}$$

Also

$$\begin{aligned} s_n^2 &= E\left[V_n^2\right] \\ &= E\left(\sigma^2 \boldsymbol{c}^T \left(\frac{1}{n} \sum_{t=1}^{n} \boldsymbol{x}_t \boldsymbol{x}_t^T \right) \boldsymbol{c} \right) \\ &\to \sigma^2 \boldsymbol{c}^T \boldsymbol{A} \boldsymbol{c} \text{ as } n \to \infty. \end{aligned}$$

Thus

$$s_n^{-2} V_n^2 \xrightarrow{p} \left(\sigma^2 \boldsymbol{c}^T \boldsymbol{A} \boldsymbol{c} \right)^{-1} V_n^2 \xrightarrow{p} 1 \text{ as } n \to \infty,$$

and condition (4.2.19) holds.

Proof that condition (4.2.20) is satisfied.

We wish to show that for each fixed $\eta > 0$,

$$a_n' = \sum_{k=1}^{n} E\left[\xi_{nk}^2 I\left(|\xi_{nk}| > \eta\right)\right] \xrightarrow{p} 0.$$

Using the Markov inequality (see e.g. Williams, 1991, p.59),

$$I\left(|\xi_{nk}| > \eta\right) \leq \frac{\xi_{nk}^{p}}{\eta^{p}},$$

where $p \in \mathbb{Z}^{+}$. In our case, we choose $p = 2$ to simplify the calculations, as for this choice of p we will end up with the fourth moment. Thus

$$I\left(|\xi_{nk}| > \eta\right) \leq \frac{\xi_{nk}^{2}}{\eta^{2}},$$

and it follows that

$$\begin{aligned} a_n' &\leq \sum_{k=1}^{n} \eta^{-2} E\left[\xi_{nk}^{4} | \mathcal{F}_{n,k-1}\right] \\ &= \eta^{-2} n^{-2} \sum_{k=1}^{n} E\left[(\boldsymbol{c}^T \boldsymbol{x}_k)^4 \varepsilon_k^4 | \mathcal{F}_{n,k-1}\right] \\ &= \eta^{-2} n^{-2} \mu_4(\varepsilon) \sum_{k=1}^{n} (\boldsymbol{c}^T \boldsymbol{x}_k)^4 = a_n, \end{aligned}$$

where $\mu_4(\varepsilon) = E[\varepsilon_k^4]$, and the independence of ε_k and $\boldsymbol{x}_k$ along with the i.i.d. property of ε_k have been used. To show that $a_n \xrightarrow{p} 0$, it is sufficient to establish that $E[a_n] \to 0$ as $n \to \infty$; see e.g. Chung (2001, Theorem 4.1.4).

But the first four cumulants of $\boldsymbol{c}^T \boldsymbol{x}_k$ are uniformly bounded in k (Lemma 4.2.4), and therefore

$$\sup_{k} E\left[(\boldsymbol{c}^T \boldsymbol{x}_k)^4\right] \leq C < \infty.$$

This leads to

$$E[s_n] \leq \frac{1}{\eta^2} . \frac{1}{n^2} \mu_4(\varepsilon) . nC \quad = \quad O\left(\frac{1}{n}\right) \text{ for each fixed } \eta > 0.$$

Therefore (4.2.20) holds. Thus, we have proved that convergence holds for each fixed $\boldsymbol{c}$, so by the Cramér-Wold device mentioned above we may conclude that

$$\frac{1}{\sqrt{n}} \sum_{t=1}^{n} \boldsymbol{x}_t \varepsilon_t \xrightarrow{d} N_{p+1}\left(\boldsymbol{0}, \sigma^2 \boldsymbol{A}\right).$$

Hence Lemma 4.2.8 is proved. □

4.2.4 Proof of Theorem 4.2.2

A benefit of providing explicit bounds in the proof of Theorem 4.2.1 is that we are able to see how the proof in the bootstrap case follows in similar fashion. Recall that the residuals are defined by

$$\hat{\varepsilon}_t = y_t - \boldsymbol{x}_t^T \hat{\boldsymbol{\alpha}} = \varepsilon_t - \boldsymbol{x}_t^T \left(\hat{\boldsymbol{\alpha}} - \boldsymbol{\alpha}_0\right),$$

using the fact that $y_t = \boldsymbol{x}_t^T \boldsymbol{\alpha}_0 + \varepsilon_t$; so, in vector form,

$$\begin{aligned} \hat{\boldsymbol{\varepsilon}} &= \boldsymbol{\varepsilon} - \widetilde{\boldsymbol{X}} \left(\widetilde{\boldsymbol{X}}^T \widetilde{\boldsymbol{X}}\right)^{-1} \widetilde{\boldsymbol{X}}^T \boldsymbol{\varepsilon} \\ &= \left[\boldsymbol{I}_n - \widetilde{\boldsymbol{X}} \left(\widetilde{\boldsymbol{X}}^T \widetilde{\boldsymbol{X}}\right)^{-1} \widetilde{\boldsymbol{X}}^T\right] \boldsymbol{\varepsilon}, \end{aligned}$$

where $\boldsymbol{I}_n$ is the $n \times n$ identity matrix and $\widetilde{\boldsymbol{X}} = [\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n]^T$. Therefore, since $\boldsymbol{x}_t$ contains the constant term (recall that by definition the first component of $\boldsymbol{x}_t$ is 1), it follows that

$$\frac{1}{n} \mathbf{1}_n^T \hat{\boldsymbol{\varepsilon}} = \frac{1}{n} \mathbf{1}_n^T \left[\boldsymbol{I}_n - \widetilde{\boldsymbol{X}} \left(\widetilde{\boldsymbol{X}}^T \widetilde{\boldsymbol{X}}\right)^{-1} \widetilde{\boldsymbol{X}}^T\right] \boldsymbol{\varepsilon} = 0,$$

where $\mathbf{1}_n$ is the n-vector of ones. Also,

$$\begin{aligned} \frac{1}{n} \sum_{t=1}^{n} \hat{\varepsilon}_t^2 &= \frac{1}{n} \sum_{t=1}^{n} \varepsilon_t^2 + \frac{1}{n} \left[\left\{\sqrt{n} \left(\hat{\boldsymbol{\alpha}} - \boldsymbol{\alpha}_0\right)\right\}^T \left(\frac{1}{n} \sum_{t=1}^{n} \boldsymbol{x}_t \boldsymbol{x}_t^T\right) \left\{\sqrt{n} \left(\hat{\boldsymbol{\alpha}} - \boldsymbol{\alpha}_0\right)\right\}\right] \\ &\quad - \frac{2}{n} \left(\frac{1}{\sqrt{n}} \sum_{t=1}^{n} \varepsilon_t \boldsymbol{x}_t^T\right) \left\{\sqrt{n} \left(\hat{\boldsymbol{\alpha}} - \boldsymbol{\alpha}_0\right)\right\} \\ &= \frac{1}{n} \sum_{t=1}^{n} \varepsilon_t^2 + O_p\left(n^{-1}\right), \end{aligned}$$

because, as was shown in the proof of Theorem 4.2.1,

$$\begin{aligned} \left\| \frac{1}{\sqrt{n}} \sum_{t=1}^{n} \varepsilon_t \boldsymbol{x}_t \right\| &= O_p(1), \\ \left\| \sqrt{n} \left(\hat{\boldsymbol{\alpha}} - \boldsymbol{\alpha}_0\right) \right\| &= O_p(1) \end{aligned}$$

and

$$\left\| \frac{1}{n} \sum_{t=1}^{n} \boldsymbol{x}_t \boldsymbol{x}_t^T \right\| = O_p(1).$$

Similar but slightly more elaborate calculations show that

$$\frac{1}{n} \sum_{t=1}^{n} \hat{\varepsilon}_t^r = \frac{1}{n} \sum_{t=1}^{n} \varepsilon_t^r + O_p\left(n^{-1}\right), \quad r = 3, 4.$$

Therefore the first four moments of the $\hat{\varepsilon}_t$ agree with the sample moments of the $\varepsilon_1, \ldots, \varepsilon_t$ up to an error term of order $O_p\left(n^{-1}\right)$. Moreover, since $E\left|\varepsilon_t\right|^4 < \infty$, by assumption, it follows from the strong law of large numbers that

$$\frac{1}{n} \sum_{t=1}^{n} \hat{\varepsilon}_t^r \xrightarrow{p} E\left[\varepsilon_t^r\right], \quad r = 2, 3, 4.$$

Let us now return to the proof of Theorem 4.2.1. In view of the above, the bootstrap analogue of the bound in Lemma 4.2.4 converges in probability to the RHS in Lemma 4.2.4. Likewise, the bootstrap analogy of the bounds used in Lemma 4.2.5 converges in probability to the bounds used in the proof of Theorem 4.2.1. Similar components apply to the application of Theorem 4.2.7 in the bootstrap case.

4.2.5 Extension to Portmanteau Statistic

At the cost of further technical detail it is possible to extend Theorems 4.2.1 and 4.2.2 to the portmanteau statistic itself. Following McLeod (1978, formula(34)) and Katayama (2008), we may write

$$\hat{\boldsymbol{r}} = \boldsymbol{r} + \boldsymbol{X}\left(\hat{\boldsymbol{\alpha}} - \boldsymbol{\alpha}_0\right) + O_p\left(n^{-1}\right),$$

where $\boldsymbol{X}$ is defined in (3.2.2). Using the martingale central limit theorem stated in Theorem 4.2.7 above, we can establish the joint normality of $\sqrt{n}\left(\boldsymbol{r}^T, \left(\hat{\boldsymbol{\alpha}} - \boldsymbol{\alpha}_0\right)^T\right)^T$, and the asymptotic normality of $\sqrt{n}\hat{\boldsymbol{r}}$ follows; see McLeod (1978, Theorem 1) where it is

shown that

$$\sqrt{n}\hat{\boldsymbol{r}} \xrightarrow{d} N_m\left(\boldsymbol{0}_m, \boldsymbol{I}_m - \boldsymbol{X}\mathbb{J}^{-1}\boldsymbol{X}^T\right).$$

Making use of similar lemmas to those proved earlier in this section, the asymptotic normality of $\sqrt{n}\hat{\boldsymbol{r}}^*$, the bootstrap analogue of $\sqrt{n}\hat{\boldsymbol{r}}$, can be proved. Novelty of our results is that proof of central limit theorem using the martingale theory which can be easily generalized for the dynamic bootstrap.

4.3 Higher-Order Accuracy

In the previous chapter it was shown numerically that the use of the dynamic bootstrap for approximating the null distribution of the portmanteau statistic leads to excellent accuracy. It is natural to ask whether this good performance can be explained in theoretical terms. To provide some idea of what theoretical results one might hope to obtain, we shall look at what happens in the multivariate i.i.d. case. Firstly, we show how good performance of dynamic bootstrap can be proved in a multivariate i.i.d. case. In Section 4.3.2 we discuss the possible way of proving these results for the portmanteau test, where a non-i.i.d. version of the results proved in the case of multivariate i.i.d. case is required. The particular source we use here is Fisher et al. (1996, Appendix B); see also Hall (1992) and reference therein. The full details are rather involved and we only give a brief sketch. Subsequently, we discuss what would be involved in proving parallel results for the portmanteau test in the time series setting.

4.3.1 The Multivariate i.i.d. Case

Suppose that we observe i.i.d. random vectors $\boldsymbol{X}_1, \ldots, \boldsymbol{X}_n$. Let $T = T_n(\boldsymbol{X}_1, \ldots, \boldsymbol{X}_n)$ denote a statistic which has an asymptotic χ^2_d distribution as $n \to \infty$ under some null hypothesis H_0. That is, under H_0,

$$P\left(T > \chi^2_{d,1-\alpha}\right) \to \alpha \text{ as } n \to \infty$$

for each $\alpha \in (0,1)$, where $\chi^2_{d,\alpha}$ is the α-quantile of χ^2_d. Since the asymptotic distribution of T under H_0 does not depend on any unknowns, T is said to be (asymptotically) pivotal under H_0. The desirability of using pivotal statistics in the bootstrap setting has been discussed by Hall (1992) and, for example, Fisher et al. (1996). The theoretical advantage of using pivotal statistics is that they generally achieve higher order of (theoretical) accuracy than is achieved by non-pivotal statistics.

In broad generality there is usually a multivariate central limit theorem underlying a statistic with a χ^2 limit distribution. In regular situations, we can express an asymptotically χ^2_d statistic T in the form

$$T = \boldsymbol{R}^T\boldsymbol{R}, \quad \boldsymbol{R} = \boldsymbol{R}_0 + n^{-1/2}\boldsymbol{R}_1 + n^{-1}\boldsymbol{R}_2 + O_p(n^{-3/2}) \tag{4.3.1}$$

where each $\boldsymbol{R}_i$ is a vector with components $R_i^1, \ldots, R_i^d$, and each R_i^j is a function of the form $n^{-1}\sum_{k=1}^n P_i^j(\boldsymbol{X}_k)$, where each P_i^j is a polynomial; see Fisher et al. (1996, Appendix B). In (4.3.1), as $n \to \infty$, $\boldsymbol{R} \xrightarrow{p} \boldsymbol{R}_o$, from which we deduce that $\boldsymbol{R}_0$ is asymptotically standard d-variate normal i.e. $N_d(\boldsymbol{0}_d, \boldsymbol{I}_d)$ under H_0, because $T = \boldsymbol{R}^T\boldsymbol{R} \xrightarrow{d} \chi^2_d$.

A key requirement for what follows is that the Edgeworth expansion given below for $f_n(\boldsymbol{x})$, the density of $\boldsymbol{R}$ at $\boldsymbol{R} = \boldsymbol{x}$, can be rigorously justified:

$$f_n(\boldsymbol{x}) = \phi_d(\boldsymbol{x})\left\{1 + n^{-1/2}p_1(\boldsymbol{x}) + n^{-1}p_2(\boldsymbol{x}) + n^{-3/2}p_3(\boldsymbol{x}) + n^{-2}E_n(\boldsymbol{x})\right\} \tag{4.3.2}$$

where $p_i(\boldsymbol{x})$, $i = 1,2,3$ are multivariate polynomials related to Hermite polynomials (see e.g. Sen et al., 2010, p.198), $\phi_d(\boldsymbol{x})$ is the $N_d(\boldsymbol{0}_d, \boldsymbol{I}_d)$ density, and the remainder term $E_n(\boldsymbol{x})$ satisfies

$$\sup_n \int |E_n(\boldsymbol{x})|\,\phi_d(\boldsymbol{x})d\boldsymbol{x} < \infty.$$

The coefficients of the polynomials $p_i(\boldsymbol{x})$ depend on lower-order cumulants of $\boldsymbol{R}$. For further details of Edgeworth expansions, see Bhattacharya and Rao (1976), Bhattacharya and Ghosh (1978), McCullagh (1987) and Hall (1992). An important further point is that the polynomials $p_1(\boldsymbol{x})$ and $p_3(\boldsymbol{x})$ are odd functions of $\boldsymbol{x}$, while $p_2(\boldsymbol{x})$ is an even function of $\boldsymbol{x}$.

Following Fisher et al. (1996), we now derive an expansion for $P(T \leq \widetilde{C})$, for any $\widetilde{C} > 0$. Write $B_1 = \left\{\boldsymbol{x} \in \mathbb{R}^d : \boldsymbol{x}^T\boldsymbol{x} \leq 1\right\}$ for the unit ball in $\mathbb{R}^d$. Then, using (4.3.2),

$$\begin{aligned} P\left[T \leq \widetilde{C}\right] &= P\left[\left\{\boldsymbol{R} : \|\boldsymbol{R}\|^2 \leq \widetilde{C}\right\}\right] \\ &= \int_{\boldsymbol{x} \in \widetilde{C}^{1/2}B_1} f_n(\boldsymbol{x})d\boldsymbol{x} \\ &= \int_{\boldsymbol{x} \in \widetilde{C}^{1/2}B_1} \phi_d(\boldsymbol{x})\left\{1 + n^{-1/2}p_1(\boldsymbol{x}) + n^{-1}p_2(\boldsymbol{x}) + n^{-3/2}p_3(\boldsymbol{x}) + n^{-2}E_n(\boldsymbol{x})\right\}d\boldsymbol{x} \\ &= \int_{\boldsymbol{x} \in \widetilde{C}^{1/2}B_1} \phi_d(\boldsymbol{x})\left\{1 + +n^{-1}p_2(\boldsymbol{x}) + n^{-2}E_n(\boldsymbol{x})\right\}d\boldsymbol{x}, \end{aligned}$$

which can be written as

$$P\left[T \leq \widetilde{C}\right] = P\left[\chi^2_d \leq \widetilde{C}\right] + n^{-1}q_1(d, \widetilde{C}, \boldsymbol{\kappa}) + O(n^{-2}). \tag{4.3.3}$$

Note that the $n^{-1/2}$ and $n^{-3/2}$ terms make a zero contribution because (i) $p_1(\boldsymbol{x})$ and $p_3(\boldsymbol{x})$ are odd functions and (ii) the unit ball B_1 is symmetric about the origin and (iii) $\phi_d(\boldsymbol{x})$ is an even function. In (4.3.3), $\boldsymbol{\kappa}$ is a vector of standardised cumulants of $\boldsymbol{R}$ which determine the coefficients of the multivariate polynomial $p_2(\boldsymbol{x})$.

Suppose now that we have B bootstrap samples, obtained by resampling the $\boldsymbol{X}_i$'s randomly with replacement, with equal probability n^{-1}, thereby obtaining $T^*_1, \ldots, T^*_B$ the values of statistic T for B bootstrap samples. Write $\hat{\chi}^2_{d,\alpha} = T^*_{([B\alpha]+1)}$, where $T^*_{(1)} \leq \ldots \leq T^*_{(B)}$ are the ordered values of $T^*_1, \ldots, T^*_B$. It is shown by Fisher et al. (1996, Appendix B, Section B5) that in the i.i.d. case under mild conditions,

$$P\left[T \leq \hat{\chi}^2_{d,\alpha}\right] = \alpha + O(n^{-2}) \tag{4.3.4}$$

under H_0; a more detailed argument is given by Hall (1992). Thus, in words: bootstrapping the asymptotically pivotal statistic T with a χ^2_d limit distribution results in a decrease in the error in the CDF approximation from $O(n^{-1})$ to $O(n^{-2})$.

4.3.2 The Portmanteau Statistic

The question we consider now is whether the analogue (4.3.4) holds for the portmanteau statistic in the time series setting. First of all, we note that the results in Fisher et al.

(1996, Appendix B) indicate that for (4.3.4) to have a chance of holding, T must be asymptotically pivotal under the null hypothesis. However, as we have seen earlier, the well-known variants of the portmanteau statistic, including those due to Box-Pierce, Ljung-Box and Katayama, are not asymptotically pivotal, at least when m stays fixed. However, it is possible to use an asymptotically pivotal version of the statistic, namely

$$T = n\hat{\boldsymbol{r}}^T \left(\boldsymbol{I} - \boldsymbol{C}\right)^{-1} \hat{\boldsymbol{r}}, \tag{4.3.5}$$

as defined in (3.3.2).

If version T in (4.3.5) of the portmanteau statistic is used, then it appears that most of the steps leading to (4.3.4) are similar to those in the i.i.d. case apart from justification of the Edgeworth expansions (4.3.2). Rigorous justification of Edgeworth expansions in the time series context is far more challenging than in the multivariate i.i.d. case. Götze and Hipp (1983) were the first authors to establish rigorous Edgeworth expansions for dependent data in some generality. The results are not easy to apply but a more recent paper by the same authors, Götze and Hipp (1994), focuses more specifically on time series. The latter paper works on sample means of a nonlinear functions of blocks of data: specifically, if $[y_t]_{t\geq 1}$ is a stationary time series, and $X = h\left(y_t, y_{t+1}, \ldots, y_{t+p-1}\right)$, where $h : \mathbb{R}^p \to \mathbb{R}$ is a nonlinear function, then Götze and Hipp (1994) develop Edgeworth expansions for the sample mean

$$\overline{X}_n = \frac{1}{n-p} \sum_{t=1}^{n-p+1} h\left(y_t, y_{t+1}, \ldots, y_{t+p-1}\right).$$

This framework does not include the situation under consideration here for either T or the bootstrap version T^*. It is an open question whether Götze and Hipp (1994) results can be extended in the direction under consideration here.

Other work on Edgeworth expansions in time series includes Maekawa (1985), who focuses on the ordinary least squares estimator in the ARMA$(1,1)$ model. In a substantial body of work, Lahiri has considered Edgeworth expansions for weakly dependent data in a sequence of papers, the most recent of which is Lahiri (2010). However, as far as we are aware, none of the research mentioned above (including that of Götz and Hipp) develops rigorous Edgeworth expansion theory for bootstrap distribution in the

time series context considered here. As noted above, this is an essential requirement for establishing the analogue of (4.3.4) rigorously. It will be interesting to see if further progress on this problem can be made in the future.

4.4 Improved Monti's Test

Monti's portmanteau test (Monti, 1994) is based on the first m partial autocorrelations. Like the Ljung-Box test (Ljung and Box, 1978), the null asymptotic distribution of Monti's test is χ^2_{m-p-q} under an ARMA(p,q) process. This approximation depends on m and stationarity of the true model. As we have noticed in Section 3.2.2, there are situations, e.g. where m is small and the process is near the stationary boundary, when this approximation becomes poor.

Katayama (2008) suggested a bias correction term in the Ljung-Box statistic. We have already looked into this bias corrected statistic with some numerical examples in Section 3.3. Following the idea of Katayama (2008), we suggest a bias correction term in Monti's test. We have already seen in our numerical examples that this bias correction term is able to correct the bias in Monti's test under the challenging conditions i.e. small m and near stationarity boundary; see Figure 3.3. In the following section we will give a derivation of Monti's bias correction term. This derivation is sketched on the lines of Katayama (2008).

Suppose that $\{y_t\}$ is the time series generated by a stationary ARMA(p,q) process,

$$\alpha(L)y_t = \beta(L)\varepsilon_t, \quad t = 0, \pm 1, \pm 2, \ldots \tag{4.4.1}$$

where $p+q > 0$ and $\{\varepsilon_t\}$ is i.i.d. $(0,\sigma^2)$. As $\alpha(L) = 1 - \sum_{i=1}^{p} \alpha_i L^i$ and $\beta(L) = 1 + \sum_{i=1}^{q} \beta_i L^i$, it follows that (4.4.1) can be written as:

$$(1 - \sum_{i=1}^{p} \alpha_i L^i) y_t = (1 + \sum_{i=1}^{q} \beta_i L^i)\varepsilon_t.$$

Let $\boldsymbol{\theta}_0 = (\boldsymbol{\alpha}_0, \boldsymbol{\beta}_0)$ denote the vector of true values of the parameters and $\hat{\boldsymbol{\theta}} = (\hat{\boldsymbol{\alpha}}, \hat{\boldsymbol{\beta}})$ is the nonlinear least squares estimate of $\boldsymbol{\theta}_0$ obtained from the observed time

series. The residuals $\hat{\varepsilon}_t = \varepsilon_t(\hat{\boldsymbol{\theta}})$ from the fitted model will be the linear combination of $y_1, y_2, \ldots, y_t$ with weights equal to the AR-representation coefficients based on nonlinear least squares estimates $\hat{\boldsymbol{\theta}}$ i.e. $\hat{\varepsilon}_t = \sum_{i=0}^{t-1} \hat{\pi}_i y_{t-i}$, where $\hat{\pi}_i$'s are defined as

$$\hat{\pi}(L) = \frac{\hat{\alpha}(L)}{\hat{\beta}(L)} = \sum_{i=0}^{\infty} \hat{\pi}_i L^i.$$

The $\hat{\pi}_i$ weights can be computed efficiently using Algorithm 5; see Section 3.2.1.

Monti (1994) has suggested a diagnostic test for time series models based on the partial autocorrelation defined as

$$Q_m^*(\hat{\boldsymbol{\omega}}) = n(n+2) \sum_{k=1}^{m} \frac{\hat{\omega}_k^2}{n-k} = \hat{\boldsymbol{\omega}}^T \boldsymbol{V}^2 \hat{\boldsymbol{\omega}},$$

where $\hat{\boldsymbol{\omega}} = (\hat{\omega}_1, \ldots, \hat{\omega}_m)$ be the vector of the first m partial autocorrelations with the kth element given by

$$\hat{\omega}_k = \frac{\hat{r}_k - \hat{\boldsymbol{r}}_{k-1}^T \hat{\boldsymbol{R}}_{k-1}^{-1} \hat{\boldsymbol{r}}'_{k-1}}{1 - \hat{r}_{k-1}^T \hat{\boldsymbol{R}}_{k-1}^{-1} \hat{\boldsymbol{r}}_{k-1}},$$

where $\hat{\boldsymbol{R}}_k = \left(\hat{\boldsymbol{r}}_{|i-j|}\right)_{i,j=1,\ldots,k}$ is the $k \times k$ Toeplitz matrix, $\hat{\boldsymbol{r}}'_k = (\hat{r}_k, \ldots, \hat{r}_1)^T$, and

$$\boldsymbol{V} = diag\left(\sqrt{\frac{n(n+2)}{n-1}}, \sqrt{\frac{n(n+2)}{n-2}}, \ldots, \sqrt{\frac{n(n+2)}{n-m}}\right).$$

Monti (1994) has shown that $\boldsymbol{V}^{\frac{1}{2}}\boldsymbol{\omega} \sim N(\boldsymbol{0}_m, (\boldsymbol{I} - \boldsymbol{C}))$, where $\boldsymbol{C} = \boldsymbol{X}\mathbb{J}^{-1}\boldsymbol{X}^T$, $\mathbb{J}$ is the Fisher's information matrix defined in (3.3.1). Each (i,j)th element of the partitioned matrix of $\boldsymbol{X}$ is given

$$\boldsymbol{X} = (-\alpha_{i-j}^* \vdots -\beta_{i-j}^*),$$

where α_i^* and β_i^* are defined in (3.2.3) and (3.2.4) respectively. Also, α_i^* and β_i^* can be computed efficiently using Algorithm 4.

4.4.1 Bias Term in Monti's Test

Consider $\boldsymbol{r} = (r_1, \ldots, r_m)^T$ and $\boldsymbol{\omega} = (\omega_1, \ldots, \omega_m)^T$ are the vectors of the first m autocorrelations and partial autocorrelations respectively. Monti (1994, Lemma 1) has shown that

$$\boldsymbol{\omega} = \boldsymbol{r} + O_p\left(\frac{1}{n}\right), \tag{4.4.2}$$

and

$$\hat{\boldsymbol{\omega}} = \hat{\boldsymbol{r}} + O_p\left(\frac{1}{n}\right), \tag{4.4.3}$$

where $\hat{\boldsymbol{r}}$ and $\hat{\boldsymbol{\omega}}$ are sample analogues of $\boldsymbol{r}$ and $\boldsymbol{\omega}$ respectively. So following result (34) in the proof of McLeod (1978),

$$\hat{\boldsymbol{r}} = \boldsymbol{r} + \boldsymbol{X}(\hat{\boldsymbol{\theta}}_n - \boldsymbol{\theta}_0) + O_p\left(\frac{1}{n}\right), \tag{4.4.4}$$

where $\boldsymbol{\theta}_0$ and $\hat{\boldsymbol{\theta}}_n$ are the true value and least squares estimate of $\boldsymbol{\theta} = (\boldsymbol{\alpha}, \boldsymbol{\beta})$. Then we can write, from (4.4.3) and (4.4.4),

$$\hat{\boldsymbol{\omega}} = \boldsymbol{r} + \boldsymbol{X}(\hat{\boldsymbol{\theta}}_n - \boldsymbol{\theta}_0) + O_p\left(\frac{1}{n}\right). \tag{4.4.5}$$

From (4.4.2) and (4.4.5)

$$\hat{\boldsymbol{\omega}} = \boldsymbol{\omega} + \boldsymbol{X}(\hat{\boldsymbol{\theta}}_n - \boldsymbol{\theta}_0) + O_p\left(\frac{1}{n}\right).$$

Working along the lines of Katayama (2008, formulae (10)-(13)), we can obtain the following bias corrected Monti's statistic:

$$Q_m^{**}(\hat{\boldsymbol{\omega}}) = Q_m^{*}(\hat{\boldsymbol{\omega}}) - B_{m,n}^{*}(\hat{\boldsymbol{\omega}}).$$

The extra positive random variable $B_{m,n}^{*}(\hat{\boldsymbol{\omega}})$ given by

$$\hat{B}_{m,n}^{*}(\hat{\boldsymbol{\omega}}) \quad = \quad \hat{\boldsymbol{\omega}}^T \boldsymbol{V} \hat{\boldsymbol{D}} \boldsymbol{V} \hat{\boldsymbol{\omega}},$$

where $\hat{\boldsymbol{\omega}}^T = (\hat{\omega}_1, \hat{\omega}_2, \ldots, \hat{\omega}_m)$, $\hat{\boldsymbol{D}} = \hat{\boldsymbol{X}}(\hat{\boldsymbol{X}}^T\hat{\boldsymbol{X}})^{-1}\hat{\boldsymbol{X}}^T$, where $\hat{\boldsymbol{X}}$ is a sample analogue of $\boldsymbol{X}$ formed by the coefficients in the reciprocal of estimated AR and MA polynomials. Algorithm 6 can then be used to obtain $\hat{B}^*_{m,n}(\hat{\boldsymbol{\omega}})$ efficiently. There is also the possibility of using the asymptotically pivotal statistic

$$\hat{\boldsymbol{\omega}}^T \left(\boldsymbol{I}_m - \boldsymbol{C}\right)^{-1} \hat{\boldsymbol{\omega}}$$

which has a limiting χ^2_{m-p-q} distribution under the null hypothesis: Section 3.3. This follows from the fact that $\sqrt{n}\,(\hat{\boldsymbol{\omega}} - \boldsymbol{\omega}_0)$ has the same asymptotic covariance matrix, $\boldsymbol{I}_m - \boldsymbol{C}$, as $\sqrt{n}\,(\hat{\boldsymbol{r}} - \boldsymbol{r}_0)$, where $\boldsymbol{\omega}_0$ and $\boldsymbol{r}_0$ are the true values of $\boldsymbol{\omega}$ and $\boldsymbol{r}$ respectively.

4.5 Conclusion

In this chapter, we have proved central limit theorems for the least squares estimator and dynamic bootstrap least squares estimator of time series models using a number of basic lemmas and martingale limit theory. These results show that the dynamic bootstrap least squares estimator has the same limit distribution as that of the least squares estimator. This is an important result and provides a basis for better performance of dynamic bootstrap methods for time series models.

We also gave a discussion to link this theory to the numerical results obtained in Section 2.5, where the dynamic bootstrap method showed good performance in approximating the finite sample distribution of portmanteau tests. We have provided only an outline on which the good performance of dynamic bootstrap methods can be proved but it requires a rigorous Edgeworth expansion for dynamic bootstrap distribution which is quite challenging to develop.

Finally, we made a novel suggestion to correct the bias in Monti (1994) test using the partial autocorrelations, which is analogous to the Ljung-Box statistic. There is also an asymptotically pivot version of this statistic which parallels the development of Section 3.3.

CHAPTER 5

Lasso Methods for Regression Models

5.1 Introduction

In this chapter, we study the lasso (Tibshirani, 1996) and adaptive lasso (Zou, 2006) for regression models. The theoretical properties of these lasso-type methods are well studied in the past decade. For example, Fan and Li (2001) have discussed the relationship between the penalized least squares and subset selection and also studied the variable selection properties for lasso-type methods. Zhao and Yu (2006) has also studied model selection consistency for the lasso and derived a condition, based on the covariance matrix of the predictors, to achieve this consistency. This same condition is also independently derived by Zou (2006). The theoretical properties of lasso-type methods are very appealing but there are still a number of unanswered questions including some issues in their practical application, e.g. the selection of the tuning parameter.

As discussed by Fan and Li (2001), penalised regression methods such as the lasso, ideally, possess two oracle properties:

1. the zero components (and only the zero components) are estimated as exactly zero with probability approaches 1 as $n \to \infty$, where n is the sample size; and

2. the non-zero parameters are estimated as efficiently well as when the correct submodel is known.

The tuning parameter plays a vital role in consistent variable selection. It controls the degree of shrinkage of the estimator. We compare the performance of lasso-type methods using different tuning parameter selectors suggested in the literature.

The oracle properties of these procedures are studied for different models and under various conditions e.g. the necessary condition for consistent selection discussed in Zhao and Yu (2006) and Zou (2006). We will demonstrate numerically that when this condition fails the adaptive lasso can still do correct variable selection while the lasso cannot.

Some of the literature on the application of the lasso in regression has focused on very high-dimensional settings. In this chapter we focus on the lasso in a modest number of dimensions as this seems more relevant to the applications of the lasso in the multivariate time series context considered in the next chapter. The rest of this chapter is organised as follows.

Section 5.2 describes shrinkage procedures and their implementation in regression models. In Section 5.3 we will discuss the necessary condition for the oracle performance of lasso-type methods. Section 5.4 discusses various methods for choosing the appropriate value of the tuning parameter and its effect on the performance of lasso-type procedures. Section 5.5 gives some numerical results on the performance of lasso methods for regression models. We end this chapter in Section 5.6 with discussion and conclusions about the performance of these lasso-type methods under various conditions.

5.2 Shrinkage Methods

The ready availability of fast and powerful computers, combined with rapid technological advances in methods of automated data collection, have led to the routine production of massive datasets, e.g. in bioinformatics. There are many real-life examples where we are dealing with a very large number of predictors, and this naturally leads to consideration of high-dimensional settings.

Traditional statistical estimation procedures such as ordinary least squares (OLS)

tend to perform poorly in high-dimensional problems. In particular, although OLS estimators typically have low bias, they tend to have high prediction variance, and may be difficult to interpret (Brown, 1993). In such situations it is often beneficial to use shrinkage i.e. shrink the estimator towards the zero vector, which in effect involves introducing some bias so as to decrease the prediction variance, with the net result of reducing the mean squared error of prediction.

There are several shrinkage methods suggested in the literature including ridge regression (Hoerl and Kennard, 1970). The paper by Tibshirani (1996), in which he suggested the lasso, is a big breakthrough in the field of sparse model estimation which performs the variable selection and coefficient shrinkage simultaneously. Other shrinkage methods include non-negative garotte (Breiman, 1995), smoothly clipped absolute deviation (SCAD) (Fan and Li, 2001), elastic net (Zou and Hastie, 2005), adaptive lasso (Zou, 2006), Dantzig selector (Candes and Tao, 2007), variable inclusion and selection algorithm (VISA) (Radchenko and James, 2008). Many other methods have been suggested in the literature but lasso-type methods are currently popular among researchers (Knight and Fu, 2000; Fan and Li, 2001; Wang and Leng, 2007; Hsu et al., 2008). The group lasso was originally suggested by Bakin (1999) in his PhD Thesis from The Australian National University, Canberra. This technique selects a group of variables; rather than individual variables, for more details see e.g. Yuan and Lin (2006), Zhao and Kulasekera (2006).

Most recently, James et al. (2009) proposed an algorithm DASSO (Dantzig selector with sequential optimization) to obtain the entire coefficient path for the Dantzig selector and they also investigated the relationship between the lasso and Dantzig selector. Hesterberg et al. (2008) have given a good survey of L_1 penalised regression. Very recent papers by Fan and Lv (2008), Fan and Lv (2009) and Lv and Fan (2009) are good reference for variable selection especially in high dimension setting. In the following paragraphs we will define the linear model and some notations used and referred to frequently in the later sections.

Let $(\boldsymbol{x}_1^T, y_1), \ldots, (\boldsymbol{x}_n^T, y_n)$ be n independent and identically distributed random vec-

tors, assumed to satisfy the linear model

$$y_i = \boldsymbol{x}_i^T \boldsymbol{\beta} + \varepsilon_i, \tag{5.2.1}$$

such that $y_i \in \mathbb{R}$ is the response variable, $\boldsymbol{x}_i = (x_{i1}, \ldots, x_{ip})^T \in \mathbb{R}^p$ is the p-dimensional set of predictors, the ε_i's are independently and identically distributed with mean 0 and variance σ^2 and $\boldsymbol{\beta} = (\beta_1, \ldots, \beta_p)$ is the set of parameters.

We define $\mathcal{A} = \{j : \beta_j \neq 0\}$ and $\mathcal{A}^c = \{j : \beta_j = 0\}$. Assume that only p_0 ($p_0 < p$) parameters are non-zero i.e. $\beta_j \neq 0$ for $j \in \mathcal{A}$ where $|\mathcal{A}| = p_0$ and $|.|$ stands for the number of elements in the set i.e. the cardinality of the set. Thus we can define $\boldsymbol{\beta}_{\mathcal{A}} = \{\beta_j : j \in \mathcal{A}\}$ and $\boldsymbol{\beta}_{\mathcal{A}^c} = \{\beta_j : j \in \mathcal{A}^c\}$. Also assume that $\frac{1}{n}\boldsymbol{X}^T\boldsymbol{X} \xrightarrow{p} \boldsymbol{C}$, where $\boldsymbol{X} = (\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n)^T$ is the design matrix and $\boldsymbol{C}$ is a positive definite matrix. We can define a partition of the matrix $\boldsymbol{C}$ as

$$\boldsymbol{C} = \begin{bmatrix} \boldsymbol{C}_{11} & \boldsymbol{C}_{12} \\ \boldsymbol{C}_{21} & \boldsymbol{C}_{22} \end{bmatrix} \tag{5.2.2}$$

where $\boldsymbol{C}_{11}$ is the $p_0 \times p_0$ submatrix corresponding to the active predictors $\{\boldsymbol{x}_j : j \in \mathcal{A}\}$. The least squares estimator estimates the zero coefficients as non-zero in the model defined above. We would like a method which is consistent in variable selection i.e. which correctly classifies the active (i.e. non-zero coefficients) and non-active (i.e. zero coefficients) predictors. This is an important property of lasso-type methods as mentioned by Knight and Fu (2000).

5.2.1 The Lasso

Tibshirani (1996) proposed a new shrinkage method named least absolute shrinkage and selection operator, abbreviated as lasso. The lasso shrinks some coefficients while setting others exactly to zero, and thus theoretical properties suggest that the lasso potentially enjoys the good features of both subset selection and ridge regression (Tibshirani,

1996). The lasso estimator of $\boldsymbol{\beta}$ is defined by

$$\hat{\boldsymbol{\beta}}^* = \text{argmin} \sum_{i=1}^{n} \left(y_i - \sum_{j=1}^{p} \beta_j x_{ij} \right)^2 \quad \text{subject to } \sum_{j=1}^{p} |\beta_j| \leq t,$$

or equivalently,

$$\hat{\boldsymbol{\beta}}^* = \text{argmin} \left\{ \sum_{i=1}^{n} \left(y_i - \sum_{j=1}^{p} \beta_j x_{ij} \right)^2 + \lambda \sum_{j=1}^{p} |\beta_j| \right\},$$

where t and λ are user-defined tuning parameters and control the amount of shrinkage. Smaller values of t and larger values of λ result in a higher amount of shrinkage. For detailed discussion on the selection of the tuning parameter see Section 5.4.

5.2.2 Characterisation of the Components

Variable selection is an important property of shrinkage methods. The lasso is a convex procedure and sets some of the components exactly to zero. In this section, we will look at how the model coefficients behave under the lasso. First we will look at a simple one dimensional example to explain why the lasso sometimes gives solutions which are exactly zero. Then we will move to the general case.

Consider a model

$$f(x) \quad = \quad (x+1)^2 + \lambda |x|$$

and the first order derivative

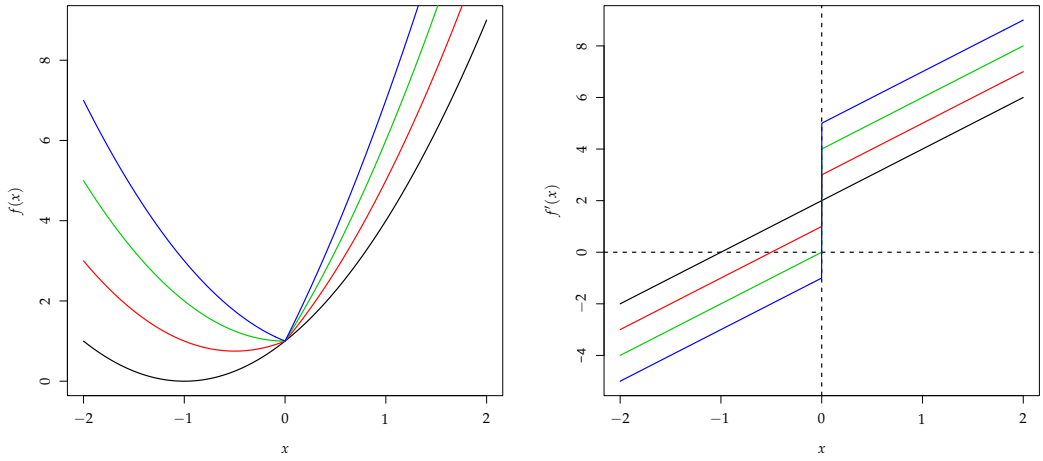
$$f'(x) \quad = \quad 2(x+1) + \lambda \text{sgn}(x),$$

where $x \in \mathbb{R}$, and $\text{sgn}(x) = -1, 0, 1$ for $x \leq 0$, $x = 0$ and $x > 0$ respectively, and $\lambda \geq 0$. The lasso will set those x to zero for which $f'(x)$ changes sign when x passes through origin. As $f'(x) \geq 0$ when $x \geq 0$, thus to set any of the x to zero the lasso needs to have

$f'(x) < 0$, when x passes through origin. Take $x < 0$. Then

$$\begin{aligned} f'(x) < 0 &\Leftrightarrow 2(x+1) + \lambda \text{sgn}(x) < 0 \\ &\Leftrightarrow 2(x+1) - \lambda < 0 \\ &\Leftrightarrow 2(x+1) < \lambda. \end{aligned}$$

Therefore $f'(x) < 0$ for all $x < 0$ when $\lambda > 2$. We conclude that $f(x)$ has a (nonstationary) global minimum at $x = 0$ if and only if $\lambda > 2$.



(a) $f(x) = (x+1)^2 + \lambda |x|$ (b) $f'(x) = 2(x+1) + \lambda \text{sgn}(x)$

Figure 5.1: Plots of $f(x)$ and $f'(x)$ for various choices of the tuning parameter. Key: $\lambda = 0$; $\lambda = 1$; $\lambda = 2$; $\lambda = 3$.

Figures 5.1(a) and 5.1(b) give plots of $f(x)$ and $f'(x)$ respectively for various choices of λ. It can be seen that for $\lambda > 2$, $f'(x)$ does become negative for all $x < 0$ but not when $\lambda \leq 2$.

Now we move to the general case and we consider the linear regression model

$$\boldsymbol{y} = \boldsymbol{X}\boldsymbol{\beta} + \boldsymbol{\varepsilon},$$

as defined in (5.2.1), but written in vector/matrix form here, with active set $\mathcal{A} = \{1, \ldots, p_0\}$. We define the lasso objective function as

$$L = (\boldsymbol{y} - \boldsymbol{X}\boldsymbol{\beta})^T (\boldsymbol{y} - \boldsymbol{X}\boldsymbol{\beta}) + \lambda \sum_{j=1}^{p} |\beta_j|.$$

Letting $\hat{\boldsymbol{\beta}}^* = \arg\min L$, thus

$$\begin{aligned} S_k &= \frac{\partial L}{\partial \beta_k} \\ &= -2\boldsymbol{x}_k^T (\boldsymbol{y} - \boldsymbol{X}\boldsymbol{\beta}) + \lambda \text{sgn}(\beta_k). \end{aligned}$$

If S_k changes sign when β_k passes through the origin then $\hat{\beta}_k^* = 0$; and if not then $\hat{\beta}_k^* \neq 0$. We write $\mathcal{A}_n^* = \left\{ j : \hat{\beta}_j^* \neq 0 \right\}$. Thus, at $\hat{\boldsymbol{\beta}}^*$,

$$\begin{aligned} j \in \mathcal{A}_n^* \quad &\text{if} \quad -2\boldsymbol{x}_k^T (\boldsymbol{y} - \boldsymbol{X}\hat{\boldsymbol{\beta}}^*) + \lambda \text{sgn}(\hat{\beta}_k) = 0 \\ j \notin \mathcal{A}_n^* \quad &\text{if} \quad \left| -2\boldsymbol{x}_k^T (\boldsymbol{y} - \boldsymbol{X}\hat{\boldsymbol{\beta}}^*) \right| \leq \lambda. \end{aligned}$$

As we have discussed earlier, the choice of λ is very important as it controls the degree of shrinkage. As $\lambda \to 0$, the OLS estimator is obtained and for λ sufficiently large, all the coefficients are zero.

5.2.3 LARS Algorithm

Efron et al. (2004) developed an efficient algorithm known as least angle regression (LARS) algorithm for finding the solution path of the lasso method, where the solution path is the set of values of $\hat{\boldsymbol{\beta}}^*(\lambda)$ as λ varies. Efron et al. (2004) also showed that both forward stagewise linear regression and the lasso are variants of the LARS. ("L" for least, "A" for angle, "R" for regression and "S" suggests "Lasso" and "Stagewise"). LARS cleverly organizes the calculations and thus the computational cost of the entire p steps is of the same order as that required for the usual least squares solution for the full model, though LARS modified for the lasso solution requires some additional steps (Efron et al., 2004). LARS, like classic forward selection, starts with all coefficients equal to zero.

Hastie et al. (2007) described the LARS algorithm to obtain the lasso solution as follows:

1. Standardise the predictors to have zero mean and unit variance. Start with the residual $\boldsymbol{r} = \boldsymbol{y} - \bar{\boldsymbol{y}}$, $\beta_1, \ldots, \beta_p = 0$.

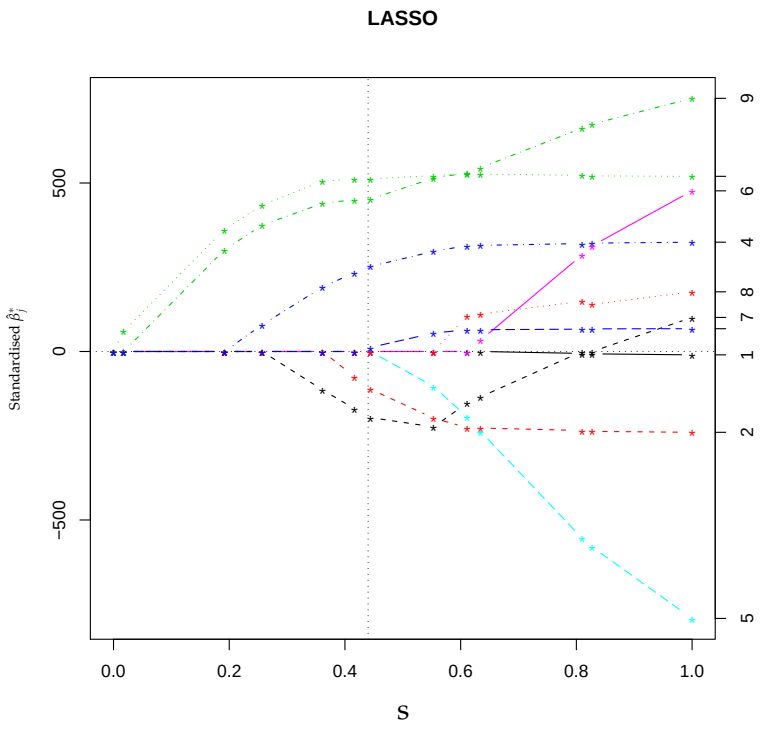


Figure 5.2: Example of Lasso solution path using the LARS algorithm

2. Find the predictor $\boldsymbol{x}_j$ most correlated with $\boldsymbol{r}$.

3. Move β_j from 0 towards its least squares coefficient $(\boldsymbol{x}_j, \boldsymbol{r})$, until some other competitor $\boldsymbol{x}_k$ has as much correlation with the current residual as does $\boldsymbol{x}_j$.

4. Move (β_j, β_k) in the direction defined by their joint least squares coefficient of the current residual on $(\boldsymbol{x}_j, \boldsymbol{x}_k)$, until some other competitor $\boldsymbol{x}_l$ has as much correlation with the current residual.

5. If a non-zero coefficient hits zero, drop it from $\mathcal{A}$ and recompute the current joint least squares direction.

6. Continue in this way until all p predictors have been entered in the model and we arrive at the full least squares solution.

Figure 5.2 is an example of the entire lasso solution path obtained using the *lars* package (Hastie and Efron, 2007) in R. This figure shows the solution path of the lasso obtained for the diabetes data discussed in Efron et al. (2004), as a function of the standardised lasso bound $s \in [0, 1]$. An important point to note is that the tuning parameter

used is not $\lambda \in [0, \infty]$ but a standardised quantity s defined as

$$s = \frac{||\hat{\boldsymbol{\beta}}^*||_1}{||\hat{\boldsymbol{\beta}}||_1} \in [0, 1], \tag{5.2.3}$$

where $\hat{\boldsymbol{\beta}}$ is the ordinary least squares estimate and $\hat{\boldsymbol{\beta}}^*$ is the lasso estimate of $\boldsymbol{\beta}$ for a specified value of λ and $||.||_1$ stands for the l_1 norm. It should be noticed that low values of s correspond to high values of λ thus resulting in a large amount of shrinkage. The vertical dotted line at $s = 0.44$ is the value of the tuning parameter chosen using cross-validation (CV), described in Section 5.4. It can be seen that all the coefficients are set to 0 at $s = 0$ and the predictors enter the solution sequentially as s increases. The lasso solution at $s = 1$ corresponds to the least squares estimates. This example clearly shows the importance of the tuning parameter in picking the correct solution from the entire solution path. We will discuss this issue in detail in Section 5.4.

5.2.4 The Adaptive Lasso

Zou (2006) proposed a new version of the lasso, named the adaptive lasso, by using adaptive weights which result in different penalisation for the coefficients appearing in the L_1 penalty term. The adaptive lasso can be defined as

$$\hat{\boldsymbol{\beta}}^{**} = \text{argmin} \left\{ \sum_{i=1}^{n} (y_i - \sum_{j=1}^{p} \beta_j x_{ij})^2 + \lambda \sum_{j=1}^{p} w_j |\beta_j| \right\},$$

where $(w_1, \ldots, w_p)$ are the adaptive weights. Zou has shown that if the weights are efficiently chosen in a data-dependent way then the adaptive lasso can achieve the oracle properties. He suggested the use of estimated weights, $\hat{w}_j = |b_j|^{-\gamma}$, where $\boldsymbol{b} = \{b_j : j = 1, \ldots, p\}$ is a root-n-consistent estimator of $\boldsymbol{\beta}$ and $\gamma > 0$ is a user-chosen constant.

The choice of $\hat{w}_j$ is very important and Zou (2006) suggested using ordinary least squares estimates while γ can be chosen by k-fold cross-validation. Zou (2006) has also noted that the adaptive lasso, like the lasso, is a convex optimisation problem and so does not suffer from having more than one local minimum, and its global minimum can be obtained by the LARS algorithm (Efron et al., 2004) after a simple modification,

we give in Algorithm 8, to accommodate the adaptive weights.

More recently, Pötscher and Schneider (2009) have studied the finite-sample and the large-sample distribution of the adaptive lasso. They have focused on the two important aspects of the adaptive lasso: (1) tuning to perform conservative model selection and (2) tuning to perform consistent model selection. They have shown that the finite-sample distribution of the adaptive lasso is highly non-normal and are often multimodal. Their results show asymptotic results with a fixed tuning parameter (i.e. tuning parameter not changing with sample size) can give the wrong picture of the adaptive lasso estimator's actual behaviour. They have also discussed scenarios when $\lambda \to \infty$ but $n^{1/2}\lambda \to \lambda_0$ when $0 \leq \lambda \leq \infty$ it is impossible to estimate the distribution function as none of the estimators is uniformly consistent.

Zou (2006) has studied whether the standard lasso has the oracle properties discussed by Fan and Li (2001). He showed that there are some scenarios e.g. when condition (5.2.4) given below does not hold, the lasso variable selection is not consistent. The oracle properties of other shrinkage methods are also studied in the literature. Fan and Li (2001) has studied the asymptotic properties of the SCAD and showed that penalized likelihood methods have some local maximisers for which the oracle properties hold.

Zou (2006) also gave a necessary and almost sufficient condition for the consistency of lasso variable selection. This condition, named as the irrepresentable condition, was also found independently by Zhao and Yu (2006). We will call this condition the Zhao-Yu-Zou condition (ZYZ condition). Assuming $\boldsymbol{C}_{11}$ is invertible, the ZYZ condition can be stated as

$$\left| \left[\boldsymbol{C}_{21} \boldsymbol{C}_{11}^{-1} \boldsymbol{s}_{\beta(\mathcal{A})} \right]_r \right| \leq 1, \quad r = 1, \ldots, p - p_0, \tag{5.2.4}$$

where $\boldsymbol{C}_{11}$, $\boldsymbol{C}_{21}$ are the partitions of $\boldsymbol{C}$ defined in (5.2.2), $\boldsymbol{s}_{\beta(\mathcal{A})} = \{\text{sgn}(\beta_j) : j \in \mathcal{A}\}$ and p_0 is the number of elements in $\mathcal{A}$.

In general, lasso-type methods are more effective than conventional methods, e.g. ordinary least squares, when the true model is sparse. If sparsity is not known to be present then there are not many advantages of using lasso-type methods as the

shrinkage results in biased estimates for the nonzero components (Hsu et al., 2008).

5.3 ZYZ Condition

The ZYZ condition (5.2.4) discussed by Zhao and Yu (2006) and Zou (2006) is a necessary condition on the matrix $\boldsymbol{C}$ defined in (5.2.2) for consistent variable selection. The ZYZ condition is always satisfied for an orthogonal design but there are some scenarios where this condition fails. Zhao and Yu (2006) and Zou (2006) have presented some examples where this condition fails, in which case, the lasso is inconsistent in variable selection. However, Zou (2006) has shown that the adaptive lasso has the oracle properties for the linear regression model, so that variable selection is consistent.

An important point to note is that the ZYZ condition is an asymptotic condition. The condition requires $\lambda \xrightarrow{p} 0$, which refers to large sample sizes $(n \to \infty)$. For finite sample sizes, the ZYZ condition does not always guarantee good variable selection.

When using the *lars* package in R for the implementation of the adaptive lasso, we notice that the theoretical properties are not shown in the simulated examples. As showed by Zou (2006) the adaptive lasso is consistent in variable selection even where the ZYZ condition fails for the standard lasso, but we failed to approach the variable selection oracle property of the adaptive lasso in the numerical example when the sample size becomes large. These strange results for the adaptive lasso lead us to look in depth to the LARS algorithm and we noticed that normalisation (see Section 5.3.1), if done after introduction of the adaptive weights, nullifies the effect of adaptive weights.

In this section we will show how use of adaptive weights makes the ZYZ condition hold even when it originally fails.

Assume $n^{-1}\boldsymbol{X}^T\boldsymbol{X} = \boldsymbol{C}^{(n)} \xrightarrow{p} \boldsymbol{C}$ and is partitioned as indicated in (5.2.2). The adaptive lasso rescales the design matrix $\boldsymbol{X}$ using some data-driven adaptive weights $\{w_j : j = 1, \ldots, p\}$. We can rearrange and partition the weight matrix, $\boldsymbol{W}$, as

$$\boldsymbol{W} = \begin{pmatrix} \boldsymbol{W}_{11} & \mathbf{0} \\ \mathbf{0} & \boldsymbol{W}_{22} \end{pmatrix},$$

where $\boldsymbol{W}_{11} = \text{diag}(w_j^{-1}; j \in \mathcal{A})$ and $\boldsymbol{W}_{22} = \text{diag}(w_j^{-1}; j \in \mathcal{A}^c)$.

Writing $\widetilde{\boldsymbol{X}} = \boldsymbol{X}\boldsymbol{W}$, we can define $\widetilde{\boldsymbol{C}}^{(n)} = n^{-1}\widetilde{\boldsymbol{X}}^T\widetilde{\boldsymbol{X}} \xrightarrow{p} \widetilde{\boldsymbol{C}}$. We can partition $\widetilde{\boldsymbol{C}}^{(n)}$ as

$$\widetilde{\boldsymbol{C}}^{(n)} = \begin{pmatrix} \widetilde{\boldsymbol{C}}_{11}^{(n)} & \widetilde{\boldsymbol{C}}_{12}^{(n)} \\ \widetilde{\boldsymbol{C}}_{21}^{(n)} & \widetilde{\boldsymbol{C}}_{22}^{(n)} \end{pmatrix}.$$

Now,

$$\begin{aligned} \widetilde{\boldsymbol{C}}^{(n)} &= \frac{1}{n}\widetilde{\boldsymbol{X}}^T\widetilde{\boldsymbol{X}} \\ &= \boldsymbol{W}^T\boldsymbol{C}^{(n)}\boldsymbol{W} \\ &= \begin{pmatrix} \boldsymbol{W}_{11}\boldsymbol{C}_{11}^{(n)}\boldsymbol{W}_{11} & \boldsymbol{W}_{11}\boldsymbol{C}_{12}^{(n)}\boldsymbol{W}_{22} \\ \boldsymbol{W}_{22}\boldsymbol{C}_{21}^{(n)}\boldsymbol{W}_{11} & \boldsymbol{W}_{22}\boldsymbol{C}_{22}^{(n)}\boldsymbol{W}_{22} \end{pmatrix} \end{aligned} \tag{5.3.1}$$

Take

$$\begin{aligned} \widetilde{\boldsymbol{C}}_{21}^{(n)}\widetilde{\boldsymbol{C}}_{11}^{(n)^{-1}}\boldsymbol{s}_{\beta(\mathcal{A})} &= \left(\boldsymbol{W}_{22}\boldsymbol{C}_{21}^{(n)}\boldsymbol{W}_{11}\right)\left(\boldsymbol{W}_{11}\boldsymbol{C}_{11}^{(n)}\boldsymbol{W}_{11}\right)^{-1}\boldsymbol{s}_{\beta(\mathcal{A})} \\ &= \boldsymbol{W}_{22}\boldsymbol{C}_{21}^{(n)}(\boldsymbol{C}_{11}^{(n)})^{-1}\boldsymbol{W}_{11}^{-1}\boldsymbol{s}_{\beta(\mathcal{A})}, \end{aligned}$$

where $\boldsymbol{s}_{\beta(\mathcal{A})}$ is defined in (5.2.4). If the weights $\{w_j\}$ are chosen appropriately (typical choices are inverse powers of absolute values of least squares estimates or ridge estimates or lasso estimates) then,

$$w_j = \frac{1}{|\hat{\beta}_j|^\gamma} \xrightarrow{p} \begin{cases} 1/|\beta_j|^\gamma, & j \in \mathcal{A} \\ \infty, & j \notin \mathcal{A}. \end{cases} \tag{5.3.2}$$

As $\boldsymbol{W}_{11} = \text{diag}\left(w_j^{-1} : j \in \mathcal{A}\right)$, $\boldsymbol{W}_{11}^{-1} = \text{diag}\left(w_j : j \in \mathcal{A}\right)$, so we can say, when in general $|\beta_j| >> 1$ for $j \in \mathcal{A}$, the elements of $\boldsymbol{W}_{11}^{-1}$ will be bounded by some finite value say k^*. Moreover, since $\boldsymbol{W}_{22} = \text{diag}\left(w_j^{-1} : j \notin \mathcal{A}\right)$, it can be easily concluded from (5.3.2) that $\boldsymbol{W}_{22} \xrightarrow{p} \boldsymbol{0}_{p-p_0}$, the $(p-p_0) \times (p-p_0)$ matrix of zeros. So for an appropriate choice of the adaptive lasso weights, we can say that componentwise

$$\left|\left[\boldsymbol{W}_{22}\boldsymbol{C}_{21}^{(n)}(\boldsymbol{C}_{11}^{(n)})^{-1}\boldsymbol{W}_{11}\boldsymbol{s}_{\beta(\mathcal{A})}\right]_r\right| \longrightarrow 0, \quad r = 1, \ldots, p-p_0$$

thus we can conclude that componentwise

$$\left|\left[\boldsymbol{W}_{22}\boldsymbol{C}_{21}^{(n)}(\boldsymbol{C}_{11}^{(n)})^{-1}\boldsymbol{W}_{11}\boldsymbol{s}_{\beta(\mathcal{A})}\right]_r\right| \leq 1, \quad r = 1,\ldots,p-p_0 \tag{5.3.3}$$

always holds, at least asymptotically. So the adaptive lasso always satisfies the ZYZ condition asymptotically.

5.3.1 Normalisation after Rescaling by the Adaptive Weights

We have mentioned earlier that, under certain conditions, normalisation of the design matrix often improves the performance of the lasso. As penalized least squares methods are not scale equivariant, it is recommended to normalize the predictors so that each variable has unit L_2 norm. Such a scaling is also the default option of the *lars* package in R.

To provide insight into the effect of normalisation, we consider a simple case. Suppose we have p predictors $\{\boldsymbol{x}_j : j = 1,\ldots,p\}$ for the model defined in (5.2.1) such that $n^{-1}\boldsymbol{X}^T\boldsymbol{X} \xrightarrow{p} \boldsymbol{C}$. LARS uses $\boldsymbol{x}_j/h_j$ to normalise the predictors, where $h_j = \sqrt{\sum_{i=1}^n x_{ij}^2}; j = 1,\ldots,p$.

Let $\widetilde{\boldsymbol{Z}}$ be the normalised design matrix of $\widetilde{\boldsymbol{X}}$, which can be defined as

$$\widetilde{\boldsymbol{Z}} = \widetilde{\boldsymbol{X}}\boldsymbol{D}, \tag{5.3.4}$$

where $\boldsymbol{D} = \text{diag}\left(1/h_1,\ldots,1/h_p\right)$. For illustrative purposes we consider

$$h_j = \begin{cases} h_1^* & \text{for all } j \in \mathcal{A} \\ h_2^* & \text{for all } j \in \mathcal{A}^c \end{cases}.$$

Thus $\boldsymbol{D}$ can be partitioned as $\boldsymbol{D} = \begin{pmatrix} \boldsymbol{D}_{11} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{D}_{22} \end{pmatrix}$, where $\boldsymbol{D}_{11} = h_1^{*^{-1}}\boldsymbol{I}_{p_0}$ and $\boldsymbol{D}_{22} = h_2^{*^{-1}}\boldsymbol{I}_{p-p_0}$. We can write the covariance matrix for the normalised predictors defined in

(5.3.4) as $\widetilde{\boldsymbol{C}}_Z^{(n)} = n^{-1}\widetilde{\boldsymbol{Z}}^T\widetilde{\boldsymbol{Z}}$ as follows:

$$\begin{aligned}\widetilde{\boldsymbol{C}}_Z^{(n)} &= \begin{pmatrix} \boldsymbol{D}_{11}\widetilde{\boldsymbol{C}}_{11}^{(n)}\boldsymbol{D}_{11} & \boldsymbol{D}_{11}\widetilde{\boldsymbol{C}}_{12}^{(n)}\boldsymbol{D}_{22} \\ \boldsymbol{D}_{22}\widetilde{\boldsymbol{C}}_{21}^{(n)}\widetilde{\boldsymbol{C}}_{11}^{(n)} & \boldsymbol{D}_{22}\widetilde{\boldsymbol{C}}_{22}^{(n)}\boldsymbol{D}_{22} \end{pmatrix} \\ &= \begin{pmatrix} \widetilde{\boldsymbol{C}}_{11(Z)}^{(n)} & \widetilde{\boldsymbol{C}}_{12(Z)}^{(n)} \\ \widetilde{\boldsymbol{C}}_{21(Z)}^{(n)} & \widetilde{\boldsymbol{C}}_{22(Z)}^{(n)} \end{pmatrix}, \text{say,}\end{aligned}$$

where $\widetilde{\boldsymbol{C}}_{ij(Z)}^{(n)} = (h_i^* h_j^*)^{-1}\widetilde{\boldsymbol{C}}_{ij}^{(n)}$, $i, j = 1, 2$.

Now take

$$\begin{aligned}\widetilde{\boldsymbol{C}}_{21(Z)}^{(n)}(\widetilde{\boldsymbol{C}}_{11(Z)}^{(n)})^{-1}\boldsymbol{s}_{\beta(\mathcal{A})} &= \left(\boldsymbol{D}_{22}\widetilde{\boldsymbol{C}}_{21}^{(n)}\boldsymbol{D}_{11}\right)\left(\boldsymbol{D}_{11}\widetilde{\boldsymbol{C}}_{11}^{(n)}\boldsymbol{D}_{11}\right)^{-1}\boldsymbol{s}_{\beta(\mathcal{A})} \\ &= \boldsymbol{D}_{22}\widetilde{\boldsymbol{C}}_{21}^{(n)}(\widetilde{\boldsymbol{C}}_{11}^{(n)})^{-1}\boldsymbol{D}_{11}^{-1}\boldsymbol{s}_{\beta(\mathcal{A})} \\ &= h_1^* h_2^{*^{-1}}\widetilde{\boldsymbol{C}}_{21}^{(n)}(\widetilde{\boldsymbol{C}}_{11}^{(n)})^{-1}\boldsymbol{s}_{\beta(\mathcal{A})}.\end{aligned}$$

Using $\widetilde{\boldsymbol{C}}_{11}^{(n)} = \boldsymbol{W}_{11}\boldsymbol{C}_{11}^{(n)}\boldsymbol{W}_{11}$ and $\widetilde{\boldsymbol{C}}_{21}^{(n)} = \boldsymbol{W}_{22}\boldsymbol{C}_{21}^{(n)}\boldsymbol{W}_{11}$, we get

$$\widetilde{\boldsymbol{C}}_{21(Z)}^{(n)}(\widetilde{\boldsymbol{C}}_{11(Z)}^{(n)})^{-1}\boldsymbol{s}_{\beta(\mathcal{A})} = h_1^* h_2^{*^{-1}}\left(\boldsymbol{W}_{22}\boldsymbol{C}_{21}^{(n)}(\boldsymbol{C}_{11}^{(n)})^{-1}\boldsymbol{W}_{\mathbf{11}}\right)\boldsymbol{s}_{\beta(\mathcal{A})}.$$

For the necessary condition for consistent variable selection to hold, we require

$$\left|\left[h_1^* h_2^{*^{-1}}\left(\boldsymbol{W}_{22}\boldsymbol{C}_{21}^{(n)}(\boldsymbol{C}_{11}^{(n)})^{-1}\boldsymbol{W}_{\mathbf{11}}\right)\boldsymbol{s}_{\beta(\mathcal{A})}\right]_r\right| \leq 1, \quad r = 1, \ldots, p - p_0.$$

Using the result in (5.3.3), this will lead to two different scenarios:

- if $h_1^* \leq h_2^*$, then the ZYZ condition always holds;
- if $h_1^* > h_2^*$, then normalisation can lead to failure of the ZYZ condition thus making the variable selection inconsistent.

For example, consider Model 0, $\beta_0 = (5.6, 5.6, 5.6, 0)^T$, studied by Zou (2006). The

covariance matrix used for simulation of predictors is

$$\boldsymbol{C} = \begin{bmatrix} 1 & -0.39 & -0.39 & 0.23 \\ -0.39 & 1 & -0.39 & 0.23 \\ -0.39 & -0.39 & 1 & 0.23 \\ 0.23 & 0.23 & 0.23 & 1 \end{bmatrix}.$$

We have $\left|\boldsymbol{C}_{21}\boldsymbol{C}_{11}^{-1}\boldsymbol{s}_{\beta(\mathcal{A})}\right| = 3.1363 > 1$, thus the ZYZ condition fails. Suppose $\boldsymbol{C}^{(n)}$ be the covariance matrix of the simulated set of predictors, $\boldsymbol{x}_i$:

$$\boldsymbol{C}^{(n)} = \begin{bmatrix} 1.0428311 & -0.4203259 & -0.3738564 & 0.2409415 \\ -0.4203259 & 0.9585507 & -0.3345396 & 0.2163182 \\ -0.3738564 & -0.3345396 & 0.9631588 & 0.2878907 \\ 0.2409415 & 0.2163182 & 0.2878907 & 1.0548909 \end{bmatrix}.$$

We observed that $\left|\boldsymbol{C}_{21}^{(n)}(\boldsymbol{C}_{11}^{(n)})^{-1}\boldsymbol{s}_{\beta(\mathcal{A})}\right| = 3.161134 > 1$. Thus the ZYZ condition fails so the lasso variable selection will be inconsistent.

Now if we apply the adaptive lasso, we need to rescale the predictors $\widetilde{\boldsymbol{x}}_j = \boldsymbol{x}_j / w_j$ using the adaptive weights, w_j. Here, for example, we use estimated weights $\hat{w}_j = |\hat{\beta}_j|^{-1}$, for $j = 1, \ldots, p$, where $\hat{\beta}_j$ is the least squares estimate of β_j, i.e. we choose the tuning parameter γ to be 1. Now the covariance matrix, $\widetilde{\boldsymbol{C}}^{(n)}$, of the $\widetilde{\boldsymbol{x}}_j$'s is given by

$$\widetilde{\boldsymbol{C}}^{(n)} = \begin{bmatrix} 0.169194693 & -0.080898724 & -0.067396782 & 0.004310662 \\ -0.080898724 & 0.218853480 & -0.071542606 & 0.004591009 \\ -0.067396782 & -0.071542606 & 0.192927391 & 0.005722972 \\ 0.004310662 & 0.004591009 & 0.005722972 & 0.002081131 \end{bmatrix}$$

and we observed that $\left|\widetilde{\boldsymbol{C}}_{21}^{(n)}(\widetilde{\boldsymbol{C}}_{11}^{(n)})^{-1}\boldsymbol{s}_{\beta(\mathcal{A})}\right| = 0.3185492 < 1$. Thus the ZYZ condition holds and leads the adaptive lasso to consistent variable selection. Now if we normalise the predictors after rescaling by the adaptive weights, the effect of the adaptive weights

is nullified and the resulting covariance matrix after normalisation is given as

$$\widetilde{\boldsymbol{C}}_z^{(n)} = \begin{bmatrix} 6.9640450 \times 10^{-3} & -2.068320 \times 10^{-4} & -1.109767 \times 10^{-3} & 8.745434 \times 10^{-4} \\ -2.068320 \times 10^{-4} & 3.475609 \times 10^{-5} & -7.317432 \times 10^{-5} & 5.785579 \times 10^{-5} \\ -1.1097671 \times 10^{-3} & -7.317432 \times 10^{-5} & 1.270880 \times 10^{-3} & 4.644906 \times 10^{-4} \\ 8.745434 \times 10^{-4} & 5.785579 \times 10^{-5} & 4.644906 \times 10^{-4} & 2.081131 \times 10^{-3} \end{bmatrix}$$

and we observe that $\left|\widetilde{\boldsymbol{C}}_{21(z)}(\widetilde{\boldsymbol{C}}_{11(z)}^{(n)})^{-1}\boldsymbol{s}_{\beta(\mathcal{A})}\right| = 9.645727 > 1$. Hence if predictors are normalised after introducing adaptive weights the adaptive lasso will result into the standard lasso.

The general case is less transparent but, even so, this illustrative example throws some light in to the effect of normalisation.

The use of adaptive weights makes the adaptive lasso an oracle procedure. Therefore it is crucial to determine at which stage we should normalise the predictors, if required. We observe that normalisation nullifies the effect of adaptive weights if it is done after the introduction of adaptive weights. In Algorithm 8, we elaborate the Zou (2006) Algorithm 1 to obtain the adaptive lasso estimates for normalised predictors.

Algorithm 8: The LARS algorithm for the adaptive lasso.

Step 1 Standardise the predictors $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_p$ so that each has mean 0 and variance 1.
Step 2 Estimate the weights $\hat{w}_j, j = 1, \ldots, p$ using the normalised predictors obtained in Step 1 above.
Step 3 Define $\boldsymbol{x}_j^* = \boldsymbol{x}_j / \hat{w}_j, j = 1, \ldots, p$.
Step 4 Solve the lasso problem for all λ

$$\hat{\boldsymbol{\beta}}^* = \text{argmin} \left\{ \sum_{i=1}^{n} \left(y_i - \sum_{j=1}^{p} \boldsymbol{x}_j^* \beta_j \right)^2 + \lambda \sum_{j=1}^{p} |\beta_j| \right\}.$$

Step 5 Output $\hat{\beta}_j^{**} = \hat{\beta}_j^* / \hat{w}_j$.

5.4 Selection of Tuning Parameter

Selection of the tuning parameter is very important as it can have a big influence on the performance of the estimator. Cross-validation is considered the simplest and most

widely used method for minimisation the prediction error (Hastie et al., 2001). In the literature, cross-validation (CV) is commonly used for estimating the tuning parameter. It is defined later in this section. The most common forms of cross-validation are k-fold, leave-one-out and the generalized cross-validation. The *lars* package uses k-fold cross-validation. We can describe the k-fold cross-validation as below:

1. Data consisting of n observations are divided at random into k mutually exclusive subsamples, known as k-folds.

2. The entire solution path is obtained as a function of the standardized tuning parameter $s \in [0, 1]$ using the LARS algorithm, while omitting the ith fold, where $i = 1, \ldots, k$.

3. The fitted model is then used for prediction of the omitted ith subsample and the prediction error is obtained against each choice of the tuning parameter $s \in [0, 1]$.

4. The value of s which minimizes the prediction error is considered the optimal choice of the tuning parameter.

Typical choices of k are 5 or 10. The choice $k = n$ is known as leave-one-out cross-validation, in this case we have n subsamples and for the ith subsample the fit is computed using all the data after omitting ith observation. Leave-one-out cross-validation is computationally very expensive. Generalized cross-validation provides an approximation to leave-one-out cross-validation. It is used when a linear model is fitted under a squared-error loss function. See Hastie et al. (2001) for more details.

The theory suggests that consistent variable selection depends very much on the selection of the tuning parameters. We will show and discuss later in Section 5.5 how the choice of tuning parameter affects the performance of the lasso and adaptive lasso. Our results (Section 5.5) show that when the tuning parameter is selected using cross-validation, the lasso and adaptive lasso do not appear to be consistent in variable selection, as independently showed by Wang and Leng (2009). Leng et al. (2006) have shown that if the prediction accuracy criterion is used to select the tuning parameter then lasso-type procedures cannot be consistent in variable selection.

We have noticed that the oracle performance of the lasso can be achieved if a reliable method of tuning parameter selection is used. Recently, papers by Wang and Leng (2009) and Hall et al. (2009a) confirmed our conclusions about the poor performance of cross-validation based on numerical results. Wang and Leng (2009) suggested a Bayesian information criterion (BIC) type criterion to choose the value of the tuning parameter.

The BIC has previously been used as a model selection tool. As in model building, we have several candidate models and adding new parameters to a model will increase the likelihood, but by including more parameters in the model, the model becomes more complex and the estimates also tend to have greater variance. In order to address this problem, Schwarz (1978) suggested a Bayesian information criterion (BIC) for the selection of a better model which achieves a suitable trade-off between simplicity (fewer parameters) and goodness of fit (greater likelihood). In the Gaussian case this takes the form given as

$$BIC = \log(\hat{\sigma}^2) + p \times \frac{\log(n)}{n},$$

where $\hat{\sigma}^2$ is the residual variance and p is the number of parameters. The candidate model which minimizes the BIC is selected. Note that $\log(\hat{\sigma}^2)$ is proportional to a maximised Gaussian likelihood. Wang et al. (2007) defined a BIC as follows:

$$BIC_{\mathcal{S}} = \log(\hat{\sigma}^2_{\mathcal{S}}) + |\mathcal{S}| \times \frac{\log(n)}{n} \times C_n,$$

where $|\mathcal{S}|$ is the size of the model i.e. the number of non-zero parameters in the model, $\hat{\sigma}^2_{\mathcal{S}} = SSE_{\mathcal{S}}/n$, $C_n > 0$ and $SSE_{\mathcal{S}}$ is the sum of squares of error for the non-zero component of model. For $C_n = 1$ the modified BIC of Wang et al. (2007) reduces to the traditional BIC of Schwarz (1978).

Suppose p_0 is the size of the true model, i.e. the number of non-zero parameters in the true model and $|\mathcal{S}|$ is the size of an arbitrary overfitted model i.e. $\mathcal{S}_T \subset \mathcal{S}$ and $|\mathcal{S}| > p_0$. Under a condtion on the size of non-zero coefficients and standard conditions of finite fourth order moments, Wang and Leng (2009) showed that $P(BIC_{\mathcal{S}} > BIC_{\mathcal{S}_T}) \longrightarrow 1$ for any overfitted model, $\mathcal{S}$. Thus, the BIC is consistent in differen-

tiating the true model from every overfitted model. Using this property of the BIC, Wang and Leng (2009) defined a modified BIC for the selection of the optimal value of the tuning parameter λ:

$$BIC_\lambda = \log(\hat{\sigma}^2_\lambda) + |\mathcal{S}_\lambda| \times \frac{\log n}{n} \times C_n, \tag{5.4.1}$$

where $\hat{\sigma}^2_\lambda = SSE_\lambda / n$, $SSE_\lambda = \sum_{i=1}^{n} \sum_{j=1}^{p} \left(y_{ij} - \sum_{j=1}^{p} \boldsymbol{x}_j^T \hat{\beta}_\lambda \right)^2$ is the sum of squared error, $\mathcal{S}_\lambda = \{j : \hat{\beta}_{j,\lambda} \neq 0\}$, $\hat{\beta}_{j,\lambda}$ is the estimate for some chosen value of λ. Importantly, $C_n > 0$ is a constant, which must be very carefully chosen. Wang and Leng used $C_n = \log \log p$ in their simulation study when the number of parameters diverge with sample size. In our study, we have tried several choices, for more discussion, see Section 5.5.2.

5.5 Numerical Results

In this section we look at the oracle properties (see Section 5.1) of the lasso (Tibshirani, 1996) and adaptive lasso (Zou, 2006). The theoretical properties of the lasso and adaptive lasso suggest that these methods are consistent in variable selection under some conditions, see Section 5.3. We compare the performance of these two shrinkage methods looking at the following properties:

(1) consistency in variable selection, and

(2) prediction performance.

For (1), we look at the probability of containing the true model on the solution path ($PTSP$) of these shrinkage methods. This measure has been used by Zou (2006). The solution path is the entire set of estimates corresponding to various choices of the tuning parameter. We obtain this solution path using the *lars* package in R. The solution path is said to contain the true model if it results in a correct estimated model (CM) for some choice of the tuning parameter, measure CM is defined more precisely later in this section. We define $PTSP$ as the proportion of times we get the CM out of N Monte Carlo runs. For an oracle performance, $PTSP \xrightarrow{p} 1$ as $n \to \infty$.

Convergence of $PTSP$ to 1 in probability supports theoretical consistent variable selection but to achieve it in practice requires the right choice of the tuning parameter. Selection of the appropriate value of the tuning parameter is very challenging as there is no precise theoretical answer to this question yet. In this study, we compare two methods, k-fold cross-validation and the BIC, in their selection of the value of the tuning parameter. We define two measures we will use to assess and compare the tuning parameters selectors' performance.

Model size (MS)

As we have defined earlier, model size, in the linear regression context, is the number of non-zero components in the model. For the simplicity of presentation, we assume that model (5.2.1) has $p_0 < p$, say, non-zero components i.e. $\{\beta_j \neq 0 : j \in \mathcal{A}\}$ then $|\mathcal{A}| = p_0$ while $|\mathcal{S}_F| = p$, where $\mathcal{A}$ and $|\mathcal{S}_F|$ are model size for true model and full model respectively. An oracle procedure, say μ, should have the model size $|\mathcal{S}_\mu| = |\mathcal{A}| = p_0$. Thus this measure guarantees that the prediction procedure is shrinking exactly the same number of estimates to zero as in the true model. In our results, we present the median MS (MMS) for the prediction procedure resulting from the M replicates. For an oracle procedure $MMS \xrightarrow{p} p_0$.

Correct model (CM)

The correct model is the measure we use to determine if the procedure is correctly shrinking the zero and non-zero components of the model. For oracle performance, the estimated model should have $\{\hat{\beta}_j = 0 \text{ for } j \in \mathcal{A}\}$ and $\{\hat{\beta}_j \neq 0 \text{ for } j \in \mathcal{A}^c\}$ i.e.

$$CM = \{\hat{\beta}_j = 0 : j \in \mathcal{A}, \hat{\beta}_j \neq 0 : j \in \mathcal{A}^c\}. \tag{5.5.1}$$

In our Monte Carlo study for each of these two methods, we compute and compare the percent of correct models (PCM). For an oracle procedure $MMS \xrightarrow{p} p_0$ and $PCM \xrightarrow{p} 100$. The measures MMS and PCM are also used by Wang and Leng (2009).

For (2), we compute the median of relative model error ($MRME$) of the lasso and adaptive lasso estimates, when the tuning parameter is selected by k-fold cross-validation and the BIC. The measure $MRME$ is used by Wang et al. (2007). We define the measure

MRME as follows.

Median of relative model error (*MRME*)

As defined in Fan and Li (2001), if $\{(\boldsymbol{x}_i, y_i) : i = 1, \ldots, n\}$ are assumed to be a random sample from the distribution $(\boldsymbol{X}, \boldsymbol{y})$. For a prediction procedure $\hat{\mu}(\boldsymbol{x})$ the prediction error can be defined as

$$PE(\hat{\mu}) = E\{y - \hat{\mu}(\boldsymbol{x})\}^2.$$

It should be noted that the expectation is taken only for the new data $(\boldsymbol{x}, y)$. Thus the prediction error, assuming $\boldsymbol{x}$ being random, can be further decomposed into two components as

$$PE(\hat{\mu}) = E\{y - E(y|\boldsymbol{x})\}^2 + E\{E(y|\boldsymbol{x}) - \hat{\mu}(\boldsymbol{x})\}^2.$$

The second component of the prediction error, due to lack of fit, is called model error. For the model (5.2.1), the model error can be defined as

$$ME(\hat{\mu}) = (\hat{\boldsymbol{\beta}} - \boldsymbol{\beta})^T E(\boldsymbol{x}\boldsymbol{x}^T)(\hat{\boldsymbol{\beta}} - \boldsymbol{\beta}), \tag{5.5.2}$$

where $\hat{\boldsymbol{\beta}}$ are the estimates used in the prediction procedure $\hat{\mu}(\boldsymbol{x})$. Now we can define the relative model error as the ratio of the model error for any prediction procedure $\hat{\mu}(\boldsymbol{x})$ to the model error for least squares. The median of the relative model error (*MRME*) for N Monte Carlo runs is obtained to assess the average lack of fit in the prediction procedure.

Ideally, a model should have a low *MRME*. In order to have a standard reference for the comparison, we define the oracle relative model error (*ORME*) as a ratio of oracle model error, where we have knowledge of the zero components of the model and the non-zero components have been replaced by the least square estimates, to the model error of least squares estimates. The *MRME* for each model was compared to the *ORME* and the model with *MRME* closest to the *ORME* is considered as the best prediction procedure.

We study the following three examples:

Model 0:

Suppose $p = 4$ and $\beta_0 = (5.6, 5.6, 5.6, 0)^T$, we consider this example to observe the effect on the lasso and adaptive lasso consistency in variable selection when the ZYZ condition does not hold. Using the partitioning of $\boldsymbol{C}$ defined in (5.2.2), we consider $\boldsymbol{C}_{11} = (1-\rho_1)\boldsymbol{I} + \rho_1 \boldsymbol{J}_1$, where $\boldsymbol{I}$ is the identity matrix, $\boldsymbol{J}_1$ is the matrix of 1's and $\boldsymbol{C}_{12} = \rho_2 \mathbf{1}$, where $\mathbf{1}$ is the vector of 1's. In this model, we chose $\rho_1 = -0.39$ and $\rho_2 = 0.23$. This model is the same as that studied in Zou (2006) to illustrate the inconsistent lasso solution path.

Model 1:

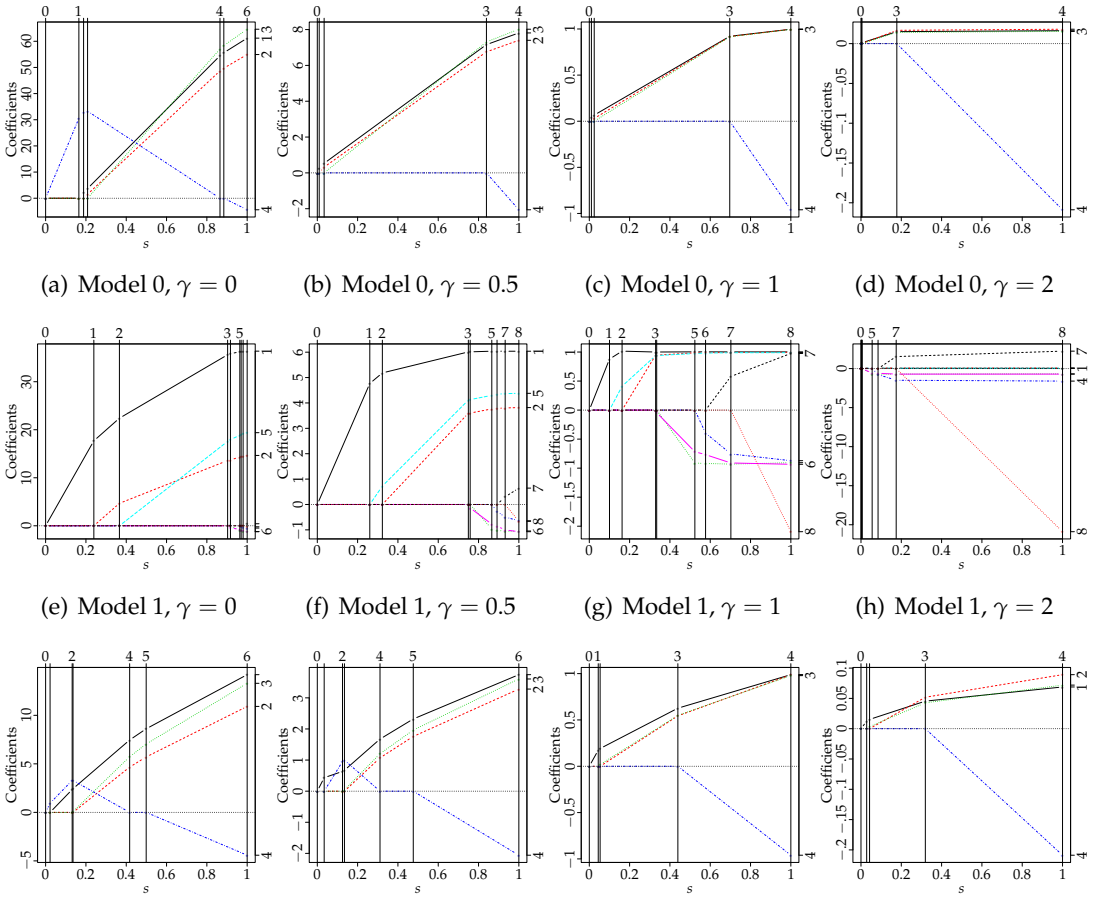
Suppose $\beta_0 = (3, 1.5, 0, 0, 2, 0, 0, 0)^T$ and $\boldsymbol{C} = \left\{(0.5^{|i-j|}); i, j = 1, \ldots, p\right\}$. The ZYZ condition holds for this choice of $\boldsymbol{C}$. This model was also studied by Fan and Li (2001), Zou (2006) and Hall et al. (2009a).

Model 2:

Suppose $\beta_0 = (0.85, 0.85, 0.85, 0)^T$ and $\boldsymbol{C}$ is the same as for Model 0. We have considered this example to compare with the results obtained in Model 0, where we have relatively large effects.

For all of the three examples, we designed a Monte Carlo study of 100 runs. For each Monte Carlo run, we simulate the linear model $\boldsymbol{y} = \boldsymbol{X}\boldsymbol{\beta} + \boldsymbol{\varepsilon}$ for the fixed set of parameters given above, where $\boldsymbol{X} \sim N_p(\mathbf{0}, \boldsymbol{C})$. In the Gaussian case, $\varepsilon_i \sim N(0, \sigma^2)$, we have considered the choices $\sigma = 1, 3, 6$ and 9.

In the next section, we will see if the numerical results support the conclusion that the lasso and adaptive lasso are consistent in variable selection. We will give results for *PTSP* to compare variable selection done by these lasso-type methods, without involving tuning parameter selection. In the second part of the next section, we will give results for *PCM*, *MMS* and *MRME*, which are obtained after selecting the tuning parameter. We will use k-fold cross-validation and BIC for the selection of tuning parameter. These results will also throw some light on how possible it is, in practice, to



(a) Model 0, $\gamma = 0$ (b) Model 0, $\gamma = 0.5$ (c) Model 0, $\gamma = 1$ (d) Model 0, $\gamma = 2$

(e) Model 1, $\gamma = 0$ (f) Model 1, $\gamma = 0.5$ (g) Model 1, $\gamma = 1$ (h) Model 1, $\gamma = 2$

(i) Model 2, $\gamma = 0$ (j) Model 2, $\gamma = 0.5$ (k) Model 2, $\gamma = 1$ (l) Model 2, $\gamma = 2$

Figure 5.3: Solution path of the lasso ($\gamma = 0$) and adaptive lasso ($\gamma = 0.5, 1,$ and 2) for the three models defined in Section 5.5. Model 0: $\boldsymbol{\beta}_0 = (5.6, 5.6, 5.6, 0)^T$. Model 1: $\boldsymbol{\beta}_0 = (3, 1.5, 0, 0, 2, 0, 0, 0)^T$. Model 2: $\boldsymbol{\beta}_0 = (0.85, 0.85, 0.85, 0)^T$. Key: ■ solid $\boldsymbol{x}_1$, ■ $\boldsymbol{x}_2$, ■ $\boldsymbol{x}_3$, ■ $\boldsymbol{x}_4$, ■ $\boldsymbol{x}_5$, ■ $\boldsymbol{x}_6$, ■ dashed $\boldsymbol{x}_7$, ■ $\boldsymbol{x}_8$.

achieve these oracle properties.

5.5.1 Variable Selection

To be consistent in variable selection is an important property of the shrinkage methods. The consistency or otherwise of the lasso selection depends on some model features e.g. the ZYZ condition (5.2.4).

In this section, we give results for the solution path and the probability that it contains the true model. These results are shown in Figure 5.3 and Figure 5.4. We consider $n = 50000$ and assume $\varepsilon \sim N(0, 1)$ in the following results.

Figure 5.3 gives the solution paths of the lasso and adaptive lasso for the three

models obtained using the LARS algorithm. The horizontal axis corresponds to the standardised tuning parameter, $s \in [0, 1]$, defined in (5.2.3), and the vertical axis gives the estimates of the model parameters in standardised units. The vertical lines show the steps of the LARS algorithm when a variable enters or leaves a model. Different colours and styles of lines correspond to the variables in the model as defined in the caption of the figure. In the following paragraphs, we will discuss the solution paths for each of the models defined above.

Model 0 in Figure 5.3:

We can define the following:

$$\mathcal{A} = \{1, 2, 3\} \quad \text{and} \quad \mathcal{A}^c = \{4\}.$$

For this model the ZYZ condition, the necessary condition for consistent variable selection, fails and thus, as shown by Zou (2006), the standard lasso cannot be consistent in variable selection. Figure 5.3(a) shows the solution path for the lasso and it can be observed that the variable in the non-active set, $\boldsymbol{x}_4$, enters the model even for values of s very close to zero and remains in the model except for a small range of values of s where its coefficient changes its sign. This shows that the lasso solution path does not contain the true model over a wide range of values of the tuning parameter which, as we will see in next section, makes it harder to select a value for the tuning parameter corresponding to the true model.

Figures 5.3(b)-(d) show the solution path of the adaptive lasso for choices of γ considered. It can be observed that the general pattern of the solution path in all these cases is similar. The predictors, $\boldsymbol{x}_1$, $\boldsymbol{x}_2$ and $\boldsymbol{x}_3$, corresponding to the active set, $\mathcal{A}$, enter the model first while the predictor, $\boldsymbol{x}_4$, for the non-active set never enters the model except near $s = 1$, which is the least squares estimate. These results show that the adaptive lasso can be correct in variable selection if an appropriate value of the tuning parameter is selected. In the next section, we will compare some popular tuning parameter selectors and see if this theoretical property of consistent variable selection can be achieved in practice.

Model 1 in Figure 5.3:

We can define the following

$$\mathcal{A} = \{1, 2, 5\} \quad \text{and} \quad \mathcal{A}^c = \{3, 4, 6, 7, 8\}.$$

For this model the ZYZ condition holds. Now we will see if the lasso and adaptive lasso both can do consistent variable selection. From Figure 5.3(e)-(f), we can see that the solution paths of the lasso and adaptive lasso contain the true model but the standard lasso is performing better in the sense that picking up the correct model from the solution path is less challenging as for a wide range of tuning parameter values, s, it sets the non-active predictors to zero and they become non-zero only when s reaches near to 1. As γ increases, the band of s for which the solution path contains the correct model becomes narrower which makes selecting the tuning parameter harder. Moreover, it can be observed that the larger the value of γ, the smaller the value of s is required to shrink the non-active predictors to zero.

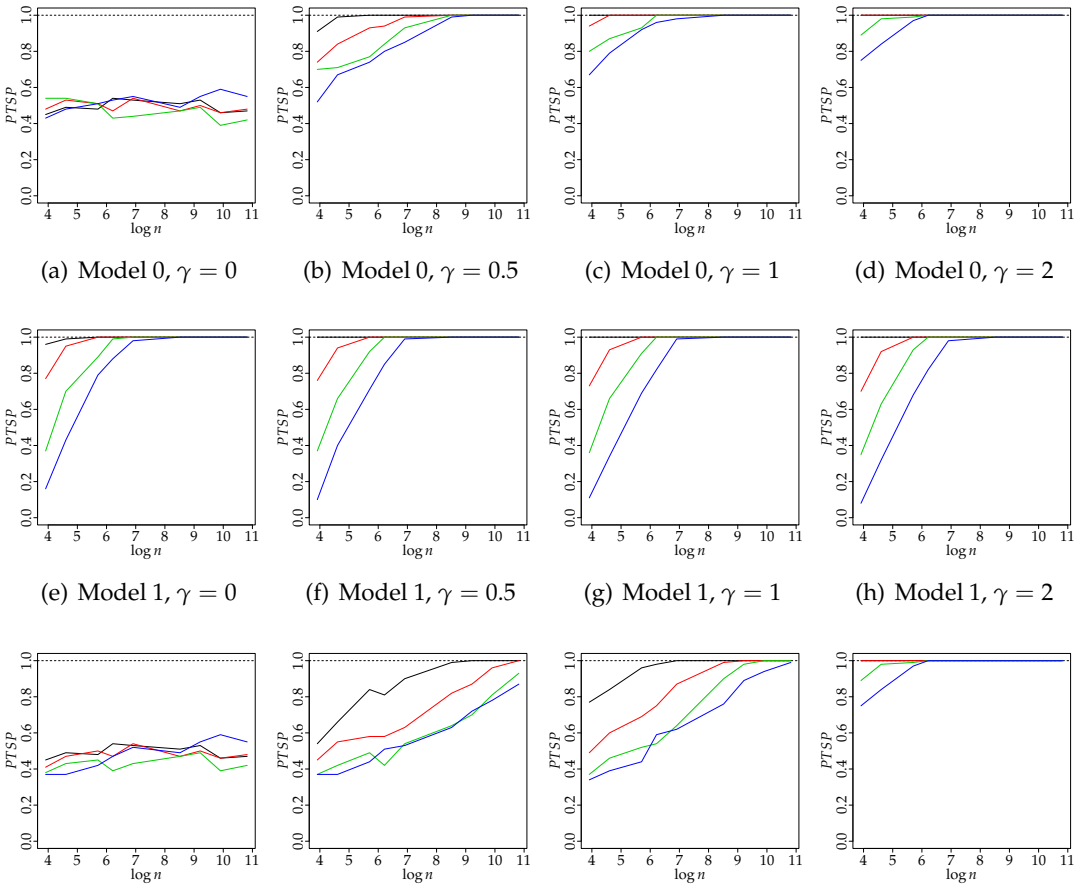
Model 2 in Figure 5.3:

We can define the following

$$\mathcal{A} = \{1, 2, 3\} \quad \text{and} \quad \mathcal{A}^c = \{4\}.$$

This model has the same construction as for Model 0 but the effects are small in their absolute values, so this is more challenging for lasso methods. The results shown in Figure 5.3(i)-(l) lead to the same conclusions as for Model 0 but the solution path shows that relatively small values of s need to be selected for correct variable selection. Also, the adaptive lasso for $\gamma = 0.5$ shows that it can be incorrect unless a moderate value of s is selected.

Having studied the solution paths of the lasso and adaptive lasso, we now have some idea which method can be correct in variable selection. Now in the rest of this section, we will give results on the basis of 100 Monte Carlo runs and will look at some empirical results for the performance measures defined earlier at the start of this section.



(a) Model 0, $\gamma = 0$ (b) Model 0, $\gamma = 0.5$ (c) Model 0, $\gamma = 1$ (d) Model 0, $\gamma = 2$

(e) Model 1, $\gamma = 0$ (f) Model 1, $\gamma = 0.5$ (g) Model 1, $\gamma = 1$ (h) Model 1, $\gamma = 2$

(i) Model 2, $\gamma = 0$ (j) Model 2, $\gamma = 0.5$ (k) Model 2, $\gamma = 1$ (l) Model 2, $\gamma = 2$

Figure 5.4: Probability, based on 100 runs, that solution paths of the lasso ($\gamma = 0$) and adaptive lasso ($\gamma = 0.5, 1,$ and 2) for the three models defined in Section 5.5. Model 0: $\beta_0 = (5.6, 5.6, 5.6, 0)^T$. Model 1: $\beta_0 = (3, 1.5, 0, 0, 2, 0, 0, 0)^T$. Model 2: $\beta_0 = (0.85, 0.85, 0.85, 0)^T$. The error distribution is $\varepsilon_i \sim N(0, \sigma^2)$; see also the caption for Figure 5.3. Key: ■ $\sigma = 1$; ■ $\sigma = 3$; ■ $\sigma = 6$; ■ $\sigma = 9$.

We now consider a selection of sample sizes ranging from $n = 50$ to $n = 50000$, $(50, 100, 300, 500, 1000, 5000, 10000, 20000, 50000)$ to study the performance of these methods for small sizes and also for their asymptotic behaviour. We assume $\varepsilon_i \sim N(0, \sigma^2)$, where $\sigma = 1$,3,6 and 9 are the choices of error standard deviation.

Figure 5.4 gives the plots for the lasso and adaptive lasso showing the empirical probability of containing the true model for each of the three models defined earlier. In these plots, the horizontal axis corresponds to sample size on a logarithmic scale and the vertical axis corresponds to the empirical probability that the true model lies on the solution path.

Model 0 in Figure 5.4:

Figure 5.4(a) shows the empirical probability of containing the true model for the standard lasso, which confirms our earlier finding in the study of the solution path that the lasso cannot be consistent in variable selection for Model 0 as the ZYZ condition fails for this model. We can see that this probability varies between 0.4 and 0.6 and does not converge to 1 even for sample sizes as large as $n = 50000$. The results do not differ much for different choices of error variance.

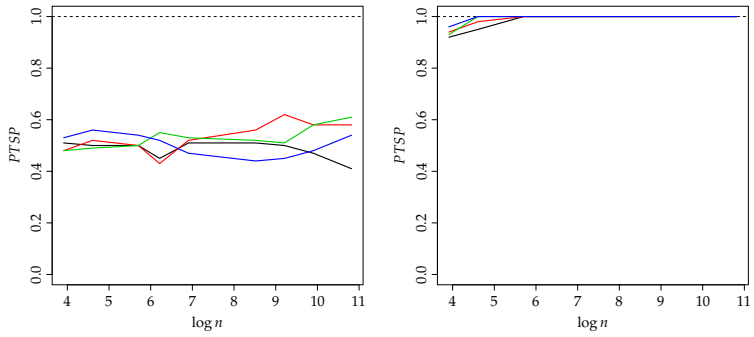
For the adaptive lasso, Figures 5.4(b)-(d), show that the probability is converging to 1 and the larger the value of γ, the smaller the sample size is required to be to get the probability exactly one. This shows that the adaptive lasso can be consistent in variable selection if an appropriate value of the tuning parameter is selected. However, the result that the adaptive lasso is doing well for larger values of γ should be interpreted with caution. We have noticed in our earlier results on solution paths shown in Figure 5.3 that with an increase in γ, the range of values of s which correspond to the true model decreases thereby making it harder for the tuning parameter selector to pick an appropriate value of the tuning parameter. We will discuss this in detail later in this section.

Model 1 in Figure 5.4:

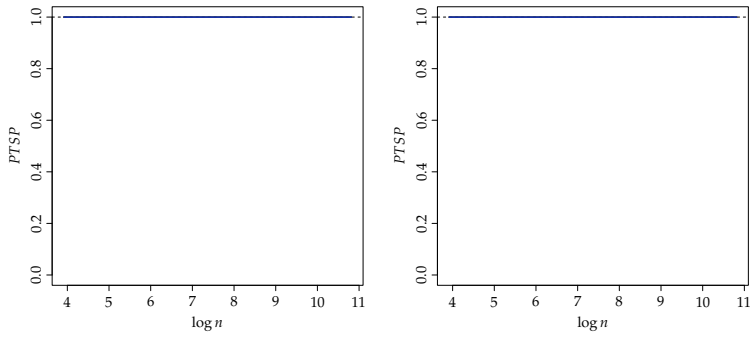
In this case, the lasso and adaptive lasso for all choices of γ does not differ much and the probability for all of them is converging to one. These results in conjunction with the results shown in Figure 5.3 suggest that it is sometimes easier to select the correct value of the tuning parameter for the lasso as compared to the adaptive lasso.

Model 2 in Figure 5.4:

For Model 2, we have small effects and the ZYZ condition also fails. As we noticed earlier from Figure 5.3, this situation becomes more challenging. Now it can be seen from the results shown in Figure 5.4(j),(k), that, in general, the probability for the adaptive lasso when $\gamma = 0.5$ and 1 converges to one at a rate slower than in the case of Model 0. But the results for the adaptive lasso when $\gamma = 2$ do not differ much for the two models.



(a) Model 0, $\gamma = 0$ (b) Model 0, $\gamma = 0.5$



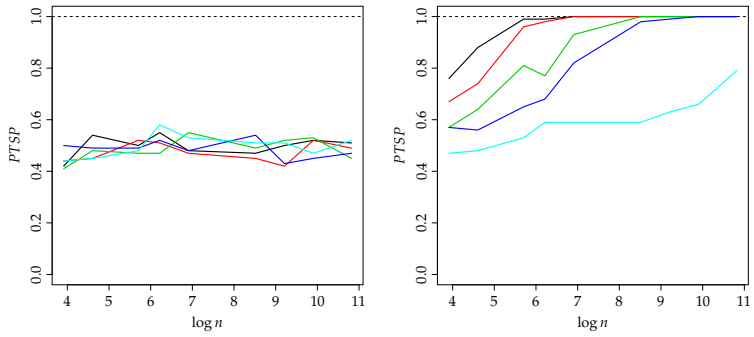
(c) Model 0, $\gamma = 1$ (d) Model 0, $\gamma = 2$

Figure 5.5: PTSP: Probability, based on 100 runs, that solution path of the lasso ($\gamma = 0$) and adaptive lasso (($\gamma = 0.5, 1,$ and 2) for Model 0 defined in Section 5.5. Model 0: $\beta_0 = (5.6, 5.6, 5.6, 0)^T$. The error distribution is $\varepsilon_i \sim t_\nu$. Key: $\nu = 5$; $\nu = 10$; $\nu = 20$; $\nu = \infty$

Before we give results for other performance measures based on the tuning parameter selector, we give some results for non-Gaussian errors. First we assume $\varepsilon_i \sim t_\nu$, where t_ν represents a Student's t-distribution with ν degrees of freedom. We consider $\nu = 5$, 10, 20 and ∞. The smaller the value of ν is, the heavier the tails of the error distribution and $\nu = \infty$ corresponds to the normal distribution.

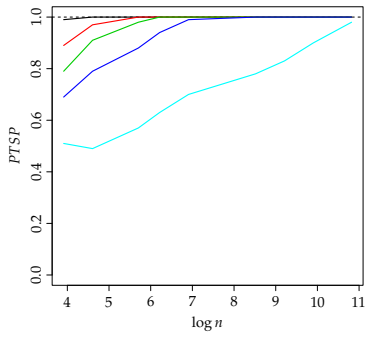
As expected, results in Figure 5.5 show that, due to failure of the ZYZ condition, the lasso is not consistent and, as is the case for Gaussian errors, the probability that the true model lies on the solution path is not converging to 1. For the adaptive lasso, the results match with Gaussian errors with $\sigma = 1$. Obviously, scaling the error with a factor, say $a > 1$, will result in poor performance as seen in the Gaussian case.

Now consider $\varepsilon_i \sim \chi^2_\nu$, where $\nu = 1$, 2, 5, 10 and 100. We know that for small ν the χ^2_ν distribution is skewed to the right, but as ν increases the χ^2_ν distribution approaches

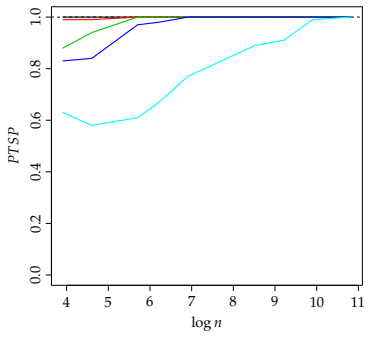


(a) Model 0, $\gamma = 0$

(b) Model 0, $\gamma = 0.5$



(c) Model 0, $\gamma = 1$



(d) Model 0, $\gamma = 2$

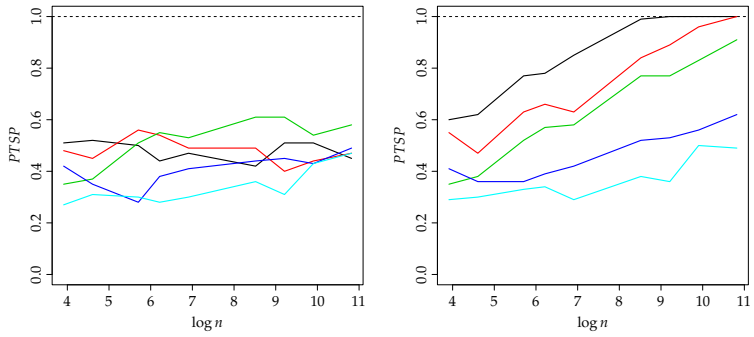
Figure 5.6: PTSP: Probability, based on 100 runs, that solution path of the lasso and the adaptive lasso ($(\gamma = 0.5, 1, \text{and } 2)$ for Model 0 defined in Section 5.5. Model 0: $\beta_0 = (5.6, 5.6, 5.6, 0)^T$. The error distribution is $\varepsilon_i \sim \chi^2_\nu$. Key: $\nu = 1$; $\nu = 2$; $\nu = 5$; $\nu = 10$, $\nu = 100$.

symmetry but the variance increases.

Figure 5.6 gives the probability for the lasso and adaptive lasso of containing the true model on their solution paths. These results confirm the earlier findings that larger error variance makes variable selection more challenging. Though for smaller choices of ν, e.g. $\nu = 1$, the chi-square distribution is extremely skewed but the adaptive lasso is still consistent in variable selection.

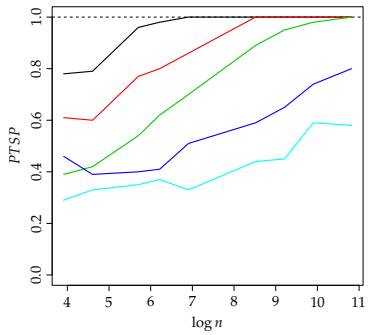
Finally, we consider the lognormal case in which $\varepsilon_i \sim e^{\sigma z_i}$, where $z_i \sim N(0, 1)$. We consider the choices $\sigma = 1$ through 5.

Figure 5.7 shows the plots of empirical probability of doing consistent variable selection of the lasso and adaptive lasso when the error term has a log-normal distribution. For the log-normal distribution, as σ increases the distribution moves away from symmetry and the variance is also increased. Thus, as our previous findings suggest,

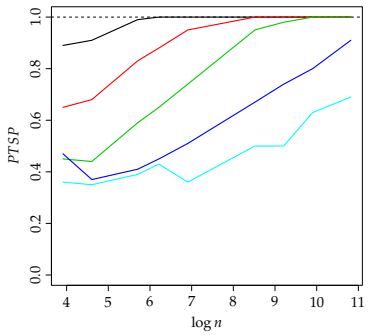


(a) Model 0, $\gamma = 0$

(b) Model 0, $\gamma = 0.5$



(c) Model 0, $\gamma = 1$



(d) Model 0, $\gamma = 2$

Figure 5.7: PTSP: Probability, based on 100 runs, that solution path of the lasso and adaptive lasso ($(\gamma = 0.5, 1, \text{and } 2)$ for Model 0 defined in Section 5.5. Model 0: $\beta_0 = (5.6, 5.6, 5.6, 0)^T$. The error distribution is $\varepsilon_i = \exp(\sigma z_i)$, where $z_i \sim N(0,1)$. Key: $\sigma = 1$; $\sigma = 2$; $\sigma = 3$; $\sigma = 4$; $\sigma = 5$.

we can expect, the log-normal distribution with larger values of σ to be more challenging for the lasso methods applied to variable selection, and this fact is evident in these results.

In the next section, we will compare the tuning parameter selectors and will also see if oracle properties of lasso-type methods can be achieved in practice.

5.5.2 Estimation of the Tuning Parameter

As we have discussed earlier in Section 5.5.1, when the ZYZ condition fails, the lasso is not consistent in variable selection but the adaptive lasso is. In cases where the ZYZ condition holds, the lasso and adaptive lasso theoretically do consistent variable selection but consistency can only be achieved if we have a method which can select an appropriate value of the tuning parameter. If the tuning parameter is not appropriately

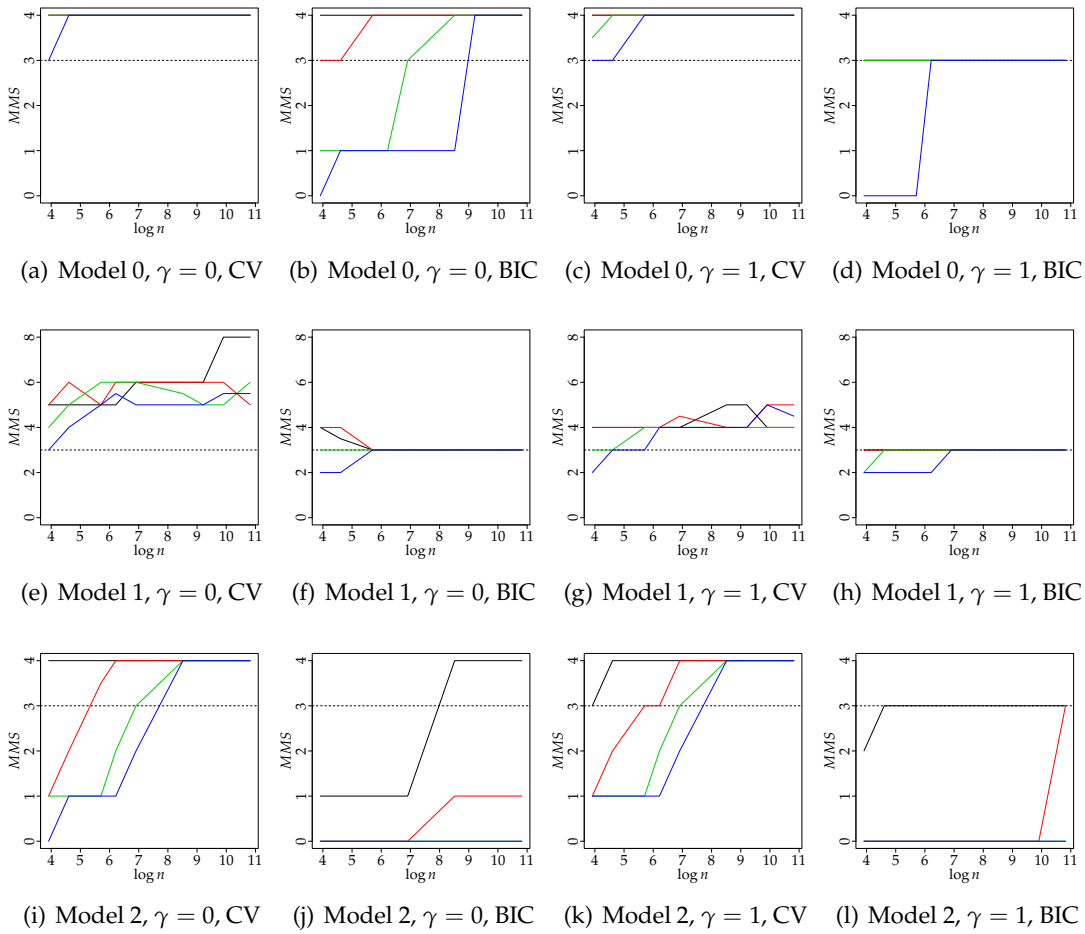
selected, even though the solution path contains the true model, it is likely we will select an incorrect model. Tibshirani (1996) also noted in a simulation example that though the lasso solution path contains the true model, only for a small fraction of possible choices of tuning parameter $s \in [0, 1]$ the lasso does pick the correct model.

The discussion above shows the importance of tuning parameter selection. In this section, we compare two methods used for tuning parameter selection: (1) k-fold cross-validation and (2) the Bayesian information criterion (BIC). These methods are defined in Section 5.4. We use 5-fold cross-validation as suggested by Zou (2006). In their numerical results, Fan and Li (2001) have found that 5-fold cross-validation and generalised cross-validation perform similarly. For the BIC, defined in (5.4.1), we have used several values for C_n, e.g. $C_n = 1$, 5, and 10. We noticed in our numerical study that all of these considered choices of C_n fail to work as n increases. This may be due to failure of Wang and Leng (2009, condition 4), for these fixed choices of C_n, that requires a condition on the size of non-zero parameters and that of C_n. We also observe from our numerical results that each of the considered fixed choice of C_n works up to a certain sample size and then the results drop down in performance. We notice that the larger the sample size the larger the value of C_n is required and vice versa.

These results lead us to the conclusion that the performance of the BIC approach is highly dependent on the value of C_n and we need a value of C_n which increases at a certain rate with n. The results for these fixed values of C_n intuitively guided us to the use of $C_n = \sqrt{n}/p$, where n is the sample size and p is the number of predictors. In the rest of the section, we give the results for this choice of C_n.

As we have discussed earlier, smaller values of the tuning parameter, s, lead to a greater amount of shrinkage and this results in setting more of the estimates to exactly zero. First we look at the median model size for the value of s chosen by each of the tuning parameter selectors viz cross-validation and the BIC.

Figure 5.8, shows the plots of median model size. The measure model size is defined earlier at the start of Section 5.5. To illustrate the results, we give the results for the lasso ($\gamma = 0$) and adaptive lasso ($\gamma = 1$). In this figure, the dashed horizontal line corresponds to the true model size while different coloured lines correspond to the



(a) Model 0, $\gamma = 0$, CV (b) Model 0, $\gamma = 0$, BIC (c) Model 0, $\gamma = 1$, CV (d) Model 0, $\gamma = 1$, BIC

(e) Model 1, $\gamma = 0$, CV (f) Model 1, $\gamma = 0$, BIC (g) Model 1, $\gamma = 1$, CV (h) Model 1, $\gamma = 1$, BIC

(i) Model 2, $\gamma = 0$, CV (j) Model 2, $\gamma = 0$, BIC (k) Model 2, $\gamma = 1$, CV (l) Model 2, $\gamma = 1$, BIC

Figure 5.8: MMS: Median model size, based on 100 Monte Carlo runs, for the lasso ($\gamma = 0$) and adaptive lasso ($\gamma = 1$) using CV (5-fold cross-validation) and BIC ($C_n = \sqrt{n}/p$) for tuning parameter selection for the three models defined in Section 5.5. Model 0: $\beta_0 = (5.6, 5.6, 5.6, 0)^T$. Model 1: $\beta_0 = (3, 1.5, 0, 0, 2, 0, 0, 0)^T$. Model 2: $\beta_0 = (0.85, 0.85, 0.85, 0)^T$. The error distribution is $\varepsilon_i \sim N(0, \sigma^2)$; see also the caption for Figure 5.3. Key: ■ $\sigma = 1$; ■ $\sigma = 3$; ■ $\sigma = 6$; ■ $\sigma = 9$.

model sizes for the different choices of error variance.

Model 0 in Figure 5.8:

Figure 5.8(a),(b) give the median model size for the lasso for cross-validation and the BIC respectively. It can be noticed that the lasso cross-validation leads to $MMS = 4$, which is the full model. If we look at the model size plot for the lasso with the BIC, we can see even the BIC does not lead to the true model size. If we read this result in relation to the solution path of the lasso for Model 0 as shown in Figure 5.3(a), we can see for the lasso the non-active predictor enters the estimated model first and for most of the part of the lasso solution path, it remains are non-zero.

As we have already concluded in the previous discussion on the solution path concerning Figure 5.3, the adaptive lasso picks up the predictors in the right order, so an appropriate choice of tuning parameter will lead to the true model size. Now we can see from Figure 5.8(c),(d) that cross-validation is again resulting in overfitted models, while with the BIC we get the true model size except for small choices of sample size. Interestingly, the BIC for the adaptive lasso does not produce any overfitted model.

Model 1 in Figure 5.8:

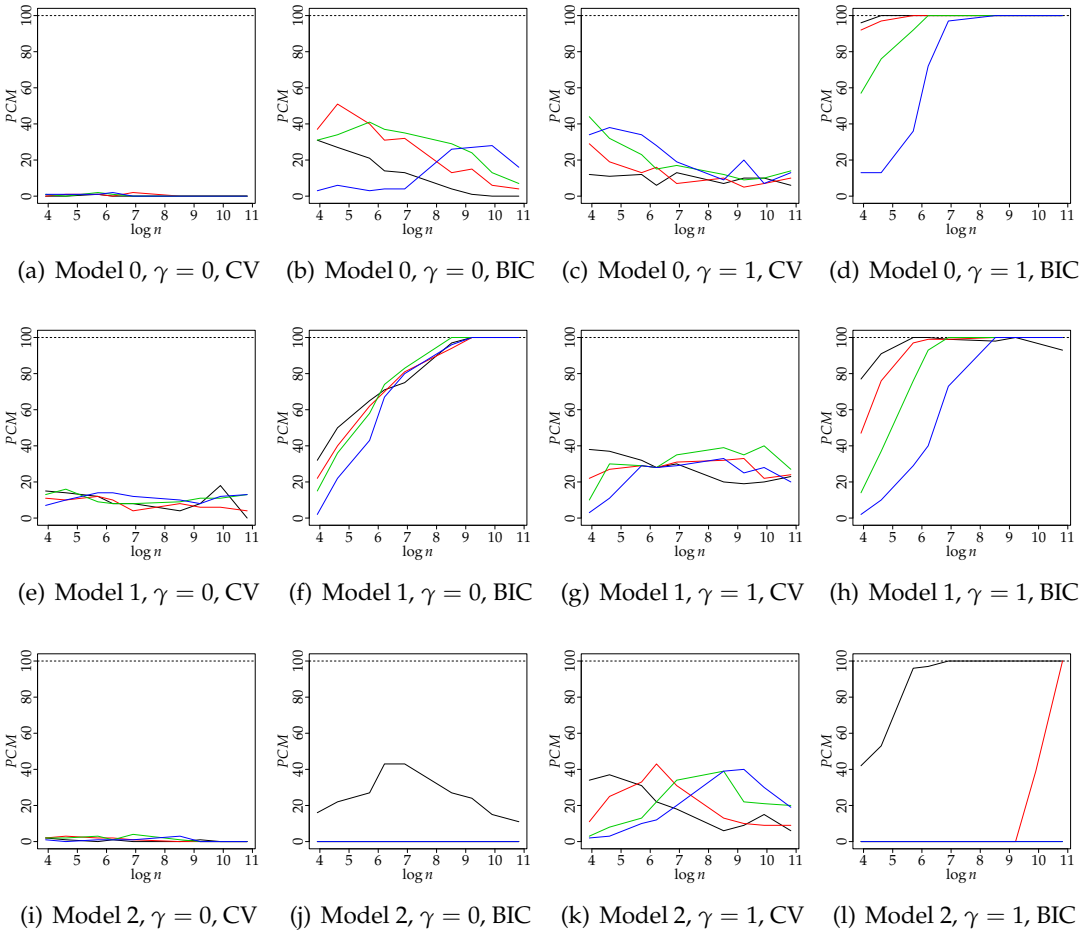
This is the model for which the ZYZ condition holds and we have seen earlier that both the lasso and adaptive lasso can do consistent variable selection for this model. We have also noticed that picking up the correct model from the lasso solution path is often easier than for the adaptive lasso. Figure 5.8(e)-(h) confirm the earlier findings that both the lasso and adaptive lasso tuning parameters selected by cross-validation result in overfitted models, while the BIC results converge to true model size. Note that for the lasso, even for larger error variance, the size of the fitted model converges to the true model size for smaller sample size.

Model 2 in Figure 5.8:

This model contains small effects and also the ZYZ condition does not hold. These two facts make the variable selection very challenging in this case. If we compare these results with those of Model 0, we notice that in this case we are seeing more underfitted models, especially for the larger choices of error variance. For the adaptive lasso with smaller error variance, $\sigma^2 = 1$, with the BIC we are able to achieve the true model but not for the larger choices of error variance.

From the discussion on model size, we note that cross-validation is not the right method to select the tuning parameter and results in some incorrect shrinkage to zero for the active predictors. We have also observed that the BIC is leading to the true model size while cross-validation not. Now we further confirm these results by looking at the percent of correct models identified by these tuning parameter selectors. The measure CM is defined in (5.5.1). PCM is the percentage of times we end up with correct model in N Monte Carlo runs. This measure help us to confirm, when we have

achieved the true model size, whether the active predictors have been selected in the model.



(a) Model 0, $\gamma = 0$, CV (b) Model 0, $\gamma = 0$, BIC (c) Model 0, $\gamma = 1$, CV (d) Model 0, $\gamma = 1$, BIC

(e) Model 1, $\gamma = 0$, CV (f) Model 1, $\gamma = 0$, BIC (g) Model 1, $\gamma = 1$, CV (h) Model 1, $\gamma = 1$, BIC

(i) Model 2, $\gamma = 0$, CV (j) Model 2, $\gamma = 0$, BIC (k) Model 2, $\gamma = 1$, CV (l) Model 2, $\gamma = 1$, BIC

Figure 5.9: PCM: Percentage of correct models identified, based on 100 Monte Carlo runs, for the lasso ($\gamma = 0$) and adaptive lasso ($\gamma = 1$) using CV (5-fold cross-validation) and the BIC ($C_n = \sqrt{n}/p$) for tuning parameter selection for the three models defined in Section 5.5. Model 0: $\beta_0 = (5.6, 5.6, 5.6, 0)^T$. Model 1: $\beta_0 = (3, 1.5, 0, 0, 2, 0, 0, 0)^T$. Model 2: $\beta_0 = (0.85, 0.85, 0.85, 0)^T$. The error distribution is $\varepsilon_i \sim N(0, \sigma^2)$; see also the caption for Figure 5.3. Key: $\sigma = 1$; $\sigma = 3$; $\sigma = 6$; $\sigma = 9$.

Figure 5.9 show the plots of percentage of correct model identified. We give the results for the lasso ($\gamma = 0$) and adaptive lasso ($\gamma = 1$) when the tuning parameter is selected by 5-fold cross-validation and the BIC ($C_n = \sqrt{n}/p$). The horizontal axis corresponds to the sample size on a logarithmic scale and the vertical axis corresponds to the percent of correct models. Ideally, these plots should match with the corresponding plots of probability of containing the true model on the solution path shown in Figure 5.4. For example, Figure 5.4(c) shows that for the adaptive lasso ($\gamma = 1$), the probability of containing the true model on the solution path converges to one for each choice

of the error variance. Now if we compare this with Figure 5.9(c), which shows the percentage of correct models identified using cross-validation, we can see this percentage is very low and even decreases to zero as sample size increases. In contrast, a comparison of Figure 5.4(c) with Figure 5.9(d) shows that for the BIC we can do consistent variable selection with percentage approaching 1.

Similar kinds of conclusions can be made for the other two models. It is found that cross-validation fails to select the appropriate value of the tuning parameter thus resulting in the selection of an incorrect model from the lasso and adaptive lasso solution path.

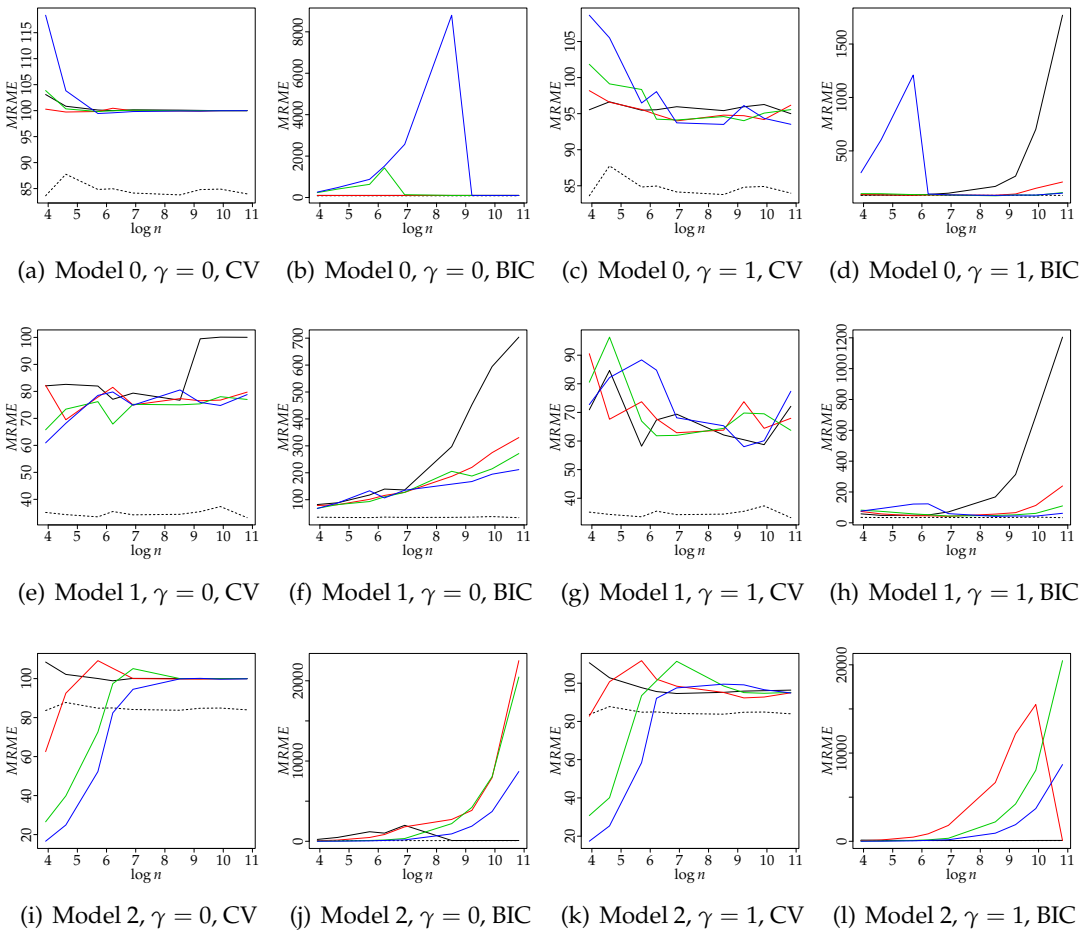
From the above discussion we note that the oracle property of consistent variable selection can be achieved for the lasso if the ZYZ condition holds, while the adaptive lasso can do the consistent variable selection even if the ZYZ condition does not hold in the standard lasso. We also found that an appropriate value of the tuning parameter can be selected if a tuning parameter selector based on the BIC is used.

In the following paragraphs we will give some results on the performance measure Median of Relative Model error ($MRME$) and will compare it with corresponding Oracle Relative Model Error ($ORME$). $MRME$ and $ORME$ are defined earlier at the start of Section 5.5 along with the definition of Model error (ME) given in (5.5.2).

Figure 5.10 gives the plots of MRME for the lasso and adaptive lasso. These plots are for the models corresponding to the value of the tuning parameter selected by cross-validation and the BIC. The dotted line is the ORME. In general, we can see that the MRME for cross-validation is lower than for the BIC. Moreover, we notice, in the case of the BIC, the MRME for the adaptive lasso is lower than the lasso when the ZYZ condition fails.

Now we give some results for the distribution of the tuning parameter selected by cross-validation and the BIC.

Figure 5.11 gives for Model 0 boxplots for the distribution of the tuning parameter, s, selected by cross-validation and the BIC. It can be noticed that for the lasso with error standard deviation, $\sigma = 1$, the distribution of s is centered around 1, with a very low range. This means that for Model 0, the lasso with cross-validation and BIC will pro-



(a) Model 0, $\gamma = 0$, CV (b) Model 0, $\gamma = 0$, BIC (c) Model 0, $\gamma = 1$, CV (d) Model 0, $\gamma = 1$, BIC

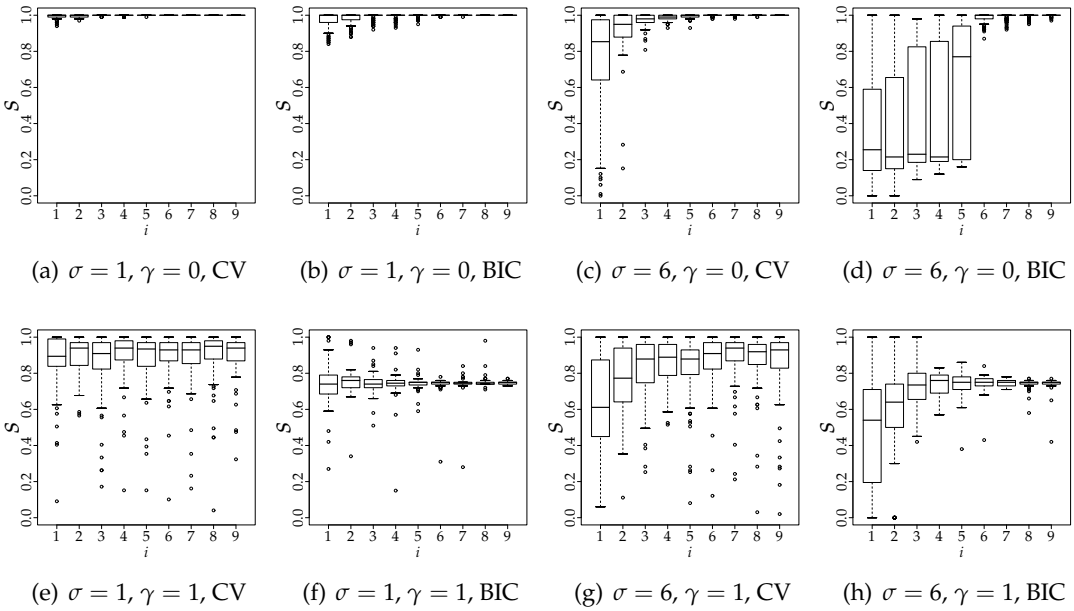
(e) Model 1, $\gamma = 0$, CV (f) Model 1, $\gamma = 0$, BIC (g) Model 1, $\gamma = 1$, CV (h) Model 1, $\gamma = 1$, BIC

(i) Model 2, $\gamma = 0$, CV (j) Model 2, $\gamma = 0$, BIC (k) Model 2, $\gamma = 1$, CV (l) Model 2, $\gamma = 1$, BIC

Figure 5.10: MRME: Median of relative model error, based on 100 Monte Carlo runs, for the lasso ($\gamma = 0$) and adaptive lasso ($\gamma = 1$) using CV (5-fold cross-validation) and the BIC ($C_n = \sqrt{n}/p$) for tuning parameter selection for the three models defined in Section 5.5. Model 0: $\beta_0 = (5.6, 5.6, 5.6, 0)^T$. Model 1: $\beta_0 = (3, 1.5, 0, 0, 2, 0, 0, 0)^T$. Model 2: $\beta_0 = (0.85, 0.85, 0.85, 0)^T$. The error distribution is $\varepsilon_i \sim N(0, \sigma^2)$; see also the caption for Figure 5.3. Key: $\sigma = 1$; $\sigma = 3$; $\sigma = 6$; $\sigma = 9$.

duce least squares estimates. This result is also supported by our earlier finding from the lasso solution path and other performance measures based on tuning parameter selectors. As the error variance increases, we can see the tuning parameter selectors, especially the BIC, start picking up tuning parameter values away from 1 and close to 0, which results in some shrinkage for the lasso estimates. The result of this change can be seen in Figure 5.9(b), where we can clearly see higher percentage of correct models for $\sigma = 6$ as compared to $\sigma = 1$.

For the adaptive lasso, cross-validation is selecting a value of s, which is further away from 1 as compared to the lasso. Due to this, a relatively higher percentage of correct models for the adaptive lasso can be seen in Figure 5.9(c), as compared to Figure



(a) $\sigma = 1, \gamma = 0$, CV (b) $\sigma = 1, \gamma = 0$, BIC (c) $\sigma = 6, \gamma = 0$, CV (d) $\sigma = 6, \gamma = 0$, BIC

(e) $\sigma = 1, \gamma = 1$, CV (f) $\sigma = 1, \gamma = 1$, BIC (g) $\sigma = 6, \gamma = 1$, CV (h) $\sigma = 6, \gamma = 1$, BIC

Figure 5.11: Box plots for tuning parameters for the lasso ($\gamma = 0$) and adaptive lasso ($\gamma = 1$) using CV (5-fold cross-validation) and the BIC ($C_n = \sqrt{n}/p$) for tuning parameter selection for the model defined in Section 5.5. Model 0: $\beta_0 = (5.6, 5.6, 5.6, 0)^T$. The error distribution is $\varepsilon_i \sim N(0, \sigma^2)$ where $\sigma = 1$ and 6. Considered choices of sample size are $n_i = (50, 100, 300, 500, 1000, 5000, 10000, 20000, 50000)$.

5.9(a). But it can be clearly noticed that the BIC is selecting a value of s, which is smaller than that selected by cross-validation and thus shrinking some of the estimates exactly to zero. So the advantage of selecting an appropriate value of s by the BIC can be clearly seen in Figure 5.9(d).

5.6 Conclusion

In this chapter we have compared the performance of the lasso and adaptive lasso. Our results show that the ZYZ condition is an important condition for consistent variable selection for the lasso and adaptive lasso. We have seen that the lasso can be consistent in variable selection when the ZYZ condition holds provided that an appropriate value of the tuning parameter is selected. It should be noted that the ZYZ condition always holds for the adaptive lasso due to the use of adaptive weights and thus it showed consistent variable selection in all the cases.

The numerical results suggest that cross-validation is not a reliable method especially if the primary objective is variable selection. In all situations considered, our

results suggest that both the lasso and adaptive lasso using cross-validation as a tuning parameter selector leads to inconsistent variable selection. In contrast, the BIC has shown its capability to choose a value for the tuning parameter which correctly shrinks the coefficients of non-active predictors to zero.

CHAPTER 6

Lasso Methods for Time Series Models

6.1 Introduction

Time series models are of importance in many fields. More recently, there has been growing interest in multivariate as opposed to univariate time series. Serial dependence of univariate time series provides an important basis for constructing time series models. In multivariate time series, in addition to the serial dependence of each component of the time series, the interdependence between different component time series needs to be accounted for in model building.

The theoretical properties of the lasso (Tibshirani, 1996) and adaptive lasso (Zou, 2006) are potentially appealing for time series models. However, although lasso-type methods are widely studied and discussed for the regression problem, there is relatively little literature available on the application of lasso-type methods in the time series context. Perhaps this is because it is not clear at the outset how best to use lasso-type methods in the multivariate time series setting. Most of the applications of lasso-type methods in the time series context are in the field of network identification, see e.g. Fujita et al. (2007). Haufe et al. (2008) discussed the application of shrinkage methods to estimate sparse vector autoregressive models in the context of causal discovery. They suggested a method based on the group lasso and compared its performance with ridge regression and the lasso. For details see Section 1.1.

The success of shrinkage methods for regression models leads us to explore the use of these methods for multivariate time series. When modelling real data we often focus on sparse models, especially in high dimensional settings. Although we use VAR(1) i.e. vector autoregressive model with lag 1, as the basis for our approach, it is important point to note that in our theoretical results we have assumed only a stationarity condition for the time series model. The VAR(1) model structure is used for convenience but our results hold for much more general classes of stationary and non-stationary time series models e.g. seasonal and non-seasonal vector ARMA models.

In Chapter 5, we studied properties of the lasso and adaptive lasso for linear regression models. In this chapter, we study the lasso and adaptive lasso in multivariate time series problems. The oracle properties of the adaptive lasso are proved for multivariate time series models under a stationarity condition. We also compare the lasso and adaptive lasso variable selection procedures for different models.

The structure of this chapter is follows: Section 6.2 gives some important definitions. Least squares estimates for VAR(1) are obtained in Section 6.3. In Section 6.4 a theorem which presents a necessary condition, similar to the ZYZ condition discussed in Chapter 5 for consistent variable selection using the lasso is stated and proved. In Section 6.5 a statement and proof of the oracle property of the adaptive lasso for multivariate time series models is given. Finally, in Section 6.6 we look at some examples of the application of lasso-type methods to time series models.

6.2 Some Definitions

6.2.1 Centred Multivariate Time Series

Consider a p-dimensional multivariate time series $\{\boldsymbol{y}_t\}_{t=1}^n$ where $\boldsymbol{y}_t = \left(y_{1t}, \ldots, y_{pt}\right)^T$. Consider a VAR(1) model which can be defined as

$$\boldsymbol{y}_t = \boldsymbol{\beta}_0 + \boldsymbol{B}\boldsymbol{y}_{t-1} + \boldsymbol{\varepsilon}_t, \quad t = 2, \ldots, n, \tag{6.2.1}$$

where $\boldsymbol{\beta}_0$ is a constant vector, $\boldsymbol{B} = [b_{ij}]_{i,j=1}^p = (\boldsymbol{b}_1, \ldots, \boldsymbol{b}_p)^T$ is $p \times p$ coefficient matrix and $\{\boldsymbol{\varepsilon}_t\}$ is a white noise process with zero mean and covariance matrix $\boldsymbol{\Sigma}_\varepsilon$.

Let us define the active sets i.e. sets of non-zero coefficients in model (6.2.1), as $\mathcal{A} = \{(i,j) : b_{ij} \neq 0\} \subseteq \{1,\ldots,p\} \times \{1,\ldots,p\}$ and, moreover, suppose that for each $i = 1,\ldots,p$, we define $\mathcal{A}(i) = \{j : b_{ij} \neq 0\} \subseteq \{1,\ldots,p\}$. Then $\mathcal{A} = \{(i,j) : j \in \mathcal{A}(i), i = 1,\ldots,p\}$.

The L_1 penalty lasso estimator, $\hat{\boldsymbol{\beta}}^*$, of $\boldsymbol{\beta} = (\boldsymbol{\beta}_0, \boldsymbol{B})$ can be defined as

$$\hat{\boldsymbol{\beta}}^* = \arg\min \left[\sum_{t=1}^{n-1} \sum_{i=1}^{p} (y_{t+1,i} - \beta_{0i} - \boldsymbol{b}_i^T \boldsymbol{y}_t)^2 + \lambda_n \sum_{i,j=1}^{p} |b_{ij}| \right],$$

where λ_n varies with n. It is important to note that components of $\boldsymbol{\beta}_0$ are not penalized. Let us now write penalized sum of squares as

$$L(\boldsymbol{\beta}_0, \boldsymbol{B}) = \sum_{t=1}^{n-1} \|\boldsymbol{y}_{t+1} - \boldsymbol{\beta}_0 - \boldsymbol{B}\boldsymbol{y}_t\|^2 + \lambda_n \sum_{i,j=1}^{p} |b_{ij}|.$$

Setting

$$\frac{\partial L}{\partial \boldsymbol{\beta}_0} = \mathbf{0}_p,$$

gives

$$\sum_{t=1}^{n-1} (\boldsymbol{y}_{t+1} - \boldsymbol{\beta}_0 - \boldsymbol{B}\boldsymbol{y}_t) = \mathbf{0}_p,$$

where $\mathbf{0}_p$ is the p-vector of zeros. For given $\boldsymbol{B}$, this yields

$$\hat{\boldsymbol{\beta}}_0^*(\boldsymbol{B}) = \bar{\boldsymbol{y}}_2 - \boldsymbol{B}\bar{\boldsymbol{y}}_1,$$

where $\bar{\boldsymbol{y}}_2 = (n-1)^{-1} \sum_{t=1}^{n-1} \boldsymbol{y}_{t+1}$ is the mean vector of last $n-1$ observations and $\bar{\boldsymbol{y}}_1 = (n-1)^{-1} \sum_{t=1}^{n-1} \boldsymbol{y}_t$ is the mean vector of first $n-1$ observations. Now we can redefine the penalized sum of squares evaluated at $\hat{\boldsymbol{\beta}}_0^{\,*}(\boldsymbol{B})$ as

$$L(\hat{\boldsymbol{\beta}}_0^*(\boldsymbol{B}), \boldsymbol{B}) = \sum_{t=1}^{n-1} \|(\boldsymbol{y}_{t+1} - \bar{\boldsymbol{y}}_2) - \boldsymbol{B}(\boldsymbol{y}_t - \bar{\boldsymbol{y}}_1)\|^2 + \lambda_n \sum_{i,j=1}^{p} |b_{ij}|.$$

Thus, working with the centerings shown above, we can omit $\boldsymbol{\beta}_0$ and without loss of generality we can work with only autoregressive parameter matrix $\boldsymbol{B}$. So the reduced

form of model (6.2.1) for centered time series, $\{\boldsymbol{y}_t\}$ can be defined as

$$\boldsymbol{y}_t = \boldsymbol{B}\boldsymbol{y}_{t-1} + \boldsymbol{\varepsilon}_t, \quad t = 2, \ldots, n, \tag{6.2.2}$$

where $\boldsymbol{B} = (\boldsymbol{b}_1, \ldots, \boldsymbol{b}_p)^T$ is the matrix of autoregressive coefficients. From this point onwards, we will work only with centered series of the form (6.2.2).

6.2.2 Karesh-Kuhn-Tucker Optimality Conditions

In optimization theory, the question of whether a given stationary point is a local minimum of the objective function often arises. The Karesh-Kuhn-Tucker (KKT) optimality conditions may be used to address this question. Here we briefly define the KKT conditions in the lasso context as there is a non-standard aspect.

Suppose we have a nonlinear programming problem:

$$\text{minimize } f(\boldsymbol{x})$$

subject to the inequality constraint

$$g_j(\boldsymbol{x}) \geq 0, \quad \text{for } j = 1, \ldots, J.$$

According to the KKT conditions for an inequality constrained problem, a point $\boldsymbol{x}^\star$ is a local minimum if a set of non-negative λ_j's may be found such that

$$\nabla f(\boldsymbol{x}^\star) - \sum_{j=1}^{J} \lambda_j \nabla g_j(\boldsymbol{x}^\star) = 0. \tag{6.2.3}$$

In the lasso context it is necessary to consider modified KKT conditions because of the non-differentiability of the penalty term when coefficients b_{ij} pass through zero. Specifically, consider the partial derivatives corresponding to components which are exactly zero at the optimum. The modified KKT condition for these partial derivatives is that they change sign at the optimum. We will make use of these modified KKT conditions in the proof of Theorems 6.4.1 and 6.5.1. See the example in Section 5.2.2 for further insight.

In general, KKT conditions are necessary conditions but not sufficient. To prove the sufficiency of KKT conditions some further restrictions are required. For more details on KKT conditions see e.g. Nocedal and Wright (1999).

6.3 Least Squares Estimates of the Multivariate Time Series

Let $\boldsymbol{y}_1, \ldots, \boldsymbol{y}_n$ be a random sample from a centered stationary p-dimensional multivariate time series $\{\boldsymbol{y}_t\}$ defined in (6.2.2). Assume $Cov(\boldsymbol{y}_t) = \boldsymbol{C}$ and $Cov(\boldsymbol{y}_{t+1}, \boldsymbol{y}_t) = \boldsymbol{D}$.

Let us first obtain the least squares estimates for the vector autoregressive model, VAR(1). Note that we are not going to assume that the VAR(1) model is the true model. We work with a general stationary sequence $\{\boldsymbol{y}_t\}_{t\geq 1}$, which is only required to satisfy mild conditions which are stated later. The model sum of squares of residuals can be defined as

$$\mathcal{M}(\boldsymbol{B}) = \sum_{t=1}^{n-1} ||\boldsymbol{y}_{t+1} - \boldsymbol{B}\boldsymbol{y}_t||^2 = \sum_{t=1}^{n-1}\sum_{i=1}^{p}(y_{t+1,i} - \boldsymbol{b}_i^T\boldsymbol{y}_t)^2 \tag{6.3.1}$$

over $\boldsymbol{b}_1, \ldots, \boldsymbol{b}_p$. Differentiating (6.3.1) with respect to $\boldsymbol{b}_i$ and equating to zero gives

$$2\sum_{t=1}^{n-1} \boldsymbol{y}_t \left(y_{t+1,i} - \hat{\boldsymbol{b}}_i^T\boldsymbol{y}_t\right) = \boldsymbol{0}_p.$$

Dividing through by $(n-1)$ and on simplifying we obtain

$$\hat{\boldsymbol{b}}_i = \left(\frac{1}{n-1}\sum_{t=1}^{n-1} \boldsymbol{y}_t\boldsymbol{y}_t^T\right)^{-1}\left(\frac{1}{n-1}\sum_{t=1}^{n-1} \boldsymbol{y}_t y_{t+1,i}\right)$$

Therefore,

$$\begin{aligned}\hat{\boldsymbol{B}} &= \left[\hat{\boldsymbol{b}}_1, \ldots, \hat{\boldsymbol{b}}_p\right]^T \\ &= \left[\frac{1}{n-1}\sum_{t=1}^{n-1} \boldsymbol{y}_{t+1}\boldsymbol{y}_t^T\right]\left[\frac{1}{n-1}\sum_{t=1}^{n-1} \boldsymbol{y}_t\boldsymbol{y}_t^T\right]^{-1} \\ &= \boldsymbol{D}_n\boldsymbol{C}_n^{-1},\end{aligned}$$

where $\boldsymbol{C}_n$ and $\boldsymbol{D}_n$ are sample analogues of $\boldsymbol{C} = Cov(\boldsymbol{y}_t)$ and $\boldsymbol{D} = Cov(\boldsymbol{y}_{t+1}, \boldsymbol{y}_t)$ defined

by

$$\boldsymbol{C}_n = \frac{1}{n-1}\sum_{t=1}^{n-1} \boldsymbol{y}_t \boldsymbol{y}_t^T \text{ and } \boldsymbol{D}_n = \frac{1}{n-1}\sum_{t=1}^{n-1} \boldsymbol{y}_{t+1} \boldsymbol{y}_t^T. \tag{6.3.2}$$

Let us now consider least squares estimate of $\boldsymbol{B}$ in a submodel consisting of only non-zero autoregressive coefficients. Following through the previous calculation, we can write the least squares estimates for the submodel $\mathcal{A}$ as

$$\hat{\boldsymbol{b}}_{i,\mathcal{A}(i)} = \left[\left\{ \boldsymbol{C}_n^{-1} \left(\frac{1}{n-1}\sum_{t=1}^{n-1} \boldsymbol{y}_{t,\mathcal{A}(i)} y_{t+1,i} \right) \right\}_j : j \in \mathcal{A}(i) \right]^T,$$

where $[\boldsymbol{a}]_j$ is the jth component of a vector $\boldsymbol{a}$.

Using the results of Hsu et al. (2008), we can write the model (6.2.2) in the regression form

$$\boldsymbol{y} \equiv \boldsymbol{Z}\boldsymbol{\beta} + \boldsymbol{E}, \tag{6.3.3}$$

where

$$\boldsymbol{\beta} = vec(\boldsymbol{B}) \tag{6.3.4}$$

is $(p^2 \times 1)$, $\boldsymbol{y} = vec(\boldsymbol{y}_2, \ldots, \boldsymbol{y}_n)$ is $(p(n-1) \times 1)$, $\boldsymbol{E} = vec(\boldsymbol{\varepsilon}_2, \ldots, \boldsymbol{\varepsilon}_n)$ is $(p(n-1) \times 1)$, $\boldsymbol{Z} \equiv \boldsymbol{z}^T \otimes \boldsymbol{I}_p$ is $(p(n-1) \times p^2)$ where $\boldsymbol{z} = (\boldsymbol{y}_1, \ldots, \boldsymbol{y}_{n-1})$ is $(p \times (n-1))$. In the above $vec(.)$ is the vectorization formed by stacking the columns of a matrix and $\otimes$ is the Kronecker product. The Kronecker product $A \otimes B$ of two matrices $A = \{a_{ij}\}_{i,j=1}^{p,q}$ and $B = \{b_{kl}\}_{k,l=1}^{r,s}$ is the $pq \times rs$ matrix $\{a_{ij}B\}_{i,j=1}^{p,q}$; see e.g. Mardia et al. (1979). Assume that

$$\frac{1}{n}\boldsymbol{Z}^T\boldsymbol{Z} \xrightarrow{p} \boldsymbol{\Gamma} = \boldsymbol{C} \otimes \boldsymbol{I}_p, \tag{6.3.5}$$

where $\boldsymbol{C} = E\left[\boldsymbol{y}_t \boldsymbol{y}_t^T\right]$. This is a very mild assumption in the regression context as discussed by Knight and Fu (2000), Zou (2006) and Zhao and Yu (2006). In the time series context it also holds under mild conditions provided long range dependence is not present.

We have $k = p^2$ parameters and we can assume that $k_0 < k$ is number of non-zero parameters in the true model (6.3.3). Let

$$\boldsymbol{C} \otimes \boldsymbol{I}_p = \begin{bmatrix} (\boldsymbol{C} \otimes \boldsymbol{I}_p)_{\mathcal{A},\mathcal{A}} & (\boldsymbol{C} \otimes \boldsymbol{I}_p)_{\mathcal{A},\mathcal{A}^c} \\ (\boldsymbol{C} \otimes \boldsymbol{I}_p)_{\mathcal{A}^c,\mathcal{A}} & (\boldsymbol{C} \otimes \boldsymbol{I}_p)_{\mathcal{A}^c,\mathcal{A}^c} \end{bmatrix}, \tag{6.3.6}$$

where $(\boldsymbol{C} \otimes \boldsymbol{I}_p)_{\mathcal{A},\mathcal{A}} = \{c_{(i,j),(r,s)} : (i,j),(r,s) \in \mathcal{A}\}$ is a $k_0 \times k_0$ matrix, $(\boldsymbol{C} \otimes \boldsymbol{I}_p)_{\mathcal{A},\mathcal{A}^c} = \{c_{(i,j),(r,s)} : (i,j) \in \mathcal{A}, (r,s) \in \mathcal{A}^c\}$ is a $(k-k_0) \times k_0$ matrix, $(\boldsymbol{C} \otimes \boldsymbol{I}_p)_{\mathcal{A}^c,\mathcal{A}} = (\boldsymbol{C} \otimes \boldsymbol{I}_p)^T_{\mathcal{A},\mathcal{A}^c}$, and $(\boldsymbol{C} \otimes \boldsymbol{I}_p)_{\mathcal{A}^c,\mathcal{A}^c} = \{c_{(i,j),(r,s)} : (i,j),(r,s) \in \mathcal{A}^c\}$ is a $(k-k_0) \times (k-k_0)$ matrix. The rows and columns $(\boldsymbol{C} \otimes \boldsymbol{I}_p)_{ij}$ are ordered using the ordering defined as follows:

$$(i_1,j_1) \begin{cases} < (i_2,j_2) & \text{if} \quad i_1 < i_2, \text{ or } i_1 = i_2 \text{ and } j_1 < j_2 \\ = (i_2,j_2) & \text{iff} \quad i_1 = i_2, j_1 = j_2 \\ > (i_2,j_2) & \text{if} \quad i_1 > i_2, \text{ or } i_1 = i_2 \text{ and } j_1 > j_2. \end{cases}$$

In the next section, we will prove a necessary condition for lasso-type methods to achieve consistent variable selection for time series models. This condition closely parallels the ZYZ condition in the regression case but the form of the condition is more complicated in the multivariate time series setting.

We will use the following notations: $\boldsymbol{B}^\dagger = [b^\dagger_{ij} : i,j = 1,\ldots,p]$, $\hat{\boldsymbol{B}}^* = [\hat{b}^*_{ij} : i,j = 1,\ldots,p]$ and $\hat{\boldsymbol{B}}^{**} = [\hat{b}^{**}_{ij} : i,j = 1,\ldots,p]$ as the true value, the lasso estimate and the adaptive lasso estimate, respectively, of parameter matrix $\boldsymbol{B} = [b_{ij} : i,j = 1,\ldots,p]$. Moreover, we define $\boldsymbol{B}^* = [b^*_{ij} : i,j = 1,\ldots,p]$ as a limiting value of the lasso estimates $\hat{\boldsymbol{B}}^* = [\hat{b}^*_{ij} : i,j = 1,\ldots,p]$.

6.4 Consistency of Lasso Variable Selection

Zou (2006) showed for linear regression models that lasso variable selection is inconsistent if the ZYZ condition (5.2.4) fails; see Section 5.5 for detailed discussion. In this section we prove the inconsistency of the lasso in variable selection for stationary multivariate time series models when the multivariate time series analogue of the ZYZ condition does not hold. Importantly, the multivariate time series analogue of ZYZ condition, like regression version, is a necessary condition so holding ZYZ condition

does not imply consistent variable selection.

Lasso estimates for the model (6.2.2) can be defined as

$$\hat{\boldsymbol{\beta}}^* \quad = \quad \arg\min \sum_{t=1}^{n-1} \sum_{i=1}^{p} (y_{t+1,i} - \boldsymbol{b}_i^T \boldsymbol{y}_t)^2 \text{ subject to } \sum_{i,j=1}^{p} |b_{ij}| \leq \omega,$$

where $\hat{\boldsymbol{\beta}}^* = vec(\hat{\boldsymbol{B}}^*)$ is defined in similar fashion as $\hat{\boldsymbol{\beta}}$ defined in (6.3.4). We shall often define the problem in following way:

$$\hat{\boldsymbol{\beta}}^* \quad = \quad \arg\min \left\{ \sum_{t=1}^{n-1} \sum_{i=1}^{p} (y_{t+1,i} - \boldsymbol{b}_i^T \boldsymbol{y}_t)^2 + \lambda_n \sum_{i,j=1}^{p} |b_{ij}| \right\}, \tag{6.4.1}$$

where λ_n is the user-defined tuning parameter that controls the amount of shrinkage. The lasso selection is consistent if and only if $\lim_{n\to\infty} P(\mathcal{A}_n^* = \mathcal{A}) = 1$, where

$$\mathcal{A} \quad = \quad \{(i,j) : b_{ij} \neq 0\}, \tag{6.4.2}$$

$$\mathcal{A}_n^* \quad = \quad \left\{(i,j) : \hat{b}_{ij}^* \neq 0\right\} \tag{6.4.3}$$

and $\hat{b}_{ij}^*$ is the lasso estimate of b_{ij}. We define

$$\boldsymbol{\beta}_{\mathcal{A}} = \{b_{ij} : (i,j) \in \mathcal{A}\} \tag{6.4.4}$$

as the non-zero coefficients in the model and

$$\hat{\boldsymbol{\beta}}_{\mathcal{A}}^* = \{\hat{b}_{ij}^* : (i,j) \in \mathcal{A}\} \tag{6.4.5}$$

the non-zero coefficients in the estimated lasso model. We assume $|\boldsymbol{\beta}_{\mathcal{A}}| = k_0$, where $|.|$ stands for the cardinality.

The following assumptions about the process $\{\boldsymbol{y}_t\}_{t\geq 1}$ will be needed:

(A1) The sequence $\{\boldsymbol{y}_t\}_{t\geq 1}$ is stationary.

(A2) The mean is zero, i.e. $E(\boldsymbol{y}_t) = \boldsymbol{0}_p$.

(A3) As $n \to \infty$,

$$\frac{1}{n} \sum_{t=1}^{n-1} \boldsymbol{y}_t \boldsymbol{y}_t^T \xrightarrow{p} \boldsymbol{C},$$

where $\boldsymbol{C}$ is of full rank; and

$$\frac{1}{n}\sum_{t=1}^{n-1}\boldsymbol{y}_{t+1}\boldsymbol{y}_t^T \xrightarrow{p} \boldsymbol{D} = \left[\boldsymbol{d}_1,\ldots,\boldsymbol{d}_p\right]^T.$$

(A4) As $n \to \infty$,

$$\boldsymbol{G} \xrightarrow{d} N_{p^2}\left(\boldsymbol{0}_{p^2}, \boldsymbol{V}\right),$$

where $\boldsymbol{G} = \left[\boldsymbol{G}_1^T,\ldots,\boldsymbol{G}_p^T\right]^T$, such that

$$\boldsymbol{G}_i = n^{-1/2}\sum_{t=1}^{n-1}\left(y_{t+1,i} - \boldsymbol{y}_t^T\boldsymbol{b}_i^\dagger\right)\boldsymbol{y}_t,$$

the $\boldsymbol{b}_i^\dagger$ are defined by

$$\boldsymbol{B}^\dagger = \left[\boldsymbol{b}_1^\dagger,\ldots,\boldsymbol{b}_p^\dagger\right] = \boldsymbol{D}\boldsymbol{C}^{-1},$$

and

$$\boldsymbol{V} = Cov(\boldsymbol{G}). \tag{6.4.6}$$

Comments: Assumption (A1) and (A2) are not necessary but they simplify the presentation. Assumptions (A3) and (A4) hold under mild moment conditions provided long-range dependence is not present. A general result which implies (A3) and (A4) under weak conditions is given by Hannan (1976). Note also that, with the given choice of the $\boldsymbol{b}_i^\dagger$, $E(\boldsymbol{G}) = \boldsymbol{0}_{p^2}$.

In the following theorem, we derive an asymptotic necessary condition for consistent variable selection for the model (6.2.2). This theorem is modeled on Theorem 1 of Zou (2006), but some new issues arise because of the time series structure.

Theorem 6.4.1 (Condition for consistent variable selection)**.** *Suppose that the multivariate time series $\{\boldsymbol{y}_t\}_{t\geq 1}$ satisfies conditions (A1)-(A4). If the lasso estimator $\hat{\boldsymbol{\beta}}^*$ in (6.4.1) is such that*

$$\lim_{n\to\infty} P(\mathcal{A}_n^* = \mathcal{A}) = 1, \tag{6.4.7}$$

where $\mathcal{A}$ and $\mathcal{A}_n^*$ are as defined above in (6.4.2) and (6.4.3) respectively, then there exists a sign vector $\boldsymbol{s}$ (whose components are of the form $sgn(x)$ for suitable x) such that, componentwise,

$$\left| \left((\boldsymbol{C} \otimes \boldsymbol{I}_p)_{\mathcal{A}^c,\mathcal{A}} (\boldsymbol{C} \otimes \boldsymbol{I}_p)^{-1}_{\mathcal{A},\mathcal{A}} \boldsymbol{s} \right)_r \right| \leq 1, \quad r = 1, \ldots, p^2 - k_0, \tag{6.4.8}$$

where k_0 is the cardinality of $\mathcal{A}$, $(\boldsymbol{a})_r$ is the rth component of the vector $\boldsymbol{a}$. The sign vector $\boldsymbol{s} = sgn\left[(\boldsymbol{C} \otimes \boldsymbol{I}_p)_{\mathcal{A},\mathcal{A}}(\boldsymbol{\beta}^*_{\mathcal{A}} - \boldsymbol{\beta}^\dagger_{\mathcal{A}})\right]$, where $\boldsymbol{\beta}^*_{\mathcal{A}}$ and $\boldsymbol{\beta}^\dagger_{\mathcal{A}}$ are defined in a similar fashion to $\boldsymbol{\beta}_{\mathcal{A}}$ in (6.4.4).

Proof. As noted by Zou (2006), lasso variable selection can be consistent only in one of the following three scenarios;

$$n^{-1}\lambda_n \longrightarrow \infty, \tag{6.4.9}$$

$$n^{-1}\lambda_n \longrightarrow \lambda_0,\ 0 < \lambda_0 < \infty, \tag{6.4.10}$$

$$n^{-1}\lambda_n \longrightarrow 0 \text{ but } n^{-1/2}\lambda_n \longrightarrow \infty. \tag{6.4.11}$$

If none of the above conditions hold, then the effect of the lasso penalty term is asymptotically negligible relative to the sum of squares term, and consequently

$$\lim_{n\to\infty} \mathcal{A}_n^* = \{(i,j) : i,j = 1, \ldots, p\}, \tag{6.4.12}$$

meaning we end up with a value of λ_n which corresponds to the least squares estimates i.e. none of the estimates shrink exactly to zero. This is because, for any given n, the least squares estimator of each b_{ij} will be non-zero with probability 1, when the distribution of $\boldsymbol{y}_t$ is continuous.

Define

$$S_n(\boldsymbol{B}) = \sum_{i=1}^{p} \sum_{t=1}^{n-1} \left(y_{t+1,i} - \boldsymbol{b}_i^T \boldsymbol{y}_t \right)^2 + \lambda_n \sum_{i,j=1}^{p} \left| b_{ij} \right|. \tag{6.4.13}$$

Now for each of the three scenarios stated above, we will look at the conditions required by the lasso to achieve consistency in variable selection.

Scenario (6.4.9): $n^{-1}\lambda_n \to \infty$

Dividing (6.4.13) by λ_n, we obtain

$$\lambda_n^{-1} S_n(\boldsymbol{B}) = \frac{n}{\lambda_n}\left[\frac{1}{n}\sum_{i=1}^{p}\sum_{t=1}^{n-1}\left(y_{t+1,i} - \boldsymbol{b}_i^T \boldsymbol{y}_t\right)^2\right] + \sum_{i,j=1}^{p}\left|b_{ij}\right|. \tag{6.4.14}$$

By assumption (A3), the term on the RHS of (6.4.14) in the square bracket [.], and the first derivative of this term with respect to each b_{ij}, are both $O_p(1)$. Therefore, since by hypothesis $n/\lambda_n \to 0$, it follows that each $\hat{b}_{ij}^* = 0$ with probability 1 for n sufficiently large. But this contradicts the assumption that $P(\mathcal{A}_n = \mathcal{A}) \to 1$. So we ignore this case.

Scenario (6.4.10): $n^{-1}\lambda_n \to \lambda_0 \in (0, \infty)$

Dividing (6.4.13) by n and defining $\overline{S}_n(\boldsymbol{B}) = n^{-1} S_n(\boldsymbol{B})$:

$$\begin{aligned}
\overline{S}_n(\boldsymbol{B}) &= \frac{1}{n}\sum_{i=1}^{p}\sum_{t=1}^{n-1}\left(y_{t+1,i} - \boldsymbol{b}_i^T \boldsymbol{y}_t\right)^2 + \frac{\lambda_n}{n}\sum_{i,j=1}^{p}\left|b_{ij}\right| \\
&= \left(\frac{1}{n}\sum_{i=1}^{p}\sum_{t=1}^{n-1} y_{t+1,i}^2\right) + \sum_{i=1}^{p}\boldsymbol{b}_i^T\left(\frac{1}{n}\sum_{t=1}^{n-1}\boldsymbol{y}_t\boldsymbol{y}_t^T\boldsymbol{b}_i\right\} \\
&\quad - 2\sum_{i=1}^{p}\boldsymbol{b}_i^T\left(\frac{1}{n}\sum_{t=1}^{n-1} y_{t+1,i}\boldsymbol{y}_t\right) + \frac{\lambda_n}{n}\sum_{i,j=1}^{p}\left|b_{ij}\right| \\
&= tr(\boldsymbol{C}_n) + \sum_{i=1}^{p}\boldsymbol{b}_i^T\boldsymbol{C}_n\boldsymbol{b}_i - 2\sum_{i=1}^{p}\boldsymbol{b}_i^T\boldsymbol{d}_{i,n} + n^{-1}\lambda_n\sum_{i=1}^{p}\left|b_{ij}\right|,
\end{aligned}$$

where $\boldsymbol{C}_n$ is defined in (6.3.2) and the $\boldsymbol{d}_{i,n}$ are the columns of the matrix $\boldsymbol{D}_n$ in (6.3.2). Letting $n \to \infty$, using the assumption $n^{-1}\lambda_n \to \lambda_0$, and using (A3), we obtain

$$\begin{aligned}
\overline{S}_n(\boldsymbol{B}) &\xrightarrow{p} tr(\boldsymbol{C}) + \sum_{i=1}^{p}\boldsymbol{b}_i^T\boldsymbol{C}\boldsymbol{b}_i - 2\sum_{i=1}^{p}\boldsymbol{b}_i^T\boldsymbol{d}_i + \lambda_0\sum_{i,j=1}^{p}\left|b_{ij}\right| \\
&= V_1(\boldsymbol{B}),
\end{aligned}$$

say. Then

$$\hat{\boldsymbol{B}}^* = \left[\hat{\boldsymbol{b}}_1^*, \ldots, \hat{\boldsymbol{b}}_p^*\right]^T \xrightarrow{p} \arg\min V_1(\boldsymbol{B}) = \boldsymbol{B}^*,$$

say.

Now pick a pair $(i, j) \in \mathcal{A}$. The derivative of $\overline{S}_n(\boldsymbol{B})$ with respect to b_{ij} and evaluated

at $\hat{\boldsymbol{b}}_i^*$ is given by

$$\frac{\partial}{\partial b_{ij}}\overline{S}_n(\boldsymbol{B}) = 2\boldsymbol{e}_j^T\boldsymbol{C}_n\left(\hat{\boldsymbol{b}}_i^* - \boldsymbol{C}_n^{-1}\boldsymbol{d}_{i,n}\right) + n^{-1}\lambda_n \text{sgn}\left(\hat{b}_{ij}^*\right), \tag{6.4.15}$$

where $\boldsymbol{e}_j$ is the $p \times 1$ vector whose jth component is 1 and whose other components are zero.

Now consider the event $\{(i,j) \in \mathcal{A}_n^*\}$, where $(i,j) \in \mathcal{A}$. This event is a subset of the event that the RHS of (6.4.15) is exactly zero. Therefore,

$$P\left[(i,j) \in \mathcal{A}_n^*\right] \leq P\left[\left|2\boldsymbol{e}_j^T\boldsymbol{C}_n\left(\hat{\boldsymbol{b}}_i^* - \boldsymbol{C}_n^{-1}\boldsymbol{d}_{i,n}\right)\right| = n^{-1}\lambda_n\right].$$

Now as $n \to \infty$, $\boldsymbol{C}_n \xrightarrow{p} \boldsymbol{C}$, $\boldsymbol{C}_n^{-1}\boldsymbol{d}_{i,n} \xrightarrow{p} \boldsymbol{b}_i^\dagger$, $\hat{\boldsymbol{b}}_i^* \xrightarrow{p} \boldsymbol{b}_i^*$ and $n^{-1}\lambda_n \to \lambda_0$.
Therefore

$$2\boldsymbol{e}_j^T\boldsymbol{C}_n\left(\hat{\boldsymbol{b}}_i^* - \boldsymbol{C}_n^{-1}\boldsymbol{d}_{i,n}\right) \xrightarrow{p} \left\{2\boldsymbol{C}\left(\boldsymbol{b}_i^* - \boldsymbol{b}_i^\dagger\right)\right\}_j,$$

the jth component of the vector $2\boldsymbol{C}(\boldsymbol{b}_i^* - \boldsymbol{b}_i^\dagger)$. Consequently, given that $P(\mathcal{A}_n^* = \mathcal{A}) \to 1$ as $n \to \infty$, it follows that, for $(i,j) \in \mathcal{A}$, $P\left[(i,j) \in \mathcal{A}_n^*\right] \to 1$, and we may conclude that

$$\left|\left\{2\boldsymbol{C}\left(\boldsymbol{b}_i^* - \boldsymbol{b}_i^\dagger\right)\right\}_j\right| = \lambda_0. \tag{6.4.16}$$

Now similarly for $(i,j) \notin \mathcal{A}$, then for $P\left\{(i,j) \notin \mathcal{A}_n^*\right\} \to 1$, we get

$$\left|\left\{2\boldsymbol{C}\left(\boldsymbol{b}_i^* - \boldsymbol{b}_i^\dagger\right)\right\}_j\right| \leq \lambda_0. \tag{6.4.17}$$

We define $\boldsymbol{B}_{\mathcal{A}}^* = \left[b_{ij}^*\right]_{(i,j)\in\mathcal{A}}$ and $\boldsymbol{B}_{\mathcal{A}^c}^* = \left[b_{ij}^*\right]_{(i,j)\in\mathcal{A}^c}$. Similarly, $\boldsymbol{B}_{\mathcal{A}}^\dagger = \left[b_{ij}^\dagger\right]_{(i,j)\in\mathcal{A}}$ and $\boldsymbol{B}_{\mathcal{A}^c}^\dagger = \left[b_{ij}^\dagger\right]_{(i,j)\in\mathcal{A}^c}$. Note that $b_{ij}^* = 0 = b_{ij}^\dagger$ for $(i,j) \in \mathcal{A}^c$, because $\lim_{n\to\infty} P\left(\mathcal{A}_n^* = \mathcal{A}\right) \to 1$ in the former case, and $(i,j) \in \mathcal{A}^c$ implies $b_{ij}^\dagger = 0$, by definition.

Take

$$\left(\boldsymbol{C} \otimes \boldsymbol{I}_p\right)\left(\boldsymbol{\beta}^* - \boldsymbol{\beta}^\dagger\right) = \begin{bmatrix} \left(\boldsymbol{C} \otimes \boldsymbol{I}_p\right)_{\mathcal{A},\mathcal{A}}\left(\boldsymbol{\beta}_{\mathcal{A}}^* - \boldsymbol{\beta}_{\mathcal{A}}^\dagger\right) + \left(\boldsymbol{C} \otimes \boldsymbol{I}_p\right)_{\mathcal{A},\mathcal{A}^c}\left(\boldsymbol{\beta}_{\mathcal{A}^c}^* - \boldsymbol{\beta}_{\mathcal{A}^c}^\dagger\right) \\ \left(\boldsymbol{C} \otimes \boldsymbol{I}_p\right)_{\mathcal{A}^c,\mathcal{A}}\left(\boldsymbol{\beta}_{\mathcal{A}}^* - \boldsymbol{\beta}_{\mathcal{A}}^\dagger\right) + \left(\boldsymbol{C} \otimes \boldsymbol{I}_p\right)_{\mathcal{A}^c,\mathcal{A}^c}\left(\boldsymbol{\beta}_{\mathcal{A}^c}^* - \boldsymbol{\beta}_{\mathcal{A}^c}^\dagger\right) \end{bmatrix}.$$

As $\boldsymbol{\beta}^*_{\mathcal{A}^c} = \boldsymbol{\beta}^\dagger_{\mathcal{A}^c} = \mathbf{0}$, thus

$$(\boldsymbol{C} \otimes \boldsymbol{I}_p)\left(\boldsymbol{\beta}^* - \boldsymbol{\beta}^\dagger\right) = \begin{bmatrix} (\boldsymbol{C} \otimes \boldsymbol{I}_p)_{\mathcal{A},\mathcal{A}} \left(\boldsymbol{\beta}^*_{\mathcal{A}} - \boldsymbol{\beta}^\dagger_{\mathcal{A}}\right) \\ (\boldsymbol{C} \otimes \boldsymbol{I}_p)_{\mathcal{A}^c,\mathcal{A}} \left(\boldsymbol{\beta}^*_{\mathcal{A}} - \boldsymbol{\beta}^\dagger_{\mathcal{A}}\right) \end{bmatrix}.$$

Using the assumption given in (6.4.10) and result obtained in (6.4.16), we can write

$$(\boldsymbol{C} \otimes \boldsymbol{I}_p)_{\mathcal{A},\mathcal{A}} \left(\boldsymbol{\beta}^*_{\mathcal{A}} - \boldsymbol{\beta}^\dagger_{\mathcal{A}}\right) = \frac{\lambda_0}{2} \boldsymbol{s}^*. \tag{6.4.18}$$

Consequently,

$$\boldsymbol{\beta}^*_{\mathcal{A}} - \boldsymbol{\beta}^\dagger_{\mathcal{A}} = \frac{\lambda_0}{2} (\boldsymbol{C} \otimes \boldsymbol{I}_p)^{-1}_{\mathcal{A},\mathcal{A}} \boldsymbol{s}^*, \tag{6.4.19}$$

where $\boldsymbol{s}^* = \text{sgn}\left[(\boldsymbol{C} \otimes \boldsymbol{I}_p)_{\mathcal{A},\mathcal{A}} \left(\boldsymbol{\beta}^*_{\mathcal{A}} - \boldsymbol{\beta}^\dagger_{\mathcal{A}}\right)\right]$. Similarly, by using (6.4.17) we can write

$$\left|\left\{(\boldsymbol{C} \otimes \boldsymbol{I}_p)_{\mathcal{A}^c,\mathcal{A}} \left(\boldsymbol{\beta}^*_{\mathcal{A}} - \boldsymbol{\beta}^\dagger_{\mathcal{A}}\right)\right\}_r\right| \leq \frac{\lambda_0}{2}, \tag{6.4.20}$$

where $r = 1, \ldots, |\mathcal{A}^c| = p^2 - k_0$. Substituting from (6.4.19) to (6.4.20), we get

$$\left|\left\{\frac{\lambda_0}{2} (\boldsymbol{C} \otimes \boldsymbol{I}_p)_{\mathcal{A}^c,\mathcal{A}} (\boldsymbol{C} \otimes \boldsymbol{I}_p)^{-1}_{\mathcal{A},\mathcal{A}} \boldsymbol{s}^*\right\}_r\right| \leq \frac{\lambda_0}{2},$$

which implies that

$$\left|\left\{(\boldsymbol{C} \otimes \boldsymbol{I}_p)_{\mathcal{A}^c,\mathcal{A}} (\boldsymbol{C} \otimes \boldsymbol{I}_p)^{-1}_{\mathcal{A},\mathcal{A}} \boldsymbol{s}^*\right\}_r\right| \leq 1 \text{ for } r = 1, \ldots, p^2 - k_0. \tag{6.4.21}$$

It is important to note that the above inequality holds componentwise.

Now we will prove the same necessary condition under the third scenario i.e. assuming condition (6.4.11)

Scenario (6.4.11): $n^{-1}\lambda_n \to 0$ but $n^{-1/2}\lambda_n \to \infty$

Under these conditions

$$\frac{n}{\lambda_n}\left(\hat{\boldsymbol{b}}^*_i - \boldsymbol{b}^\dagger_i\right) \xrightarrow{p} \arg\min(V_3), \tag{6.4.22}$$

where

$$V_3(\boldsymbol{u}) = \sum_{i=1}^{p}\left(\boldsymbol{u}_i^T \boldsymbol{C}\boldsymbol{u}_i\right) + \sum_{i,j=1}^{p}\left[u_{ij}\text{sgn}\left(b_{ij}^{\dagger}\right) I\left(b_{ij}^{\dagger} \neq 0\right) + |u_{ij}| I\left(b_{ij}^{\dagger} = 0\right)\right], \quad (6.4.23)$$

$I(.)$ is the indicator function and $\boldsymbol{u}_i, i = 1, \ldots, p$ are non-random. To see this write

$$\boldsymbol{b}_i = \boldsymbol{b}_i^{\dagger} + \frac{\lambda_n}{n}\boldsymbol{u}_i \quad i = 1, \ldots, p, \quad (6.4.24)$$

and substitute this into (6.4.13). Then we can define the following quantity

$$S_n(\boldsymbol{u}) = \sum_{i=1}^{p}\sum_{t=1}^{n-1}\left\{y_{t+1,i} - \left(\boldsymbol{b}_i^{\dagger} + \frac{\lambda_n}{n}\boldsymbol{u}_i\right)^T \boldsymbol{y}_t\right\}^2 + \lambda_n \sum_{i,j=1}^{p}\left|b_{ij}^{\dagger} + \frac{\lambda_n}{n}u_{ij}\right|.$$

Assuming $\hat{\boldsymbol{u}}^* = \arg\min V_3^{(n)}(\boldsymbol{u})$ then we can write

$$\hat{\boldsymbol{b}}_i^* = \boldsymbol{b}_i^{\dagger} + \frac{\lambda_n}{n}\hat{\boldsymbol{u}}_i^*, \quad i = 1, \ldots, p.$$

Consider

$$\begin{aligned} S_n(\boldsymbol{u}) - S_n(\boldsymbol{0}) &= \sum_{i=1}^{p}\sum_{t=1}^{n-1}\left[\left\{\left(y_{t+1,i} - (\boldsymbol{b}_i^{\dagger})^T\boldsymbol{y}_t\right) - \frac{\lambda_n}{n}\boldsymbol{u}_i^T\boldsymbol{y}_t\right\}^2 - \left(y_{t+1,i} - \boldsymbol{b}_i^{\dagger}\boldsymbol{y}_t\right)^2\right] \\ &\quad + \lambda_n \sum_{i,j=1}^{p}\left(\left|b_{ij}^{\dagger} + \frac{\lambda_n}{n}u_{ij}\right| - \left|b_{ij}^{\dagger}\right|\right). \end{aligned}$$

After some algebraic manipulation, we can write

$$\begin{aligned} S_n(\boldsymbol{u}) - S_n(\boldsymbol{0}) &= \sum_{i=1}^{p}\sum_{t=1}^{n-1}\left\{\frac{\lambda_n^2}{n^2}\boldsymbol{u}_i^T\boldsymbol{y}_t\boldsymbol{y}_t^T\boldsymbol{u}_i - 2\frac{\lambda_n}{n}\left(y_{t+1,i} - (\boldsymbol{b}_i^{\dagger})^T\boldsymbol{y}_t\right)\boldsymbol{y}_t^T\boldsymbol{u}_i\right\} \\ &\quad + \lambda_n \sum_{i,j=1}^{p}\left(\left|b_{ij}^{\dagger} + \frac{\lambda_n}{n}u_{ij}\right| - \left|b_{ij}^{\dagger}\right|\right), \\ &= \sum_{i=1}^{p}\left\{\frac{\lambda_n^2}{n}\boldsymbol{u}_i^T\boldsymbol{C}_n\boldsymbol{u}_i - 2\frac{\lambda_n}{\sqrt{n}}\boldsymbol{G}_i^T\boldsymbol{u}_i\right\} + \lambda_n \sum_{i,j=1}^{p}\left(\left|b_{ij}^{\dagger} + \frac{\lambda_n}{n}u_{ij}\right| - \left|b_{ij}^{\dagger}\right|\right). \end{aligned}$$

We can define $V_3^{(n)}(\boldsymbol{u}) = n\lambda_n^{-2}\left[S_3(\boldsymbol{u}) - S_3(\boldsymbol{0})\right]$, and thus we can write

$$V_3^{(n)}(\boldsymbol{u}) = \sum_{i=1}^{p}\boldsymbol{u}_i^T\boldsymbol{C}_n\boldsymbol{u}_i - 2\frac{\sqrt{n}}{\lambda_n}\sum_{i=1}^{p}\boldsymbol{G}_i^T\boldsymbol{u}_i + \frac{n}{\lambda_n}\sum_{i,j=1}^{p}\left(\left|b_{ij}^{\dagger} + \frac{\lambda_n}{n}u_{ij}\right| - \left|b_{ij}^{\dagger}\right|\right) \quad (6.4.25)$$

As we know, $\boldsymbol{C}_n \xrightarrow{p} \boldsymbol{C}$, so $\boldsymbol{u}_i^T \boldsymbol{C}_n \boldsymbol{u}_i \xrightarrow{p} \boldsymbol{u}_i^T \boldsymbol{C} \boldsymbol{u}_i$. Now $\boldsymbol{G}_i$ is $O_p(1)$, thus under this scenario condition and $\lambda_n/\sqrt{n} \to \infty$, so using Slutsky's theorem, we conclude

$$\frac{\sqrt{n}}{\lambda_n} \boldsymbol{G}_i^T \boldsymbol{u}_i \xrightarrow{p} 0. \tag{6.4.26}$$

Thus the second term on RHS of (6.4.25) converges to zero in probability. Now for the third term we have two different cases. If $(i,j) \in \mathcal{A}$, i.e $b_{ij}^{\dagger} \neq 0$, then

$$\frac{n}{\lambda_n} \left(\left| b_{ij}^{\dagger} + \frac{\lambda_n}{n} u_{ij} \right| - \left| b_{ij}^{\dagger} \right| \right) \xrightarrow{p} u_{ij} \text{sgn}(b_{ij}), \tag{6.4.27}$$

and if $(i,j) \notin \mathcal{A}$ i.e. $b_{ij}^{\dagger} = 0$ then

$$\frac{n}{\lambda_n} \left(\left| b_{ij}^{\dagger} + \frac{\lambda_n}{n} u_{ij} \right| - \left| b_{ij}^{\dagger} \right| \right) = |u_{ij}|. \tag{6.4.28}$$

Thus we can conclude from (6.4.26)-(6.4.28) and (6.4.25) that

$$V_3^{(n)}(\boldsymbol{u}) \xrightarrow{p} V_3(\boldsymbol{u})$$

and

$$\hat{\boldsymbol{u}}^* = \frac{n}{\lambda_n} \left(\hat{\boldsymbol{\beta}}^* - \boldsymbol{\beta}^{\dagger} \right) \xrightarrow{p} \boldsymbol{u}^{\dagger} = \arg\min V_3(\boldsymbol{u}), \tag{6.4.29}$$

where $V_3(\boldsymbol{u})$ is as defined in (6.4.23).

The above result (6.4.29) is an important result and we will use it to derive the condition for consistent lasso variable selection. We can write (6.4.23) as

$$\begin{aligned} V_3(\boldsymbol{u}_{\mathcal{A}}, \boldsymbol{u}_{\mathcal{A}^c}) &= \begin{bmatrix} \boldsymbol{u}_{\mathcal{A}} & \boldsymbol{u}_{\mathcal{A}^c} \end{bmatrix} \begin{bmatrix} \boldsymbol{C}_{\mathcal{A},\mathcal{A}} & \boldsymbol{C}_{\mathcal{A},\mathcal{A}^c} \\ \boldsymbol{C}_{\mathcal{A}^c,\mathcal{A}} & \boldsymbol{C}_{\mathcal{A}^c,\mathcal{A}^c} \end{bmatrix} \begin{bmatrix} \boldsymbol{u}_{\mathcal{A}} \\ \boldsymbol{u}_{\mathcal{A}^c} \end{bmatrix} + \sum_{(i,j)\in\mathcal{A}} u_{ij} s_{ij} + \sum_{(i,j)\notin\mathcal{A}} |u_{ij}| \\ &= \boldsymbol{u}_{\mathcal{A}} \boldsymbol{C}_{\mathcal{A},\mathcal{A}} \boldsymbol{u}_{\mathcal{A}} + \boldsymbol{u}_{\mathcal{A}} \boldsymbol{C}_{\mathcal{A},\mathcal{A}^c} \boldsymbol{u}_{\mathcal{A}^c} + \boldsymbol{u}_{\mathcal{A}^c} \boldsymbol{C}_{\mathcal{A}^c,\mathcal{A}} \boldsymbol{u}_{\mathcal{A}} + \boldsymbol{u}_{\mathcal{A}^c} \boldsymbol{C}_{\mathcal{A}^c,\mathcal{A}^c} \boldsymbol{u}_{\mathcal{A}^c} \\ &\quad + \sum_{(i,j)\in\mathcal{A}} u_{ij} s_{ij} + \sum_{(i,j)\notin\mathcal{A}} |u_{ij}|. \end{aligned}$$

Consider $(i,j) \in \mathcal{A}$:

$$\frac{\partial}{\partial u_{ij}} V_3(\boldsymbol{u}_{\mathcal{A}}, \boldsymbol{u}_{\mathcal{A}^c}) = 2\boldsymbol{e}_j^T \boldsymbol{C}_{\mathcal{A},\mathcal{A}} \boldsymbol{u}_{\mathcal{A}} + 2\boldsymbol{e}_j^T \boldsymbol{C}_{\mathcal{A},\mathcal{A}^c} \boldsymbol{u}_{\mathcal{A}^c} + \boldsymbol{e}_j \boldsymbol{s},$$

where $\boldsymbol{e}_j$ is the $k_0 \times 1$ vector whose jth component is 1 and whose other components are zero. Thus setting $\frac{\partial}{\partial \boldsymbol{u}} V_3(\boldsymbol{u}_{\mathcal{A}}, \boldsymbol{u}_{\mathcal{A}^c}) = 0$, gives

$$\begin{aligned} 2\boldsymbol{C}_{\mathcal{A},\mathcal{A}}\boldsymbol{u}_{\mathcal{A}} + 2\boldsymbol{C}_{\mathcal{A},\mathcal{A}^c}\boldsymbol{u}_{\mathcal{A}^c} + \boldsymbol{s} &= 0 \\ 2\left(\boldsymbol{C}_{\mathcal{A},\mathcal{A}}\boldsymbol{u}_{\mathcal{A}} + \boldsymbol{C}_{\mathcal{A},\mathcal{A}^c}\boldsymbol{u}_{\mathcal{A}^c}\right) &= -\boldsymbol{s} \\ \Longrightarrow \hat{\boldsymbol{u}}^*_{\mathcal{A}} &= -\boldsymbol{C}^{-1}_{\mathcal{A},\mathcal{A}}\left(\boldsymbol{C}_{\mathcal{A},\mathcal{A}^c}\boldsymbol{u}_{\mathcal{A}^c} + \frac{1}{2}\boldsymbol{s}\right). \end{aligned}$$

Now we consider $(i,j) \in \mathcal{A}^c$. Then $\hat{\boldsymbol{u}}^*_{\mathcal{A}^c} = \arg\min \widetilde{V}_3(\boldsymbol{u}_{\mathcal{A}^c})$, where,

$$\widetilde{V}_3(\boldsymbol{u}_{\mathcal{A}^c}) = \boldsymbol{u}^T_{\mathcal{A}^c}\left(\boldsymbol{C}_{\mathcal{A}^c,\mathcal{A}^c} - \boldsymbol{C}_{\mathcal{A}^c,\mathcal{A}}\boldsymbol{C}^{-1}_{\mathcal{A},\mathcal{A}}\boldsymbol{C}_{\mathcal{A},\mathcal{A}^c}\right)\boldsymbol{u}_{\mathcal{A}^c} - \boldsymbol{u}^T_{\mathcal{A}^c}\boldsymbol{C}_{\mathcal{A}^c,\mathcal{A}}\boldsymbol{C}^{-1}_{\mathcal{A},\mathcal{A}}\boldsymbol{s} + \sum_{(i,j)\in\mathcal{A}^c}|u_{ij}|.$$

Take

$$\begin{aligned} \frac{\partial}{\partial u_{ij}}\widetilde{V}_3(\boldsymbol{u}_{\mathcal{A}^c}) &= 2\boldsymbol{e}^T_j\left(\boldsymbol{C}_{\mathcal{A}^c,\mathcal{A}^c} - \boldsymbol{C}_{\mathcal{A}^c,\mathcal{A}}\boldsymbol{C}^{-1}_{\mathcal{A},\mathcal{A}}\boldsymbol{C}_{\mathcal{A},\mathcal{A}^c}\right)\boldsymbol{u}_{\mathcal{A}^c} \\ &\quad - \boldsymbol{C}_{\mathcal{A}^c,\mathcal{A}}\boldsymbol{C}^{-1}_{\mathcal{A},\mathcal{A}}\boldsymbol{s} + \operatorname{sgn}|u_{ij}|. \end{aligned} \tag{6.4.30}$$

Using (6.4.29), which implies that $\hat{\boldsymbol{u}}^*_{\mathcal{A}^c} \xrightarrow{p} \boldsymbol{u}^\dagger_{\mathcal{A}^c} = \boldsymbol{0}$, it is seen that the first term on RHS of (6.4.30) goes to zero and according to the modified KKT optimality conditions, defined in Section 6.2.2, the RHS of (6.4.30) requires the change of sign, so we can write

$$\left|\left((\boldsymbol{C}\otimes\boldsymbol{I}_p)_{\mathcal{A}^c,\mathcal{A}}(\boldsymbol{C}\otimes\boldsymbol{I}_p)^{-1}_{\mathcal{A},\mathcal{A}}\boldsymbol{s}^\dagger\right)_r\right| \leq 1, \quad r = 1,\ldots,p^2 - k_0, \tag{6.4.31}$$

where $\boldsymbol{s}^\dagger = \operatorname{sgn}(\boldsymbol{\beta}^\dagger_{\mathcal{A}})$.

Thus (6.4.21) and (6.4.31) jointly prove that lasso variable selection cannot be consistent unless (6.4.8) holds. □

6.5 Adaptive Lasso

We have looked at the oracle properties of the adaptive lasso in the regression context in Chapter 5. For an appropriate value of the tuning parameter, the adaptive lasso has shown to achieve the oracle properties. The adaptive lasso estimator $\hat{\boldsymbol{\beta}}^{**} =$

$\left[\hat{b}_{ij}^{**} : i,j = 1,\ldots,p\right]$ of $\boldsymbol{\beta} = [b_{ij} : i,j = 1,\ldots,p]$ for the model (6.2.2) can be defined as

$$\hat{\boldsymbol{\beta}}^{**} \quad = \quad \arg\min \left\{ \sum_{t=1}^{n-1} \sum_{i=1}^{p} (y_{t+1,i} - \boldsymbol{b}_i^T \boldsymbol{y}_t)^2 + \lambda_n \sum_{i,j=1}^{p} w_{ij} |b_{ij}| \right\},$$

where w_{ij} is an adaptive weight for each b_{ij} and λ_n is the user-defined tuning parameter which controls the amount of shrinkage. Zou (2006) proved in his Theorem 2 that for a suitable choice of λ_n, the adaptive lasso satisfies the oracle properties in the regression context. Here we extend the same conclusions to the multivariate time series context. Our proof is modeled on that of Theorem 2 of Zou (2006).

We will define $\mathcal{A}_n^{**} = \left\{(i,j) : \hat{b}_{ij}^{**} \neq 0\right\}$ for $i,j = 1,\ldots,p$.

Theorem 6.5.1. *Suppose that $\lambda_n/\sqrt{n} \to 0$, $\lambda_n n^{(\gamma-1)/2} \to \infty$ for some $\gamma > 0$, and conditions (A1)-(A4) of Section 6.4 are satisfied. Assume also that the weights are given by $\hat{w}_{ij} = 1/|\hat{b}_{ij}|^{\gamma}$, where the $\hat{b}_{ij}$ are the least squares estimates of the b_{ij}. Then the adaptive lasso estimates satisfy the following oracle properties.*

1. Consistency in variable selection:

$$\lim_{n\to\infty} P\left(\mathcal{A}_n^{**} = \mathcal{A}\right) = 1$$

2. Asymptotic normality:

$$\sqrt{n}\left(\hat{\boldsymbol{\beta}}_{\mathcal{A}}^{**} - \boldsymbol{\beta}_{\mathcal{A}}^{\dagger}\right) \xrightarrow{d} N\left(\mathbf{0}, \boldsymbol{\Sigma}\right),$$

where $\boldsymbol{\Sigma}$ is as defined in (6.5.7), $\hat{\boldsymbol{\beta}}_{\mathcal{A}}^{**} = \left\{\hat{b}_{ij}^{**} : (i,j) \in \mathcal{A}_n^{**}\right\}$ and $\boldsymbol{\beta}_{\mathcal{A}}^{\dagger} = \left\{b_{ij}^{\dagger} : (i,j) \in \mathcal{A}\right\}$.

Proof. First we prove the asymptotic normality of the adaptive lasso estimator $\hat{\boldsymbol{\beta}}_{\mathcal{A}}^{**}$. Consider

$$\psi_n(\boldsymbol{B}) = \sum_{t=1}^{n-1} \sum_{i=1}^{p} (y_{t+1,i} - \boldsymbol{b}_i^T \boldsymbol{y}_t)^2 + \lambda_n \sum_{i,j=1}^{p} \hat{w}_{ij} |b_{ij}|.$$

Let $\boldsymbol{b}_i = \boldsymbol{b}_i^\dagger + n^{-1/2}\boldsymbol{u}_i$ and define

$$\begin{aligned}\psi_n(\boldsymbol{u}) &= \sum_{t=1}^{n-1}\sum_{i=1}^{p}\left[y_{t+1,i} - \left(\boldsymbol{b}_i^\dagger + n^{-1/2}\boldsymbol{u}_i\right)^T \boldsymbol{y}_t\right]^2 \\ &\quad + \lambda_n \sum_{i,j=1}^{p} \hat{w}_{ij}\left|b_{ij}^\dagger + n^{-1/2}u_{ij}\right|, \end{aligned} \tag{6.5.1}$$

where $\boldsymbol{u} = \left(\boldsymbol{u}_1, \ldots, \boldsymbol{u}_p\right)^T$ is a $p \times p$ matrix. We can write

$$\begin{aligned}\hat{\boldsymbol{u}}_i^{**} &= \sqrt{n}\left(\hat{\boldsymbol{b}}_i^{**} - \boldsymbol{b}_i^\dagger\right) \\ &= \arg\min\left(\psi_n(\boldsymbol{u}) - \psi_n(\boldsymbol{0})\right) \\ &= \arg\min V_4^{(n)}(\boldsymbol{u}),\end{aligned}$$

where

$$V_4^{(n)}(\boldsymbol{u}) = \psi_n(\boldsymbol{u}) - \psi_n(\boldsymbol{0}).$$

Using (6.5.1),

$$\begin{aligned}V_4^{(n)}(\boldsymbol{u}) &= \sum_{t=1}^{n-1}\sum_{i=1}^{p}\left[y_{t+1,i} - \left(\boldsymbol{b}_i^\dagger + n^{-1/2}\boldsymbol{u}_i\right)^T \boldsymbol{y}_t\right]^2 + \lambda_n \sum_{i,j=1}^{p} \hat{w}_{ij}\left|b_{ij}^\dagger + n^{-1/2}u_{ij}\right| \\ &\quad - \sum_{t=1}^{n-1}\sum_{i=1}^{p}\left[y_{t+1,i} - (\boldsymbol{b}_i^\dagger)^T\boldsymbol{y}_t\right]^2 - \lambda_n \sum_{i,j=1}^{p} \hat{w}_{ij}\left|b_{ij}^\dagger\right|.\end{aligned}$$

After some algebraic manipulation, we get

$$\begin{aligned}V_4^{(n)}(\boldsymbol{u}) &= n^{-1}\sum_{t=1}^{n-1}\sum_{i=1}^{p}\boldsymbol{u}_i^T\boldsymbol{y}_t\boldsymbol{y}_t^T\boldsymbol{u}_i - 2n^{-1/2}\sum_{t=1}^{n-1}\sum_{i=1}^{p}\left(y_{t+1,i} - \boldsymbol{y}_t^T\boldsymbol{b}_i^\dagger\right)\boldsymbol{y}_t^T\boldsymbol{u}_i \\ &\quad + \lambda_n \sum_{i,j=1}^{p}\hat{w}_{ij}\left(\left|b_{ij}^\dagger + n^{-1/2}u_{ij}\right| - \left|b_{ij}^\dagger\right|\right).\end{aligned} \tag{6.5.2}$$

The first term on the RHS of (6.5.2) may be written as

$$n^{-1}\sum_{t=1}^{n-1}\sum_{i=1}^{p}\boldsymbol{u}_i^T\boldsymbol{y}_t\boldsymbol{y}_t^T\boldsymbol{u}_i = \sum_{i=1}^{p}\boldsymbol{u}_i^T\boldsymbol{C}_n\boldsymbol{u}_i \xrightarrow{p} \sum_{i=1}^{p}\boldsymbol{u}_i^T\boldsymbol{C}\boldsymbol{u}_i,$$

where $\boldsymbol{C}_n = n^{-1}\sum_{t=1}^{n-1}\boldsymbol{y}_t\boldsymbol{y}_t^T$, and the second term satisfies

$$n^{-1/2}\sum_{t=1}^{n-1}\sum_{i=1}^{p}\left(y_{t+1,i}-\boldsymbol{y}_t^T\boldsymbol{b}_i^\dagger\right)\boldsymbol{y}_t^T\boldsymbol{u}_i \xrightarrow{d}\sum_{i=1}^{p}\boldsymbol{u}_i^T\boldsymbol{G}_i,$$

using (A4). Consider the third term,

$$\lambda_n\sum_{i,j=1}^{p}\hat{w}_{ij}\left(\left|b_{ij}^\dagger+n^{-1/2}u_{ij}\right|-\left|b_{ij}^\dagger\right|\right)=n^{-1/2}\lambda_n\sum_{i,j=1}^{p}\hat{w}_{ij}n^{1/2}\left(\left|b_{ij}^\dagger+n^{-1/2}u_{ij}\right|-\left|b_{ij}^\dagger\right|\right).$$

If $(i,j)\in\mathcal{A}$ i.e. $b_{ij}^\dagger\neq 0$, then

$$n^{1/2}\left(\left|b_{ij}^\dagger+n^{-1/2}u_{ij}\right|-\left|b_{ij}^\dagger\right|\right)\xrightarrow{p}u_{ij}\text{sgn}(b_{ij}^\dagger).$$

Therefore, because $n^{-1/2}\lambda_n\to 0$ and $\hat{w}_{ij}=\left|\hat{b}_{ij}\right|^{-\gamma}=O_p(1)$ by the assumption of Theorem 6.5.1, therefore by Slutsky's theorem, we have

$$n^{-1/2}\lambda_n\hat{w}_{ij}n^{1/2}\left(\left|b_{ij}^\dagger+n^{-1/2}u_{ij}\right|-\left|b_{ij}^\dagger\right|\right)\xrightarrow{p}0. \tag{6.5.3}$$

Thus the third term on the RHS of (6.5.2) converges to zero in probability when $(i,j)\in\mathcal{A}$. Now consider the situation $(i,j)\notin\mathcal{A}$, i.e. $b_{ij}^\dagger=0$. In this case

$$n^{1/2}\left(\left|b_{ij}^\dagger+n^{-1/2}u_{ij}\right|-\left|b_{ij}^\dagger\right|\right)=\left|u_{ij}\right|$$

and therefore the contribution of component (i,j) to the penalty term is

$$\begin{aligned}n^{-1/2}\lambda_n\hat{w}_{ij}n^{1/2}\left(\left|b_{ij}^\dagger+n^{-1/2}u_{ij}\right|-\left|b_{ij}^\dagger\right|\right) &= n^{-1/2}\lambda_n\left|\hat{b}_{ij}\right|^{-\gamma}\left|u_{ij}\right| \\ &= n^{\gamma-1/2}\lambda_n\left|\sqrt{n}\hat{b}_{ij}\right|^{-\gamma}\left|u_{ij}\right|.\end{aligned}$$

As $\sqrt{n}\hat{b}_{ij}=O_p(1)$, and $\sqrt{n}\hat{b}_{ij}\neq 0$ with probability 1, and $n^{(\gamma-1)/2}\lambda_n\to\infty$ by the hypothesis of the theorem, it follows that

$$n^{(\gamma-1)/2}\lambda_n\left|\sqrt{n}\hat{b}_{ij}\right|^{-\gamma}\left|u_{ij}\right|\xrightarrow{p}\infty$$

unless $u_{ij}=0$. It follows that the only way we can keep $V_4^{(n)}(\boldsymbol{u})$ finite is to put $u_{ij}=0$ for all $(i,j)\in\mathcal{A}^c$. Write $\boldsymbol{u}_i=\left(\boldsymbol{u}_{i,\mathcal{A}(i)}^T,\boldsymbol{0}_{p-|\mathcal{A}(i)|}^T\right)^T$. Then, setting all these $u_{ij}=0$ results

in third term on RHS of (6.5.2) convergence to zero for $(i,j) \in \mathcal{A}^c$ i.e. for $(i,j) \notin \mathcal{A}$ we have

$$\left|b_{ij}^{\dagger} + n^{-1/2}u_{ij}\right| - \left|b_{ij}^{\dagger}\right| \xrightarrow{p} 0. \tag{6.5.4}$$

Using (6.5.3) and (6.5.4), we can say this term vanishes for all $i,j = 1,\ldots,p$. Thus we obtain

$$V_4^{(n)}(\boldsymbol{u}) \xrightarrow{p} V_4(\boldsymbol{u}) = \sum_{i=1}^{p} \boldsymbol{u}_{i,\mathcal{A}(i)}^T \boldsymbol{C}_{\mathcal{A}(i),\mathcal{A}(i)} \boldsymbol{u}_{i,\mathcal{A}(i)} - 2\sum_{i=1}^{p} \boldsymbol{u}_{i,\mathcal{A}(i)}^T \boldsymbol{G}_{i,\mathcal{A}(i)}. \tag{6.5.5}$$

Now

$$\frac{\partial V_4(\boldsymbol{u})}{\partial \boldsymbol{u}_{i,\mathcal{A}(i)}} = 0$$

implies

$$\boldsymbol{C}_{\mathcal{A}(i),\mathcal{A}(i)} \hat{\boldsymbol{u}}_{i,\mathcal{A}(i)}^{**} = \boldsymbol{G}_{i,\mathcal{A}(i)}$$

i.e.

$$\hat{\boldsymbol{u}}_{i,\mathcal{A}(i)}^{**} = \boldsymbol{C}_{\mathcal{A}(i),\mathcal{A}(i)}^{-1} \boldsymbol{G}_{i,\mathcal{A}(i)}.$$

It follows that

$$\hat{\boldsymbol{u}}_{i,\mathcal{A}(i)}^{**} = \left(\hat{\boldsymbol{u}}_{i,\mathcal{A}(i)}^{**T}, \boldsymbol{0}_{p-|\mathcal{A}(i)|}^T\right)^T, \quad i = 1,\ldots,p$$

Using Lemma 4.2.8 we can write $\boldsymbol{G} \xrightarrow{d} N(\boldsymbol{0}, \boldsymbol{V})$ and so we can write

$$\sqrt{n}\left(\hat{\boldsymbol{\beta}}_{\mathcal{A}}^{**} - \boldsymbol{\beta}_{\mathcal{A}}^{\dagger}\right) \xrightarrow{d} N_{k_0^2}\left(\boldsymbol{0}_{k_0^2}, \boldsymbol{\Sigma}\right). \tag{6.5.6}$$

where

$$\begin{aligned} \boldsymbol{\Sigma} &= \triangle \boldsymbol{\Xi} \triangle, \\ \triangle &= \operatorname{diag}\left\{\boldsymbol{C}_{\mathcal{A}(1),\mathcal{A}(1)}^{-1}, \ldots, \boldsymbol{C}_{\mathcal{A}(p),\mathcal{A}(p)}^{-1}\right\} \end{aligned} \tag{6.5.7}$$

and

$$\Xi = \left\{\Xi_{i,j}\right\}_{i,j=1}^{p}, \quad \Xi_{i,j} = Cov\left(\boldsymbol{G}_{i,\mathcal{A}(i)}, \boldsymbol{G}_{j,\mathcal{A}(j)}\right), \; i,j = 1,\ldots,p,$$

and each Ξ_{ij} is a $|\mathcal{A}_{(i)}| \times |\mathcal{A}_{(j)}|$ matrix. Note that the covariance matrix $\boldsymbol{\Sigma}$ is the same as that for the least squares estimator of the non-zero b_{ij}, where all zero b_{ij} are omitted from the estimation procedure.

Now we will show that the adaptive lasso is always consistent in variable selection. We can conclude from the above result (6.5.6) that $P((i,j) \in \mathcal{A}_n^{**}) \longrightarrow 1$, for all$(i,j) \in \mathcal{A}$; i.e. the adaptive lasso is consistent in correctly classifying the non-zero b_{ij}. Now we need to prove that $P((i,j) \in \mathcal{A}_n^{**}) \longrightarrow 0$, for all $(i,j) \notin \mathcal{A}$. This is equivalent to proving that

$$P(\hat{b}_{ij}^{**} \neq 0) \to 0, \text{ for all } (i,j) \notin \mathcal{A}.$$

Now consider the case that $(i,j) \in \mathcal{A}_n^{**}$, so we can write the $\psi_n(\boldsymbol{u})$ in the form as below:

$$\psi_n(\boldsymbol{u}) = \sum_{i=1}^{p}\sum_{t=1}^{n-1}\left(y_{t+1,i} - \boldsymbol{b}_i^T\boldsymbol{y}_t\right)^2 + \lambda_n \sum_{i,j=1}^{p} \hat{w}_{ij}\left|b_{ij}\right|.$$

Using the modified KKT optimality condition for $(i,j) \in \mathcal{A}_n^{**}$, we get

$$-2\sum_{t=1}^{n-1}\left(y_{t+1,i} - (\hat{\boldsymbol{b}}_i^{**})^T\boldsymbol{y}_t\right)y_{tj} = \pm\lambda_n\hat{w}_{ij},$$

which implies

$$\begin{aligned}\pm\frac{\lambda_n\hat{w}_{ij}}{\sqrt{n}} &= \frac{2\sum_{t=1}^{n-1}\left(y_{t+1,i} - (\hat{\boldsymbol{b}}_i^{**})^T\boldsymbol{y}_t\right)y_{tj}}{\sqrt{n}} \\ &= 2G_{ij}.\end{aligned}$$

As $n^{-1/2}\lambda_n \xrightarrow{p} 0$, $\lambda_n n^{(\gamma-1)/2} \to \infty$ by assumption, and $\sqrt{n}\hat{b}_{ij} = O_p(1)$ for $(i,j) \notin \mathcal{A}$, where $\hat{b}_{ij}$ is the ordinary least squares estimator, therefore

$$\frac{\lambda_n\hat{w}_{ij}}{\sqrt{n}} = \lambda_n n^{(\gamma-1)/2}\left|\sqrt{n}\hat{b}_{ij}\right|^{-\gamma} \xrightarrow{p} \infty.$$

However $\boldsymbol{G} = \left[G_{ij}\right]_{i,j=1}^{p}$ is normally distributed, so

$$P\left((i,j) \in \mathcal{A}_n^{**}\right) \leq P\left[\left|\left(2\boldsymbol{G}_i\right)_j\right| = \frac{\lambda_n \hat{w}_{ij}}{\sqrt{n}}\right] \longrightarrow 0 \text{ as } n \to \infty$$

where $(.)_j$ stands for the jth component. This proves the consistency. □

6.6 Numerical Results

In previous sections, theoretical results suggest that oracle properties of the lasso methods for time series models, like regression problems, can be achieved if certain conditions are satisfied, namely the time series version of ZYZ condition (6.4.8) holds and an appropriate value of the tuning parameter is selected.

Here we have considered three models. Model 0 is an example of a VAR(1) model while Model 1 and Model 2 are as studied by Hsu et al. (2008, p. 3650). We now numerically study the oracle properties of the lasso and adaptive lasso for time series models using these models.

Model 0:

$$\left(\boldsymbol{I} - \boldsymbol{A}_1 L\right) \boldsymbol{y}_t = \boldsymbol{\varepsilon}_t, \quad \boldsymbol{\varepsilon}_t \sim N\left(\boldsymbol{0}, \boldsymbol{\Sigma}_{\varepsilon 1}\right)$$

Model 1:

$$\left(\boldsymbol{I} - \boldsymbol{A}_1 L\right)\left(\boldsymbol{I} - \boldsymbol{A}_2 L^4\right) \boldsymbol{y}_t = \boldsymbol{\varepsilon}_t, \quad \boldsymbol{\varepsilon}_t \sim N\left(\boldsymbol{0}, \boldsymbol{\Sigma}_{\varepsilon 1}\right)$$

Model 2:

$$\left(\boldsymbol{I} - \boldsymbol{A}_3 L - \boldsymbol{A}_4 L^2\right) \boldsymbol{y}_t = \boldsymbol{\varepsilon}_t, \quad \boldsymbol{\varepsilon}_t \sim N\left(\boldsymbol{0}, \boldsymbol{\Sigma}_{\varepsilon 2}\right)$$

where L is the lag operator such that $L^d y_t = y_{t-d}$ and

$$\boldsymbol{A}_1 = \begin{pmatrix} a_{11} & a_{12} \\ 0 & a_{13} \end{pmatrix}, \quad \boldsymbol{A}_2 = \begin{pmatrix} a_{11} & 0 \\ 0 & a_{22} \end{pmatrix}, \quad \boldsymbol{\Sigma}_{\varepsilon 1} = \begin{pmatrix} 1 & \rho \\ \rho & 1 \end{pmatrix},$$

$$\boldsymbol{A}_3 = \begin{pmatrix} a_{31} & 0 & 0 \\ a_{32} & 0 & 0 \\ 0 & a_{33} & a_{34} \end{pmatrix}, \quad \boldsymbol{A}_4 = \begin{pmatrix} 0 & 0 & 0 \\ 0 & a_{41} & 0 \\ a_{42} & 0 & 0 \end{pmatrix}, \quad \boldsymbol{\Sigma}_{\varepsilon 2} = \begin{pmatrix} 1 & \rho & \rho \\ \rho & 1 & \rho \\ \rho & \rho & 1 \end{pmatrix}.$$

Model 0 is a two dimensional VAR(1) model, Model 1 is a two-dimensional seasonal model with period 4 and can be considered as a sparse vector autoregressive model of order 5 i.e. VAR(5), and Model 2 is a three-dimensional VAR(2) model. The elements of matrices A_1, A_2, A_3 and A_4 are randomly selected from $U(0.5, 1)$, where $U[a, b]$ represents a uniform distribution with parameters a and b.

We have

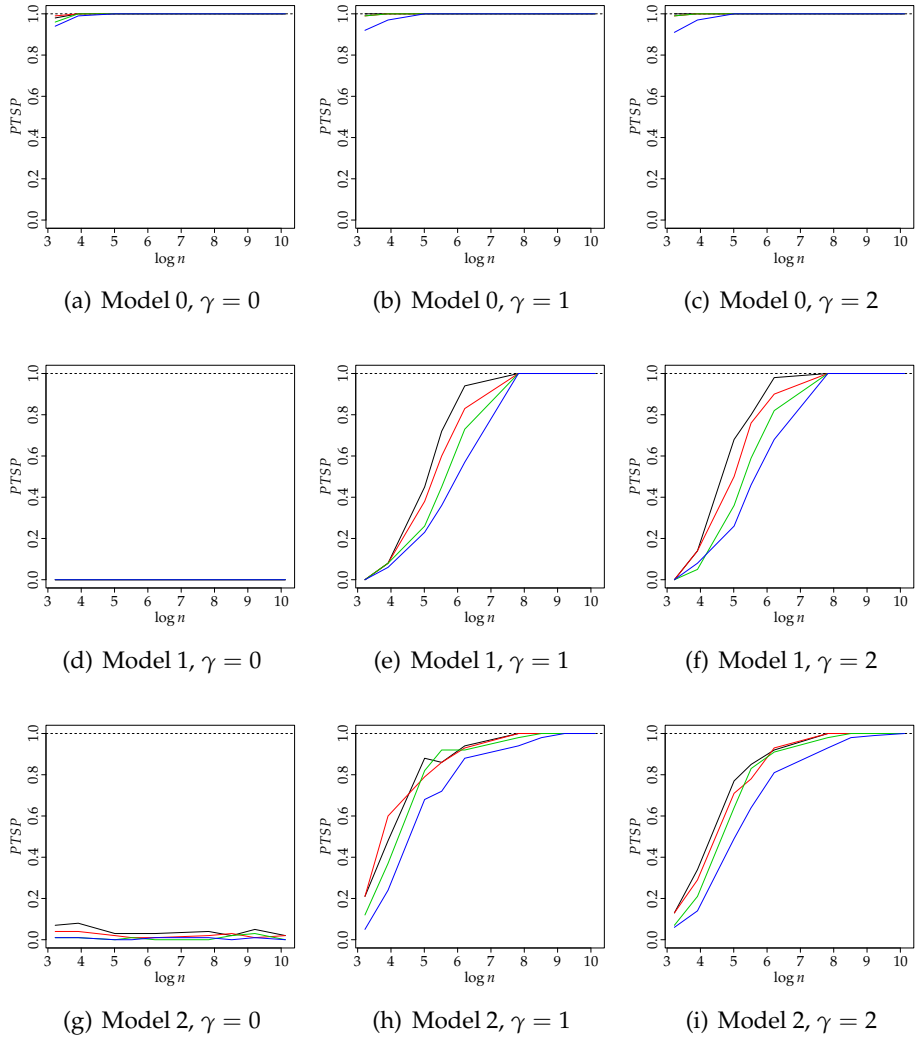
$$\boldsymbol{A}_1 = \begin{pmatrix} 0.6148036 & 0.9161782 \\ 0 & 0.8834940 \end{pmatrix}, \quad \boldsymbol{A}_2 = \begin{pmatrix} 0.9817781 & 0 \\ 00.6621272 & \end{pmatrix},$$

$$\boldsymbol{A}_3 = \begin{pmatrix} 0.9844588 & 0 & 0 \\ 0.9785033 & 0 & 0 \\ 0 & 0.6834209 & 0.781373 \end{pmatrix}, \quad \boldsymbol{A}_4 = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0.7862824 & 0 \\ 0.6825497 & 0 & 0 \end{pmatrix}.$$

For the selected above choices, the ZYZ condition holds for Model 0 but not for Model 1 and Model 2. In order to look at the effect of correlated errors, we will consider various choices in the numerical results in the next section viz. $\rho = 0$, $\rho = 0.4$, $\rho = 0.7$ and $\rho = 0.9$.

6.6.1 Variable Selection

In Section 5.5.1, we have already seen that lasso-type methods, under certain conditions, can achieve consistent variable selection for regression models. Now on the same lines, we will study these properties for multivariate time series models. To illustrate the theoretical properties we have proved in Section 6.4, we will look at the various VAR models defined above. We consider different samples sizes ranging 50 to 50000.



(a) Model 0, $\gamma = 0$ (b) Model 0, $\gamma = 1$ (c) Model 0, $\gamma = 2$

(d) Model 1, $\gamma = 0$ (e) Model 1, $\gamma = 1$ (f) Model 1, $\gamma = 2$

(g) Model 2, $\gamma = 0$ (h) Model 2, $\gamma = 1$ (i) Model 2, $\gamma = 2$

Figure 6.1: PTSP: Probability of true model lying on solution path, based on 100 runs, that solution path of the lasso and the adaptive lasso contains the true model for the three models defined in Section 6.6. Key: $\rho = 0$, $\rho = 0.4$; $\rho = 0.7$; $\rho = 0.9$.

Figure 6.1, shows the probability that the solution path of the lasso ($\gamma = 0$) and adaptive lasso ($\gamma = 1, 2$) contain the true model for 100 Monte Carlo runs. As the ZYZ condition holds for this Model 0, we can see that, for both the lasso and adaptive lasso, this probability is one or close to one even for the smallest sample size considered. It can be seen that the results are not substantially different even if the errors are correlated.

For Model 1 and Model 2, the ZYZ condition fails and we can see that the plots for both of these two models do not differ much. It can be concluded that the lasso cannot be consistent in variable selection as the probability of containing the true model on

the solution path is zero or very close to zero. In contrast, for the adaptive lasso, this probability rapidly approaches one, which indicates that the adaptive lasso is consistent in variable selection if an appropriate value of the tuning parameter is selected. As expected, we can see that for small sample sizes, uncorrelated errors are least challenging for consistent variable selection. In general, higher error correlation corresponds to lower probability but the asymptotic results are almost equivalent for all levels of correlation.

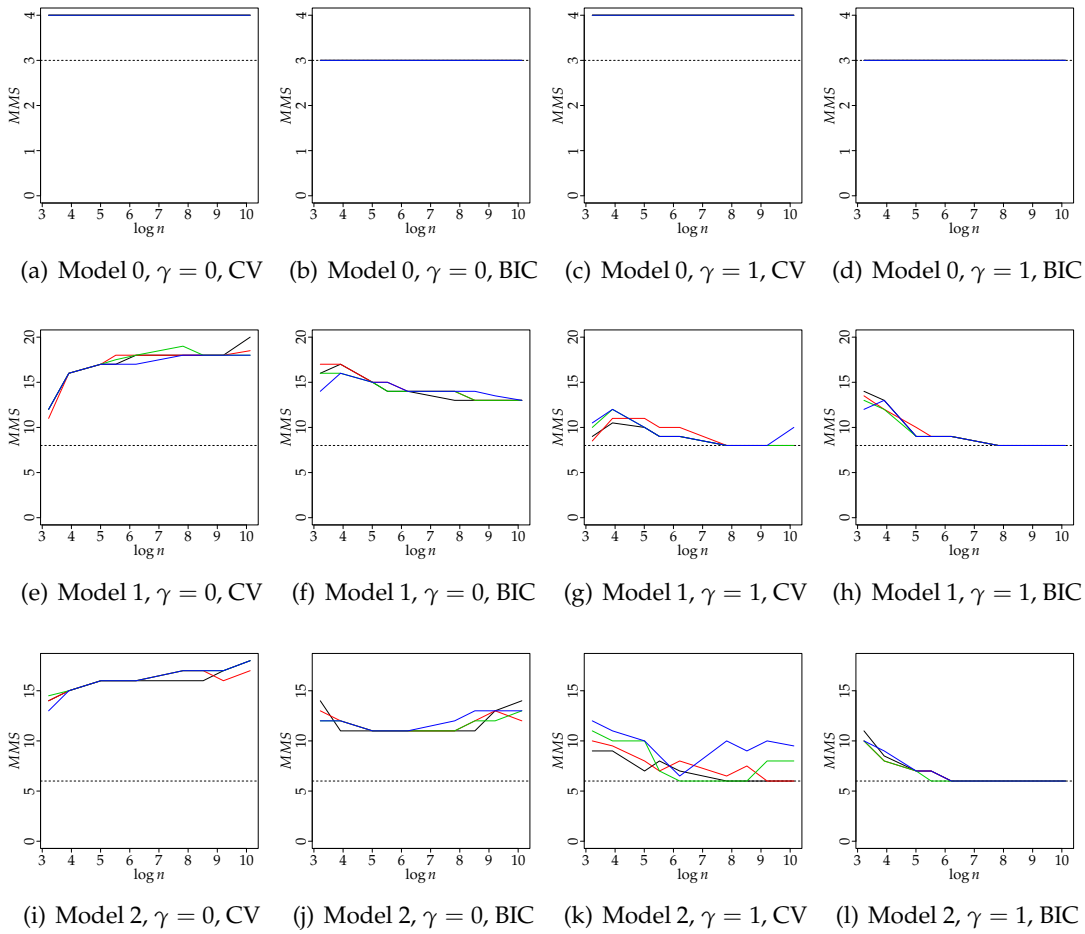
We have concluded from Figure 6.1 that for Model 0, both the lasso and adaptive lasso should be consistent in variable selection. Also for Model 1 and Model 2, we have seen that the lasso cannot be consistent in variable selection but the adaptive lasso can be. If a method is potentially consistent in variable selection in the sense that the solution path contains the correct model then consistency entirely depends on the tuning parameter selector providing an appropriate value of the tuning parameter corresponding to correct variables in model.

In the next section, as was done for regression models, we will compare the performance of 5-fold cross-validation and BIC ($C_n = \sqrt{n}/k$) in selecting the appropriate value of the tuning parameter. We will look at the performance measures MMS, PCM for consistent variable selection and $MRME$ for prediction accuracy. All these measures are defined earlier in Section 5.5.

6.6.2 Estimation of the Tuning Parameter

As stated earlier in Section 5.2, lasso-type methods shrink some model coefficients exactly to zero. This amount of shrinkage depends on the value of the tuning parameter. Higher values of the tuning parameter $s \in [0, 1]$ corresponds to less shrinkage. Suppose τ is the appropriate value of the tuning parameter for which we have $|\mathcal{S}_\tau| = k_0$ and $\mathcal{A}_n = \mathcal{A}$, where $|\mathcal{S}_\tau|$ stands for model size for τ as the value of the tuning parameter.

If a tuning parameter selector has a tendency to select a value of s, say $s > \tau$, it will result in entering non-active predictors in the model. On the other hand, if we have $|\mathcal{S}_\tau| = k_0$, it does not guarantee that the correct model is selected as it is possible for some non-active predictors in the model while some active predictors are dropped.



(a) Model 0, $\gamma = 0$, CV (b) Model 0, $\gamma = 0$, BIC (c) Model 0, $\gamma = 1$, CV (d) Model 0, $\gamma = 1$, BIC

(e) Model 1, $\gamma = 0$, CV (f) Model 1, $\gamma = 0$, BIC (g) Model 1, $\gamma = 1$, CV (h) Model 1, $\gamma = 1$, BIC

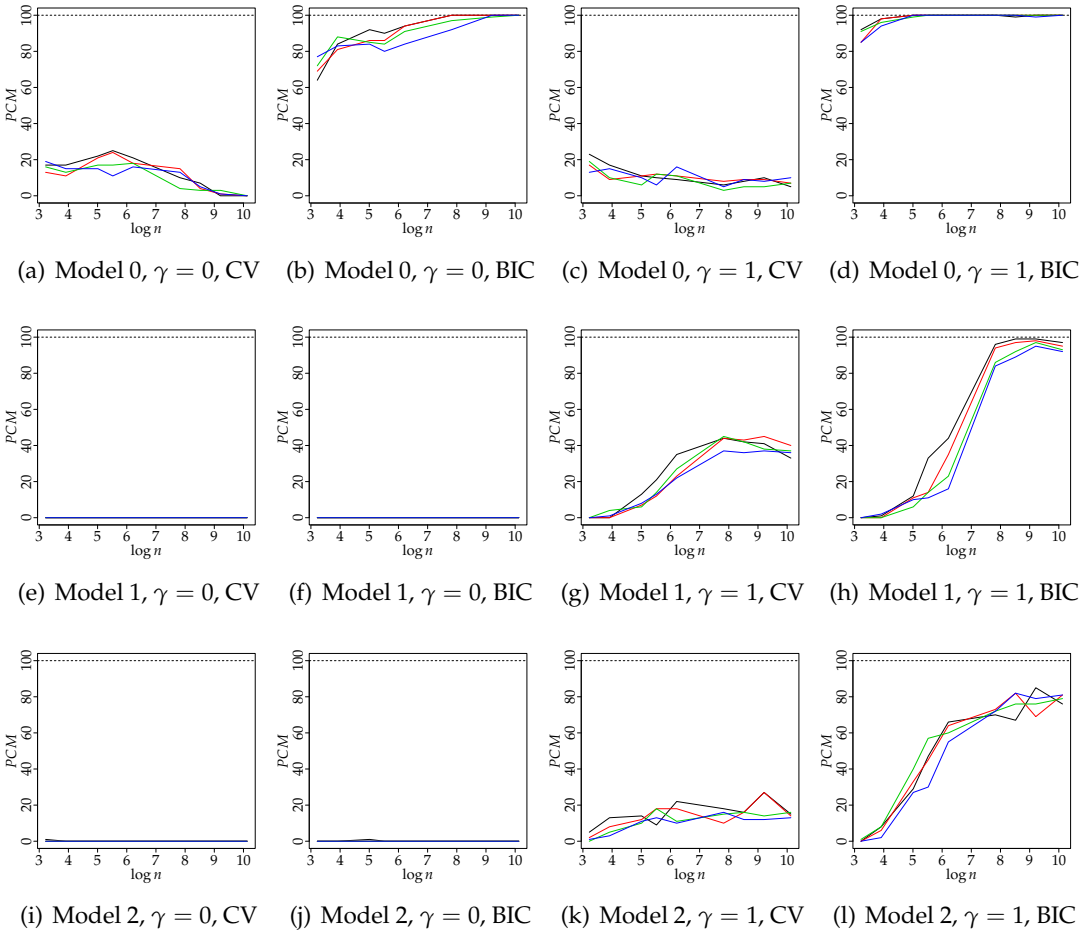
(i) Model 2, $\gamma = 0$, CV (j) Model 2, $\gamma = 0$, BIC (k) Model 2, $\gamma = 1$, CV (l) Model 2, $\gamma = 1$, BIC

Figure 6.2: MMS: Median model size estimated, based on 100 runs, by the lasso and adaptive lasso while the tuning parameter is selected by 5-fold cross-validation and Wang and Leng (2009) BIC ($Cn = \sqrt{n}/k$) for the three models defined in Section 6.6. Key: ■ $\rho = 0$, ■ $\rho = 0.4$; ■ $\rho = 0.7$; ■ $\rho = 0.9$.

This can happen especially if some of the parameters are close to zero.

In this section, first we will look at the median model size corresponding to the tuning parameter selected by cross-validation and BIC.

Figure 6.2 shows the median model size corresponding to the value of the tuning parameter selected by cross-validation and BIC for 100 Monte Carlo runs. As we concluded earlier in the discussion on the probability of containing the true model on the solution path, both the lasso and adaptive lasso are consistent in variable selection, so we should have the model size equal to k_0. It can be clearly observed that cross-validation results in overfitted models both for the lasso and adaptive lasso while BIC is providing the tuning parameter for which we are getting the true model size.



(a) Model 0, $\gamma = 0$, CV (b) Model 0, $\gamma = 0$, BIC (c) Model 0, $\gamma = 1$, CV (d) Model 0, $\gamma = 1$, BIC

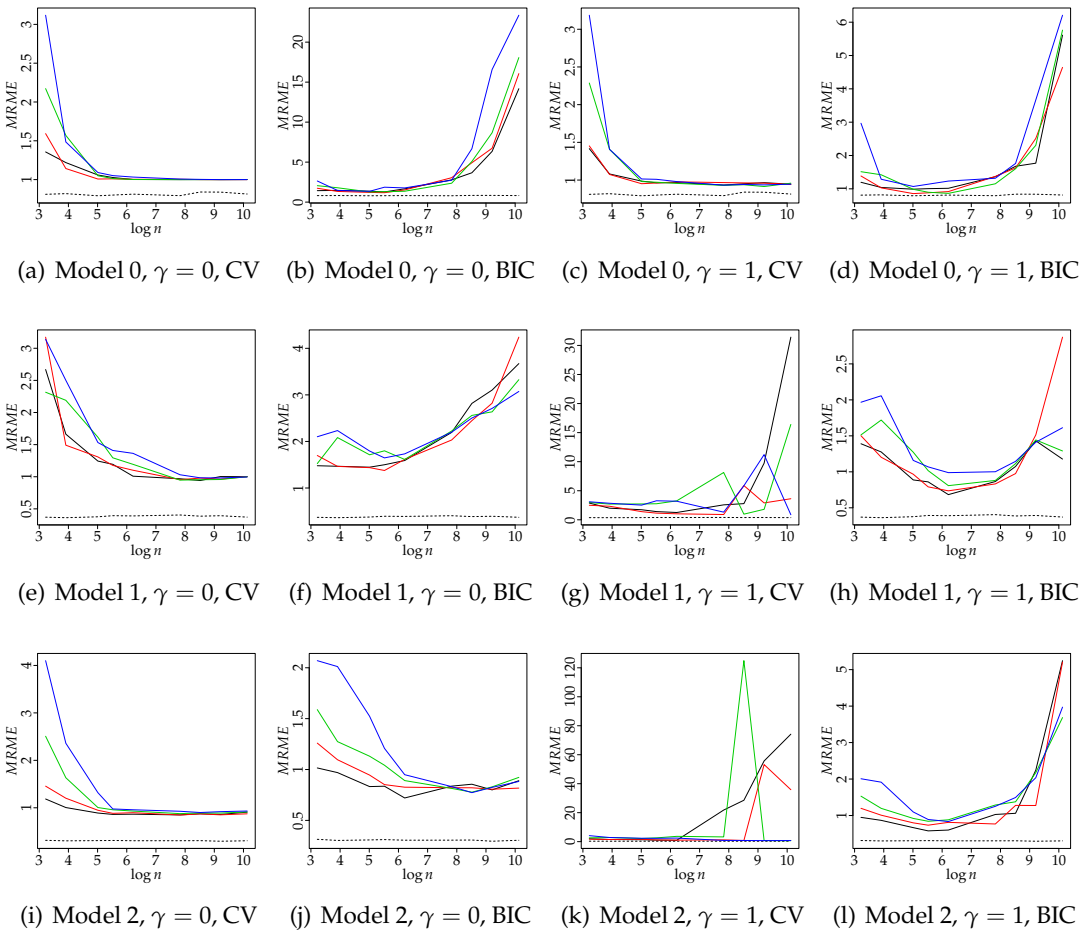
(e) Model 1, $\gamma = 0$, CV (f) Model 1, $\gamma = 0$, BIC (g) Model 1, $\gamma = 1$, CV (h) Model 1, $\gamma = 1$, BIC

(i) Model 2, $\gamma = 0$, CV (j) Model 2, $\gamma = 0$, BIC (k) Model 2, $\gamma = 1$, CV (l) Model 2, $\gamma = 1$, BIC

Figure 6.3: PCM: Percent correct models identified, based on 100 runs, by the lasso and adaptive lasso while the tuning parameter is selected by CV (5-fold cross-validation) and Wang and Leng (2009) BIC ($C_n = \sqrt{n}/k$) for the three models defined in Section 6.6. Key: $\rho = 0$, $\rho = 0.4$; $\rho = 0.7$; $\rho = 0.9$.

For Model 1 and Model 2 the lasso has not shown consistent variable selection in the previous section and in Figure 6.2 both cross-validation and BIC are providing an estimate of the tuning parameter corresponding to over-fitted models, so this situation can be considered as a consequence of the failure of the ZYZ condition. For the adaptive lasso, cross-validation results in an median model size close to the true model size but the results are more consistent when BIC is used as a tuning parameter selector.

We have seen that BIC is able to estimate a value of the tuning parameter which results in correct model size in all the three examples. For the last two models also the estimated model size using cross-validation is approaching the correct model size. Now with the plots of measure PCM, we will see, though the model size is correct, if we are getting only the active predictors in the estimated model.



(a) Model 0, $\gamma = 0$, CV (b) Model 0, $\gamma = 0$, BIC (c) Model 0, $\gamma = 1$, CV (d) Model 0, $\gamma = 1$, BIC

(e) Model 1, $\gamma = 0$, CV (f) Model 1, $\gamma = 0$, BIC (g) Model 1, $\gamma = 1$, CV (h) Model 1, $\gamma = 1$, BIC

(i) Model 2, $\gamma = 0$, CV (j) Model 2, $\gamma = 0$, BIC (k) Model 2, $\gamma = 1$, CV (l) Model 2, $\gamma = 1$, BIC

Figure 6.4: MRME: Median of relative model error, based on 100 runs, by the lasso and adaptive lasso while the tuning parameter is selected by CV (5-fold cross-validation) and Wang and Leng (2009) BIC ($C_n = \sqrt{n}/k$) for the three models defined in Section 6.6. Key: $\rho = 0$, $\rho = 0.4$; $\rho = 0.7$; $\rho = 0.9$.

Figure 6.3 shows the percentage of correct models identified by cross-validation and BIC for 100 Monte Carlo runs. As we have seen in the MMS plots that cross-validation results in overfitted models, we can see the PCM is very low with cross-validation but approaching the maximum of 100% with BIC both for the lasso and adaptive lasso. For Model 1 and Model 2, the only situation where we are getting near to 100% is the adaptive lasso with BIC as a tuning parameter selector.

The above results confirm our earlier findings in Section 5.5 that the ZYZ condition is an important condition for consistent variable selection and rescaling of the predictors with the adaptive weights always leads to the adaptive lasso satisfying this condition. Moreover, cross-validation fails to provide a value of the tuning parameter for which variable selection can be consistent.

Now we move to look at the second oracle property of prediction accuracy of the lasso methods. In the following paragraphs, we will provide plots on the results of median of relative model error (MRME); see Section 5.5 for definition.

Figure 6.4 shows the MRME for the lasso and adaptive lasso both with cross-validation and BIC as tuning parameter selector. It can be observed that with cross-validation MRME decreases as sample size increases, but we observe some large values of MRME with BIC as sample size increases. This behaviour may be due to this particular choice of C_n, as larger values of C_n will tend to result in a higher amount of shrinkage even for the active set of predictors. A more suitable choice of C_n might achieve low values of MRME even for some large choices of sample size.

6.7 Conclusion

In this chapter, we proved the necessary condition for consistent variable selection by lasso-type methods for multivariate time series models. Like for regressive models, our results suggest that ZYZ condition is an important condition for lasso-type methods to do consistent variable selection. This condition always hold for the adaptive lasso but not for the lasso.

We also proved that the adaptive lasso is an oracle procedure for time series models and our numerical results support these findings. As this condition always hold for the adaptive lasso so it does consistent variable selection if an efficient tuning parameter selector, like BIC, is used. But this consistent selection can be at the cost of increased model error.

Moreover, in the case of correlated errors, high correlation among the error terms makes the situation relatively harder but the lasso-type methods a still able to achieve the oracle properties under certain conditions.

CHAPTER 7

Summary, Conclusions and Topics for Further Research

7.1 Summary and Discussion

In this chapter, we present a summary of our findings and some possible further extensions of this project. In this thesis, we looked at two separate but related applications of time series models. In the first part, we have looked at time series diagnostic testing tools. In the second part, we first numerically studied the oracle properties, especially consistent variable selection, for linear regression models. We then studied the application of lasso methods to multivariate time series models through some theoretical results and also gave some numerical examples to illustrate the theory.

We found that the dynamic bootstrap method is the best method among the considered semi-parametric bootstrap methods in providing an approximate distribution of the diagnostic tests. Our results show that there is not any clear advantage of using transformed, or wild, residuals for bootstrapping.

We also found that both the Ljung-Box (Ljung and Box, 1978) and Monti's (Monti, 1994) test statistics suffer from location bias but our results show that the amount of bias in Monti's test is relatively low. Our results also confirm the finding that the Ljung and Box (1978) suggestion corrects the location bias in the Box and Pierce (1970) test but at the cost of increased variance. In our study comparing bootstrap methods, the dynamic bootstrap comes out superior to the fixed design bootstrap method.

Though in some cases for the $CvM_{exp,P}$ statistic proposed by Escanciano (2007), the fixed design bootstrap method has shown better performance but, in general, we cannot see any obvious advantage of using the fixed design bootstrap.

In the comparison of power properties of diagnostic tests, the portmanteau tests have shown more power against the linear alternatives while the $CvM_{exp,P}$ statistic has shown more power against non-linear alternatives. Our results suggest that the approximation of the finite sample distribution and power of these tests highly depends on the choice of these parameters, P and m.

Issues in Chapter 2, namely bias in portmanteau tests and choice of m, motivated us to have an in depth look into them. In Chapter 3, we have seen that bias in portmanteau tests is enormous when m is small and the process is near the stationarity boundary. We confirmed, as found by Katayama (2008), that Katayama's suggestion corrects the bias in the Ljung-Box test under these conditions.

The conditions mentioned above are also the situation where Monti's test also shows a large amount of bias. We made a novel suggestion, along the lines of Katayama (2008), to correct the bias in Monti (1994) suggested test which uses partial autocorrelations. Numerical results show that this suggestion works. We also gave a novel result that dynamic bootstrapping does an automatic bias correction in these portmanteau tests. As the computation of the bias correction term, especially for higher order processes, is not very simple, we suggested a novel algorithm to efficiently compute this bias correction term.

We noticed that bias arises due to poor approximation of the information matrix which depends on the choice of m. For diagnostic purposes, in order to automatically correct the bias in the challenging case of a near stationary process with small m, we made a novel suggestion to use pivotal portmanteau test with two different values of m, a relatively large value of m for the computation of the information matrix which corrects the bias and then using a small value of m, i.e. the number of autocorrelations used for diagnostic test purposes, to achieve a good approximation of the asymptotic distribution. Our numerical results showed that this novel suggestion corrects the bias as well as Katayama's suggestion does. We also looked at another suggestion by

Katayama (2009), to use a multiple test which enables the use of a range of values for m, to deal with the choice of m. We made a novel suggestion to use a hybrid bootstrap method to compute the joint significance levels of the test. Results show that our suggestion is easy to implement and performs, in some cases, better than Katayama's method.

In Chapter 2, we found from the numerical examples that the dynamic bootstrap provided a distribution of portmanteau tests which is more accurate than the first order asymptotic distribution. In Chapter 4, we provide a theoretical justification of good performance of dynamic bootstrap method. We have stated and proved a number of lemmas to show that the distribution of bootstrap least squares estimates converges in limit to that of least squares estimates. We also proved a martingale central limit theorem for the residuals. Though the result in this theorem is already proved in the literature but the use of martingale theory helps to apply these results to the dynamic bootstrap method. In this same chapter, we also gave a theoretical derivation of bias correction term we suggested in Monti's test.

Chapter 5 is the first of two chapters in the second part of our thesis, where, through numerical examples, we have looked at the oracle properties of the lasso methods using three different examples. We have compared the performance of the lasso and the adaptive lasso. Our results show that the ZYZ condition is an important condition for consistent variable selection for the lasso and adaptive lasso. We have seen that the lasso can be consistent in variable selection when the ZYZ condition holds provided that an appropriate value of the tuning parameter is selected. It should be noted that the ZYZ condition always holds for the adaptive lasso due to the use of adaptive weights and thus it showed consistent variable selection in all the cases.

Tuning parameter selection is an important practical problem and can greatly effect the performance of the lasso methods. We compared cross-validation, a popular method for tuning parameter selection, with the Wang and Leng (2009) suggestion of using the Bayesian Information Criterion (BIC). The numerical results suggest that cross-validation is not a reliable method especially if the primary objective is variable selection. In all situations considered, our results suggest that for both the lasso and adaptive lasso, using cross-validation as a tuning parameter selector, leads to inconsis-

tent variable selection. Meanwhile, the BIC has shown its capability to choose a value for the tuning parameter which correctly shrinks the coefficients of non-active predictors to zero.

The success of lasso methods for regression models motivated us to look at the properties of these methods for multivariate time series models. The results are not trivial as time series models differ in their structure to regression models because of serial dependence. In Chapter 6, we have proved the necessary condition for consistent variable selection by the lasso for time series models. We also proved that the adaptive lasso is an oracle procedure for time series models and our numerical results support these findings. An efficient method for selection of the tuning parameter is important to achieve these properties in practice and the numerical results show that with the BIC, as tuning parameter selector, we can achieve it.

7.2 Future Work

In the first part of this thesis we looked at different semi-parametric bootstrap methods for providing an approximation to the asymptotic distribution of time series diagnostic tests. We found that, among the methods considered, the dynamic bootstrap generally provided the best approximation. This work can be further extended by considering some non-parametric methods such as block bootstrap methods (Lahiri, 2003). We looked at stationary AR processes in this work, so study of other classes of stationary model e.g. general ARMA and nonlinear models would be of interest.

In our size study, in Chapter 2, we considered the case of a linear model and looked at examples of AR processes. Some other examples of mixed models and also some non-linear models will be helpful to have a further extension of the results obtained in this thesis. An important objective was to compare the performance of the CvM statistic with the Box-Pierce family of portmanteau tests, so we limited ourselves to some popular tests which are based on autocorrelations and partial autocorrelations. Obviously, there are some other tests like CvM with different weighting schemes (Escanciano, 2007) and portmanteau tests based on autocorrelations of squared residuals (McLeod and Li, 1983).

In Chapter 3, we discussed the suggestions to correct the bias in portmanteau tests in some challenging conditions. We also made novel suggestions for automatic correction of bias in the Ljung-Box test and a bias correction term in Monti's test. Bootstrap estimates of standard errors of the estimates for our novel suggested Algorithm 7 can be obtained to compare its performance with the other available estimation methods. All the results we presented in this chapter are mainly for AR(1) process. Moreover, the effect of these suggestions is only studied in correcting the size of these tests. A further study is required to look at the power properties of these bias corrected tests.

We proved the asymptotic distribution of dynamic bootstrap least squares estimates for stationary AR(p) process. An obvious further extension is to prove these results for an ARMA(p, q) process. It would also be of interest to explore higher order properties of the bootstrap distribution, as discussed in Chapter 4, although this is likely to be challenging.

In the second part of our thesis, we studied the oracle properties of the lasso and adaptive lasso. As we found that the use of adaptive weights is the key in making the adaptive lasso an oracle procedure, so obviously the choice of γ, the exponent in the weight function, is important. Zou (2006) suggested the use of cross-validation for choosing the value of γ. A detailed study along those lines could help us to obtain an optimal value of γ. In this thesis, we computed the adaptive weights using the least squares estimates. Other suggestions, for example the lasso and ridge regression estimates, can be considered and compared with these results.

Another important factor in achieving the oracle properties of lasso-type methods is the choice of the tuning parameter, which controls the size of the penalty term. Our results clearly show this fact. We compared two methods used as tuning parameter selectors viz. k-fold cross-validation and the BIC. The other forms of cross-validation such as generalized cross-validation and leave-one-out cross-validation can be studied and compared with these two methods.

In all the examples of regression and time series models we studied in Chapter 5 and Chapter 6, the use of BIC resulted in an efficient way to choose the value of the tuning parameter especially when the primary objective is variable selection. As

expected and observed from the results, the choice of C_n for the BIC is an issue and we made suggestion to use $C_n = \sqrt{n}/p$. This suggestion showed good performance in our examples, though theoretical insight into this suggestion still needs to be provided.

We looked at both oracle properties of lasso-type methods but we focused more on consistent variable selection. A detailed study of prediction error and an application to real data sets will be helpful.

Finally, in all these examples, we looked at low dimensional regression and time series models. A study of high dimensional models especially in the case of regression is an obvious matter of interest, and has potential applications in bioinformatics, for example.

References

Akaike, H. (1974). A new look at the statistical model identification. *IEEE transactions on automatic control*, 19(6):716–723.

Anderson, T. (1971). *The Statistical Analysis of Time Series*. John Wiley & Sons: New York, USA.

Anderson, T. and Walker, A. (1964). On the asymptotic distribution of the autocorrelations of a sample from a linear stochastic process. *The Annals of Mathematical Statistics*, 35(3):1296–1303.

Ansley, C. and Newbold, P. (1979). On the finite sample distribution of residual autocorrelations in autoregressive-moving average models. *Biometrika*, 66(3):547–553.

Bakin, S. (1999). Adaptive regression and model selection in data mining problems. *PhD Thesis, School of Mathematical Sciences, The Australian National University, Canberra*.

Bhattacharya, R. and Ghosh, J. (1978). On the validity of the formal Edgeworth expansion. *The Annals of Statistics*, 6(2):434–451.

Bhattacharya, R. and Rao, R. (1976). *Normal Approximation and Asymptotic Expansions*. John Wiley & Sons: New York, USA.

Bierens, H. (1982). Consistent model specification tests. *Journal of Econometrics*, 20(1):105–134.

Billingsley, P. (1979). *Probability and Measure*. John Wiley & Sons: New York, USA.

Bjorck, A. (1996). *Numerical Methods for Least Squares Problems*. Society for Industrial Mathematics: USA.

Box, G. and Jenkins, G. (1994). *Time Series Analysis, Forecasting and Control*. John Wiley & Sons: New Jersey, USA.

Box, G. and Pierce, D. (1970). Distribution of residual autocorrelations in autoregressive-integrated moving average time series models. *Journal of the American Statistical Association*, 65(332):1509–1526.

Breiman, L. (1995). Better subset selection using the non-negative garotte. *Technometrics*, 37(4):373–384.

Breiman, L. (1996). Heuristics of instability and stabilization in model selection. *The Annals of Statistics*, 24(6):2350–2383.

Brockwell, P. and Davis, R. (1991). *Time series: Theory and Methods*. Springer-Verlag: New York, USA.

Brockwell, P. and Davis, R. (2002). *Introduction to Time Series and Forecasting*. Springer-Verlag: New York, USA.

Brown, B. (1971). Martingale central limit theorems. *The Annals of Mathematical Statistics*, 42(1):59–66.

Brown, J. (1993). *Measurement, Regression and Calibration*. Oxford University Press: Oxford, UK.

Candes, E. and Tao, T. (2007). The Dantzig selector: statistical estimation when p is much larger than n. *Annals of Statistics*, 35(6):2313–2351.

Chatfield, C. (2004). *The Analysis of Time Series: An Introduction*. Chapman & Hall/CRC: London, UK.

Chernick, M. (1999). *Bootstrap Methods: A Practioner's Guide*. John Wiley & Sons: New York, USA.

Cho, H. and Fryzlewicz, P. (2010). High-dimensional variable selection via tilting. *http://stats.lse.ac.uk/fryzlewicz/tilt/tilt.pdf*.

Chung, K. (2001). *A Course in Probability*. Academic Press: San Diego, USA.

Davies, N., Triggs, C., and Newbold, P. (1977). Significance levels of the box-pierce portmanteau statistic in finite samples. *Biometrika*, 64(3):517–522.

Efron, B. (1979). Bootstrap methods: Another look at the jackknife. *The Annals of Statistics*, 7(1):1–26.

Efron, B. (1988). Computer-intensive methods in statistical regression. *SIAM Review*, 30(3):421–449.

Efron, B., Hastie, T., Johnstone, I., and Tibshirani, R. (2004). Least angle regression. *Annals of Statistics*, 32(2):407–499.

Escanciano, J. (2006a). A consistent diagnostic test for regression models using projections. *Econometric Theory*, 22(06):1030–1051.

Escanciano, J. (2006b). Goodness-of-fit tests for linear and nonlinear time series models. *Journal of the American Statistical Association*, 101(474):531–541.

Escanciano, J. (2007). Model checks using residual marked empirical process. *Statistica Sinica*, 17(1):115–138.

Fan, J. and Li, R. (2001). Variable selection via nonconcave penalized likelihood and its oracle properties. *Journal of the American Statistical Association*, 96(456):1348–1360.

Fan, J. and Lv, J. (2008). Sure independence screening for ultrahigh dimensional feature space. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 70(5):849–911.

Fan, J. and Lv, J. (2009). Non-concave penalized likelihood with NP-dimensionality. *Arxiv preprint arXiv:0910.1119*.

Fan, J. and Peng, H. (2004). Nonconcave penalized likelihood with a diverging number of parameters. *Annals of Statistics*, 32(3):928–961.

Fan, J. and Yao, Q. (2003). *Nonlinear Time Series: Nonparametric and Parametric Methods*. Springer Verlag: New York, USA.

Fisher, N., Hall, P., Jing, B., and Wood, A. (1996). Improved pivotal methods for constructing confidence regions with directional data. *Journal of the American Statistical Association*, 91(435):1062–1070.

Frank, I. and Friedman, J. (1993). A statistical view of some chemometrics regression tools. *Technometrics*, 35(2):109–135.

Fujita, A., Sato, J., Garay-Malpartida, H., Yamaguchi, R., Miyano, S., Sogayar, M., and Ferreira, C. (2007). Modeling gene expression regulatory networks with the sparse vector autoregressive model. *BMC Systems Biology*, 32(1).

Gonçalves, S. and Kilian, L. (2004). Bootstrapping autoregressions with conditional heteroskedasticity of unknown form. *Journal of Econometrics*, 123(1):89–120.

Götze, F. and Hipp, C. (1983). Asymptotic expansions for sums of weakly dependent random vectors. *Probability Theory and Related Fields*, 64(2):211–239.

Götze, F. and Hipp, C. (1994). Asymptotic distribution of statistics in time series. *The Annals of Statistics*, 22(4):2062–2088.

Gustafsson, M., Hornquist, M., and Lombardi, A. (2005). Constructing and analyzing a large-scale gene-to-gene regulatory network-lasso-constrained inference and biological validation. *IEEE/ACM Transactions on Computational Biology and Bioinformatics (TCBB)*, 2(3):254–261.

Hall, P. (1992). *The Bootstrap and Edgeworth Expansions*. Springer-Verlag: New York, USA.

Hall, P., Horowitz, J., and Jing, B. (1995). On blocking rules for the bootstrap with dependent data. *Biometrika*, 82(3):561.

Hall, P., Lee, E., and Park, B. (2009a). Bootstrap-based penalty choice for the lasso achieving oracle performance. *Statistica Sinica*, 19:449–471.

Hall, P., Titterington, D., and Xue, J. (2009b). Tilting methods for assessing the influence of components in a classifier. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 71(4):783–803.

Hannan, E. (1976). The asymptotic distribution of serial covariances. *The Annals of Statistics*, 4(2):396–399.

Hardle, W., Horowitz, J., and Kreiss, J. (2003). Bootstrap methods for time series. *International Statistical Review*, 71(2):435–459.

Hastie, T. and Efron, B. (2007). *lars: Least Angle Regression, Lasso and Forward Stagewise*. R package version 0.9-7.

Hastie, T., Taylor, J., Tibshirani, R., Walther, G., et al. (2007). Forward stagewise regression and the monotone lasso. *Electronic Journal of Statistics*, 1(1):1–29.

Hastie, T., Tibshirani, R., Friedman, J., Hastie, T., Friedman, J., and Tibshirani, R. (2001). *The Elements of Statistical Learning*. Springer-Verlag: New York, USA.

Haufe, S., Müller, K., Nolte, G., and Krämer, N. (2008). Sparse Causal Discovery in Multivariate Time Series. In *Proceedings of the NIPS08 workshop on causality*.

Hesterberg, T., Choi, N., Meier, L., and Fraley, C. (2008). Least angle and L1 penalized regression: A review. *Statistics Surveys*, 2:61–93.

Hocking, R. and Leslie, R. (1967). Selection of the best subset in regression analysis. *Technometrics*, 9(4):531–540.

Hoerl, A. and Kennard, R. (1970). Ridge regression: biased estimation for nonorthogonal problems. *Technometrics*, 12(1):55–67.

Hoeting, J., Madigan, D., Raftery, A., and Volinsky, C. (1999). Bayesian model averaging: a tutorial (with comments by M. Clyde, David Draper and EI George, and a rejoinder by the authors. *Statistical science*, 14(4):382–417.

Hong, Y. and Lee, T. (2003). Diagnostic checking for the adequacy of nonlinear time series models. *Econometric Theory*, 19(6):1065–1121.

Horowitz, J., Lobato, I., Nankervis, J., and Savin, N. (2006). Bootstrapping the Box–Pierce Q test: A robust test of uncorrelatedness. *Journal of Econometrics*, 133(2):841–862.

Hothorn, T., Bretz, F., and Westfall, P. (2008). Simultaneous Inference in General Parametric Models. *Biometrical Journal*, 3:346–363.

Hsu, N., Hung, H., and Chang, Y. (2008). Subset selection for vector autoregressive processes using Lasso. *Computational Statistics and Data Analysis*, 52(7):3645–3657.

James, G., Radchenko, P., and Lv, J. (2009). DASSO: Connections between the dantzig selector and lasso. *Journal of Royal Statistical Society, Series B*, 71(1):121–142.

James, W. and Stein, C. (1961). Estimation with quadratic loss. In *Proceedings of the Fourth Berkeley Symposium on Mathematical Statistics and Probability: held at the Statistical Laboratory, University of California, June 20-July 30, 1960*, page 361. University of California Press.

Katayama, N. (2008). An improvement of the portmanteau statistic. *Journal of Time Series Analysis*, 29(2):359–370.

Katayama, N. (2009). On multiple portmanteau tests. *Journal of Time Series Analysis*, 30(5):487–504.

Kheoh, T. and McLeod, A. (1992). Comparison of two modified portmanteau tests for model adequacy. *Computational Statistics & Data Analysis*, 14(1):99–106.

Knight, K. and Fu, W. (2000). Asymptotics for lasso-type estimators. *Annals of Statistics*, 28(5):1356–1378.

Künsch, H. (1989). The jackknife and the bootstrap for general stationary observations. *The Annals of Statistics*, 17(3):1217–1241.

Kwan, A. and Sim, A. (1996). Portmanteau tests of randomness and jenkins' variance-stabilizing transformation. *Economics Letters*, 50(1):41–49.

Lahiri, S. (1999). Theoretical comparisons of block bootstrap methods. *Annals of Statistics*, 27(1):386–404.

Lahiri, S. (2003). *Resampling Methods for Dependent Data*. Springer-Verlag: New York, USA.

Lahiri, S. (2010). Edgeworth expansions for studentized statistics under weak dependence. *The Annals of Statistics*, 38(1):388–434.

Leng, C., Lin, Y., and Wahba, G. (2006). A note on the lasso and related procedures in model selection. *Statistica Sinica*, 16(4):1273–1284.

Li, W. (2004). *Diagnostic Checks in Time Series*. Chapman & Hall/CRC: New York, USA.

Liu, R. (1988). Bootstrap procedures under some non-IID models. *The Annals of Statistics*, 16(4):1696–1708.

Ljung, G. (1986). Diagnostic testing of univariate time series models. *Biometrika*, 73(3):725–730.

Ljung, G. and Box, G. (1978). On a measure of lack of fit in time series models. *Biometrika*, 65(2):297–303.

Lv, J. and Fan, Y. (2009). A unified approach to model selection and sparse recovery using regularized least squares. *Annals of Statistics*, 37(6A):3498–3528.

MacKinnon, J. (2006). Bootstrap methods in econometrics. *The Economic Record*, 82(1):S2–S18.

Maekawa, K. (1985). Edgeworth expansion for the OLS estimator in a time series regression model. *Econometric Theory*, 1(2):223–239.

Mainassara, B. et al. (2009). Multivariate portmanteau test for structural VARMA models with uncorrelated but non-independent error terms. *MPRA Paper*.

Mann, H. and Wald, A. (1943). On the statistical treatment of linear stochastic difference equations. *Econometrica, Journal of the Econometric Society*, 11(3):173–220.

Mardia, K., Kent, J., and Bibby, J. (1979). *Multivariate Analysis*. Academic Press: London, UK.

McCullagh, P. (1987). *Tensor Methods in Statistics*. Chapman & Hall/CRC: London, UK.

McLeod, A. (1978). On the distribution of residual autocorrelations in box-jenkins models. *Journal of the Royal Statistical Society. Series B (Methodological)*, 40(3):296–302.

McLeod, A. and Li, W. (1983). Diagnostic checking ARMA fime series models using squared-residual autocorrelations. *Journal of Time Series Analysis*, 4(4):269–273.

McQuarrie, A. and Tsai, C. (1998). *Regression and Time Series Model Selection*. World Scientific Publishing Company: Singapore, Singapore.

Meteopolis, N. and Ulam, S. (1949). The Monte Carlo method. *Journal of the American Statistical Association*, 44(247):335–341.

Monti, A. (1994). A proposal for a residual autocorrelation test in linear models. *Biometrika*, 81(4):776–780.

Moran, P. (1953). The statistical analysis of the Canadian lynx cycle. I. Structure and prediction. *Australian Journal of Zoology*, 1:163–173.

Nardi, Y. and Rinaldo, A. (2008). Autoregressive process modeling via the lasso procedure. *Arxiv preprint arXiv:0805.1179*.

Nocedal, J. and Wright, S. (1999). *Numerical Optimization*. Springer-Verlag: New York, USA.

Pena, D. and Rodriguez, J. (2002). A powerful portmanteau test of lack of fit for time series. *Journal of the American Statistical Association*, 97(458):601–611.

Pierce, D. (1972). Residual correlations and diagnostic checking in dynamic-disturbance time series models. *Journal of the American Statistical Association*, 67(339):636–640.

Politis, D. and Romano, J. (1992). A circular block-resampling procedure for stationary data. *Exploring the limits of bootstrap*, pages 263–270.

Pötscher, B. and Schneider, U. (2009). On the distribution of the adaptive lasso estimator. *Journal of Statistical Planning and Inference*, 139:2775–2790.

Prothero, D. and Wallis, K. (1976). Modelling macroeconomic time series. *Journal of the Royal Statistical Society. Series A (General)*, 139(4):468–500.

Quenouille, M. (1947). A large-sample test for the goodness of fit of autoregressive schemes. *Journal of the Royal Statistical Society*, pages 123–129.

Radchenko, P. and James, G. (2008). Variable inclusion and shrinkage algorithms. *Journal of the American Statistical Association*, 103(483):1304–1315.

Robert, C. and Casella, G. (2004). *Monte Carlo Statistical Methods*. Springer-Verlag: New York, USA.

Schwarz, G. (1978). Estimating the dimension of a model. *The annals of statistics*, 6(2):461–464.

Seber, G. and Lee, A. (2003). *Linear Regression Analysis*. John Wiley & Sons: New York, USA.

Sen, P., Singer, J., and Predroso de Lima, A. (2010). *From Finite Sample to Asymptotic Methods in Statistics*. Cambridge University Press: New York, USA.

Shao, J. and Tu, D. (1995). *The Jackknife and Bootstrap*. Springer-Verlag: New York, USA.

Stute, W., Manteiga, W., and Quindimil, M. (1998). Bootstrap approximations in model checks for regression. *Journal of the American Statistical Association*, 93(441):141–149.

Tang, Z. and Mohler, R. (1988). *Bilinear Time Series: Theory and Application*. Springer-Verlag: New York, USA.

Taniguchi, M. and Kakizawa, Y. (2000). *Asymptotic Theory of Statistical Inference for Time Series*. Springer-Verlag: New York, USA.

Tibshirani, R. (1996). Regression shrinkage and selection via the lasso. *Journal of Royal Statistical Society, Series B*, 58(1):267–288.

Tibshirani, R. (1997). The lasso method for variable selection in the Cox model. *Statistics in Medicine*, 16(4):385–395.

Tong, H. (1990). *Nonlinear Time Series: a Dynamical System Approach*. Oxford University Press: Oxford, UK.

Tong, H. and Lim, K. (1980). Threshold autoregression, limit cycles and cyclical data. *Journal of the Royal Statistical Society. Series B (Methodological)*, 42(3):245–292.

Van De Geer, S. (2008). High-dimensional generalized linear models and the lasso. *Annals of Statistics*, 36(2):614–645.

Van der Vaart, A. (2000). *Asymptotic Statistics*. Cambridge University Press: Cambridge, UK.

Wang, H. and Leng, C. (2007). Unified lasso estimation via least squares approximation. *Journal of the American Statistical Association*, 102(479):1039–1048.

Wang, H. and Leng, C. (2009). Shrinkage tuning parameter selection with a diverging number of parameters. *Journal of Royal Statistical Society, Series B*, 71(3):671–683.

Wang, H., Li, R., and Tsai, C. (2007). Tuning parameter selectors for the smoothly clipped absolute deviation method. *Biometrika*, 94(3):553–568.

Wei, W. (2006). *Time Series Analysis: Univariate and Multivariate Methods*. Addison-Wesley: New York, USA.

Williams, D. (1991). *Probability with Martingales*. Cambridge University Press: Cambridge, UK.

Yuan, M. and Lin, Y. (2006). Model selection and estimation in regression with grouped variables. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 68(1):49–67.

Zhao, M. and Kulasekera, K. (2006). Consistent linear model selection. *Statistics & Probability Letters*, 76(5):520–530.

Zhao, P. and Yu, B. (2006). On model selection consistency of lasso. *The Journal of Machine Learning Research*, 7:2541–2563.

Zou, H. (2006). The adaptive lasso and its oracle properties. *Journal of the American Statistical Association*, 101(476):1418–1429.

Zou, H. and Hastie, T. (2005). Regularization and variable selection via the elastic net. *Journal of the Royal Statistical Society Series B*, 67(2):301–320.