Western Graduate&PostdoctoralStudies

Western University
Scholarship@Western

Electronic Thesis and Dissertation Repository

8-22-2014 12:00 AM

The Doubly Adaptive LASSO Methods for Time Series Analysis

Zi Zhen Liu, *The University of Western Ontario*

Supervisor: Reg J. Kulperger, *The University of Western Ontario*
Joint Supervisor: Hao Yu, *The University of Western Ontario*
A thesis submitted in partial fulfillment of the requirements for the Doctor of Philosophy degree in Statistics and Actuarial Sciences
© Zi Zhen Liu 2014

Follow this and additional works at: https://ir.lib.uwo.ca/etd

 Part of the Statistics and Probability Commons

Recommended Citation

Liu, Zi Zhen, "The Doubly Adaptive LASSO Methods for Time Series Analysis" (2014). *Electronic Thesis and Dissertation Repository*. 2321.
https://ir.lib.uwo.ca/etd/2321

This Dissertation/Thesis is brought to you for free and open access by Scholarship@Western. It has been accepted for inclusion in Electronic Thesis and Dissertation Repository by an authorized administrator of Scholarship@Western. For more information, please contact wlswadmin@uwo.ca.

THE DOUBLY ADAPTIVE LASSO METHODS FOR TIME SERIES ANALYSIS

(Thesis format: Monograph)

by

Zi Zhen Liu

Graduate Program in Statistics and Actuarial Science

A thesis submitted in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy

The School of Graduate and Postdoctoral Studies
The University of Western Ontario
London, Ontario, Canada

© Zi Zhen Liu 2014

Abstract

In this thesis, we propose a systematic approach called the doubly adaptive LASSO tailored to time series analysis, which includes four specific methods for four time series models, respectively:

The *PAC-weighted adaptive LASSO for univariate autoregressive (AR) models*. Although the LASSO methodology has been applied to AR models, the existing methods in the literature ignore the temporal dependence information embedded in AR time series data. Consequently, the methods may not reflect the characteristics of underlying AR processes, especially, the lag order of AR models. The PAC-weighted adaptive LASSO incorporates the partial autocorrelation (PAC) into the adaptive LASSO weights. The PAC-weighted adaptive LASSO estimator has asymptotic oracle properties and a Monte Carlo study shows promising results.

The *PAC-weighted adaptive positive LASSO for autoregressive conditional heteroscedastic (ARCH) models*. We have not found any results in the literature that apply the LASSO methodology to ARCH models. The PAC-weighted adaptive positive LASSO incorporates the PAC information embedded in squared ARCH process into adaptive LASSO weights. The word *positive* reflects the fact that the parameters in ARCH models are non-negative. We introduce a new concept named the surrogate of the second-order approximate likelihood, and propose a modified shooting algorithm to implement the PAC-weighted adaptive positive LASSO computationally. The PAC-weighted adaptive positive LASSO estimator has asymptotic oracle properties and a Monte Carlo study shows promising results.

The *PLAC-weighted adaptive LASSO for vector autoregressive (VAR) models*. Although the LASSO methodology has been applied to building VAR time series models, the existing methods in the literature ignore the temporal dependence information embedded in VAR time series data. Consequently, the methods may not reflect the characteristics of VAR time series data, especially, the lag order of VAR models. The PLAC-weighted adaptive LASSO incorporates the partial lag autocorrelation (PLAC) into the adaptive LASSO weights. The PLAC-weighted adaptive LASSO estimator has oracle properties and Monte Carlo studies show promising results.

The *PLAC-weighted adaptive LASSO for BEKK vector ARCH (VARCH) models*. We have not found any results in the literature that apply the LASSO methodology to VARCH processes. We focus on the BEKK VARCH models. The PLAC-weighted adaptive LASSO incorporates the PLAC information embedded in the squared BEKK VARCH process into the adaptive LASSO weights. We extend the concept of the surrogate of the second-order approximate likelihood, and propose a modified shooting algorithm to implement the PLAC-weighted adaptive LASSO computationally. We conduct a Monte Carlo study and have preliminary results from the study.

Keywords: Time series, financial time series, data mining, oracle property, LASSO, adaptive LASSO, doubly adaptive LASSO, positive LASSO, PAC-weighted adaptive LASSO, PAC-weighted adaptive positive LASSO, PLAC-weighted adaptive LASSO, autoregressive, AR(P), autoregressive conditional heteroscedastic, ARCH(q), vector autoregressive, multivariate autoregressive, VAR(p), vector ARCH, multivariate ARCH, VARCH(q), analytical score, analytical Hessian, quadratic approximation, surrogate to approximate likelihood, S&P 500, Nikkei.

Dedicated to my:

Mother Rangguo Liu
Father (Jinyuan Jia)
Brothers Jiabao Liu and Sanbao Liu
Sisters Jiaqiao Liu and (Jiayi Liu)
Wife Qingjun Zou
Daughters Dana Liu and Janet Liu

Acknowlegements

"Glory to God in the highest" (Luke 2:14 NIV)

I express my sincere gratitude to my supervisors Dr. Reg J. Kulperger and Dr. Hao Yu for their great support, encouragement, advice and guidance over the years of my doctoral research that have led to the completion of this dissertation. They suggested this interesting topic to me at the very start of my doctoral research. This dissertation would not have been possible without their initial suggestion of my research direction.

I thank very much the external examiner Dr. Zhou Zhou, and the examination committee members, Dr. Pei Yu, Dr. Ian McLeod, and Dr. Duncan Murdoch for reviewing my dissertation and offering valuable suggestions for improvement.

I thank the statistical community at the Department of Statistical and Actuarial Sciences of the University of Western Ontario. I learnt a lot from excellent lectures delivered by Dr. D. Bellhouse, Dr. W. J. Braun, Dr. W. He, Dr. R. J. Kulperger, Dr. I. McLeod, Dr. D. Murdoch, Dr. S. Provost, and Dr. H. Yu. Thanks also go to Ms. J. Bai and Ms. J. Dungavell for their logistic support.

I express my sincere gratitude to Dr. Nancy Reid for bringing me into the field of statistics by hiring me as a research assistant under her supervision. I thank the statistical community at the Department of Statistical Sciences of the University of Toronto. I learnt a lot from excellent lectures delivered by Dr. D. Brenner, Dr. M. Evans, Dr. A. Feuerverger, Dr. D. A. S. Fraser, and Dr. K. Knight, Dr. P. McDunnough, Dr. R. M. Neal, Dr. N. Reid, Dr. J. S. Rosenthal, Dr. J. Stafford, and Dr. F. Yao.

I express my deep love to my mother Rangguo Liu and my father (Jinyuan Jia) for their unfathomable, absolute and unqualified love to me. I express my deep love to my brothers and sisters Jiabao Liu, Jiaqiao Liu, (Jiayi Liu), and Sanbao Liu for their love, kindness, encouragement and help in my life. I express my deep love to my wife Qingjun Zou and my daughters Dana Liu and Janet Liu for their sacrifice and support during my extremely busy years of research.

Zi Zhen Liu
July 18, 2014
London, Ontario, Canada

Contents

List of Figures

List of Tables

List of Appendices

Chapter 1

Introduction

1.1 Parsimonious models and shrinkage

A large number of statistical models have linear structure. Say we have a data set with $x_{i1}, \cdots, x_{ip}$ being the inputs and y_i the outcome. The models with linear structure have the form

$$f(E[y_i|\boldsymbol{x}_i]) = \beta_1 x_1 + \cdots + \beta_p x_p, \tag{1.1}$$

where the input x_{ij} can be continuous, binary, or categorical, and f would have different functional forms depending on whether we have a classification or regression problem. If y_i is continuous, we often use the identity function for f, which is the linear regression model. If y_i is binary, we often use the logit function for f, which is the logistic regression model. If y_i is the number of occurrence of a rare event, we often use the Poisson regression model with the log function for f. If y_i is the hazard rate, we often use the log function for f, which is the Cox proportional hazard model. What is common in all these models is that they are all linear in the inputs $x_{i1}, \cdots, x_{ip}$. There are good reasons for these linear-structured models to be widely used. First, they are simple in functional structure and thus more interpretable than complex nonlinear models. Moreover, they often provide an adequate description of how the inputs affect the output. Finally, they sometimes outperform more complicated nonlinear models with regard to prediction.

When we fit the model to the data, we are not always satisfied with the full model, especially when the number of the inputs is large. The so-called *subset selection* or *variable selection* procedures eliminate the insignificant variables from the model while keeping those significant

ones in the model. We need parsimonious models for two reasons. The first is *prediction accuracy*. Prediction accuracy may be improved by excluding insignificant variables although the bias of estimators may be increased by doing so. With the full model, the estimators for the parameters have lower bias but larger variance. In contrast, for a parsimonious model, the estimators may have larger bias but smaller variance. We would like to sacrifice a little bit of bias but reduce the prediction variance with the net benefit of reduced mean squared error of prediction. This is the so-called the *bias-variance tradeoff*. The second reason for parsimonious models is the ease of *interpretation*. From a smaller model it is easier for us to see the inputs that have the strongest effects on the outcome.

Variable selection is a crucial but difficult problem in building statistical models. A large amount of research has been and continues to be devoted to this topic. There are a variety of subset selection strategies in the literature. (i) The *best subset* selection fits models of all possible subsets, then puts these models into categories corresponding to $0, 1, \cdots, p$ parameter models, then selects one from each category by minimal residual sum of squares which results in $p+1$ candidate models that contain $0, 1, \cdots, p$ variables, respectively, and finally chooses the model that satisfies some optimal criterion, say, the Akaike Information Criterion (AIC). This approach is feasible for moderate p. (ii) The *forward Stepwise* selection starts with the intercept, then sequentially adds into the model the variable that most improves the fit to yield $p+1$ nested candidate models, and finally chooses the model that satisfies the some optimal criterion like the AIC. (iii) The *backward Stepwise* selection starts with the full model, and sequentially removes the least significant variables. (iv) The *forward stagewise* selection starts with an intercept equal to $\bar{y}$, and includes the next variable that is most correlated with the current residual. The process continues until none of the variables have correlation with the current residuals and the final model is obtained.

Subset selection procedures are discrete in the sense that variables are either retained or discarded. They often have high variability (Breiman, 1996). Apart from instability issue they quickly become computationally infeasible as p becomes large. Methods using *shrinkage*, *regularization* or *penalization* are more continuous procedures, and thus do not have as much high variability, and can also better deal with algorithmic problems when p is large. The

penalization problem for the model (1.1) has the general form

$$\hat{\boldsymbol{\beta}} = \arg\min_{\boldsymbol{\beta}} \sum_{i=1}^{n} \ell((y_i, \boldsymbol{x}_i), \boldsymbol{\beta}) + \lambda_n P(\boldsymbol{\beta}), \tag{1.2}$$

where $\ell(y_i, \boldsymbol{x}_i'\boldsymbol{\beta})$ is a convex loss function, $P(\boldsymbol{\beta})$ is a penalty function, $\lambda_n > 0$ is the tuning parameter. Depending on the functional form of f in (1.1) there are many loss function, for example, the square error loss in linear regression, the negative log-likelihood in generalized linear models, the negative partial log-likelihood function in Cox proportional hazards model, and etc. The tuning parameter λ_n balances the loss and the penalty. It determines the bias-variance tradeoff by controlling the amount of penalty. A variety of penalty functions has been proposed in the literature. An example is the *Bridge* penalty function of Frank and Friedman (1993) defined as

$$P(\boldsymbol{\beta}) = \sum_{j=1}^{p} |\beta_j|^{\gamma}, \tag{1.3}$$

where $\gamma \geq 0$. While they did not solve for the Bridge estimator, Frank and Friedman pointed out that it is desirable to get the optimal value of γ. The Bridge penalty includes a few well known penalty functions as special cases. For $\gamma = 0$, $P(\boldsymbol{\beta})$ reduces to many well-known model selection criteria such as the Akaike Information criterion (AIC) and the Bayesian Information criterion (BIC). For $\gamma \in (0, 1]$, $P(\boldsymbol{\beta})$ is known as the *soft-thresholding* penalty (Donoho and Johnstone, 1994). Particularly for $\gamma = 1$, it is the penalty for the *least absolute shrinkage and selection operator* (LASSO) (Tibshirani, 1996). For $\gamma = 2$, it is the penalty for the *ridge regression* (Hoerl and Kennard, 1970).

In this dissertation, we focus on the the LASSO methodology only. We adapt the LASSO to the context of time series analysis and propose a doubly adaptive LASSO methodology for time series models.

1.2 The LASSO methodology

The LASSO (Tibshirani, 1996) is a celebrated breakthrough in the area of model selection. The LASSO becomes increasingly popular because its optimization objective is convex, it performs variable selection and parameter estimation simultaneously, and there exist efficient

algorithms. In this section, we review the definition, asymptotic properties, algorithms, and the irrepresentable condition, and the adaptive version of the LASSO.

1.2.1 The shrinkage mechanism

Consider the linear regression model, $y_i = \boldsymbol{x}_i\boldsymbol{\beta} + \epsilon_i$, where $\epsilon_1, \cdots, \epsilon_n$ are iid$(0, \sigma^2)$. Let $\boldsymbol{x}_i = (x_1, \cdots, x_p)'$ and $\boldsymbol{\beta} = (\beta_1, \cdots, \beta_p)$.

Definition (**The LASSO** (Tibshirani, 1996)). The LASSO estimator, denoted by $\hat{\boldsymbol{\beta}}_n^L$, is defined as

$$\hat{\boldsymbol{\beta}}_n^L = \arg\min_{\boldsymbol{\beta}} \sum_{i=1}^{n} (y_i - \boldsymbol{x}_i'\boldsymbol{\beta})^2 + \lambda_n \sum_{j=1}^{p} |\beta_j|, \tag{1.4}$$

or, equivalently,

$$\hat{\boldsymbol{\beta}}_n^L = \arg\min_{\boldsymbol{\beta}} \sum_{i=1}^{n} (y_i - \boldsymbol{x}_i'\boldsymbol{\beta})^2 \text{ subject to } \sum_{j=1}^{p} |\beta_j| \le t.$$

There is no closed form formula for $\hat{\boldsymbol{\beta}}_n^L$ in general. However, for the special case of the orthonormal design, an analytical formula exists, and we record it here as a proposition, which may shed light on our intuitive understanding of the shrinkage mechanism of the LASSO.

Proposition 1.2.1 *(**The LASSO estimator in orthonormal design** (Tibshirani, 1996)). For the orthonormal design in which $\sum_{i=1}^n \boldsymbol{x}_i\boldsymbol{x}_i' = I$ with I being the identity matrix, the LASSO estimator (1.4) is a function of $\lambda_n > 0$ in the form*

$$\hat{\beta}_j^L(\lambda_n) = \left(|\hat{\beta}_j^{ols}| - \frac{\lambda_n}{2}\right)^+ sgn(\hat{\beta}_j^{ols}), \tag{1.5}$$

for $j = 1, \cdots, p$ where $(z)^+ = \max\{z, 0\}$ and $sgn(z) = +1, 0, -1$ if $z > 0, = 0, < 0$, respectively. Also t is a function of λ_n defined by

$$t(\lambda_n) = \sum_{j=1}^{p} \left(|\hat{\beta}_j^{ols}| - \frac{\lambda_n}{2}\right)^+. \tag{1.6}$$

To get the results of Proposition 1.2.1, note that for the orthonormal design, $\sum_{i=1}^n (y_i - \boldsymbol{x}_i'\boldsymbol{\beta})^2 = \sum_{i=1}^n (y_i - \hat{y}_i)^2 + \sum_{j=1}^p (\beta_j - \hat{\beta}_j^{ols})^2$, where $\hat{\beta}_j^{ols}$ is the ordinary least squares (OLS) estimator

for β_j and $\hat{y}_i$ is the OLS predicted value. Applying the Karush-Kuhn-Tucker (KKT) theorem for the constrained optimization problem [1], we have

$$\begin{cases} (a)\ \hat{\beta}_j^L - \hat{\beta}_j^{ols} + \frac{\lambda_n}{2}\text{sgn}(\hat{\beta}_j^L) = 0,\ j = 1, \cdots, p, \\ (b)\ \lambda_n \geq 0, \\ (c)\ \lambda_n(\sum_{j=1}^p |\hat{\beta}_j^L| - t) = 0, \\ (d)\ \sum_{j=1}^p |\hat{\beta}_j^L| \leq t < \infty. \end{cases} \tag{1.7}$$

Consider the two cases. In the first case, $\lambda_n = 0$. Then from (1.7)(a) we have $\hat{\beta}_j^L = \hat{\beta}_j^{ols}$ so that from (1.7)(d) we have $\sum_{j=1}^p |\hat{\beta}_j^{ols}| \leq t < \infty$, which is not true because $\hat{\beta}_j^{ols}$ is the unconstrained minimizer.

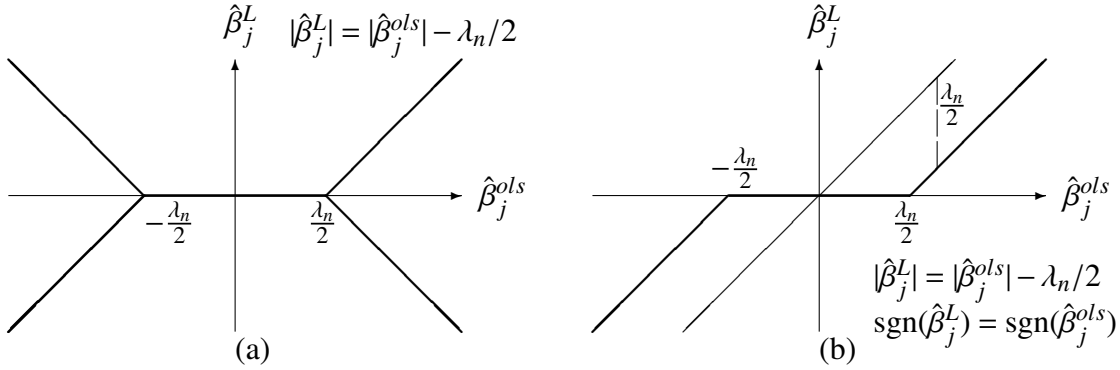


Figure 1.1: (a) Illustration of (1.11); (b) Illustration of the LASSO estimator in the orthonormal design.

Now consider the second case in which $\lambda_n > 0$. From (1.7)(c) we have

$$\sum_{j=1}^p |\hat{\beta}_j^L| = t. \tag{1.8}$$

We write (1.7)(a) as $|\hat{\beta}_j^L|\text{sgn}(\hat{\beta}_j^L) = |\hat{\beta}_j^{ols}|\text{sgn}(\hat{\beta}_j^{ols}) - \frac{\lambda_n}{2}\text{sgn}(\hat{\beta}_j^L)$ or

$$|\hat{\beta}_j^L| = |\hat{\beta}_j^{ols}|\text{sgn}(\hat{\beta}_j^L)\text{sgn}(\hat{\beta}_j^{ols}) - \frac{\lambda_n}{2}. \tag{1.9}$$

The LHS of (1.9) is non-negative. For the RHS of (1.9) to be non-negative, it is necessary that

$$\text{sgn}(\hat{\beta}_j^L) = \text{sgn}(\hat{\beta}_j^{ols}), \tag{1.10}$$

[1] **Karush-Kuhn-Tucker theorem** (Chong and Zak, 2008, p.458): Let $f, \boldsymbol{g}, \boldsymbol{h} \in C^1$. Let $\boldsymbol{x}^*$ be a regular point and a local minimizer for the problem of minimizing f subject to $\boldsymbol{h}(\boldsymbol{x}) = \boldsymbol{0}$ and $\boldsymbol{g}(\boldsymbol{x}) \leq \boldsymbol{0}$ where $f : \mathbb{R}^n \to \mathbb{R}$, $\boldsymbol{h} : \mathbb{R}^n \to \mathbb{R}^m (m < n)$ and $\boldsymbol{g} : \mathbb{R}^n \to \mathbb{R}^p$. Then there exist $\boldsymbol{\lambda}_1^* \in \mathbb{R}^m$ and $\boldsymbol{\lambda}_2^* \in \mathbb{R}^p$ such that (a) $Df(\boldsymbol{x}^*) + \boldsymbol{\lambda}_1^{*'}\boldsymbol{h}(\boldsymbol{x}^*) + \boldsymbol{\lambda}_2^{*'}\boldsymbol{g}(\boldsymbol{x}^*) = \boldsymbol{0}'$, (b) $\boldsymbol{\lambda}_2^* \geq \boldsymbol{0}$, (c) $\boldsymbol{\lambda}_2^{*'}\boldsymbol{g}(\boldsymbol{x}^*) = 0$, (d) $\boldsymbol{h}(\boldsymbol{x}^*) = \boldsymbol{0}$ and (e) $\boldsymbol{g}(\boldsymbol{x}^*) \leq \boldsymbol{0}$.

so that

$$|\hat{\beta}_j^L| = |\hat{\beta}_j^{ols}| - \frac{\lambda_n}{2}. \tag{1.11}$$

With pictorial aid it is easy to find the solution. Figure 1.1(a) is the plot of (1.11) whereas Figure 1.1(b) is the plot of (1.5) because it satisfies both (1.10) and (1.11). Substituting the $\hat{\beta}_j^L$ into (1.8) we have (1.6).

Proposition 1.2.1 gives us insight into the shrinkage mechanism of the LASSO. From Figure 1.1(b) we see clearly that the lasso shrinkage causes the estimates of the non-zero coefficients to be biased towards zero. We also see that the LASSO translates each coefficient by a constant factor $\lambda_n/2$, truncating at zero, which is known as *soft thresholding* (Donoho and Johnstone, 1994). The LASSO also performs *continuous subset selection*. Let us look at a simple example. Suppose that we have three standardized and orthonormal input variables x_1, x_2 and x_3. We assume that $\hat{\beta}_1^{ols} > \hat{\beta}_2^{ols} > \hat{\beta}_3^{ols} > 0$. We make use of Proposition 1.2.1. If $\lambda_n \geq 2\hat{\beta}_1^{ols}$, we have $\hat{\beta}_j^L = 0$, for $j = 1,2,3$, and $t = 0$. If $2\hat{\beta}_2^{ols} \leq \lambda_n < 2\hat{\beta}_1^{ols}$, we have $\hat{\beta}_1^L = \hat{\beta}_1^{ols} - \lambda_n/2$, $\hat{\beta}_j^L = 0$ for $j = 2,3$, and $0 < t \leq \hat{\beta}_1^{ols} - \hat{\beta}_2^{ols} \triangleq t_1$. If $2\hat{\beta}_3^{ols} \leq \lambda_n < 2\hat{\beta}_2^{ols}$, we have $\hat{\beta}_j^L = \hat{\beta}_j^{ols} - \lambda_n/2$ for $j = 1,2\,\hat{\beta}_3^L = 0$, and $t_1 < t \leq \hat{\beta}_1^{ols} + \hat{\beta}_2^{ols} - 2\hat{\beta}_3^{ols} \triangleq t_2$. If $0 \leq \lambda_n < 2\hat{\beta}_3^{ols}$, we have $\hat{\beta}_j^L = \hat{\beta}_j^{ols} - \lambda_n/2$ for $j = 1,2,3$, and $t_2 < t \leq \hat{\beta}_1^{ols} + \hat{\beta}_2^{ols} + \hat{\beta}_3^{ols} \triangleq t_3$. A bit of mechanical manipulation and rearranging gives us the following solution

$$\hat{\beta}_1^L = \begin{cases} t & \text{if } 0 \leq t \leq t_1 \\ \frac{1}{2}t + \frac{1}{2}t_1 & \text{if } t_1 < t \leq t_2 \\ \frac{1}{3}t + \frac{1}{2}t_1 + \frac{1}{6}t_2 & \text{if } t_2 < t \leq t_3 \end{cases},$$

$$\hat{\beta}_2^L = \begin{cases} 0 & \text{if } 0 \leq t \leq t_1 \\ \frac{1}{2}t - \frac{1}{2}t_1 & \text{if } t_1 < t \leq t_2 \\ \frac{1}{3}t - \frac{1}{2}t_1 + \frac{1}{6}t_2 & \text{if } t_2 < t \leq t_3 \end{cases},$$

$$\hat{\beta}_2^L = \begin{cases} 0 & \text{if } 0 \leq t \leq t_1 \\ 0 & \text{if } t_1 < t \leq t_2 \\ \frac{1}{3}t - \frac{1}{3}t_2 & \text{if } t_2 < t \leq t_3 \end{cases}.$$

We see that the LASSO solutions are continuous paths over the tuning parameter t, and each path is piecewise linear between thresholding points.

1.2.2 The computational algorithms

The LASSO procedure has an attractive property in terms of optimization. The objective function to be minimized is convex. Thus it does not suffer from the issue of multiple local minimal points, and the global minimum problem can be solved efficiently using a variety of algorithms, including but not limited to the quadratic programming for the LASSO (Tibshirani, 1996), the shooting algorithm for the LASSO (Fu, 1998), the homotopy algorithm for the LASSO (Osborne, Presnell and Turlach, 2000a, 2000b), the least angle regression and shrinkage (LARS) (Efron, Hastie, Johnston, and Tibshirani, 2004), the coordinate descent algorithm for the LASSO (Friedman, Hastie, Hoefling and Tibshirani, 2007). Note that in principle, the shooting algorithm belongs to the class of coordinate descent algorithms. The LARS belongs to the class of continuation methods or homotopy methods. That is why we put the definite article *the* before the name of each algorithm.

In this dissertation, we make use of the LARS algorithm of Efron, et al (2004). We also modify the shooting algorithm of Fu (1998) to minimize the LASSO regularized negative likelihood functions in Chapter 3 and Chapter 5 . Here we briefly review the LARS and the shooting algorithms.

The LARS algorithm

Perhaps the LARS algorithm (Efron, et al 2004) is the most well-known continuation algorithm in data mining. The LARS gives path solutions and the path is piece-wise linear, It is contrived with great ingenuity [2]. It is also extremely efficient so that the computational cost of the entire steps is of the same order as that of the ordinary least squares solution for the full model. See Algorithm 1 for details.

[2] In the preface to the book *The Science of Bradley Efron: Selected Papers* (Edited by Morris and Tibshirani), Tibshirani recount the story of how Efron contrived magically the *lars* algorithm pretty much single-handedly using geometric insight and analysis.

Algorithm 1: Least angle regression for the LASSO (Hastie, et al, 2009, p.74 - 76).

1. Standardize the predictors to have mean zero and unit norm. Start with the residual $\boldsymbol{r} = \boldsymbol{y} - \hat{\boldsymbol{y}}, \beta_1, \cdots, \beta_p = 0$
2. Find the predictor x_j most correlated with $\boldsymbol{r}$.
3. Move β_j from 0 towards its least-squares coefficient , until some other competitor $< \boldsymbol{x}_j, \boldsymbol{r} >$ has as much correlation with the current residual as does x_j.
4. Move β_j and 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 its variable from the active set of variables and recompute the current joint least squares direction.
6. Continue in this way until all p predictors have been entered. After $\min(n-1, p)$ steps, we arrive at the full least-squares solution.

The shooting algorithm

Fu (1998) proposed a shooting algorithm for solving the LASSO problem numerically. Let $Q(\boldsymbol{\beta}) = (\boldsymbol{y} - \boldsymbol{X}\boldsymbol{\beta})'(\boldsymbol{y} - \boldsymbol{X}\boldsymbol{\beta})$, where $\boldsymbol{y} = (y_1, \cdots, y_n)'$, and $\boldsymbol{X}$ is the design matrix. Then the LASSO estimator for the linear regression model is to minimize the objective function $Q(\boldsymbol{\beta}) + \lambda \sum_{j=1}^{p} |\beta_j|$. The first order necessary condition of optimization is $\partial Q(\boldsymbol{\beta})/\partial \boldsymbol{\beta} = -\lambda \text{sgn}(\boldsymbol{\beta})$, and $\partial Q(\boldsymbol{\beta})/\partial \boldsymbol{\beta} = -2\boldsymbol{X}'(\boldsymbol{y} - \boldsymbol{X}\boldsymbol{\beta}) = 2\boldsymbol{X}' \sum_{i=1}^{p} \boldsymbol{x}^i \beta_i - 2\boldsymbol{X}'\boldsymbol{y}$, which is the vector

$$\begin{pmatrix} 2\sum_{i=1}^{p} (\boldsymbol{x}^1)' \boldsymbol{x}^i \beta_i - 2(\boldsymbol{x}^1)'\boldsymbol{y} \\ \vdots \\ 2\sum_{i=1}^{p} (\boldsymbol{x}^j)' \boldsymbol{x}^i \beta_i - 2(\boldsymbol{x}^j)'\boldsymbol{y} \\ \vdots \\ 2\sum_{i=1}^{p} (\boldsymbol{x}^p)' \boldsymbol{x}^i \beta_i - 2(\boldsymbol{x}^p)'\boldsymbol{y} \end{pmatrix} = \begin{pmatrix} 2(\boldsymbol{x}^1)'\boldsymbol{x}^1\beta_1 + 2\sum_{i\neq 1} (\boldsymbol{x}^1)'\boldsymbol{x}^i\beta_i - 2(\boldsymbol{x}^1)'\boldsymbol{y} \\ \vdots \\ 2(\boldsymbol{x}^j)'\boldsymbol{x}^j\beta_j + 2\sum_{i\neq j} (\boldsymbol{x}^j)'\boldsymbol{x}^i\beta_i - 2(\boldsymbol{x}^j)'\boldsymbol{y} \\ \vdots \\ 2(\boldsymbol{x}^p)'\boldsymbol{x}^p\beta_p + 2\sum_{i\neq p} (\boldsymbol{x}^p)'\boldsymbol{x}^i\beta_i - 2(\boldsymbol{x}^p)'\boldsymbol{y} \end{pmatrix} \triangleq \begin{pmatrix} S_1 \\ \vdots \\ S_j \\ \vdots \\ S_p \end{pmatrix},$$

with $\boldsymbol{x}^j = (x_{1j}, \cdots, x_{nj})'$ being the jth column of $\boldsymbol{X}$. Letting

$$S_{0,j} = S_0(0, \boldsymbol{\beta}^{(-j)}, \boldsymbol{X}, \boldsymbol{y}) = 2\sum_{i\neq j} (\boldsymbol{x}^j)'\boldsymbol{x}^i\beta_i - 2(\boldsymbol{x}^j)'\boldsymbol{y},$$

where $\boldsymbol{\beta}^{(-j)}$ is the coefficient vector without β_j, we have

$$S_j = S_j(\boldsymbol{\beta}, \boldsymbol{X}, \boldsymbol{y}) = 2(\boldsymbol{x}^j)'\boldsymbol{x}^j\beta_j + S_{0,j},$$

for $j = 1, \cdots, p$.

In Figure 1.2, shoot from the point S_0 in the direction of slope $2(\boldsymbol{x}^j)'\boldsymbol{x}^j$. If no target was hit, as shown on middle figure, the solution is set to zero; if the target is hit, as shown on the left

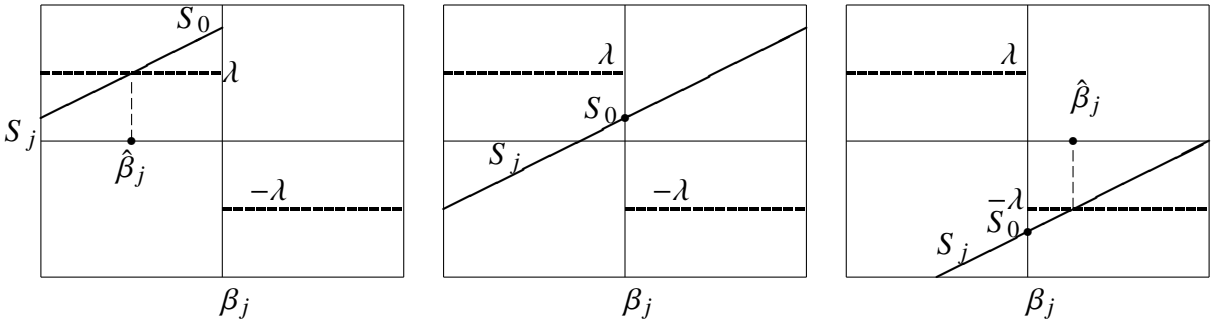


Figure 1.2: The shooting algorithm (Fu, 1998).

or right figure, the unique non-zero solution is obtained. The solution is expressible in closed form as

$$\hat{\beta}_j = \begin{cases} \frac{\lambda - S_{0,j}}{2(\boldsymbol{x}^j)'\boldsymbol{x}^j} & \text{if } S_{0,j} > \lambda, \\ 0 & \text{if } |S_{0,j}| < \lambda, \\ \frac{-\lambda - S_{0,j}}{2(\boldsymbol{x}^j)'\boldsymbol{x}^j} & \text{if } S_{0,j} < -\lambda, \end{cases}$$

for $j = 1, \cdots, p$.

1.2.3 The asymptotic properties

Knight and Fu (2000) set up a paradigm for asymptotic analysis of the whole class of Bridge estimator defined in (1.2) and (1.3) with the loss being the squared error loss, including the LASSO estimator. We follow Knight and Fu to conduct the asymptotic analysis. So we quote the following theorems from Knight and Fu (2000).

Consider the linear regression model, $y_i = \boldsymbol{x_i}\boldsymbol{\beta} + \epsilon_i$, where $\epsilon_1, \cdots, \epsilon_n$ are iid$(0, \sigma^2)$ with regularity conditions for the design:

A1: $\boldsymbol{C}_n = \frac{1}{n}\sum_{i=1}^{T} \boldsymbol{x}_i\boldsymbol{x}_i' \rightarrow \boldsymbol{C}$ with $\boldsymbol{C}$ being a positive definite $p \times p$ matrix,

A2: $\frac{1}{n}\max_{1\le i\le n} \boldsymbol{x}_i'\boldsymbol{x}_i \rightarrow 0$, as $n \rightarrow \infty$.

Let $\boldsymbol{\beta}^*$ be the true unknown parameter vector, $\hat{\boldsymbol{\beta}}_n^{ols}$ the ordinary least squares estimator for $\boldsymbol{\beta}^*$, and $\hat{\boldsymbol{\beta}}_n^{L}$ the LASSO estimator for $\boldsymbol{\beta}^*$ defined in (1.4). Recall that $\hat{\boldsymbol{\beta}}_n^{ols}$ is consistent, unbiased and $\sqrt{n}(\hat{\boldsymbol{\beta}}_n^{ols} - \boldsymbol{\beta}^*) \xrightarrow{D} N(0, \sigma^2\boldsymbol{C}^{-1})$.

Theorem 1.2.2 *(**Consistency** (Knight and Fu, 2000)). Under **A1** and **A2**, if $\lambda_n/n \to \lambda_0 \geq 0$ then*

$$\hat{\boldsymbol{\beta}}_n^L \xrightarrow{P} \underset{\boldsymbol{\beta}}{\operatorname{argmin}}(Z(\boldsymbol{\beta})),$$

where

$$Z(\boldsymbol{\beta}) = (\boldsymbol{\beta}-\boldsymbol{\beta}^*)'\boldsymbol{C}(\boldsymbol{\beta}-\boldsymbol{\beta}^*) + \lambda_0 \sum_{j=1}^{p} |\beta_j|.$$

Theorem 1.2.3 *($\sqrt{n}$-**Consistency** (Knight and Fu, 2000)). Under **A1** and **A2**, if $\lambda_n/\sqrt{n} \to \lambda_1 \geq 0$ then*

$$\sqrt{n}(\hat{\boldsymbol{\beta}}_n^L - \boldsymbol{\beta}^*) \xrightarrow{D} \underset{\boldsymbol{u}}{\operatorname{argmin}}(V_1(\boldsymbol{u})),$$

where

$$V_1(\boldsymbol{u}) = -2\boldsymbol{u}'\boldsymbol{w} + \boldsymbol{u}'\boldsymbol{C}\boldsymbol{u} + \lambda_1 \sum_{j=1}^{p} \Big\{ u_j sgn(\beta_j) I(\beta_j \neq 0) + |u_j| I(\beta_j = 0) \Big\}.$$

and $\boldsymbol{w} \sim N(\boldsymbol{0}, \sigma^2 \boldsymbol{C})$.

Remarks:

(i) By Theorem 1.2.2, if $\lambda_0 = 0$, then $\operatorname{argmin}(Z(\boldsymbol{\beta})) = \boldsymbol{\beta}^*$ and so $\hat{\boldsymbol{\beta}}_n^L$ is consistent.

(ii) By Theorem 1.2.3, if $\lambda_n = O(\sqrt{n})$, then $\hat{\boldsymbol{\beta}}_n^L$ is $\sqrt{n}$-consistent.

(iii) By Theorem 1.2.3, if $\lambda_1 = 0$, then $\sqrt{n}(\hat{\boldsymbol{\beta}}_n^L - \boldsymbol{\beta}^*)$ has the same asymptotic distribution as does $\sqrt{n}(\hat{\boldsymbol{\beta}}_n^{ols} - \boldsymbol{\beta}^*)$.

(iv) From Theorem 1.2.3, we see that if $\lambda_1 > 0$, the non-zero parameters are estimated with some asymptotic bias.

1.2.4 Selection consistency and irrepresentable conditions

Estimation consistency does not necessarily imply selection consistency. Without loss of generality, suppose that $\beta_1, \cdots, \beta_r \neq 0$ and $\beta_{r+1}, \cdots, \beta_p = 0$. Let $\mathbb{S} = \{1, 2, \cdots, r\}$. Let $\mathbb{S}^c = \{r+1, \cdots, p\}$. Let $\hat{\mathbb{S}}_n = \{j : \hat{\beta}_{j,n}^L \neq 0\}$. Let $\hat{\mathbb{S}}_n^c = \{j : \hat{\beta}_{j,n}^L = 0\}$. Let $\boldsymbol{\beta}_{\mathbb{S}} = (\beta_1, \cdots, \beta_r)'$. We rewrite the the matrix $\boldsymbol{C}$ as follows

$$\begin{pmatrix} \boldsymbol{C}_{\mathbb{S}\mathbb{S}} & \boldsymbol{C}_{\mathbb{S}\mathbb{S}^c} \\ \boldsymbol{C}_{\mathbb{S}\mathbb{S}^c} & \boldsymbol{C}_{\mathbb{S}^c\mathbb{S}^c} \end{pmatrix},$$

where $\boldsymbol{C}_{\mathbb{S}\mathbb{S}}$ is $r \times r$ matrix, $\boldsymbol{C}_{\mathbb{S}^c\mathbb{S}^c}$ is $(p-r)\times(p-r)$ matrix.

Definition (**Selection consistency** (Zou, 2006)). The LASSO variable selection is consistent if and only if $\lim_n P(\hat{\mathbb{S}}_n = \mathbb{S}) = 1$.

Proposition 1.2.4 . *Under **A1** and **A2**, if $\lambda_n/\sqrt{n} \to \lambda_1 > 0$ then*

$$\liminf_n P(\hat{\mathbb{S}}_n^c = \mathbb{S}^c) = c > 0.$$

Proposition 1.2.4 is summarized from a result of Knight and Fu (2000). For the proof, see the paragraph before Example 1 in the paper of Knight and Fu (2000). Proposition 1.2.4 says that when some of β_j's are exactly 0, the limiting distribution specified in Theorem 1.2.3 of the LASSO estimator puts positive probability at 0 if $\lambda_n = O(\sqrt{n})$.

Proposition 1.2.5 *(Zou, 2006). Under **A1** and **A2**, if $\lambda_n/\sqrt{n} \to \lambda_1 \geq 0$ then*

$$\limsup_n P(\hat{\mathbb{S}}_n = \mathbb{S}) \leq c < 1.$$

Proposition 1.2.5 is quoted from Zou (2006). For the proof, see the paper of Zou (2006). Proposition 1.2.5 says that if $\lambda_n = O(\sqrt{n})$, which is the optimal rate of convergence in estimation, then the set $\hat{\mathbb{S}}_n$ is not the true set $\mathbb{S}$ with a positive probability.

We then wonder if the LASSO could achieve selection consistency if we are willing to sacrifice the convergence rate of estimation. It turns out that the slower convergence rate of estimation does not guarantee selection consistency. The problem lies in several quite restrictive conditions (Meinshausen and Bühlmann, 2006). The main and restrictive assumption for consistent variable selection is the so-called *neighborhood stability* (Meinshausen and Bühlmann, 2006), *coherence* condition (Donoho, Elad and Temlyakov, 2006) or *irrepresentable* condition (Zhao and Yu, 2006). The irrepresentable condition concerns the design matrix $\boldsymbol{X}$ and cannot be relaxed (Meinshausen and Bühlmann, 2006). Several authors independently investigated this issue, including Zou (2006), Zhao and Yu (2006), and Meinshausen and Bühlmann (2006). Bühlmann and van de Geer (p.22 and 190-194, 2011) gives an excellent comprehensive exposition of the irrepresentable condition.

Definition (**Irrepresentable condition** (Zou, 2006; Zhao and Yu, 2006; Bühlmann and van de Geer, 2011)). Assume that $\boldsymbol{C}_{\mathbb{S}\mathbb{S}}$ is non-singular. We say that the *strong irrepresentable*

condition is met if

$$\|\boldsymbol{C}_{\mathbb{S}\mathbb{S}^c}\boldsymbol{C}_{\mathbb{S}\mathbb{S}}^{-1}\text{sgn}(\boldsymbol{\beta}_{\mathbb{S}})\|_\infty < 1. \tag{1.12}$$

We say that the *weak irrepresentable condition* is met if

$$\|\boldsymbol{C}_{\mathbb{S}\mathbb{S}^c}\boldsymbol{C}_{\mathbb{S}\mathbb{S}}^{-1}\text{sgn}(\boldsymbol{\beta}_{\mathbb{S}})\|_\infty \leq 1. \tag{1.13}$$

Theorem 1.2.6 *(**Sufficiency and essential necessity of selection consistency**(Zou, 2006; Zhao and Yu, 2006)). Under regularity assumptions **A1** and **A2**, we have*

(i) Essentially necessary condition: If $\lim_n P(\hat{\mathbb{S}}_n = \mathbb{S}) = 1$*, then the weak irrepresentable condition (1.13) follows.*

(ii) Sufficient conditions: If the strong irrepresentable condition (1.12) holds, then $\lim_n P(\hat{\mathbb{S}}_n = \mathbb{S}) = 1$.

The irrepresentable condition corresponds to a condition on the design matrix of the form

$$\|(\boldsymbol{X}'_{\mathbb{S}\mathbb{S}}\boldsymbol{X}_{\mathbb{S}\mathbb{S}})^{-1}\boldsymbol{X}'_{\mathbb{S}\mathbb{S}}\boldsymbol{X}_{\mathbb{S}^c\mathbb{S}^c}\|_\infty \leq 1-\eta \text{ for some } \eta \in (0,1].$$

This means that the least squares coefficients for the columns of $\boldsymbol{X}_{\mathbb{S}^c\mathbb{S}^c}$ on $\boldsymbol{X}_{\mathbb{S}\mathbb{S}}$ are not too large, that is, the relevant variables in $\mathbb{S}$ are not too highly correlated with the nuisance variables in $\mathbb{S}^c$. It is not so much that Theorem 1.2.6 allows us to say when the LASSO is consistent for selection and when not as that it gives us a warning message that the LASSO would perform poorly for variable selection with strongly correlated design.

A variety of remedies has been suggested to improve the performance of the LASSO, for example, the *relaxed* LASSO of Meinshausen (2007), the *smoothly clipped absolute deviation* (SCAD) of Fan and Li (2001), and so forth. The *adaptive* LASSO (Zou, 2006) is a simple yet effective remedy. The adaptive LASSO yields consistent estimators and selects variables consistently even if the irrepresentable condition fails while retaining the attractive convexity property of the LASSO.

1.2.5 The adaptive LASSO and its oracle properties

We review the definition, computational algorithm, and asymptotic properties of the adaptive LASSO of Zou (2006).

Definition (**The adaptive LASSO** (Zou, 2006)). The adaptive LASSO estimator, denoted by $\hat{\boldsymbol{\beta}}_n^{aL}$, is defined as

$$\hat{\boldsymbol{\beta}}_n^{aL} = \arg\min_{\boldsymbol{\beta}} \sum_{i=1}^{n} (y_i - \boldsymbol{x}_i'\boldsymbol{\beta})^2 + \lambda_n \sum_{j=1}^{p} \hat{w}_j |\beta_j|, \tag{1.14}$$

where

$$\hat{w}_j = \frac{1}{|\hat{\beta}_j|^\gamma} \tag{1.15}$$

for some $\gamma > 0$, and $\hat{\beta}_j$ is a $\sqrt{n}$-consistent estimate for β_j.

The analytical formula for $\hat{\boldsymbol{\beta}}_n^{aL}$ exists only for orthonormal models while there is no closed form formula for general designs. Following the same process shown in Section 1.2.1, we obtain the following results for the orthonormal models.

Proposition 1.2.7 *(**The adaptive LASSO estimator in orthonormal design**). For the orthonormal design in which $\sum_{i=1}^{n} \boldsymbol{x}_i \boldsymbol{x}_i' = I$ with I being the identity matrix, the adaptive LASSO estimator defined by (1.14) and (1.15) is a function of $\lambda_n > 0$ in the form*

$$\hat{\beta}_j^{aL}(\lambda_n) = \left(|\hat{\beta}_j^{ols}| - \frac{\lambda_n}{2|\hat{\beta}_j|^\gamma} \right)^+ sgn(\hat{\beta}_j^{ols}), \tag{1.16}$$

for $j = 1, \cdots, p$ where $(z)^+ = \max\{z, 0\}$ and $sgn(z) = +1, 0, -1$ if $z > 0, = 0, < 0$, respectively. And t is a function of λ_n defined by

$$t(\lambda_n) = \sum_{j=1}^{p} \left(|\hat{\beta}_j^{ols}| - \frac{\lambda_n}{2|\hat{\beta}_j|^\gamma} \right)^+ .$$

The adaptive LASSO estimator (1.16) for the orthonormal design is illustrated by Figure 1.3 where we set $\hat{\beta}_j = \hat{\beta}_j^{ols}$. Proposition 1.2.7 and Figure 1.3 gives us insight into the mechanism of the adaptive LASSO. We see that the adaptive LASSO shrinkage still causes the estimate of a non-zero coefficient to be biased towards zero but the bias becomes much smaller, especially when the coefficient is large, compared to the bias caused by the LASSO. Moreover, we see that a nuisance coefficient becomes easier to be truncated at zero due to the adaptive soft thresholding. As the sample size increases, the adaptive weights for zero coefficients approaches infinity while the weights for non-zero ones approaches finite constants. We then get unbiased (in asymptotic sense) estimates for significant coefficients and at the same time get

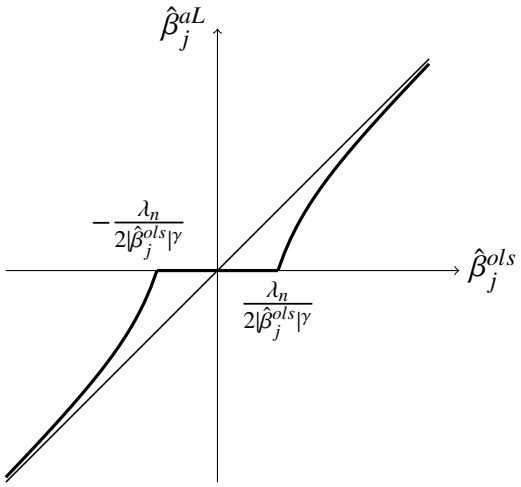


Figure 1.3: Illustration of the adaptive LASSO estimator in the orthonormal design with the adaptive weight $\hat{w}_j$ being $1/|\hat{\beta}_j^{ols}|^\gamma$.

the nuisance ones truncated at zero. In addition, the adaptive LASSO still attains continuous subset selection property of the LASSO.

The adaptive LASSO attains the attractive convexity property of the LASSO in terms of optimization. In addition, the LARS algorithm (Efron et al 2004) can be directly employed to solve the adaptive LASSO problem. Let $\boldsymbol{W} = \text{diag}(\hat{w}_1, \cdots, \hat{w}_p)$. The adaptive LASSO objective can be rewritten as

$$\left(\boldsymbol{y} - \boldsymbol{X}\boldsymbol{W}^{-1}\boldsymbol{W}\boldsymbol{\beta}\right)'\left(\boldsymbol{y} - \boldsymbol{X}\boldsymbol{W}^{-1}\boldsymbol{W}\boldsymbol{\beta}\right) + \lambda\sum_{j=1}^{p}\hat{w}_j|\beta_j| = \left(\boldsymbol{y} - \tilde{\boldsymbol{X}}\tilde{\boldsymbol{\beta}}\right)'\left(\boldsymbol{y} - \tilde{\boldsymbol{X}}\tilde{\boldsymbol{\beta}}\right) + \lambda\sum_{j=1}^{p}|\tilde{\beta}_j|,$$

where $\tilde{\boldsymbol{X}} = \boldsymbol{X}\boldsymbol{W}^{-1}$, and $\tilde{\boldsymbol{\beta}} = \boldsymbol{W}\boldsymbol{\beta}$ (i.e. $\tilde{\beta}_j = \hat{w}_j\beta_j$). Thus, the LARS algorithm for the adaptive LASSO consists of the following steps:

Algorithm 2: The LARS algorithm for the adaptive LASSO (Zou, 2006).

1 Calculte $\tilde{\boldsymbol{X}} = \boldsymbol{X}\boldsymbol{W}^{-1}$, i.e. $\tilde{\boldsymbol{x}}_j = \boldsymbol{x}_j/\hat{w}_j, j = 1, \cdots, p$.

2 Apply Algorithm 1 to obtain $\hat{\tilde{\boldsymbol{\beta}}}(\lambda) = \arg\min_{\tilde{\boldsymbol{\beta}}}\left\{(\boldsymbol{y} - \tilde{\boldsymbol{X}}\tilde{\boldsymbol{\beta}})'(\boldsymbol{y} - \tilde{\boldsymbol{X}}\tilde{\boldsymbol{\beta}}) + \lambda\sum_{j=1}^{p}|\tilde{\beta}_j|\right\}$.

3 Output $\hat{\boldsymbol{\beta}}^{aL}(\lambda) = \boldsymbol{W}^{-1}\hat{\tilde{\boldsymbol{\beta}}}$.

Fan and Li (2001) discussed nice properties that a good shrinkage estimator should provide. (i) *Unbiasedness*. The estimator is nearly unbiased when the true unknown parameter is large. (ii) *Sparsity*. The estimator has a thresholding rule that automatically truncates the nuisance

coefficients at zero to reduce model complexity. (iii) *Continuity*. The estimator avoids instability in prediction. In the same paper, they proposed the smoothly clipped absolute deviation (SCAD) penalty to remedy the selection inconsistency of the LASSO. They demonstrate that the SCAD estimator is $\sqrt{n}$-consistent. Moreover, in language similar to Donoho and Johnstone (1994), and they showed that the estimator performs as well as the oracle estimator, which knows in advance the sparsity structure of the true model. Zou (2006) showed that the adaptive LASSO also possesses these oracle properties.

Theorem 1.2.8 *((**Oracle properties of the adaptive LASSO** (Zou, 2006)). Under **A1** and **A2**, if $\lambda_n/\sqrt{n} \to 0$ and $\lambda_n n^{(\gamma-1)/2} \to \infty$ then the adaptive LASSO estimator must satisfy*
(i) Selection consistency: $\lim_n P(\hat{\mathbb{S}}_n = \mathbb{S}) = 1$.
(ii) Asymptotic normality: $\sqrt{n}(\hat{\boldsymbol{\beta}}_{\mathbb{S}}^{aL} - \boldsymbol{\beta}_{\mathbb{S}}^{*}) \xrightarrow{D} N\left(0,\ \sigma^2 \boldsymbol{C}_{\mathbb{S}\mathbb{S}}^{-1}\right)$.

1.2.6 Critiques for the oracle properties

The LASSO methodology is successful and popular in statistical modeling, especially in high dimensional data analysis, due to the fact that it performs model selection and parameter estimation simultaneously. Most existing studies have focused on the prediction, estimation and selection properties ranging from prediction consistency and estimation consistency to selection consistency with the aim of recovery of the true underlying sparse model, as we summarized in previous sections. Some important questions are less well studied. For example, a classical variable selection procedure sets a coefficient in a model to zero if it is marginally insignificant, i.e. the 95% confidence interval contains 0 whereas the LASSO sets a parameter directly to zero due to optimization of a penalized objective function, which is hard to understand from a statistical point of view. Another example concerns statistical inference. In practice, data analysts would like to assess how significant a selected variable is and to make multiple comparisons between a number of variables simultaneously. A new advance has been made recently by Lockhart, Taylor, Tibshirani and Tibshirani (2014) in the regard of testing significance for the LASSO. Yet, some of criticisms in the literature to the LASSO and shrinkage methods at large remain unanswered.

Leeb and Pötscher (2008) related the oracle properties of shrinkage estimators to the su-

perefficient Hodges' estimator, a well-known pitfall that holds only for a set of parameters with Lebesgue measure zero. They argued that the oracle properties are often a consequence of sparsity of an estimator. They showed that any estimator satisfying a sparsity property has maximal risk that converges to the supremum of the loss function; in particular, the maximal risk diverges to infinity whenever the loss function is unbounded.

Pötscher and Schneider (2009) and Pötscher and Leeb (2009) studied the distribution of the adaptive LASSO estimator (and other shrinkage estimators). They showed that while the oracle properties predict normality, the finite-sample distribution of the adaptive LASSO estimator is highly non-normal, and non-normality persists even in large samples. They argued that the oracle properties based on *fixed-parameter* asymptotics are not reliable tools to assess the estimator's actual performance. To determine if the non-normality of the finite-sample distribution really is a transient feature as $n \to \infty$ as the oracle properties suggest, one needs to study *moving-parameter* asymptotics rather than fixed-parameter asymptotics. They argued that the mathematical reason for the failure of the pointwise asymptotic distribution to capture the behaviour of the finite-sample distribution well is that the convergence of the latter to the former is not uniform in the underlying parameter. In particular, small non-zero coefficients cannot be detected consistently and their presence are related to the phenomenon of super-efficiency. Selection consistency needs the so-called *beta-min* condition (Bühlmann and van de Geer, 2011, page 35 and 187), a condition requiring some lower non-zero bound on $|\boldsymbol{\beta}^*|_{min} \triangleq \min_{j\in\mathbb{S}} |\beta_j^*|$, for example, $|\boldsymbol{\beta}^*|_{min} \gg \sqrt{s \log p/n}$ in linear regression, where $s = |\mathbb{S}|$ is the cardinality of the set $\mathbb{S}$. Pötscher and Leeb (2009) showed that the uniform convergence rate of the adaptive estimator is slower than $1/\sqrt{n}$ in the case of consistent model selection. Pötscher and Schneider (2010) also showed that the intervals based on the adaptive LASSO estimator are larger than the standard intervals by an order of magnitude in the case of consistent model selection.

Leeb and Pötscher (2003, 2005) discussed the effects of model selection on inference. They showed that the finite-sample distribution of a post-model selection estimator is typically not uniformly close to the pointwise asymptotic distribution. They claimed the *impossibilty*, namely, the finite-sample distribution of a post-model-selection estimator is typically too

complicated to be estimated. Hence, regardless of sample size the asymptotic distribution can not be safely used to replace the finite-sample distribution. Leeb and Pötscher (2005) viewed a post-model-selection estimator as a discontinuous form of shrinkage estimators. The two types of estimators show similar features in the asymptotic distributions. The finite distribution functions or the risks of the two types of estimators often can not be estimated uniformly consistently.

While they do not invalidate the LASSO methodology and shrinkage methods at large, these critiques do shed light on some critical issues in the area of shrinkage methods and definitely provide motivation for further investigation.

1.3 Literature review of the LASSO methodology in time series analysis

As of now we have not found any research results in the literature that apply the LASSO methodology to build the autoregressive conditional heteroscedastic (ARCH) model of Engle (1982) and multivariate ARCH models.

There exist a lot of research examples that utilize the LASSO methodology to build autoregressive (AR) models and vector AR models. In this section we briefly review these existing results. Readers are notified that our review is not a complete list. For example, we do not touch upon the applications of the LASSO to time series regression model, frequency-domain analysis, change-point models, and non-parametric time series analysis. We do not touch upon the Bayesian LASSO and the fused LASSO.

For a linear regression model with autoregressive errors (REGAR) with fixed autoregressive (AR) order, Wang, Li, and Tsai (2007) adapted the LASSO to the REGAR models to shrink both the regression coefficients and the autoregressive (AR) coefficients. Yoon, Park, and Lee (2013) applied three shrinkage methods, the adaptive LASSO, the bridge, and the SCAD to the REGAR model, proposed computational algorithm, studied asymptotic properties such as consistency, selection consistency, and asymptotic normality, and compared the performances

of the three estimators. Chand (2011) implemented LASSO-type shrinkage methods to linear regression and time series models in his dissertation.

Autoregressive models with infinite variance are important in modeling heavy tailed time series. Tang, Zhou, and Wu (2012) proposed a self-weight composite quantile regression (SWCQR) and applied the adaptive LASSO on SWCQR for estimation and selection of infinite variance autoregressive models. Xu, Xiang, Wang and Lin (2012) applied the adaptive LASSO penalty to the least absolute deviation loss function and they reported that the proposed method is able to consistently identify the true model and at the same time produce efficient estimators. Xu et. al (2012) also provided a unified way to conduct variable selection for AR models with finite or infinite variance.

Nardi and Rinaldo (2011) applied the LASSO to the AR process whose maximal lag order p grows with sample size n at certain rate. They referred this scheme as a double asymptotic framework. The AR model with an increasing p lies between a fixed order AR and an infinite-order AR process. They showed that the Lasso procedure is particularly adequate for this double asymptotic scheme. They derived theoretical results establishing nice asymptotic properties, under a much faster rate of growth of the AR order. In particular, model selection consistency, estimation consistency, and prediction consistency hold if the maximal lag p grows with n as $p = o(n)$, $p = o(n^{1/2})$, and $p = o(n^{1/3})$, respectively. Medeiros and Mendes (2012) studied the asymptotic properties of the adaptive LASSO in sparse high-dimensional linear time-series models where both the number of autoregressive variables can increase with the number of observations and might be larger than the number of observations. They showed that the adaptive LASSO has oracle properties even when the errors are non-Gaussian and conditionally heteroskedastic.

Most existing applications of shrinkage estimators focus on the stationary AR processes. Some recent research extend the literature by applying shrinkage methods to nonstationary AR processes. Kock (2012) applied the adaptive LASSO to both stationary and non-stationary AR models. He showed that the adaptive LASSO has oracle efficiency. In particular, his results

imply that the adaptive LASSO is able to discriminate between stationary and non-stationary AR processes and thereby constitutes an addition to the set of unit root tests. He also studied the finite properties of the adaptive LASSO using the AR(1) model. Caner and Knight (2013) applied the Bridge estimators to nonstationary AR processes, and proposed a novel way to test nonstationarity of AR processes. The method of Caner and Knight (2013) can select the correct model with probability tending to 1, and select the optimal lag length and unit root simultaneously, thereby outperforming the existing unit root tests.

Park and Sakaori (2013) prosed the lag weighted LASSO. Their method imposes different penalties on each coefficient based on weights that reflect not only the coefficients size but also the lag effects. They reported that the lag weighted LASSO is superior to both the LASSO and the adaptive LASSO in forecast accuracy. They modified the adaptive LASSO weight as

$$w_{j,l} = \frac{1}{(|\hat{\beta}_{j,l}|\alpha(1-\alpha)^l)^\gamma},$$

where $0 < \alpha < 1$, l represents the l-th lag. They constructed this weight formula based on the assumption that the the effects of autoregressors decay geometrically as the lag length increases. Interestingly enough, their method shares the similar spirit as our methodology.

In the literature the LASSO methodology has been applied to multivariate (vector) autoregressive processes of order p, abbreviated as VAR(p). Valdés-Sosa et al. (2005) used sparse VAR(1) models to estimate brain functional connectivity where the LASSO is applied to achieve sparsity of VAR(1) models. Fujita, et al (2007) applied sparse VAR model to estimate gene regulatory networks based on gene expression profiles obtained from time-series microarray experiments where sparsity was reported to have been achieved by LASSO.

Hsu, Huang and Chang (2007) applied the LASSO to achieve subset selection for VAR models of high order. In their methodology, they first used AIC or Bayesian Information Criterion (BIC) to select the optimal lag order p_{aic} or p_{bic}. They proposed the top-down, bottom-up and hybrid strategies to reduce the full VAR(p_{aic}) or VAR(p_{aic}) models. The performance of the several strategies was compared. Ren and Zhang (2010) applied the adaptive LASSO to

achieve subset selection for VAR models with higher lag order. Ren and Zhang (2010) first used AIC or Hannan and Quinn (HQ) criterion to determine the optimal lag order p_{aic} or p_{hq} and then the adaptive LASSO was applied to reduce the full VAR(p_{aic}) or VAR(p_{hq}) models.

Haufe, Muller, Nolte, and Kramer (2008) applied the grouped LASSO to VAR models. Song and Bickel (2011) proposed an integrated approach for large VAR processes that yields three types of estimators; that is, the adaptive LASSO with (i) universal grouping, (ii) no grouping, and (iii) segmented grouping. Kock and Callot (2012) investigated oracle efficient estimation and forecasting of the adaptive LASSO and the adaptive group LASSO for VAR models.

1.4 The doubly adaptive LASSO for time series models

In this section, we explain our source of motivation. We also present the general idea underlying our methodology, and discuss how to choose tuning parameter and weighting parameters.

1.4.1 Motivation

Although the LASSO and the adaptive LASSO have been successfully applied to AR and VAR models, some aspects of existing methods are not very satisfactory for time series data analysts.

(i) Suppose that we have time series data generated from AR(p) model but we do not know the true order p. We arbitrarily guess a large value for the order h and we assume that $h > p$. The LASSO and adaptive LASSO often include in the model the autoregressive variables with lags beyond the true order albeit the model is sparse. This is not surprising because time series random variables are temporally dependent. Both the LASSO and adaptive LASSO are conservative and reluctant to discard the autoregressive variables with lags beyond p. Thus, the existing methods first determine the right order using some criteria such as the AIC, BIC, and Hannan and Quinn (HQ). Then the LASSO methodology is applied to shrink some intermediate coefficients to zero. This is good but it is definitely better if we could let the LASSO to determine the order for us.

(ii) It does make sense to have a time series model to reflect the natural assumption that the effects of autoregressors decay as the lag length increases, although the decay patterns are not necessarily geometrical.

(iii) There are no applications of the LASSO methodology to the ARCH and VARCH models. It is desirable if we could extend the literature of the LASSO methodology to the area of volatility models.

These facts motivate us to propose the *doubly adaptive LASSO* tailored to the time series analysis, which is the theme of this dissertation.

1.4.2 The doubly adaptive LASSO (daLASSO)

For time series data $y_1, \cdots, y_T$, the doubly adaptive LASSO estimators take the form

$$\hat{\boldsymbol{\theta}}^{daL} = \arg\min_{\boldsymbol{\theta}} \sum_{t=1}^{T} \ell(\boldsymbol{y}_t, \boldsymbol{\theta}) + \lambda_T \sum_{j} \hat{w}_{T,j} |\theta_j|,$$

where $\lambda_T > 0$ is the tuning parameter, $\boldsymbol{\theta}$ is the coefficient vector in a time series model, $\ell(y_i, \boldsymbol{x}_i'\boldsymbol{\theta})$ is the loss function, which is the squared error loss for AR and VAR models or the negative log-likelihood function for ARCH and VARCH models, and the adaptive weight $\hat{w}_{T,j}$ is defined as the product of the two weights [3], namely,

$$\begin{aligned} \hat{w}_{T,j} &= \hat{w}_j^Z \hat{w}_j^B \ , \\ \hat{w}_j^Z &= \frac{1}{|\hat{\beta}|^{\gamma_1}} \end{aligned} \tag{1.17}$$

and $\hat{w}_j^B$, say, for the AR models is

$$\hat{w}_j^B = \frac{1}{(\sum_{i=j}^{h} |\hat{\rho}_{ii}|^{\gamma_0})^{\gamma_2}} \ , \tag{1.18}$$

where $\hat{\rho}_{ii}$ is the partial autocorrelation at lag i, and γ_0, γ_1 and γ_2 are some non-negative constants called weighting parameters. The formula (1.17) is borrowed from Zou (2006) (denoted

[3] An examiner suggested an alternative way: the maximum of the two.

by superscript Z). We borrow the idea in Box-Pierce test statistic and Monti (1994) test statistics [4] (denoted by superscript B) to construct formula (1.18). In weight formula for $\hat{w}_{T,j}$, we let $\hat{w}_j^Z$ make use of magnitude information of the coefficient, and we let $\hat{w}_j^B$ make use of decay structure and lag order information of the corresponding autoregressive variable. We use *doubly adaptive* to emphasize this form.

In this dissertation the doubly adaptive LASSO is actually the general name for specific four methods: the partial autocorrelation or PAC-weighted adaptive LASSO for AR model, the PAC-weighted adaptive positive LASSO for ARCH model, the partial lag autocorrelation matrix norm or PLAC-weighted adaptive LASSO for VAR model, and the PLAC-weighted adaptive LASSO for BEKK VARCH model.

1.4.3 Determining optimal values for tuning and weighting parameters

The adaptive Lasso and the doubly adaptive Lasso yield a path of possible solutions defined by the continuum depending on the values of the hyperparameters which represent the amount of shrinkage. The choice of the weighting parameters γ_0, γ_1, and γ_2 and the tuning parameter λ_T determines the tradeoff between model fit and model sparsity. We desire a good value for these parameters unknown a priori to satisfy certain criteria. In the literature, a variety of criteria have been proposed for such selection. Some of well-known criteria include cross validation (CV) (e.g. leave-one-out CV, 5-fold CV), generalized cross validation (Craven and Wahba, 1979, Tibshirani, 1996, Fan and Li, 2010), Mallow's C_p (Mallows, 1973), AIC (Akaike, 1973, 1974), Bayesian information criterion (BIC) (Schwarz, 1978), final prediction error (FPE) (Akaike, 1969, 1971) and HQ (Hannan and Quinn, 1979).

Perhaps the CV is the most commonly used method. However, it is important to note that CV picks values of hyperparameters that result in predictive optimality. So the values chosen by CV are not usually the same values as those that are likely to recover the true model. Indeed, it was proved (Meinshausen and Bühlmann, 2006) that the prediction-optimal value of

[4] Box-Pierce portmanteau test statistic is defined as $Q_{BP} = T\sum_{i=1}^{h}|\hat{\rho}(i)|^2$, where $\hat{\rho}(i)$ is the estimated autocorrelation at lag i. Monti portmanteau test statistic is defined as $Q_M = T(T+2)\sum_{i=1}^{h}\frac{|\hat{\rho}_{ii}|^2}{T-i}$, where $\hat{\rho}_{ii}$ is the estimated partial autocorrelation at lag i.

the tuning parameter does not result in model selection consistency. Generally speaking, we often need a larger penalty for variable selection and a smaller penalty for good prediction. When CV is used, the LASSO often selects too many variables, which is good in the variable screening situation, but not good for variable selection.

We also note that the CV scheme is difficult to implement for time series analysis due to the nature of temporal dependence present in time series data. In univariate time series, the problem may be not that serious and we may implement CV, as demonstrated in Chapter 2. But the CV is quite difficult to implement for multivariate time series data.

In this dissertation, except for univariate AR models in Chapter 2, we use the BIC to choose the optimal values of tuning and weighting parameters. Many authors have used the BIC for this purpose in the literature including Caner and Knight (2013), Wagener and Dette (2012), Wang and Leng(2007), and Wang, et al (2007). Note that we apply double penalization when we use the BIC to choose hyperparameters. The first is L_1 penalization from the LASSO, which yields the path solution by the LASSO,

$$\hat{\boldsymbol{\theta}}((\lambda_T, \gamma_0, \gamma_1, \gamma_2)) = \arg\min_{\boldsymbol{\theta}} \sum_{t=1}^{T} \ell((y_t, \boldsymbol{x}_t), \boldsymbol{\theta}) + \lambda_T \sum_j \hat{w}_{T,j}((\lambda_T, \gamma_0, \gamma_1, \gamma_2))|\theta_j|,$$

and the the second is the penalization from the BIC, which yields optimal values for these hyperparameter.

$$(\lambda_T, \gamma_0, \gamma_1, \gamma_2)^* = \arg\min_{\Lambda} \mathrm{BIC}((\lambda_T, \gamma_0, \gamma_1, \gamma_2)) = -2\ell_T(\hat{\boldsymbol{\theta}}((\lambda_T, \gamma_0, \gamma_1, \gamma_2))) + |\hat{\mathbb{S}}_T| \log(T).$$

where $|\hat{\mathbb{S}}_T|$ is the cardinality of the set $\hat{\mathbb{S}}$. Then the solution $\hat{\boldsymbol{\theta}}^{daL}$ is read off from the path against $(\lambda_T, \gamma_0, \gamma_1, \gamma_2)^*$.

1.5 Thesis organization

The remaining of this thesis are organized in the following.

In Chapter 2, we will propose the partial autocorrelation or PAC-weighted adaptive LASSO for univariate autoregressive process with lag order p fixed (AR(p)). We will prove the asymp-

totic oracle properties of the PAC-weighted adaptive LASSO estimator, conduct Monte Carlo study on the performance of the doubly adaptive estimator. The proposed methodology shows promising results for modelling stationary AR(p) processes, and show some application examples for real world time series data analysis.

In Chapter 3, we will propose the partial autocorrelation or PAC-weighted adaptive positive LASSO for univariate autoregressive conditional heteroscedastic process with lag order q fixed (ARCH(q)). We will prove the asymptotic oracle properties of the PAC-weighted adaptive positive LASSO estimator, propose a computational algorithm based on the quadratic approximation of likelihood function, conduct Monte Carlo study on the performance of the doubly adaptive LASSO estimator, and apply the methodology to analysis of some financial time series data such as the US S&P 500 index returns and the Japanese Nikkei returns.

In Chapter 4, we will review the concept and algorithm of the partial lag autocorrelation (PLAC) matrix developed by Heyse (1985), and then propose the PLAC-weighted adaptive LASSO for multivariate autoregressive process with lag order p fixed (VAR(p)). We will prove the asymptotic oracle properties of the PLAC-weighted adaptive positive LASSO estimator, conduct Monte Carlo study on the performance of the doubly adaptive LASSO estimator, and show an application example for real world time series data analysis.

In Chapter 5, we will propose the PLAC-weighted adaptive LASSO for BEKK multivariate autoregressive conditional heteroscedastic with lag order q fixed (VARCH(q)). We will propose a computational algorithm based on the quadratic approximation of likelihood function for which we derive the analytical score gradient and analytical Hessian matrix. We will conduct Monte Carlo study on the performance of the doubly adaptive LASSO estimator.

In Chapter 6, we will give a general discussion and present our future research plan.

Appendix A contains some concepts and theorems in probability. Appendix B contains some definitions and formulae in matrix calculus. Appendix C records the details of partial lag autocorrelation matrix including computational algorithm. Appendix D contains detailed derivations of analytical score and Hessian for BEKK VARCH(q) models.

Chapter 2

The Doubly Adaptive LASSO for AR(p) Models

2.1 Introduction

We recall that under quite general conditions a second-order stationary process with constant mean can be approximated well by an autoregressive (AR) model, which specifies that the output variable depends linearly on its own past values. Let $\{y_t\}$ be a stationary stochastic process. Let $\mathcal{F}_t$ be the information available at t. $\mathcal{F}_{t-1} \equiv \{y_{t-1}, y_{t-2}, \cdots\}$ denotes the past history of a stationary stochastic process. By specifying the stationary process as an AR(p) model , we implicitly assume that only the most recent values $y_{t-1}, \cdots, y_{t-p}$ matters for specifying the dynamics of y_t so that $\mathcal{F}_{t-1} \approx \{y_{t-1}, y_{t-2}, \cdots, y_{t-p}\}$. It is also reasonable to assume that some autoregressors between y_{t-1} and y_{t-p} do not matter either. In other words, we desire a sparse AR(p) with the order p sufficiently large but finite. Due to its successful application in high dimensional linear regression model, Cox proportional hazards model and other areas, the LASSO may be naturally the first choice for many time series data analysts if they like to build a sparse AR(P) model by shrinking irrelevant autoregressive coefficients to zero. In fact, there have been quite a few results in the literature that employed the LASSO methodology to build AR(p) models, as we reviewed in Section 1.3.

We start with a review on some basic concepts regarding the AR(p) process, and classic procedure for building an AR(p) model. In Section 2.3 we review the adaptive LASSO of Zou (2006) for the situation in which the AR order is is known a priori or has been identified al-

ready, and then propose the doubly adaptive LASSO for the situation in which the AR order is unknown or difficult to identify a priori, as is the usual case. In Section 2.4 we study the asymptotic properties of the doubly adaptive LASSO estimators. The algorithmic implementation is discussed in Section 2.5. Results from simulation study are summarized in Section 2.6. Examples of real data analysis using the doubly adaptive LASSO procedure are presented in Section 2.7.

2.2 The AR(p) process and standard modelling procedure

Definition (**The AR(p) process**). The time series $\{y_t\}, t \in \mathbb{Z} = \{0, \pm 1, \pm 2, \cdots\}$ is said to be an AR(p) process if it is stationary, and it is the solution of the specification

$$y_t = \phi_1 y_{t-1} + \cdots + \phi_p y_{t-p} + a_t, \quad t \in \mathbb{Z}, \tag{2.1}$$

where $\phi_1, \cdots, \phi_p$ are unknown parameters, $a_t \sim \text{WN}\left(0, \sigma_a^2\right)$. We say that $\{y_t\}$ is an AR(p) process with mean μ if $\{y_t - \mu\}$ is an AR(p) process.

In this thesis, for convenience and without loss of generality, we deal with only the demeaned AR(p) process.

Recall that for the stationary process $\{y_t\}$ the *autocovariance* between y_t and y_{t+k} is

$$\gamma(k) = \text{Cov}(y_t, y_{t+k}) = E[(y_t - \mu)(y_{t+k} - \mu)],$$

and the *autocorrelation* between y_t and y_{t+k} is

$$\rho(k) = \frac{\gamma(k)}{\gamma(0)},$$

where $\gamma(0) = \text{VAR}[y_t] = \text{VAR}[y_t] = \sigma_a^2$. Note that $\rho(0) = 1$ and $\rho(k) < 1 \forall k \neq 0$. The *partial autocorrelation coefficient* (PAC) at lag k, ρ_{kk}, is the autocorrelation between y_t and y_{t+k} after their dependency on the intervening variables $y_{t+1}, \cdots, y_{t+k-1}$ has been removed, namely,

$$\rho_{kk} = \text{Cor}\left(y_t,\ y_{t+k} | y_{t+1},\ \cdots, y_{t+k-1}\right). \tag{2.2}$$

Note that $\rho_{00} = 1$ and $\rho_{11} = \rho(1)$. Using Durbin's recursive algorithm, we compute ρ_{kk} for $|k| > 1$. Starting with $\rho_{11} = \rho(1)$, compute recursively

$$\begin{cases} \rho_{kk} = \frac{\rho(k) - \sum_{j=1}^{k-1} \rho_{k-1,j}\rho_{k-j}}{1 - \sum_{j=1}^{k-1} \rho_{k-1,j}\rho(j)} \\ \rho_{kj} = \rho_{k-1,j} - \rho_{kk}\rho_{k-1,k-j}, \quad j = 1, \cdots, k-1 \end{cases} . \tag{2.3}$$

To estimate ρ_{kk} using observed data $y_1, y_2, \cdots, y_T$, we estimate $\rho(k)$ by the sample autocorrelation defined as

$$\hat{\rho}(k) = \frac{\hat{\gamma}(k)}{\hat{\gamma}(0)} = \frac{\sum_{t=1}^{T-k}(y_t - \bar{y})(y_{t+k} - \bar{y})}{\sum_{t=1}^{T}(y_t - \bar{y})^2},$$

for $k = 0, 1, 2, \cdots$. Starting with $\hat{\rho}_{11} = \hat{\rho}(1)$, compute recursively via Durbin's algorithm (2.3) to get $\hat{\rho}_{kk}$ for $k = 2, 3, \cdots$.

The sample PAC coefficients have a nice asymptotic property. The variables $\sqrt{T}\hat{\rho}_{p+1,p+1}$, $\sqrt{T}\hat{\rho}_{p+2,p+2}$, $\cdots$ are asymptotically iid(0, 1) (Quenouille, 1949, 1957). On a sample partial correlogram, a plot of $\hat{\rho}_{kk}$ versus k, there would display a sharp cutoff at lag p, and $\hat{\rho}_{kk}$ for $k > p$ appear insignificant. So the lag at which the PAC function cuts off is the indicated lag order of the AR model.

Estimation of the AR(p) model

Given the order p there are a variety of approaches to estimating the parameters (see, for example, Hamilton p.117 - 146, 1994). If the distribution of the innovation process $\{a_t\}$ is known, we may obtain the maximum likelihood estimates MLE by maximizing the log-likelihood function. Through the Yule-Walker equations we may obtain the method-of-moments estimator. Maximizing the Gaussian quasi-likelihood may yield qausi maximum likelihood estimates (QMLE) if the normal distribution is used as a proxy for the unknown innovation distribution $\{a_t\}$. Another possibility is to treat $y_t = \phi_1 y_{t-1} + \cdots + \phi_p y_{t-p} + a_t, t = 1, \cdots, T$ as regression equations and employ the ordinary least squares (OLS) method for estimation. The OLS estimator has downward bias, which is known as Hurwicz bias (Hurwicz, 1950). However, the OLS estimator has nice asymptotic properties such as consistency (Hurwicz bias vanishes as $T \to \infty$) and asymptotic normality under some regularity conditions (see, for example, Hayashi p.109 - 117, 2000).

Model selection via minimizing criteria

A sequence of AR models are estimated with sequentially increasing orders $1, 2, ..., h$ with h sufficiently large. Then the model that minimizes some criterion is chosen. Some frequently used criteria include the final prediction error (FPE) (Akaike, 1969), the Akaike information criterion (AIC) (Akaike, 1974), the Bayesian information criterion (BIC) (Schwarz, 1978), and the HQ criteria (Hannan and Quinn, 1979).

Box-Jenkins methodology

Perhaps the most popular procedure for time series data analysis is the Box-Jenkins methodology, which starts with identification of the AR lag order. Parameter estimation follows the a priori identification. A variety of methods has been proposed for order identification. De Gooijer, Abraham, Gould and Robinson (1985) reviewed and discussed the most important of the order determination methods in their survey paper. Choi (1992) devoted a monograph to the identification of ARMA models. A popular method employed by time series data analysts is via the the partial autocorrelation (PAC) function using the cut-off property of the partial autocorrelation functions on the sample partial correlogram (e.g. Hipel, McLeod, and Lennox, 1977).

Subset selection

Because the true order is generally unknown a priori, the problem of the criterion-based model selection approaches is that a nested structure is enforced on the various models, in the sense that only models in which the first h coefficients are non-zero are considered. McLeod and Zhang (2006) propose a subsets selection method to circumvent this problem by examining the problem in partial autocorrelation space.

2.3 The adaptive and doubly adaptive LASSO

Classical approaches we reviewed in the previous section consist of several separate steps, and quickly become computationally infeasible as the AR order grows. In this section, we use the LASSO methodology to model the AR(p) process. There are two situations. If the order is

known in advance or has been identified already, we recommend the adaptive LASSO of Zou (2006). If the order is not known in advance or difficult to identify, we propose the doubly adaptive LASSO, or partial autocorrelation or PAC-weighted adaptive LASSO. By employing the PAC-weighted adaptive LASSO we want to get order identification, subset selection and parameter estimation properly done in one go.

2.3.1 The doubly adaptive LASSO when p is unknown

Suppose that we observe a time series $y_1, y_2, \cdots, y_T$, which is a realization of a stationary AR process with the true order p as well as true parameters $\boldsymbol{\phi}^o = (\phi_1^o, \cdots, \phi_p^o)$ unknown. For this situation we propose the doubly adaptive LASSO approach for a sparse estimator. We first set our guess of the AR order to be h, a sufficiently large positive integer[1]. Since the initial values $y_0, \cdots, y_{-h+1}$ are not available, we use $y_1, \cdots, y_h$ as a presample, hence the effective sample size is $T - h$. Now, having the data, we formulate the following AR(h) model

$$y_t = \phi_1 y_{t-1} + \cdots + \phi_h y_{t-h} + a_t, \;\; t = h+1, \cdots, T. \tag{2.4}$$

Let

$$\boldsymbol{\phi} = (\phi_1, \cdots, \phi_h)', \text{ and} \tag{2.5}$$

$$\boldsymbol{x}_t = (y_t, \, y_{t-1} \cdots, \, y_{t-h+1})', \tag{2.6}$$

and we may write the model equivalently as

$$y_t = \boldsymbol{x}_{t-1}' \boldsymbol{\phi}, \;\; t = h+1, \cdots, T. \tag{2.7}$$

Let

$$\boldsymbol{y} = (y_{h+1}, \cdots, y_T)', \tag{2.8}$$

$$\boldsymbol{a} = (a_{h+1}, \cdots, a_T)', \text{ and} \tag{2.9}$$

$$\boldsymbol{X} = (\boldsymbol{x}_h, \cdots, \boldsymbol{x}_{T-1}) = \begin{pmatrix} y_h & y_{h+1} & \cdots & y_{T-1} \\ y_{h-1} & y_h & \cdots & y_{T-2} \\ \vdots & \vdots & & \vdots \\ y_1 & y_2 & \cdots & y_{T-h} \end{pmatrix}_{h \times (T-h)}, \tag{2.10}$$

and we may write the same model (2.4) compactly in matrix form as

$$\boldsymbol{y} = \boldsymbol{X}' \boldsymbol{\phi} + \boldsymbol{a}. \tag{2.11}$$

[1] h is set to be quite large, for instance, $h = \kappa T^{\alpha}$, $0 \le \alpha \le 1$ for some constant κ.

Definition (**The doubly adaptive LASSO**). The doubly adaptive LASSO or PAC-weighted adaptive LASSO estimator, denoted by $\hat{\boldsymbol{\phi}}_T^{daL}$, is defined as

$$\hat{\boldsymbol{\phi}}_T^{daL} = \arg\min_{\phi} \left\{ (\boldsymbol{y} - \boldsymbol{X}'\boldsymbol{\phi})'(\boldsymbol{y} - \boldsymbol{X}'\boldsymbol{\phi}) + \lambda_T \sum_{j=1}^{h} \hat{w}_{T,j} |\phi_j| \right\}. \tag{2.12}$$

where

$$\hat{w}_{T,j} = \frac{1}{|\tilde{\phi}_j|^{\gamma_1} \left(\sum_{i=j}^{h} |\hat{\rho}_{ii}|^{\gamma_0} \right)^{\gamma_2}} = \frac{1}{|\tilde{\phi}_j|^{\gamma_1} A_j^{\gamma_2}}, \tag{2.13}$$

$$A_j = \sum_{i=j}^{h} |\hat{\rho}_{ii}|^{\gamma_0}, \tag{2.14}$$

for $j = 1, \cdots, h$, $\tilde{\phi}_j$ is any consistent estimate for ϕ_j, $\hat{\rho}_{ii}$ is the estimate for ρ_{ii} defined in (2.2), and $\gamma_0 > 0$, $\gamma_1 \geq 0$, and $\gamma_2 \geq 0$ are some fixed constants.

Remark 1. Both the LASSO (Tibshirani, 1996) and the adaptive LASSO (Zou, 2006) are special cases of the doubly adaptive LASSO. In former case, $\gamma_1 = \gamma_2 = 0$, and in latter case, $\gamma_2 = 0$.

Remark 2. In the doubly adaptive LASSO procedure the PAC information and the Y-W or OLS estimates of the AR(h) model work in tandem to perform subset selection and parameter estimation simultaneously. The basic idea can be elucidated from the following points:

Firstly, note that A_j is the tailed cumulative sum of PAC coefficients to power γ_0 from jth lag to the maximum lag h, and $A_1 \geq \cdots \geq A_p \geq \cdots \geq A_h$. Hence, $\hat{w}_{T,j}$ is decreasing with increasing j. Therefore monotonically increasing penalties are imposed on ϕ_j's as j increases from 1 to h. Consequently, depending on the structure of the PAC, an AR term with smaller lag is more likely to be included in the model.

Secondly, due to the cutoff property of the PAC function, namely, that the value of $|\hat{\rho}_{ii}|$ for $i = p+1, p+2 \cdots, h$ are relatively tiny, it is expected that the A_j will exhibit a sharp jump at $j = p$ as j goes from h backwards to p, the true order of AR process. Consequently, the AR terms with lags greater than p get much more penalties so that they are more likely to be excluded from the model, and the true order of the ARCH process is thus automatically identified.

Finally, $|\tilde{\phi}_j|^{\gamma_1}$ imposes larger penalty on ϕ_j if the corresponding AR term is not significant. This is obvious because if an AR term is not important, the consistently estimated value of the corresponding coefficient is close to zero, and the penalty is close to ∞. Consequently, the insignificant AR terms get more penalties so that they are more likely to be excluded from the model whereas the significant AR terms are more likely to be included in the model.

Remark 3. Let $\boldsymbol{\phi}^o$ be the unknown true parameter vector, that is,

$$\boldsymbol{\phi}^o = (\phi_1^o, \cdots, \phi_p^o)^{'}. \tag{2.15}$$

Using the PAC-weighted adaptive LASSO, we actually estimate the extended true parameter vector, $\boldsymbol{\phi}^*$, defined as

$$\boldsymbol{\phi}^* = (\phi_1^*, \cdots, \phi_p^*, \phi_{p+1}^*, \cdots, \phi_h^*)^{'} = (\phi_1^o, \cdots, \phi_p^o, 0, \cdots, 0)^{'} \tag{2.16}$$

It is clear that the AR(p) process with the true parameter vector $\boldsymbol{\phi}^o$ and the AR(h) process with the extended true parameter vector $\boldsymbol{\phi}^*$ are equivalent.

2.3.2 The adaptive LASSO when p is known

Suppose that the true order p is known or has been identified. Then we set $h = p$ and $\gamma_2 = 0$ in (2.13). We use $y_1, \cdots, y_p$ as a presample, hence the effective sample size is $T - p$. The doubly adaptive LASSO reduces to the adaptive LASSO.

2.4 Asymptotic properties of the doubly adaptive LASSO

The adaptive LASSO and the doubly adaptive LASSO methods yield biased estimators. In this section, however, we show that with properly chosen values for γ_0, γ_1, and γ_2 in (2.13), together with a proper choice of λ_T, the doubly adaptive LASSO enjoys desirable asymptotic properties. We actually study the asymptotic properties of the doubly adaptive LASSO estimator for the extended true parameter vector $\boldsymbol{\phi}^*$ in (2.16) instead of $\boldsymbol{\phi}^o$ in (2.15).

First, we clarify notations. Let $\mathbb{S}$ be the set of the true nonzero coefficient, i.e. $\mathbb{S} = \{j : \phi_j^* \neq 0\} = \text{supp}(\boldsymbol{\phi}^*) \subset \{1, 2, \cdots, h\}$ with h being set large enough such that $h > p$. Let $\mathbb{S}^c = \{1, 2, \cdots, h\} \setminus$

$\mathbb{S}$. Let $s = |\mathbb{S}|$ be the cardinality of the set $\mathbb{S}$. The model sparsity implies that $s < p$. Let $\tilde{\phi}_j$ be any consistent estimate for the true ϕ_j^*, say the OLS or Yule-Walker estimate. Let $\hat{\phi}_{T,j}^{daL}$ be the doubly adaptive LASSO estimate for ϕ_j^*. Let $\hat{\mathbb{S}}_T = \{j : \hat{\phi}_{T,j}^{daL} \neq 0\}$ and $\hat{\mathbb{S}}_T^c = \{1, 2, \cdots, h\} \setminus \hat{\mathbb{S}}_T$. Let $\boldsymbol{\phi}_{\mathbb{S}}^*$ be the s-dimensional vector for true underlying nonzero parameters, and $\boldsymbol{\phi}_{\mathbb{S}^c}^*$ be the vector for true underlying null parameters, i.e. $\boldsymbol{\phi}_{\mathbb{S}}^* = \{\phi_j^* : j \in \mathbb{S}\}$ and $\boldsymbol{\phi}_{\mathbb{S}^c}^* = \{\phi_j^* : j \in \mathbb{S}^c\}$. Let $\hat{\boldsymbol{\phi}}_{T,\mathbb{S}}^{daL}$ be the vector for the PAC-weighted adaptive LASSO estimate for $\boldsymbol{\phi}_{\mathbb{S}}^*$ and $\hat{\boldsymbol{\phi}}_{T,\mathbb{S}^c}^{daL}$ the vector for PAC-weighted adaptive LASSO estimate for null vector $\boldsymbol{\phi}_{\mathbb{S}^c}^*$, i.e. $\hat{\boldsymbol{\phi}}_{T,\mathbb{S}}^{daL} = \{\hat{\phi}_{T,j}^{daL} : j \in \mathbb{S}\}$ and $\hat{\boldsymbol{\phi}}_{T,\mathbb{S}^c}^{daL} = \{\hat{\phi}_{T,j}^{daL} : j \in \mathbb{S}^c\}$. Let $\hat{\boldsymbol{\phi}}_{\hat{\mathbb{S}}_T}^{daL}$ be the vector for nonzero estimates from the doubly adaptive LASSO and $\hat{\boldsymbol{\phi}}_{\hat{\mathbb{S}}_T^c}^{daL}$ the vector for null estimates, i.e. $\hat{\boldsymbol{\phi}}_{\hat{\mathbb{S}}_T}^{daL} = \{\hat{\phi}_{T,j}^{daL} : j \in \hat{\mathbb{S}}_T\}$ and $\hat{\boldsymbol{\phi}}_{\hat{\mathbb{S}}_T^c}^{daL} = \{\hat{\phi}_{T,j}^{daL} : j \in \hat{\mathbb{S}}_T^c\}$.

Proposition 2.4.1 *(**The condition for the ergodic stationarity**). The AR(h) process specified by (2.1) is ergodic stationary if and only if the corresponding characteristic equation satisfies the stability condition, namely,*

$$1 - \phi_1 z - \cdots - \phi_p z^h \neq 0, \quad \textit{for } |z| \leq 1.$$

See Hayashi (2000) p.374 for proof.

Let Γ be the covariance matrix of $\boldsymbol{x}_t$ in (2.6), namely,

$$\Gamma = E[\boldsymbol{x}_t \boldsymbol{x}_t'] = \begin{pmatrix} \sigma_a^2 & \gamma(1) & \cdots & \gamma(h-1) \\ \gamma(1) & \sigma_a^2 & \cdots & \gamma(h-2) \\ \vdots & \vdots & & \vdots \\ \gamma(h-1) & \gamma(h-2) & \cdots & \sigma_a^2 \end{pmatrix}_{h \times h}. \tag{2.17}$$

Γ is symmetric and can be partitioned as follows

$$\Gamma = \begin{pmatrix} \Gamma_{\mathbb{S}\mathbb{S}} & \Gamma_{\mathbb{S}\mathbb{S}^c} \\ \Gamma_{\mathbb{S}^c\mathbb{S}} & \Gamma_{\mathbb{S}^c\mathbb{S}^c} \end{pmatrix},$$

where the ordering is retained according to the lag index of $\boldsymbol{x}_t$ within each partition.

Assumptions:

A0: The coefficients vector $\boldsymbol{\phi}$ belongs to a compact parameter space.

A1: $\forall \boldsymbol{\phi}$ in the parameter space, $1 - \phi_1 z - \cdots - \phi_p z^h \neq 0$ for $|z| \leq 1$.

A2: The process a_t is a strong white noise, i.e. $E[a_t] = 0$, a_t and a_s are independent for $s \neq t$, and $E[a_t^4] < M < \infty$.

A3: $\Gamma_{\mathbb{SS}}$ is not singular and therefore invertible.

Remarks:

1) **A0** is always assumed.

2) **A1** ensures that $\{\boldsymbol{x}_t\}$ is ergodic stationary.

3) No normality of a_t is assumed.

4) **A2** requires the existence of fourth moments of $\{y_t\}$.

Lemma 2.4.2 . *Under **A1** and **A2**, we have*

(i) $\frac{1}{T-h}\boldsymbol{X}\boldsymbol{X}' \xrightarrow{a.s.} \Gamma$,

(ii) $\frac{1}{T-h}\boldsymbol{X}\boldsymbol{a} \xrightarrow{a.s.} \mathbf{0}$, *and*

(iii) $\frac{1}{\sqrt{T-h}}\boldsymbol{X}\boldsymbol{a} \xrightarrow{D} \boldsymbol{w} \sim N(\mathbf{0}, \sigma_a^2\Gamma)$.

Proof (i) It is easy to check that $\boldsymbol{X}\boldsymbol{X}' = \sum_{t=h}^{T-1} \boldsymbol{x}_t\boldsymbol{x}_t'$. By **A1**, $\boldsymbol{x}_t$ is ergodic stationary. By Theorem A.3.1 for ergodicity of functions, $\boldsymbol{x}_t\boldsymbol{x}_t'$ is also ergodic stationary. By Ergodic Theorem A.3.2, we have

$$\frac{1}{T-h}\boldsymbol{X}\boldsymbol{X}' \xrightarrow{a.s.} E[\boldsymbol{x}_t\boldsymbol{x}_t'] = \Gamma.$$

(ii) It is not very hard to check that $\boldsymbol{X}\boldsymbol{a} = \sum_{t=h+1}^{T} \boldsymbol{x}_{t-1}a_t$. Since $\boldsymbol{x}_t$ is ergodic stationary by **A1**, so is $\boldsymbol{x}_{t-1}a_t$ by Theorem A.3.1 for ergodicity of functions. By Ergodic Theorem A.3.2, we have

$$\frac{1}{T-h}\boldsymbol{X}\boldsymbol{a} \xrightarrow{a.s.} E[\boldsymbol{x}_{t-1}a_t],$$

where $E[\boldsymbol{x}_{t-1}a_t] = E\left[[\boldsymbol{x}_{t-1}a_t|\mathcal{F}_{t-1}]\right] = \boldsymbol{x}_{t-1}E[a_t|\mathcal{F}_{t-1}] = \mathbf{0}$.

(iii) Let $\boldsymbol{\nu}_t = \boldsymbol{x}_{t-1}a_t$. Then $\{\boldsymbol{\nu}_t\}$ is a vector martingale difference (MDS) because $E[\boldsymbol{\nu}_t|\mathcal{F}_{t-1}] = \mathbf{0}$. By **A1**, **A2**, and Theorem A.4.1, the CLT for the MDS (Billingsley, 1961), we have

$$\frac{1}{\sqrt{T-h}}\sum_{t=h+1}^{T} \boldsymbol{\nu}_t \xrightarrow{D} N(\mathbf{0}, \ \Sigma_\nu),$$

where $\Sigma_\nu = \text{Var}[\boldsymbol{\nu}_t] = \text{Var}[\boldsymbol{x}_{t-1}a_t] = E[\boldsymbol{x}_{t-1}\boldsymbol{x}_{t-1}'a_t^2] = \sigma_a^2\Gamma$. ∎

Definition (**Estimation consistency**). An estimator $\hat{\boldsymbol{\phi}}_T$ is said to be consistent for $\boldsymbol{\phi}^*$ if

$$\| \hat{\boldsymbol{\phi}}_T - \boldsymbol{\phi}^* \| \xrightarrow{P} 0 \text{ as } T \to \infty.$$

Theorem 2.4.3 ***(Estimation Consistency)***. *Let* $a_T = \sqrt{T-h}\min_{j\in\mathbb{S}}\left(|\tilde{\phi}_{T,j}|^{\gamma_1}A_j^{\gamma_2}\right)$. *If* $\lambda_T = o_p(a_T)$, *then under assumptions* ***A0 – A2***, $\hat{\boldsymbol{\phi}}_T^{daL}$ *must satisfy:*

$$\| \hat{\boldsymbol{\phi}}_T^{daL} - \boldsymbol{\phi}^* \| = O_p\left((T-h)^{-1/2}\right).$$

Proof Let $\Psi_T(\boldsymbol{\phi})$ by defined as

$$\Psi_T(\boldsymbol{\phi}) = \| \boldsymbol{y} - \boldsymbol{X}'\boldsymbol{\phi}) \|^2 + \lambda_T \sum_{j=1}^{h} \hat{w}_{T,j}|\phi_j|, \tag{2.18}$$

where $\boldsymbol{X}$ is defined by (2.10) and $\boldsymbol{y}$ by (2.8). Following Fan and Li (2001), we show that for every $\epsilon > 0$ there exists a sufficiently large C such that

$$\mathbb{P}\left(\inf_{\|\boldsymbol{u}\|\geq C} \Psi_T\left(\boldsymbol{\phi}^* + \boldsymbol{u}/\sqrt{T-h}\right) > \Psi_T(\boldsymbol{\phi}^*)\right) \geq 1-\epsilon,$$

which implies that with probability at least $1-\epsilon$ that there exists a minimum in the ball $\{\boldsymbol{\phi}^* + \boldsymbol{u}/\sqrt{T-h} : \|\boldsymbol{u}\| \leq C\}$. Hence there exists a local minimizer such that $\| \hat{\boldsymbol{\phi}}_T^{daL} - \boldsymbol{\phi}^* \| = O_p((T-h)^{-1/2})$. Observe that

$$\begin{aligned}
&\Psi_T\left(\boldsymbol{\phi}^* + \boldsymbol{u}/\sqrt{T-h}\right) - \Psi_T\left(\boldsymbol{\phi}^*\right) \\
&= \left\|\boldsymbol{y} - \boldsymbol{X}'\left(\boldsymbol{\phi}^* + \boldsymbol{u}/\sqrt{T-h}\right)\right\|^2 - \left\|\boldsymbol{y} - \boldsymbol{X}'\boldsymbol{\phi}^*\right\|^2 + \lambda_T \sum_{j=1}^{h} \hat{w}_{T,j}\left(\left|\phi_j^* + \frac{u_j}{\sqrt{T-h}}\right| - \left|\phi_j^*\right|\right) \\
&= \boldsymbol{u}'\left(\frac{1}{T-h}\boldsymbol{X}\boldsymbol{X})\right)\boldsymbol{u} - 2\boldsymbol{u}'\left(\frac{1}{\sqrt{T-h}}\boldsymbol{X}\boldsymbol{a}\right) + \lambda_T \sum_{j\in\mathbb{S}} \hat{w}_{T,j}\left(\left|\phi_j^* + \frac{u_j}{\sqrt{T-h}}\right| - \left|\phi_j^*\right|\right) + \lambda_T \sum_{j\notin\mathbb{S}} \hat{w}_{T,j}\frac{|u_j|}{\sqrt{T-h}} \\
&\geq \boldsymbol{u}'\left(\frac{1}{T-h}\boldsymbol{X}\boldsymbol{X}'\right)\boldsymbol{u} - 2\boldsymbol{u}'\left(\frac{1}{\sqrt{T-h}}\boldsymbol{X}\boldsymbol{a}\right) + \lambda_T \sum_{j\in\mathbb{S}} \hat{w}_{T,j}\left(\left|\phi_j^* + \frac{u_j}{\sqrt{T-h}}\right| - \left|\phi_j^*\right|\right) \\
&\geq \boldsymbol{u}'\left(\frac{1}{T-h}\boldsymbol{X}\boldsymbol{X}'\right)\boldsymbol{u} - 2\boldsymbol{u}'\left(\frac{1}{\sqrt{T-h}}\boldsymbol{X}\boldsymbol{a}\right) - \lambda_T \sum_{j\in\mathbb{S}} \hat{w}_{T,j}\frac{|u_j|}{\sqrt{T-h}}
\end{aligned}$$

Consider the third term, which can be expressed as

$$\begin{aligned}
\lambda_T \sum_{j=1}^{h} \hat{w}_{T,j}\frac{|u_j|}{\sqrt{T-h}} &= \frac{\lambda_T}{\sqrt{T-h}} \sum_{j\in\mathbb{S}} \left|\tilde{\phi}_j\right|^{-\gamma_1} A_j^{-\gamma_2}|u_j| \\
&\leq \frac{\lambda_T}{\sqrt{T-h}}\left(\min_{j\in\mathbb{S}}\left(|\tilde{\phi}_j|^{\gamma_1}A_j^{\gamma_2}\right)\right)^{-1} \| \boldsymbol{u} \| \\
&= \frac{\lambda_T}{a_T} \| \boldsymbol{u} \| = o_p(1) \| \boldsymbol{u} \| .
\end{aligned}$$

For the second term, by Lemma (2.4.2) (iii), we have

$$\boldsymbol{u}'\left(\frac{1}{\sqrt{T-h}}\right)\boldsymbol{X}\boldsymbol{a} = \boldsymbol{u}' o_p(\mathbf{1}) \leq o_p(1) \parallel \boldsymbol{u} \parallel .$$

For the first term, in view of Lemma (2.4.2) (i), we have

$$\frac{1}{T-h}\boldsymbol{X}\boldsymbol{X}' \to \Gamma \text{ a.s..}$$

So the first term is a quadratic form in $\boldsymbol{u}$.

Then it follows that in probability,

$$\Psi_T\left(\boldsymbol{\phi}^* + \boldsymbol{u}/\sqrt{T-h}\right) - \Psi_T\left(\boldsymbol{\phi}^*\right) \geq \boldsymbol{u}^T\Gamma\boldsymbol{u} - 2o_p(1) \parallel \boldsymbol{u} \parallel .$$

Therefore, for any $\epsilon > 0$, there exists a sufficiently large C such that the quadratic term dominates the other terms with probability $\geq 1-\epsilon$. ∎

Let us look at a condition for Theorem 3.4. Observe that

$$\begin{aligned} A_j &= \left(\sum_{i=j}^{h}|\hat{\rho}_{ii}|^{\gamma_0}\right)^{\gamma_2} \geq \left(\sum_{i=p}^{h}|\hat{\rho}_{ii}|^{\gamma_0}\right)^{\gamma_2} = \left(|\hat{\rho}_{pp}|^{\gamma_0} + \sum_{i=p+1}^{h}|\hat{\rho}_{ii}|^{\gamma_0}\right)^{\gamma_2} \\ &\simeq \left(|\hat{\rho}_{pp}|^{\gamma_0} + (h-p)(T-h)^{-\gamma_0/2}O_p(1)\right)^{\gamma_2}, \end{aligned}$$

$$A_j \leq \left(\sum_{i=1}^{h}|\hat{\rho}_{ii}|^{\gamma_0}\right)^{\gamma_2} = \left(\sum_{i=1}^{p}|\hat{\rho}_{ii}|^{\gamma_0} + \sum_{i=p+1}^{h}|\hat{\rho}_{ii}|^{\gamma_0}\right)^{\gamma_2} \simeq \left(\sum_{i=1}^{p}|\hat{\rho}_{ii}|^{\gamma_0} + (h-p)(T-h)^{-\gamma_0/2}O_p(1)\right)^{\gamma_2},$$

for $j \in \mathbb{S}$. Also $\tilde{\phi}_{T,j} \xrightarrow{P} \phi_j^*$ for $j \in \mathbb{S}$. Hence, $a_T = \sqrt{T-h}O_p(1)$. So the condition $\lambda_T = o_p(a_T)$ in Theorem 3.4 is satisfied if the condition $\lambda_T = o_p\left(\sqrt{T-h}\right)$ in Thereom 1.2.2 is satisfied. Therefore, we may conclude that the LASSO, the adaptive LASSO and the doubly adaptive LASSO are all able to achieve estimation consistency under the same asymptotic condition $\lambda_T = o_p\left(\sqrt{T-h}\right)$. Their performance may be different in finite samples; we need to compare their finite sample properties.

Proposition 2.4.4 . *Let* $a_T = \sqrt{T-h}\min_{j\in\mathbb{S}}\left(|\tilde{\phi}_{T,j}|^{\gamma_1}A_j^{\gamma_2}\right)$, *and* $b_T = \sqrt{T-h}\max_{j\notin\mathbb{S}}\left(|\tilde{\phi}_{T,j}|^{\gamma_1}A_j^{\gamma_2}\right)$. *If* $\lambda_T = o_p(a_T)$ *and* $\lambda_T/b_T \xrightarrow{P} \infty$, *then under assumptions* ***A0*** *–* ***A3****, we have:*

$$\begin{cases} \sqrt{T-h}\left(\hat{\boldsymbol{\phi}}_{T,\mathbb{S}}^{daL} - \boldsymbol{\phi}_{\mathbb{S}}^*\right) \xrightarrow{D} N\left(\mathbf{0},\ \sigma_a^2(\Gamma_{\mathbb{S}\mathbb{S}})^{-1}\right) \\ \sqrt{T-h}\left(\hat{\boldsymbol{\phi}}_{T,\mathbb{S}^c}^{daL} - \boldsymbol{\phi}_{\mathbb{S}^c}^*\right) \xrightarrow{D} \mathbf{0} \end{cases},$$

as $T \to \infty$.

Proof We follow the methodology of Knight and Fu (2000) and Zou (2006).

Let $\boldsymbol{\phi} = \boldsymbol{\phi}^* + \boldsymbol{u}/\sqrt{T-h}$ and define

$$\Psi_T(\boldsymbol{u}) = \left\| \boldsymbol{y} - \boldsymbol{X}\left(\boldsymbol{\phi}^* + \frac{\boldsymbol{u}}{\sqrt{T-h}}\right)\right\|^2 + \lambda_T \sum_{j=1}^{h} \hat{w}_{T,j}\left|\phi_j^* + \frac{u_j}{\sqrt{T-h}}\right|.$$

Let the reparameterized objective function be defined as

$$V_T(\boldsymbol{u}) = \Psi_T(\boldsymbol{u}) - \Psi_T(\boldsymbol{0}).$$

Then the minimizing objective is equivalent to minimizing $V_T(\boldsymbol{u})$ with respect to $\boldsymbol{u}$. Let $\hat{\boldsymbol{u}}_T = \arg\min V_T(\boldsymbol{u})$, then

$$\hat{\boldsymbol{\phi}}_T^{daL} = \boldsymbol{\phi}^* + \hat{\boldsymbol{u}}_T/\sqrt{T-h},$$

or

$$\hat{\boldsymbol{u}}_T = \sqrt{T-h}\left(\hat{\boldsymbol{\phi}}_T^{daL} - \boldsymbol{\phi}^*\right).$$

Observe that

$$V_T(\boldsymbol{u}) = \boldsymbol{u}'\left(\frac{1}{T-h}\boldsymbol{X}\boldsymbol{X}'\right)\boldsymbol{u} - 2\boldsymbol{u}'\left(\frac{1}{\sqrt{T-h}}\boldsymbol{X}\boldsymbol{a}\right) + \frac{\lambda_T}{\sqrt{T-h}}\sum_{j=1}^{h}\hat{w}_{T,j}\sqrt{T-h}\left(\left|\phi_j^* + \frac{u_j}{\sqrt{T-h}}\right| - \left|\phi_j^*\right|\right).$$

By Lemma (2.4.2) we have $\frac{1}{T-h}\boldsymbol{X}\boldsymbol{X}' \xrightarrow{a.s.} \Gamma$, and $\frac{1}{\sqrt{T-h}}\boldsymbol{X}\boldsymbol{a} \xrightarrow{D} \boldsymbol{w} \sim N(\boldsymbol{0}, \sigma_a^2\Gamma)$. Consider the limiting behaviour of the third term. First, by the conditions required in the theorem, we have $\lambda_T\hat{w}_{T,j}/\sqrt{T-h} \le \lambda_T/\left(\sqrt{T-h}\min_{j\in\mathbb{S}}\left(\left|\tilde{\phi}_j\right|^{\gamma_1}A_j^{\gamma_2}\right)\right) = \lambda_T/a_T \xrightarrow{P} 0$ for $j \in \mathbb{S}$ and $\frac{\lambda_T}{\sqrt{T-h}}w_{T,j} = \frac{\lambda_T}{\sqrt{T-h}}|\tilde{\phi}_j|^{-\gamma_1}A_j^{-\gamma_2} \ge \lambda_T/\left(\sqrt{T-h}\max_{j\notin\mathbb{S}}\left(\left|\tilde{\phi}_j\right|^{\gamma_1}A_j^{\gamma_2}\right)\right) = \lambda_T/b_T \xrightarrow{P} \infty$ for $j \notin \mathbb{S}$. In summary, we have

$$\frac{\lambda_T}{\sqrt{T-h}}\hat{w}_{T,j} = \frac{\lambda_T}{\sqrt{T-h}\left|\tilde{\phi}_j\right|^{\gamma_1}A_j^{\gamma_2}} \xrightarrow{P} \begin{cases} 0 & \text{if } j \in \mathbb{S} \\ \infty & \text{if } j \notin \mathbb{S}. \end{cases}$$

Secondly, we have

$$\sqrt{T-h}\left(\left|\phi_j^* + \frac{u_j}{\sqrt{T-h}}\right| - \phi_j^*\right) \to \begin{cases} u_j\operatorname{sgn}(\phi_j^*) & \text{if } j \in \mathbb{S} \ \ (\phi_j^* = 0) \\ |u_j| & \text{if } j \notin \mathbb{S} \ \ (\phi_j^* \neq 0) \end{cases}$$

By Slutsky's theorem, we have the following limiting behaviour of the third term

$$\frac{\lambda_T}{\sqrt{T-h}}\hat{w}_{T,j}\sqrt{T-h}\left(\left|\phi_j^* + \frac{u_j}{\sqrt{T-h}}\right| - \left|\phi_j^*\right|\right) \xrightarrow{P} \begin{cases} 0 & \text{if } \forall j \in \mathbb{S} \\ 0 & \text{if } u_j = 0, \ \forall j \notin \mathbb{S} \\ \infty & \text{otherwise.} \end{cases}$$

Thus, we have $V_T(\boldsymbol{u}) \to V(\boldsymbol{u})$ for every $\boldsymbol{u}$, where

$$\begin{aligned} V(\boldsymbol{u}) &= \begin{pmatrix} \boldsymbol{u}'_{\mathbb{S}} & \boldsymbol{u}'_{\mathbb{S}^c} \end{pmatrix} \begin{pmatrix} \Gamma_{\mathbb{S}\mathbb{S}} & \Gamma_{\mathbb{S}\mathbb{S}^c} \\ \Gamma_{\mathbb{S}^c\mathbb{S}} & \Gamma_{\mathbb{S}^c\mathbb{S}^c} \end{pmatrix} \begin{pmatrix} \boldsymbol{u}_{\mathbb{S}} \\ \boldsymbol{u}_{\mathbb{S}^c} \end{pmatrix} - 2 \begin{pmatrix} \boldsymbol{u}'_{\mathbb{S}} & \boldsymbol{u}'_{\mathbb{S}^c} \end{pmatrix} \begin{pmatrix} \boldsymbol{w}_{\mathbb{S}} \\ \boldsymbol{w}_{\mathbb{S}^c} \end{pmatrix} \\ &\quad + \sum_{j \in \mathbb{S}^c} \frac{\lambda_T}{\sqrt{T-h}} \hat{w}_{T,j} \sqrt{T-h} \left(\left| \phi^*_j + \frac{u_j}{\sqrt{T-h}} \right| - \left| \phi^*_j \right| \right) \\ &= \begin{cases} \boldsymbol{u}'_{\mathbb{S}} \Gamma_{\mathbb{S}\mathbb{S}} \boldsymbol{u}_{\mathbb{S}} - 2 \boldsymbol{u}'_{\mathbb{S}} \boldsymbol{w}_{\mathbb{S}} & \text{if } \boldsymbol{u}_{\mathbb{S}^c} = \boldsymbol{0} \\ \infty & \text{otherwise.} \end{cases} \end{aligned}$$

$V_T(\boldsymbol{u})$ is convex with the unique minimum at $\left((\Gamma_{\mathbb{S}\mathbb{S}})^{-1} \boldsymbol{w}_{\mathbb{S}}, \ \boldsymbol{0} \right)'$. Following the epiconvergence results of Geyer (1994) and Knight and Fu (2000), $\operatorname{argmin}_{\boldsymbol{u}} V_T(\boldsymbol{u}) \xrightarrow{D} \operatorname{argmin}_{\boldsymbol{u}} V(\boldsymbol{u})$, [2] we have

$$\begin{cases} \hat{\boldsymbol{u}}_{\mathbb{S}} \xrightarrow{D} (\Gamma_{\mathbb{S}\mathbb{S}})^{-1} \boldsymbol{w}_{\mathbb{S}} \\ \hat{\boldsymbol{u}}_{\mathbb{S}^C} \xrightarrow{D} \boldsymbol{0} \end{cases},$$

or

$$\begin{cases} \sqrt{T-h} \left(\hat{\boldsymbol{\phi}}^{daL}_{T,\mathbb{S}} - \boldsymbol{\phi}^*_{\mathbb{S}} \right) \xrightarrow{D} N(\boldsymbol{0}, \ \sigma_a^2 (\Gamma_{\mathbb{S}\mathbb{S}})^{-1}) \\ \sqrt{T-h} \left(\hat{\boldsymbol{\phi}}^{daL}_{T,\mathbb{S}^c} - \boldsymbol{\phi}^*_{\mathbb{S}^c} \right) \xrightarrow{D} \boldsymbol{0} \end{cases}.$$

∎

Proposition 2.4.4 is very interesting. Imagine a *Teacher-Student dual* in which the teacher generates 500 data sets from a sparse AR(p) model and the student fits sparse AR(p) models for the teacher. The teacher will give the student a good mark if the student could statistically identify the sparsity structure and estimate the coefficients with $\sqrt{T}$-consistency. Because the set $\mathbb{S}$ is unknown for the student, therefore, the student does not know $\hat{\boldsymbol{\phi}}^{daL}_{T,\mathbb{S}}$ whereas the teacher knows everything. In particular, the teacher knows the set $\mathbb{S}$, and he thus knows $\hat{\boldsymbol{\phi}}^{daL}_{T,\mathbb{S}}$. Proposition 2.4.4 is therefore useful for the teacher, the data generator, but of little use for the student, the data analyst.

Corollary 2.4.5 . *Let* $a_T = \sqrt{T-h} \min_{j \in \mathbb{S}} \left(|\tilde{\phi}_{T,j}|^{\gamma_1} A_j^{\gamma_2} \right)$, *and* $b_T = \sqrt{T-h} \max_{j \notin \mathbb{S}} \left(|\tilde{\phi}_{T,j}|^{\gamma_1} A_j^{\gamma_2} \right)$. *If* $\lambda_T = o_p(a_T)$ *and* $\lambda_T / b_T \xrightarrow{P} \infty$, *then under assumptions* ***A0*** *–* ***A3****, we have that*

$$\mathbb{P}\left(j \in \hat{\mathbb{S}}_T \right) \to 1 \ \textit{if} \ j \in \mathbb{S},$$

as $T \to \infty$.

[2] In fact, since V_T can be infinite, we can no longer define convergence via uniform convergence on compact sets but instead defined it via epiconvergence which allows for extended real-valued functions (Knight and Fu, 2000).

Proof By Theorem A.5.1, the $\sqrt{T-h}$-normality of $\hat{\boldsymbol{\phi}}^{daL}_{T,\mathbb{S}}$ in Proposition 2.4.4 implies that $\|\hat{\boldsymbol{\phi}}^{daL}_{T,\mathbb{S}} - \boldsymbol{\phi}^*_{\mathbb{S}}\| = O_p\left(1/\sqrt{T-h}\right)$. Thus, $\hat{\boldsymbol{\phi}}^{daL}_{T,\mathbb{S}} \xrightarrow{P} \boldsymbol{\phi}^*_{\mathbb{S}}$, which implies that $\forall\, j \in \mathbb{S}$, we have $\mathbb{P}\left(j \in \hat{\mathbb{S}}_T\right) \to 1$, as $T \to \infty$. ∎

Fan and Li (2001) discussed the oracle properties of a sparse estimator in the language of Donoho and Johnstone (1994). Heuristically, an oracle procedure can perform as well asymptotically as if the true submodel were known in advance. We extend the notion of the oracle properties of an estimator to the context of AR times series models.

Definition (**Oracle properties**) . The doubly adaptive positive LASSO estimator $\hat{\boldsymbol{\phi}}^{daL}_T$ for $\boldsymbol{\phi}^*$ is said to have the oracle properties if, with probability tending to 1, it could (i) identify the true sparsity pattern, i.e. $\lim P(\hat{\mathbb{S}}_T = \mathbb{S}) = 1$, (ii) identify the true lag order of the AR process, i,e, $\lim P(\hat{p}^{daL}_T = p) = 1$, and (iii) have an optimal estimation rate of the coefficients as $T \to \infty$.

The following theorem says that the doubly adaptive LASSO procedure is an oracle procedure.

Theorem 2.4.6 *(**Oracle properties of** $\hat{\boldsymbol{\phi}}^{daL}_T$). Let* $a_T = \sqrt{T-h}\min_{j\in\mathbb{S}}\left(|\tilde{\phi}_{T,j}|^{\gamma_1}A_j^{\gamma_2}\right)$, *and* $b_T = \sqrt{T-h}\max_{j\notin\mathbb{S}}\left(|\tilde{\phi}_{T,j}|^{\gamma_1}A_j^{\gamma_2}\right)$. *If* $\lambda_T = o_p(a_T)$ *and* $\lambda_T/b_T \xrightarrow{P} \infty$, *then under assumptions* ***A0*** *–* ***A3***, $\hat{\boldsymbol{\phi}}^{daL}_T$ *must satisfy:*

(i) Selection Consistency: $\mathbb{P}\left(\hat{\mathbb{S}}_T = \mathbb{S}\right) \to 1$ *as* $T \to \infty$.

(ii) Identification consistency: $\mathbb{P}\left(\hat{p}^{daL}_T = p\right) \to 1$, *and*

(ii) Asymptotic Normality: $\sqrt{T-h}\left(\hat{\boldsymbol{\phi}}^{daL}_{\hat{\mathbb{S}}_T} - \boldsymbol{\phi}^*_{\mathbb{S}}\right) \xrightarrow{D} N(\mathbf{0},\ \sigma_a^2(\Gamma_{\mathbb{S}\mathbb{S}})^{-1})$ *as* $T \to \infty$.

Proof (i) In view of Corollary 2.4.5, we know that $\forall\, j \in \mathbb{S}$, $P(j \in \hat{\mathbb{S}}_T) \to 1$. So it suffices to show that $\forall m \notin \mathbb{S}$, $P(m \in \hat{\mathbb{S}}_T) \to 0$. Now, we follow the methodology of Zou (2006).

Consider the event $\{m \in \hat{\mathbb{S}}_T\}$. The KKT conditions for optimality entail that

$$2\boldsymbol{X}_{(m,\cdot)}\left(\boldsymbol{y} - \boldsymbol{X}'\hat{\boldsymbol{\phi}}^{daL}_T\right) = \lambda_T \hat{w}_{T,m}\mathrm{sgn}\left(\hat{\phi}^{daL}_{T,m}\right),$$

where the subscript $(m, \cdot)$ denotes the m-th row of a matrix. If $\lambda_T/b_T \xrightarrow{P} \infty$, we have

$$\frac{\lambda_T}{\sqrt{T-h}}\hat{w}_{T,m} = \frac{\lambda_T}{\sqrt{T-h}}\frac{1}{|\tilde{\phi}_m|^{\gamma_1}A_m^{\gamma_2}} \geq \frac{\lambda_T}{b_T} \xrightarrow{P} \infty,$$

whereas

$$\frac{\boldsymbol{X}_{(m,\cdot)}\left(\boldsymbol{y}-\boldsymbol{X}'\hat{\boldsymbol{\phi}}_T^{daL}\right)}{\sqrt{T-h}}=\left(\frac{\boldsymbol{X}_{(m,\cdot)}\boldsymbol{X}'}{T-h}\right)\sqrt{T-h}\left(\boldsymbol{\phi}^*-\hat{\boldsymbol{\phi}}_T^{daL}\right)+\frac{\boldsymbol{X}_{(m,\cdot)}\boldsymbol{a}}{\sqrt{T-h}}.$$

Note that $\boldsymbol{X}_{(m,\cdot)}\boldsymbol{a}$ is the m-th element of the vector $\boldsymbol{X}\boldsymbol{a}$, denoted by $(\boldsymbol{X}\boldsymbol{a})_m$. By Lemma 2.4.2, we have

$$\frac{1}{\sqrt{T-h}}(\boldsymbol{X}\boldsymbol{a})_m \xrightarrow{D} N\left(0,\ \sigma_a\Gamma_{(m,m)}\right),$$

where $\Gamma_{(m,m)}$ is the m-th diagonal element of Γ. Note also that $\boldsymbol{X}_{(m,\cdot)}\boldsymbol{X}'$ is the m-th row of the matrix $\boldsymbol{X}\boldsymbol{X}'$, denoted by $(\boldsymbol{X}\boldsymbol{X}')_{(m,\cdot)}$. By Lemma 2.4.2, we have

$$\frac{1}{T-h}(\boldsymbol{X}\boldsymbol{X}')_{(m,\cdot)} \xrightarrow{a.s.} \Gamma_{(m,\cdot)}.$$

By Slutsky's theorem and the results of (i), we see that

$$\frac{1}{T-h}\boldsymbol{X}_{(m,\cdot)}\boldsymbol{X}'\sqrt{T-h}\left(\boldsymbol{\phi}^*-\hat{\boldsymbol{\phi}}_T^{daL}\right)\xrightarrow{D}\ \Gamma_{(m,\cdot)}\boldsymbol{z},$$

where $\boldsymbol{z}$ is a normally-distributed vector, and thus $\Gamma_{(m,\cdot)}\boldsymbol{z}$ a normally-distributed scalar variable. Therefore,

$$P(m\in\hat{\mathbb{S}}_T)\le P\left(2\boldsymbol{X}_{(m,\cdot)}\left(\boldsymbol{y}-\boldsymbol{X}'\hat{\boldsymbol{\phi}}_T^{daL}\right)=\lambda_T\hat{w}_m\mathrm{sgn}\left(\hat{\phi}_{T,m}^{daL}\right)\right)\to 0.$$

(ii) The AR order estimated by the doubly adaptive LASSO is

$$\hat{p}_T^{daL}=\min\left\{j:\ \hat{\phi}_k^{daL}=0\ \forall k=j+1,j+2,\ \cdots,\ h\right\},$$

or equivalently,

$$\hat{p}_T^{daL}=\min\left\{k:\ k\in\hat{\mathbb{S}}_T^c\ \forall k=j+1,j+2,\ \cdots,\ h\right\}. \tag{2.19}$$

The true order p of the AR model is

$$p=\min\{k:\ k\in\mathbb{S}^c\ \forall k=j+1,j+2,\ \cdots,\ h\}. \tag{2.20}$$

We have from (i) that $\hat{\mathbb{S}}_T^c\to\mathbb{S}^c$ in probability, so the RHS of (2.19) and (2.20) are equal in probability. Therefore, $\lim\mathbb{P}(\hat{p}_T^{daL}=p)=1$.

(iii) From (i), we have that $\lim\mathbb{P}\left(\hat{\boldsymbol{\phi}}_{\hat{\mathbb{S}}_T}^{daL}=\hat{\boldsymbol{\phi}}_{T,\mathbb{S}}^{daL}\right)\to 1$. Then, the asymptotic normality of $\hat{\boldsymbol{\phi}}_{\hat{\mathbb{S}}_T}^{daL}$ follows from Proposition 2.4.4. ∎

We continue the story of the *Teacher-Student dual*. The student knows $\hat{\mathbb{S}}_T$ and $\hat{\boldsymbol{\phi}}^{daL}_{\hat{\mathbb{S}}_T}$. Theorem 2.4.6 assures that apart from estimating the coefficients with $\sqrt{T}$-consistency, the student could statistically identify the sparsity structure as if he knew $\mathbb{S}$. Theorem 2.4.6 is therefore useful particularly useful for the student, the data analyst.

Let us look at a condition in Proposition 2.4.4, Corollary 2.4.5 and Theorem 2.4.6. Observe that

$$A_j = \left(\sum_{i=j}^{h} |\hat{\rho}_{ii}|^{\gamma_0} \right)^{\gamma_2} \leq \left(\sum_{i=p+1}^{h} |\hat{\rho}_{ii}|^{\gamma_0} \right)^{\gamma_2} \simeq (h-p)^{\gamma_2} (T-h)^{-\gamma_0\gamma_2/2} O_p(1)$$

for $j > p$, and

$$A_j \leq \left(\sum_{i=1}^{h} |\hat{\rho}_{ii}|^{\gamma_0} \right)^{\gamma_2} = \left(\sum_{i=1}^{p} |\hat{\rho}_{ii}|^{\gamma_0} + \sum_{i=p+1}^{h} |\hat{\rho}_{ii}|^{\gamma_0} \right)^{\gamma_2} \simeq \left(\sum_{i=1}^{p} |\hat{\rho}_{ii}|^{\gamma_0} + (h-p)(T-h)^{-\gamma_0/2} O_p(1) \right)^{\gamma_2}$$

for $j < p$ and $j \in \mathbb{S}^c$. Also $\tilde{\phi}_{T,j} \xrightarrow{P} (T-h)^{-1/2} O_p(1)$ for $j \in \mathbb{S}^c$. Hence, $\sqrt{T-h}|\tilde{\phi}_{T,j}|^{\gamma_1} A_j^{\gamma_2} \simeq (h-p)^{\gamma_2}(T-h)^{(1-\gamma_1-\gamma_0\gamma_2)/2} O_p(1) = (T-h)^{(1-\gamma_1-\gamma_0\gamma_2)/2} O_p(1)$ for $j > p$, and $\sqrt{T-h}|\tilde{\phi}_{T,j}|^{\gamma_1} A_j^{\gamma_2} \simeq (T-h)^{(1-\gamma_1-\gamma_0\gamma_2)/2} O_p(1) = (T-h)^{(1-\gamma_1)/2} O_p(1)$ for $j < p$ and $j \in \mathbb{S}^c$. Recall that Theorem 1.2.8 needs a condition, $\lambda_T/(T-h)^{(1-\gamma_1)/2} \to \infty$; if this condition is satisfied, the condition $\lambda_T/b_T \xrightarrow{P} \infty$ is also satisfied in Proposition 2.4.4, Corollary 2.4.5 and Theorem 2.4.6. Notice also that $\lambda_T/(T-h)^{(1-\gamma_1-\gamma_0\gamma_2)/2} \xrightarrow{P} \infty$ does not imply $\lambda_T/(T-h)^{(1-\gamma_1)/2} \to \infty$.

Remarks:

(**1**) Although the asymptotic distributions of $\hat{\boldsymbol{\phi}}^{daL}_{T,\mathbb{S}}$ and $\hat{\boldsymbol{\phi}}^{daL}_{\hat{\mathbb{S}}_T}$ are identical, $\hat{\boldsymbol{\phi}}^{daL}_{T,\mathbb{S}}$ and $\hat{\boldsymbol{\phi}}^{daL}_{\hat{\mathbb{S}}_T}$ represent different identities; $\hat{\boldsymbol{\phi}}^{daL}_{T,\mathbb{S}}$ is the doubly adaptive LASSO estimator for the vector of the true non-zero parameters we do not know in advance whereas $\hat{\boldsymbol{\phi}}^{daL}_{\hat{\mathbb{S}}_T}$ is the vector for non-zeros estimated by the doubly adaptive LASSO. The delicate difference between $\hat{\boldsymbol{\phi}}^{daL}_{T,\mathbb{S}}$ and $\hat{\boldsymbol{\phi}}^{daL}_{\hat{\mathbb{S}}_T}$ can be understood via the thought experiment of the Teacher-Student dual.

(**2**) In the literature, the oracle properties concern $\hat{\boldsymbol{\phi}}^{daL}_{T,\mathbb{S}}$, as shown by Theorem 1.2.8, which we argue is not quite correct because we, as data analysts, do not really know $\hat{\boldsymbol{\phi}}^{daL}_{T,\mathbb{S}}$ from the start to the end. The oracle properties we discuss here concern $\hat{\boldsymbol{\phi}}^{daL}_{\hat{\mathbb{S}}_T}$ rather than $\hat{\boldsymbol{\phi}}^{daL}_{T,\mathbb{S}}$.

(**3**) Proposition 2.4.4 concerns $\hat{\boldsymbol{\phi}}^{daL}_{T,\mathbb{S}}$, the daLASSO estimators for the true non-zero parameters, which are unknown in advance whereas Theorem 2.4.6 concerns $\hat{\boldsymbol{\phi}}^{daL}_{\hat{\mathbb{S}}_T}$, the non-zeros estimated by the doubly adaptive LASSO.

(**4**) Estimation consistency is necessary for oracle properties whereas oracle properties are sufficient for the former.

(**5**) Under the same asymptotic condition for tuning parameter λ_T (and other regularity conditions), the LASSO, the adaptive LASSO and the doubly adaptive LASSO all have estimation consistency property.

(**6**) Under the same asymptotic condition for tuning parameter λ_T (and other regularity conditions), the adaptive LASSO and the doubly adaptive LASSO both have oracle properties.

(**7**) The LASSO, the adaptive LASSO and the doubly adaptive LASSO estimator might behaviour quite differently when finite samples are used. We need to investigate and compare their finite sample properties.

2.5 Computation algorithms for the doubly adaptive LASSO

Given values of λ_T, γ_0, γ_1, and γ_2, the PAC-weighted adaptive LASSO procedure is implemented via the *lars* algorithm (Efron et al., 2004). The *lars* algorithm is very efficient, requiring the same order of computational cost as that of a single least squares fit. The doubly adaptive LASSO methodology yields a path of possible solutions defined by the continuum over tuning and weighting parameters. The choice of λ_T, γ_0, γ_1, and γ_2 plays a crucial role in the implementation of the doubly adaptive LASSO since it determines the tradeoff between model fit and model sparsity.

Although the BIC criterion has been reported to be the best for the choice of tuning and weighting parameters, other criteria may also be applicable. Madigan and Ridgeway (2004) reported that the C_p performs as well as the cross-validation in linear regression. McQuarrie and Tsai (p. 251-290, 1998) suggested the leaving-one-out cross-validation (LOOCV) or leaving-one-block-out cross-validation (LOBOCV) for nonparametric model selection in time series

analysis. Here we use the Mallows' C_p to choose the optimal value for λ_T, and the LOBOCV to determine the optimal value of γ_0, γ_1 and γ_2.

Choosing λ_T

The *lars* package offers a simple statistic, the Mallows' C_p, which can be used for model selection. We adapt the C_p used in linear regression to AR models

$$C_k = \frac{\text{SSE}_k}{s^2} - (T-h) + 2\text{df}, \tag{2.21}$$

where $k \in \{1, \cdots, h\}$ denotes the number of autoregressors in the fitted model, SSE_k is the sum of the squared errors, i.e. $\text{SSE}_k = \sum_{t=h+1}^{T}(y_t - \hat{E}_k[y_t])^2$ with $\hat{E}_k[y_t]$ being the predicted value for y_t from a sparse AR model fitted via the doubly adaptive LASSO, $s^2 = \text{SSE}_h/(T-2h) = \sum_{t=h+1}^{T}(y_t - \hat{E}_h[y_t])^2/(T-2h)$ with $\hat{E}_h[y_t]$ being the predicted value for y_t from the full AR(h) model, and

$$\text{df} = \frac{\sum_{t=h+1}^{T} cov(\hat{E}_k[y_t],\, y_t)}{s^2},$$

which is roughly the number of non-zero parameters in the model.

Choosing γ_0, γ_1 and γ_2

Recall that for the linear regression model $y_i = \boldsymbol{z_i}\boldsymbol{\beta} + \epsilon_i, i = 1, \cdots, n$, one measure for the performance of a fitted model is based on its prediction ability. The best model is the one that minimizes the *mean squared error of prediction* (MSEP). To estimate the MSEP, Allen (1974) suggested the so-called *leaving-one-out cross validation* (LOOCV) approach. The i-th observation is removed from the data set, and the remaining $(n-1)$ observations are used to fit the model. The estimated coefficients vector is denoted as $\boldsymbol{\beta}_{(i)}$ with (i) indicating that the ith observation is removed from the data. the prediction error $e_{(i)} = y_i - \hat{y}_{(i)}$ where $\hat{y}_{(i)}$ is the predicted value for y_i. Under independent errors assumption, y_i and $\hat{y}_{(i)}$ are independent, and $e_{(i)}^2$ is unbiased for MSEP. Successively removing $i = 1, \cdots, n$ gives $e_{(1)}, \cdots, e_{(n)}$. It seems that we need to fit n regression models to n data of size $n-1$ in order to get $e_{(i)}$'s. Fortunately, we do not have to fit n models because it can be shown that $e_{(i)} = e_i/(1-h_i)$, where h_i's are the diagonal elements of the projection matrix or so-called hat matrix. So it is all sufficient to fit once a

regression model to the whole data of sample size n. The LOOCV is defined as $\frac{1}{n}\sum_{i=1}^{n} e_{(i)}^2$. Thus the LOOCV can be calculated efficiently using LOOCV $= \frac{1}{n}\sum_{i=1}^{n} e_i^2/(1-h_i)^2$.

A key assumption that the observation removed is independent of the remaining ones in linear regression models fails for AR models. However, McQuarrie and Tsai (1998) showed in their simulation study that the LOOCV is still valid for AR model selection. But the direct formula for the LOOCV in the ordinary regression setting we showed in the previous paragraph is no longer available for AR models so computation of LOOCV is not as efficient. McQuarrie and Tsai (1998) also proposed a method called *leaving-one-block-out cross validation* (LOBOCV) that may reduce the temporal dependence in data. Suppose that there exists a constant b such that y_i and y_j are approximately independent for $|i-j| > b$. When leaving y_t out, one leaves out $\pm b$ additional observations around y_t, namely, the block $[y_{i-b}, \cdots, y_i, \cdots, y_{t+b}]$ in $\boldsymbol{y}$, and the block composed of columns $[\boldsymbol{x}_{i-b}, \cdots, \boldsymbol{x}_i, \cdots, \boldsymbol{x}_{t+b}]$ are removed correspondingly. The model is then fitted to the data with the block deleted. So the LOOCV and LOBOCV are defined as

$$\text{LOOCV} = \frac{1}{T-2h}\sum_{t=h+1}^{T}\left(y_t - \boldsymbol{x}_{t-1}'\hat{\boldsymbol{\phi}}_{(t)}^{daL}\right)^2,$$

$$\text{LOBOCV} = \frac{1}{T-2h}\sum_{t=h+1}^{T}\left(y_t - \boldsymbol{x}_{t-1}'\hat{\boldsymbol{\phi}}_{(t\pm b)}^{daL}\right)^2, \text{ for some } b,$$

where $\boldsymbol{x}_{t-1}'$ is defined in (2.6), $\hat{\boldsymbol{\phi}}_{(t)}^{daL}$ is the double adaptive LASSO estimate with the t-th column are removed from $\boldsymbol{X}$, and $\hat{\boldsymbol{\phi}}_{(t\pm b)}^{daL}$ with $(t-b)-(t+b)$ columns removed from $\boldsymbol{X}$. Interestingly, we use both the LOOCV and LOBOCV and we found little difference between the LOOCV and LOBOCV in choosing the parameters so we stick to the LOOCV.

Computational algorithms

Algorithm 3 is the detailed computational procedure for the doubly adaptive LASSO given the value of the triple $(\gamma_0, \gamma_1, \gamma_2)$. Algorithm 4 shows the complete computation steps via the LOOCV.

Algorithm 3: The *lars* algorithm for the doubly adaptive LASSO given $(\gamma_0, \gamma_1, \gamma_2)$.

Input: Data $\boldsymbol{y}_t, t = 1, \cdots, T$, and a specific value for $(\gamma_0, \gamma_1, \gamma_2)$.

Output: $\widehat{\boldsymbol{\phi}}_T^{daL}$ for specific $(\gamma_0, \gamma_1, \gamma_2)$.

1 START

2 Compute $\hat{w}_{T,j}$ defined by (2.13).

3 Compute $\boldsymbol{X}^* = \boldsymbol{X}\boldsymbol{W}^{-1}$, where $\boldsymbol{W} = \text{diag}[\hat{w}_1, \cdots, \hat{w}_h]$, i.e. $\boldsymbol{x}_j^* = \boldsymbol{x}_j / \hat{w}_j, j = 1, \cdots, h$.

4 Apply *lars* to obtain $\hat{\boldsymbol{\phi}}(\lambda_T) = \text{argmin}_{\boldsymbol{\phi}} \left\{ (\boldsymbol{y} - \boldsymbol{X}^* \boldsymbol{\phi})^T (\boldsymbol{y} - \boldsymbol{X}^* \boldsymbol{\phi}) + \lambda_T \sum_{j=1}^{h} |\phi_j| \right\}$.

5 Compute $\hat{\boldsymbol{\phi}}_T^{daL}(\lambda_T) = \boldsymbol{W}^{-1} \hat{\boldsymbol{\phi}}$.

6 Compute $C_p(\lambda_T)$ according to (2.21) for the whole path.

7 Output $\widehat{\boldsymbol{\phi}}_T^{daL}(\lambda_T^*)$ where λ_T^* is such that $C_p(\lambda_T^*) \leq C_p(\lambda_T)$. END

Algorithm 4: Complete algorithm for the doubly adaptive positive LASSO via the LOOCV

Input: Data: $\boldsymbol{y}_t, t = 1, \cdots, T$

Output: The doubly adaptive positive LASSO estimator $\widehat{\boldsymbol{\phi}}_T^{daL}$

1 Start: Set up a grid $\mathcal{G} = \gamma_0 \times \gamma_1 \times \gamma_2$ with $G = |\mathcal{G}|$.

2 **for** $g \leftarrow 1$ ***to*** G **do**

3 Apply Algorithm 3 to get $\widehat{\boldsymbol{\phi}}_T \left(\gamma_0^{(g)}, \gamma_1^{(g)}, \gamma_2^{(g)} \right)$.

4 Calculate LOOCV$(\gamma_0^{(g)}, \gamma_1^{(g)}, \gamma_2^{(g)})$.

5 Choose $(\gamma_0^*, \gamma_1^*, \gamma_2^*)$ such that
LOOCV$(\gamma_0^*, \gamma_1^*, \gamma_2^*) = \min\{\text{LOOCV}(\gamma_0^{(g)}, \gamma_1^{(g)}, \gamma_2^{(g)}) : \forall g = 1, \cdots, G\}$.

6 Output $\widehat{\boldsymbol{\phi}}_T^{daL} \leftarrow \widehat{\boldsymbol{\phi}}_T(\gamma_0^*, \gamma_1^*, \gamma_2^*)$.

7 End

2.6 Monte Carlo study

We use Monte Carlo to empirically assess the statistical properties of the doubly adaptive LASSO estimator with respect to AR order identification, sparse pattern recovery, and parameter estimation. We summarize the empirical minimum, maximum, mean, medium, mode (for AR lag order only), standard error, bias, MSE, MAD, and selection proportion. The definitions of empirical bias, MSE, and MAD are listed as the following

$$\widehat{Bias}(\hat{p}^{daL}) = \hat{E}[\hat{p}^{daL}] - p = \frac{1}{M} \sum_{m=1}^{M} (\hat{p}^{daL})^{(m)} - p$$

$$\widehat{MSE}(\hat{p}^{daL}) = \hat{E}[\hat{p}^{daL} - p]^2 = \frac{1}{M} \sum_{m=1}^{M} ((\hat{p}^{daL})^{(m)} - p)^2$$

$$\widehat{MAD}(\hat{p}^{daL}) = \hat{E}|\hat{p}^{daL} - p| = \frac{1}{M}\sum_{m=1}^{M}|(\hat{p}^{daL})^{(m)} - p|$$

$$\widehat{Bias}(\hat{\phi}_j^{daL}) = \hat{E}[\hat{\phi}_j^{daL}] - \phi_j^* = \frac{1}{M}\sum_{m=1}^{M}(\hat{\phi}_j^{daL})^{(m)} - \phi_j^*$$

$$\widehat{MSE}(\hat{\phi}_j^{daL}) = \hat{E}[\hat{\phi}_j^{daL} - \phi_j^*]^2 = \frac{1}{M}\sum_{m=1}^{M}\left((\hat{\phi}_j^{daL})^{(m)} - \phi_j^*\right)^2$$

$$\widehat{MAD}(\hat{\phi}_j^{daL}) = \hat{E}|\hat{\phi}_j^{daL} - \phi_j^*| = \frac{1}{M}\sum_{m=1}^{M}\left|(\hat{\phi}_j^{daL})^{(m)} - \phi_j^*\right|$$

where M denotes the total number of MC runs.

2.6.1 Performance of the daLASSO with an appropriate choice of tuning and weighting parameters using samples of different sizes

We would like to assess the performance of the doubly adaptive LASSO with an appropriate choice of tuning and weighting parameters using small, medium and large samples. We generated 10,000 data sets of 6 different sample sizes $T = 100, 250, 500, 500, 800, 1500, 2000$ from the stationary AR(15) model:

$$Y_t = 0.2Y_{t-1} + 0.1Y_{t-3} + 0.2Y_{t-5} + 0.2Y_{t-10} + 0.25Y_{t-15} + a_t \tag{2.22}$$

Pretending that we do not know the true lag order of the underlying model, corresponding to each sample size we set maximum order h to be 25, 50, 100, 150, 200, 250, respectively. We set $\gamma_0 = 4.5$, $\gamma_1 = 5$, and $\gamma_2 = 1.45$ [3] and use the doubly adaptive LASSO to fit AR models. Figure 2.1 shows the distribution of estimated AR orders corresponding to the 6 different sample sizes. Figure 2.2 shows the proportions of the coefficients of the AR model being selected corresponding to the 6 different sample size. Table 2.1 shows the empirical statistics of the AR order estimates when the sample size is quite large ($T = 2000$). Table 2.2 shows the empirical statistics of coefficients estimates and proportion of AR coefficients being selected when the sample size is quite large ($T = 2000$). We highlight a few observations.

Observations:

[3] The other choices of values for tuning and weighting parameters might also work.

Table 2.1: Empirical statistics of the doubly adaptive LASSO estimates for the AR order, based on 10,000 replications (10,000 data sets each of size T=2,000 were generated from $Y_t = 0.2Y_{t-1} + 0.1Y_{t-3} + 0.2Y_{t-5} + 0.2Y_{t-10} + 0.25Y_{t-15} + a_t$. Set $h = 250$. Set $\gamma_0 = 4.5$, $\gamma_1 = 5$, and $\gamma_2 = 1.5$. Use the C_p to choose the value of λ_T.)

True	Minimum	Maximum	Mean	Median	Mode	SE	Bias	MSE	MAD
15	10	15	15	15	15	0.05	-0.0005	0.0025	0.0005

Table 2.2: Empirical statistics of the doubly adaptive LASSO estimates for the AR coefficients, based on 10,000 replications (10,000 data sets each of size T=2,000 were generated from $Y_t = 0.2Y_{t-1} + 0.1Y_{t-3} + 0.2Y_{t-5} + 0.2Y_{t-10} + 0.25Y_{t-15} + a_t$. Set $h = 250$. Set $\gamma_0 = 4.5$, $\gamma_1 = 5$, and $\gamma_2 = 1.5$. Use the C_p to choose the value of λ_T.)

Lag	True	Minimum	Maximum	Mean	Median	SE	Bias	MSE	MAD	Proportion
1	0.2	0.114	0.278	0.2000	0.199	0.0207	-0.0012	0.0004	0.01652	1
2	0	-0.087	0.091	-0.0003	0	0.0135	-0.0003	0.0002	0.00453	0.132
3	0.1	0	0.187	0.0980	0.099	0.0224	-0.0015	0.0004	0.01689	0.990
4	0	-0.093	0.081	-0.0004	0	0.0129	-0.0004	0.0002	0.00404	0.114
5	0.2	0.116	0.309	0.2000	0.199	0.0231	-0.0006	0.0005	0.01836	1
6	0	-0.088	0.085	-0.0000	0	0.0079	-0.0001	0.0001	0.00115	0.024
7	0	-0.077	0.074	-0.0001	0	0.007	-0.0001	0.0000	0.00105	0.025
8	0	-0.087	0.081	-0.0001	0	0.0081	-0.0001	0.0001	0.00129	0.028
9	0	-0.092	0.09	-0.0001	0	0.0072	-0.0001	0.0001	0.00105	0.024
10	0.2	0.115	0.285	0.2000	0.198	0.0226	-0.0015	0.0005	0.01805	1
11	0	-0.079	0.085	-0.0000	0	0.0038	-0.0000	0.0000	0.00026	0.005
12	0	-0.096	0.063	-0.0001	0	0.0032	-0.0001	0.0000	0.0002	0.004
13	0	-0.082	0.078	-0.0001	0	0.0037	-0.0001	0.0000	0.00024	0.005
14	0	-0.09	0.075	-0.0001	0	0.0037	-0.0001	0.0000	0.00025	0.005
15	0.25	0	0.323	0.2500	0.249	0.0226	-0.0013	0.0005	0.01797	1.000

(**1**) Distributions of the AR(15) order estimates and order identification consistency. From Figure 2.1, we observe that the daLASSO chose most frequently orders *lager* than the true order 15 for the sample of small size (T = 100), chose *most frequently* the true order 15 as the sample sizes increased to 250 and over, and chose *almost always* the true order as the sample sizes increased to 1500 and over. So it is evident that as sample size gets increasing, the AR order estimated by the doubly adaptive LASSO tends to the true order (15) with probability tending to 1. In addition, the distribution of the daLASSO estimates for the AR(15) order from the small sample (T = 100 or 250) is flatter, more dispersed, and more dependent on h, ranging from 1 to h (h = 25 or 50), the distribution of the daLASSO estimates for the AR(15) order from the moderate sample (T = 500 or 800) is sharper around the mode (15) but right-skewed with long tails with a positive probability of the largest possible order that we initially guessed (25 or 50), and the distribution of the daLASSO estimates for the AR(15) order from the large samples (T = 1500 or 2000) concentrates almost all probability mass at the true order with a

small potion of probability mass at 10, and is not dependent on h. Table 2.1 provides another evidence that the daLASSO estimates for the AR order from a large sample (T = 2000) are very close to the true order.

(**2**) Variable selection consistency. As shown in Figure 2.2, the daLASSO is excellent in excluding the autoregressors beyond the true order 15; the coefficients 16 – 20 are set to 0 even when the samples of moderate size ($T = 500$) are used. Figure 2.2 also shows that the daLASSO is powerful in choosing the true sets of variables (Y_{t-1}, Y_{t-5}, Y_{t-10}, and Y_{t-15} except Y_{t-3}) with probability close to 1 even when the samples used are of moderate size ($T = 500$). The daLASSO is still conservative in the sense that zeros below the true order are falsely chosen with high probability (Y_{t-2} and Y_{t-4}) when the samples used are not large ($T < 2000$). However, both Figure 2.2 and Table 2.2 shows that the doubly adaptive LASSO can satisfactorily recover the sparsity pattern when the sample size is large ($T = 2000$). We may also see that the values for h are almost irrelevant in recovering the sparse pattern.

(**3**) Estimation consistency. Table 2.2 shows $\widehat{MSE}(\hat{\phi}_j^{daL}) \simeq 0,\ j = 1, \cdots, \hat{p}$ when the sample size is quite large ($T = 2000$), which is an evidence that $\hat{\boldsymbol{\phi}}^{daL}$ is asymptotically consistent.

In one word, the simulation shows evidences for the asymptotic properties stated in Section 2.4, that is, with the values of λ, γ_0, γ_1, and γ_2 properly chosen, the doubly adaptive LASSO can achieve identification consistency, selection consistency, and estimation consistency.

2.6.2 Performance of the daLASSO with tuning and weighting parameters being chosen via LOOCV using a sample of moderate size

In the previous subsection, we were lucky to have an appropriate choice of values for γ_0, γ_1, and γ_2. In reality, however, we are not able to determine a proper choice of values for γ_0, γ_1, and γ_2 a priori. Now we would like to assess the performance of the doubly adaptive LASSO with tuning and weighting parameters being chosen via LOOCV using a sample of moderate size. We generated $1,000$ data sets of a moderate size $T = 800$ from a stationary AR(15) model used by Nardi et. al.(2010):

$$Y_t = 0.2Y_{t-1} + 0.1Y_{t-3} + 0.2Y_{t-5} + 0.3Y_{t-10} + 0.1Y_{t-15} + a_t \tag{2.23}$$

Figure 2.1: Empirical distributions of the doubly adaptive LASSO estimates for the AR order as sample size increases, based on 10,000 replications (10,000 data sets for each of 6 different sample sizes were generated form $Y_t = 0.2Y_{t-1} + 0.1Y_{t-3} + 0.2Y_{t-5} + 0.2Y_{t-10} + 0.25Y_{t-15} + a_t$. Set $\gamma_0 = 4.5$, $\gamma_1 = 5$, and $\gamma_2 = 1.5$. The optimal value of λ_T was chosen by the C_p.)

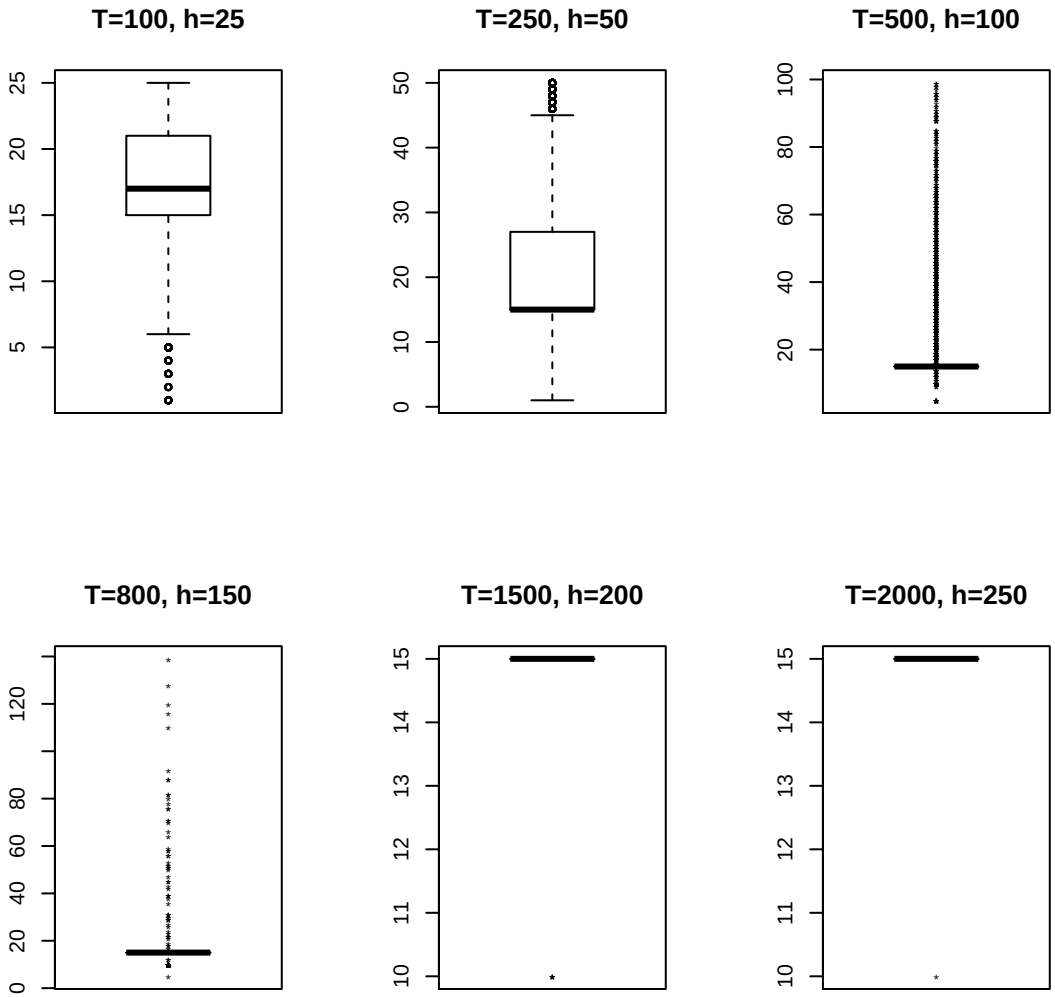
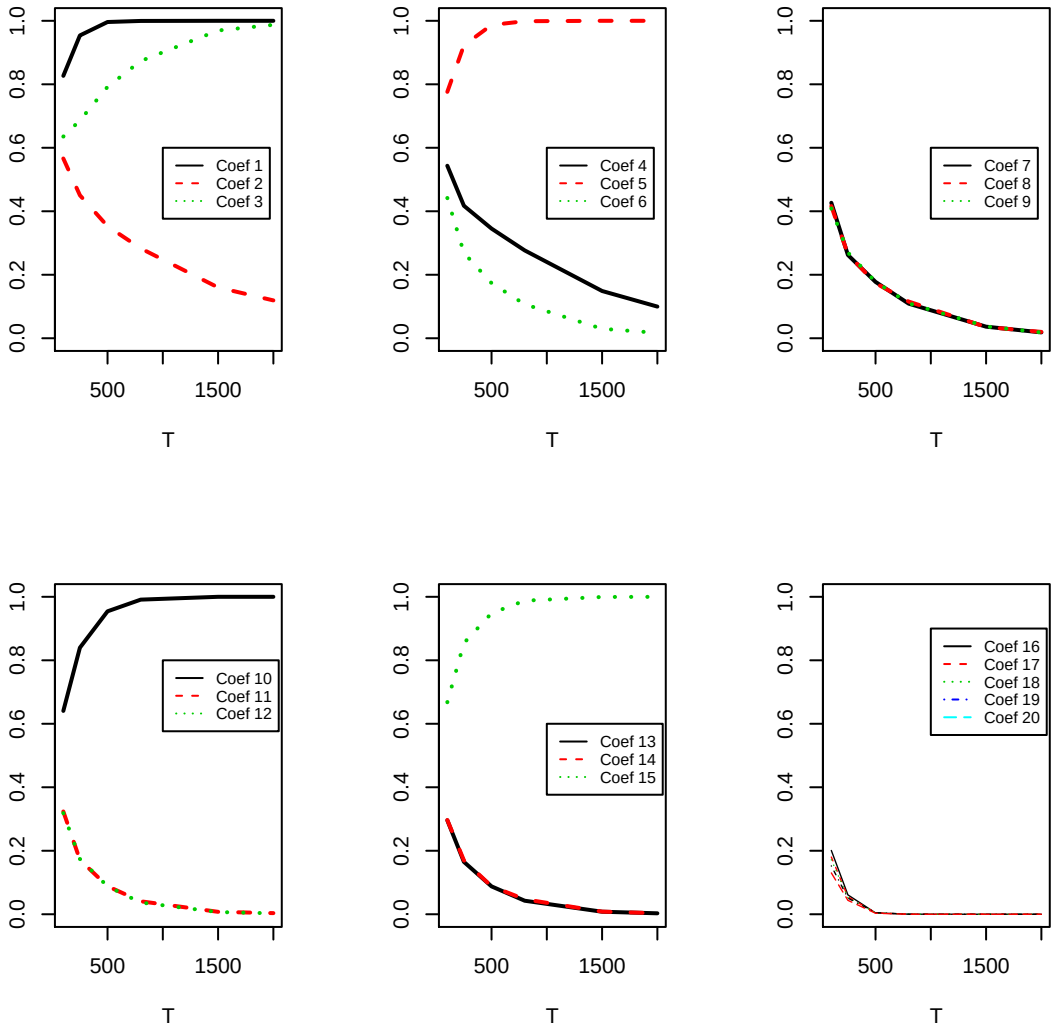


Figure 2.2: Empirical probabilities of AR coefficients being selected in the model by the doubly adaptive LASSO for as sample size increases, based on 10,000 replications (10,000 data sets for each of 6 different sample sizes $T = 100, 250, 500, 800, 1500, 2000$ were generated form $Y_t = 0.2Y_{t-1} + 0.1Y_{t-3} + 0.2Y_{t-5} + 0.2Y_{t-10} + 0.25Y_{t-15} + a_t$. Set $h = 25, 50, 100, 150, 200, 250$ accordingly with respect to the different T. Set $\gamma_0 = 4.5$, $\gamma_1 = 5$, and $\gamma_2 = 1.5$. The optimal value of λ_T was chosen by the C_p.)



Pretending we do not know the true lag order of the underlying model, we set maximum order h = 50. First, we employ the adaptive LASSO (Zou, 2006) to fit an AR models to each of the simulated 1,000 data sets of size 800. As Table 2.3 shows, the adaptive LASSO (Zou, 2006) tends to choose a model with larger AR order.

Table 2.3: Empirical statistics of the adaptive LASSO estimates for the AR order, based on 1,000 replications (1,000 data sets each of size T=800 were generated from $Y_t = 0.2Y_{t-1} + 0.1Y_{t-3} + 0.2Y_{t-5} + 0.2Y_{t-10} + 0.1Y_{t-15} + a_t$ (Nardi, 2011). Set $h = 50, \gamma_0 = \gamma_2 = 0$. The optimal value of γ_1 was chosen by the LOOCV and the optimal value of λ_T chosen by the C_p)

True	Minimum	Maximum	Mean	Median	Mode	SE	Bias	MSE	MAD
15	10	50	38.4	41	49	651.8	10.2	23.4	23.5

We use the double adaptive LASSO to fit an AR model to each of the simulated $1,000$ data sets of size 800. We choose the optimal values for γ_0, γ_1, and γ_2 via the minimum LOOCV criterion, and choose the optimal value for λ_T via the minimum C_p criterion. Table 2.4 shows some empirical statistics of the soubly adaptive LASSO estimates for *AR* order, and Table 2.5 shows some empirical statistics for coefficients estimates, and selection probabilities for the the *AR* coefficients. We highlight a few some observations from Table 2.4 and 2.5.

Table 2.4: Empirical statistics of the doubly adaptive LASSO estimates for the AR order, based on 1,000 replications (1,000 data sets each of size T=800 were generated from $Y_t = 0.2Y_{t-1} + 0.1Y_{t-3} + 0.2Y_{t-5} + 0.2Y_{t-10} + 0.1Y_{t-15} + a_t$ (Nardi, 2011). Set $h = 50$. The optimal values of γ_0, γ_1, and γ_2 were chosen by the LOOCV and the optimal value of λ_T chosen by the C_p)

True	Minimum	Maximum	Mean	Median	Mode	SE	Bias	MSE	MAD
15	10	50	23	16	15	197.0	11.8	7.6	9.7

Observations:

(**1**) Order identification. Table 2.4 shows that the mode of 1,000 AR order estimates is 15, indicating that the doubly adaptive LASSO choose the right AR order most frequently for a sample of moderate size. This is evident also in Table 2.5: The selection probabilities of AR(h) coefficients beyond the true order 15 is very small. The mean and median of 1,000 AR order estimates are 23 and 16, respectively, indicating that the distribution of AR order estimates is

skewed to the left, which is not surprising since the CV criteria tend to select a larger number of variables as it is often observed in practice.

(**2**) Variable selection. Table 2.5 shows that Y_{t-1}, Y_{t-3}, Y_{t-5}, Y_{t-10} are always selected by the doubly adaptive LASSO, which is desirable. Y_{t-10} is selected over 70% of times. It is not always selected largely because its true value is relatively small (0.1). Also, Y_{t-2}, Y_{t-4}, Y_{t-6} through Y_{t-9} are selected with over 40% of times, respectively. This is not desirable but not surprising since the CV criteria tend to select a larger number of variables.

(**3**) Coefficients Estimation. Table 2.5 shows that $\widehat{MSE}(\hat{\phi}_j^{daL}) \simeq 0,\ j = 1, \cdots, \hat{p}$, indicating that the estimation consistency is valid even for the moderate sample size $T = 800$.

2.7 Real data analysis

2.7.1 Chemical process time series

Figure 2.3 shows the data set of Series A in the text by Box et al. (1994). Cleveland (1971) fitted an $AR(1,2,7)$, where the numbers in the brackets denote the indices of AR coefficients. McLeod and Zhang (2005) fitted an $AR(1,2,6,7)$. Setting $h = 30$, using LOOCV to determine the optimal value of γ_0, γ_1, γ_2, and Mallows C_p to determine λ_T, the PAC-weighted adaptive LASSO yield a sparse $AR(1,2,6,7)$ model:

$$\hat{Y}_t = 2.7376 + 0.3616Y_{t-1} + 0.2032Y_{t-2} + 0.1142Y_{t-6} + 0.1605Y_{t-7}$$

2.7.2 Annual tree ring width

Figure 2.4 shows 771 consecutive annual tree ring width measurements on Douglas fir at Nine Mile Canyon, UT, for the years $1194 - 1964$ (McLeod and Hipel, 1995). McLeod and Hipel (1995) fitted an $AR(1,9)$ model. McLeod and Zhang (2005) fitted an $AR(1,2,9)$ model. Our adaptive LASSO yields an $AR(1,2,3,4,7,9,17)$ model:

$$\hat{Y}_t = 39.574 + 0.376Y_{t-1} + 0.102Y_{t-2} - 0.06Y_{t-3} + 0.106Y_{t-4}$$
$$+ 0.059Y_{t-7} + 0.106Y_{t-9} - 0.086Y_{t-17}$$

Table 2.5: Empirical statistics of the doubly adaptive LASSO estimates for the AR coefficients, based on 1,000 replications (1,000 data sets each of size T=800 were generated from $Y_t = 0.2Y_{t-1} + 0.1Y_{t-3} + 0.2Y_{t-5} + 0.2Y_{t-10} + 0.1Y_{t-15} + a_t$ (Nardi, 2011). Set $h = 50$. The optimal values of γ_0, γ_1, and γ_2 were chosen by the LOOCV and the optimal value of λ_T chosen by the C_p)

Lag	TRUE	Minimum	Maximum	Mean	Median	SE	Bias	MSE	MAD	Proportion
1	0.2	0.080	0.323	0.1993	0.1990	0.036	-0.0007	0.0013	0.0288	1
2	0	-0.110	0.108	-0.0009	0	0.033	-0.0009	0.0011	0.0210	0.545
3	0.1	-0.028	0.209	0.0988	0.0998	0.038	-0.0012	0.0014	0.0296	0.974
4	0	-0.154	0.110	-0.0002	0	0.035	-0.0002	0.0012	0.0224	0.532
5	0.2	0.089	0.305	0.2036	0.2027	0.039	0.0036	0.0015	0.0311	1
6	0	-0.103	0.105	-0.0018	0	0.031	-0.0018	0.0010	0.0184	0.429
7	0	-0.108	0.114	-0.0001	0	0.032	-0.0001	0.0010	0.0188	0.427
8	0	-0.139	0.107	-0.0015	0	0.034	-0.0015	0.0011	0.0199	0.437
9	0	-0.105	0.123	0.0014	0	0.031	0.0014	0.0010	0.0179	0.415
10	0.3	0.157	0.409	0.3008	0.3008	0.038	0.0008	0.0014	0.0303	1
11	0	-0.174	0.103	-0.0002	0	0.025	-0.0002	0.0006	0.0088	0.138
12	0	-0.110	0.116	0.0010	0	0.024	0.0010	0.0006	0.0084	0.135
13	0	-0.117	0.111	-0.0011	0	0.023	-0.0011	0.0005	0.0083	0.141
14	0	-0.131	0.128	-0.0012	0	0.026	-0.0012	0.0007	0.0091	0.137
15	0.1	0.000	0.230	0.0815	0.0952	0.056	-0.0185	0.0035	0.0453	0.735
16	0	-0.157	0.122	-0.0006	0	0.022	-0.0006	0.0005	0.0063	0.092
17	0	-0.157	0.104	-0.0012	0	0.020	-0.0012	0.0004	0.0056	0.085
18	0	-0.130	0.128	-0.0002	0	0.024	-0.0002	0.0006	0.0070	0.097
19	0	-0.116	0.129	0.0005	0	0.022	0.0005	0.0005	0.0065	0.097
20	0	-0.123	0.129	0.0000	0	0.023	0.0000	0.0005	0.0070	0.099
21	0	-0.112	0.135	0.0015	0	0.020	0.0015	0.0004	0.0054	0.078
22	0	-0.116	0.110	-0.0015	0	0.020	-0.0015	0.0004	0.0053	0.076
23	0	-0.138	0.113	0.0004	0	0.022	0.0004	0.0005	0.0060	0.083
24	0	-0.130	0.111	0.0008	0	0.018	0.0008	0.0003	0.0044	0.067
25	0	-0.120	0.128	0.0000	0	0.018	0.0000	0.0003	0.0039	0.051
26	0	-0.101	0.104	0.0011	0	0.016	0.0011	0.0003	0.0037	0.056
27	0	-0.119	0.119	0.0008	0	0.018	0.0008	0.0003	0.0041	0.059
28	0	-0.132	0.118	-0.0002	0	0.017	-0.0002	0.0003	0.0037	0.053
29	0	-0.113	0.115	0.0001	0	0.017	0.0001	0.0003	0.0041	0.06
30	0	-0.124	0.105	-0.0020	0	0.018	-0.0020	0.0003	0.0039	0.051
31	0	-0.099	0.125	0.0012	0	0.016	0.0012	0.0003	0.0035	0.049
32	0	-0.114	0.117	-0.0002	0	0.015	-0.0002	0.0002	0.0030	0.042
33	0	-0.111	0.132	0.0005	0	0.017	0.0005	0.0003	0.0034	0.046
34	0	-0.101	0.099	-0.0002	0	0.013	-0.0002	0.0002	0.0023	0.033
35	0	-0.127	0.108	0.0001	0	0.014	0.0001	0.0002	0.0024	0.032
36	0	-0.135	0.124	-0.0009	0	0.015	-0.0009	0.0002	0.0026	0.036
37	0	-0.114	0.095	0.0000	0	0.016	0.0000	0.0003	0.0035	0.048
38	0	-0.112	0.128	0.0000	0	0.014	0.0000	0.0002	0.0025	0.035
39	0	-0.127	0.095	0.0001	0	0.011	0.0001	0.0001	0.0015	0.021
40	0	-0.099	0.094	-0.0007	0	0.012	-0.0007	0.0001	0.0022	0.033
41	0	-0.109	0.102	-0.0005	0	0.011	-0.0005	0.0001	0.0015	0.019
42	0	-0.119	0.105	0.0000	0	0.012	0.0000	0.0001	0.0017	0.022
43	0	-0.097	0.113	-0.0001	0	0.012	-0.0001	0.0002	0.0018	0.023
44	0	-0.137	0.088	-0.0002	0	0.010	-0.0002	0.0001	0.0013	0.018
45	0	-0.125	0.100	-0.0003	0	0.012	-0.0003	0.0001	0.0017	0.02
46	0	-0.110	0.106	-0.0001	0	0.011	-0.0001	0.0001	0.0014	0.016
47	0	-0.090	0.097	0.0000	0	0.008	0.0000	0.0001	0.0008	0.011
48	0	-0.108	0.088	0.0000	0	0.007	0.0000	0.0000	0.0006	0.008
49	0	-0.106	0.114	0.0000	0	0.009	0.0000	0.0001	0.0009	0.01
50	0	-0.100	0.107	-0.0001	0	0.008	-0.0001	0.0001	0.0008	0.009

Figure 2.3: Chemical process time series (Data source: Box et al. 2004)

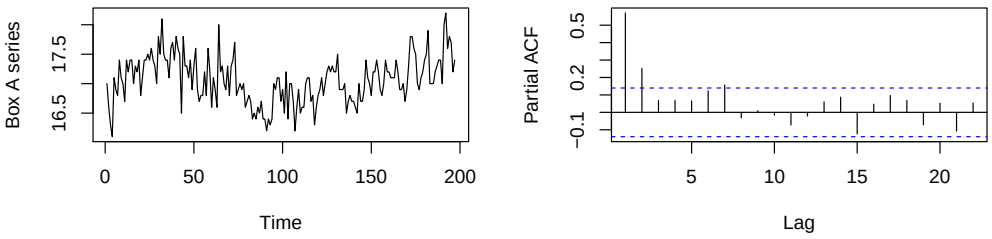
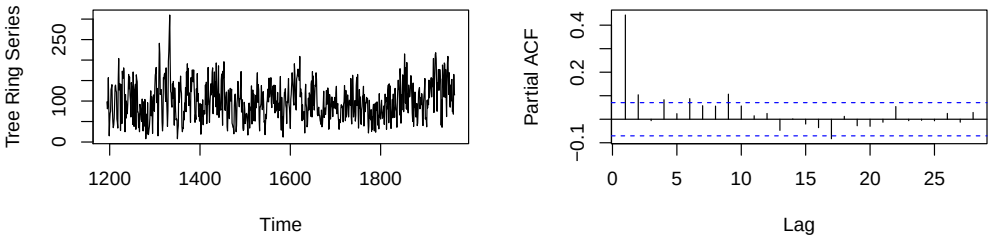


Figure 2.4: Annual tree ring width measurements on Douglas fir (1194-1964) (Data source: McLeod and Hipel, 1995)

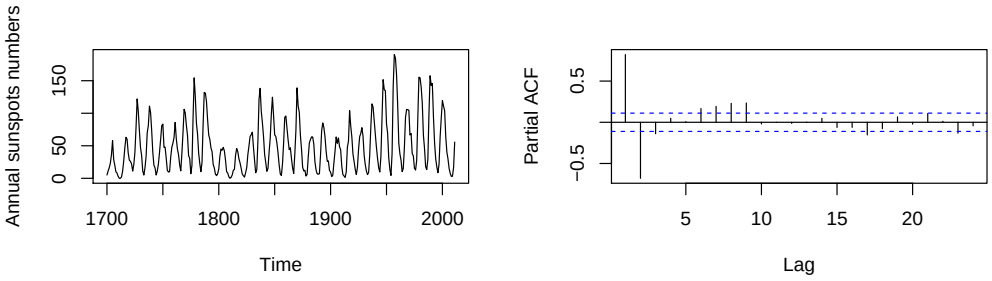


2.7.3 Annual sunspot numbers

Figure 2.5 shows 312 consecutive annual sunspots numbers for the years $1700-2011$ (Solar Influence Data Analysis Center). There exists quite a few $ARMA(p,q)$ models (Woodward and Gray, 1978). McLeod et al. (1977) proposed an $AR(1,2,9)$ model with mean 11.77 for a transformed series $2(\sqrt{y_t+1}-1)$. Our adaptive LASSO yields an $AR(1,2,3,4,5,9)$ model for y_t:

$$\hat{Y}_t = 6.521 + 1.167Y_{t-1} - 0.393Y_{t-2} - 0.172Y_{t-3} + 0.138Y_{t-4} - 0.072Y_{t-5} + 0.2Y_{t-9}$$

Figure 2.5: Annual sunspots numbers (1700-2011) (Data source: SIDC website http://sidc.be/sunspot-data/)



Chapter 3

The Doubly Adaptive Positive LASSO for ARCH(q) Models

3.1 Introduction

Financial time series have some characteristics of empirical statistical regularities dubbed as *stylized facts*. Many empirical studies have documented properties of stylized facts in financial time series data like daily stock returns. Let p_t be the stock price at time t. The continuously compounded return, called log return or simply return, at t is defined as $\epsilon_t = \log(p_t/p_{t-1})$. The return ϵ_t approximately represents relative price increase since $\epsilon_t \approx (p_t - p_{t-1})/p_{t-1}$. It is convenient to use the return to make comparisons between stocks since it is independent of monetary units. Some stylized facts of $\{p_t\}$ and $\{\epsilon_t\}$ that have been amply documented in the financial literature include but are not limited to: (i) *Stationarity*: The price series $\{p_t\}$ is generally close to a random walk without intercept whereas the return series $\{\epsilon_t\}$ is compatible with the second-order stationarity assumption; (ii) *Memory*: the return series $\{\epsilon_t\}$ has weak autocorrelation or short memory whereas the squared return series $\{\epsilon_t^2\}$ or absolute returns series $\{|\epsilon_t|\}$ has strong autocorrelation or long memory; (iii) *Volatility clustering*: The squared return series $\{\epsilon_t^2\}$ or absolute returns series $\{|\epsilon_t|\}$ tend to appear in clusters with some periods being highly volatile and other periods being tranquil; (iv) *Heteroscedasticity*: The volatility of the return series $\{\epsilon_t\}$ is not constant over time; (v) *Leptokurticity*: The return series $\{\epsilon_t\}$ generally has a heavy-tailed distribution.

The autoregressive conditional heteroscedastic (ARCH) model was proposed by Engle (1982) to capture some of these stylized facts. The ARCH model expresses the conditional

variance at time t of the return series $\{\epsilon_t\}$ as a deterministic linear function of the past observations of the squared returns. The dynamic model for conditional variances evolves over time by making use of the most recent information available. The ARCH model is a standard tool for modeling financial volatilities as well as a benchmark model for evaluating other volatility models. The ARCH model is simple and straightforward in algebraic structure yet powerful in interpretation and volatility forecasting.

Due to the long memory property of the squared or absolute returns series, the lag order of the ARCH model needs to be large enough in order for the model to have a good fit to the data and to have a good forecasting capacity. Naturally, for an ARCH model with large lag order, only a subset of ARCH autoregressors are relevant for forecasting financial volatilities. Therefore, we desire a large-order but sparse ARCH model with some of the parameters being null. The sparsity gives rise to the model selection problem. Classical model selection approaches are not only unstable (Breiman, 1996) but also computationally infeasible. Due to its successful applications in AR models, the LASSO may be naturally the first choice for many time series data analysts if they would like to build a sparse ARCH(q) model by shrinking irrelevant ARCH coefficients to zero.

Unfortunately, in the literature we have not found any results that applied the LASSO methodology to modeling ARCH processes. The curse of dimensionality that we would encounter in optimizing the (quasi) maximum likelihood function for large-order ARCH models might be the major reason for the scarcity of examples in the literature. In this chapter, we propose the doubly adaptive positive LASSO, the partial autocorrelation or PAC-weighted adaptive positive LASSO, for modelling the sparse ARCH processes. By applying the doubly adaptive LASSO procedure we may obtain identification, selection and estimation done all in one go.

We review the ARCH models and standard modeling procedure in Section 3.2. We formulate the doubly adaptive positive LASSO tailored to ARCH processes in Section 3.3. In Section 3.4 we study asymptotic properties of the doubly adaptive positive LASSO estimator. Computational details are described in 3.5. Results from numerical experiments are contained

in Section 3.6. Section 3.7 showcases real data analysis examples.

3.2 The pure ARCH(q) process and standard modelling procedure

In this section, we review the basic concepts of the ARCH model and the standard modeling methods including order identification and quasi maximum likelihood estimation.

The pure ARCH(q) process

Let $\{\epsilon_t\}, t = 0, \pm 1, \pm 2 \cdots, \pm\infty$ be a time series and $\mathcal{F}_t$ be the σ-field generated by past $\{\epsilon_t\}$, i.e. $\mathcal{F}_t = \sigma(\epsilon_t, \epsilon_{t-1}, \cdots)$. Suppose that ϵ_t is square-integrable and

$$\epsilon_t = \sigma_t \eta_t \quad \text{with} \quad \eta_t \sim iid(0, 1). \tag{3.1}$$

The time series $\{\epsilon_t\}$ is a martingale difference

$$\mathbb{E}[\epsilon_t | \mathcal{F}_{t-1}] = 0 \ \text{ a.s.},$$

with time-varying conditional variance

$$\mathbb{E}[\epsilon_t^2 | \mathcal{F}_{t-1}] = \sigma_t^2.$$

The pure ARCH(q) specification for $\sigma_t^2, \forall t \in \mathbb{Z}$ (Engle, 1982) is defined as

$$\sigma_t^2 = \alpha_0 + \alpha_1 \epsilon_{t-1}^2 + \cdots + \alpha_q \epsilon_{t-q}^2, \quad \forall t \in \mathbb{Z}, \tag{3.2}$$

where $\eta_t \perp \epsilon_{t-j}$ for $j > 0$, and $\alpha_0 > 0$, $\alpha_j \geq 0$, $j = 1, \cdots, q-1$, and $\alpha_q > 0$. The parameters are restricted to be non-negative to guarantee that the conditional variances are always non-negative. Recall that non-negativity of ARCH coefficients is necessary and sufficient for the conditional variances to be always nonnegative (Engle, 1982; Nelson and Cao 1992; Tsai and Chan 2008).

Identification

The lag order of the pure ARCH(q) model is unknown a priori. In practice, we sequentially fit a variety of candidate models ARCH(1), ARCH(2), up to ARCH(h), where h is an integer with large enough value. We then conduct diagnosis to check if the models are adequate or not. We finally choose from all adequate candidates the most parsimonious model by some criteria such as minimum BIC or AIC.

Alternatively, we may first identify the order of the ARCH(q) model, then estimate the parameters. Define the process $\{\nu_t\}$ of $\epsilon_t^2, \forall t \in \mathbb{Z}$ as

$$\nu_t = \epsilon_t^2 - \sigma_t^2.$$

It is easy to verify that $\mathbb{E}[\nu_t|\mathcal{F}_{t-1}] = 0$, $\text{cov}(\nu_t, \nu_{t-j}) = 0$ and $\text{cov}(\nu_t, \epsilon_{t-j}) = 0$, for $j > 0$. A little bit of manipulation yields

$$\epsilon_t^2 = \alpha_0 + \alpha_1 \epsilon_{t-1}^2 + \cdots + \alpha_q \epsilon_{t-q}^2 + \nu_t, \forall t \in \mathbb{Z},$$

which suggests that the AR lag order of ϵ_t^2 corresponds to the ARCH lag order of ϵ_t. So to identify the order of the ARCH process ϵ_t, we compute the sample partial autocorrelation from a realization of the AR process ϵ_t^2. From the partial correlogram for ϵ_t^2, the AR lag order of ϵ_t^2, or the ARCH lag order of ϵ_t is determined. Shin and Kang (2001) argued that, to a first-order approximation, a power transformation preserves the theoretical autocorrelation function and hence the order of a stationary ARMA process. Their result suggests that the ARCH order may also be identified by studying the absolute returns. Also see Francq and Zakonian (2010 page 109).

The quasi-maximum likelihood estimator

The standard approach is the quasi-maximum likelihood (QML) estimation which minimizes the negative quasi-log likelihood function. Let $(\epsilon_1, \cdots, \epsilon_T)$ be a realization of the ARCH process Given initial observations $\boldsymbol{\epsilon}_0 = (\epsilon_0, \epsilon_{-1}, \cdots, \epsilon_{1-q})$, the conditional Gaussian quasi-likelihood is given by

$$\mathcal{L}_T(\boldsymbol{\theta}) = \mathcal{L}_T(\boldsymbol{\theta}; \epsilon_T, \cdots, \epsilon_1, \boldsymbol{\epsilon}_0) = \prod_{t=1}^{T} \frac{1}{\sqrt{2\pi}\sigma_t} \exp\left(-\frac{\epsilon_t^2}{2\sigma_t^2}\right),$$

and the negative log conditional quasi-likelihood function $L_T(\boldsymbol{\theta})$ is defined as

$$L_T(\boldsymbol{\theta}) = \sum_{t=1}^{T} \left\{ \frac{1}{2} \log\left(\sigma_t{}^2(\boldsymbol{\theta})\right) + \frac{\epsilon_t{}^2}{2\sigma_t{}^2(\boldsymbol{\theta})} + \frac{1}{2}\log(2\pi) \right\},$$

where $\boldsymbol{\theta} = [\alpha_0, \boldsymbol{\alpha}']'$ with $\boldsymbol{\alpha} = [\alpha_1, \cdots, \alpha_q]'$.

The quasi maximum likelihood estimator is defined as any measurable solution of

$$\hat{\boldsymbol{\theta}}_T^{\text{qml}} = \underset{\boldsymbol{\theta}\in\Theta}{\operatorname{argmin}}\, L_T(\boldsymbol{\theta}).$$

Subset selection

Subset selection is the restricted optimization. First we have to know which coefficients are zero. Francq and Zakoïan (2009) proposed the method to test the nullity of the ARCH coefficients. Francq and Zakoïan (2007) also studied the asymptotic distribution of the QML estimator when the true parameter may have zero coefficients. They approximated quasi-likelihood by a quadratic function and project the asymptotic distribution of a normal vector distribution onto a convex cone.

3.3 The adaptive and doubly adaptive positive LASSO

In this section, we adapt the LASSO methodology to modeling the ARCH process. There are two situations. If the order is known in advance or has been identified already, we recommend the adaptive positive LASSO. If the order is not known in advance or difficult to identify, we propose the doubly adaptive positive LASSO, or PAC-weighted adaptive positive LASSO. We use the word *positive* following Efron et. al (2004) since the coefficients of ARCH(q) models are restricted to be nonnegative.

3.3.1 The doubly adaptive positive LASSO when q is unknown

Suppose that we have the data $\epsilon_1, \epsilon_2, \cdots, \epsilon_T$, which is a realization of the ARCH(q) process defined by (3.1) and (3.2) with the true order q and true parameters $\boldsymbol{\alpha}^o = (\alpha_0^o, \alpha_1^o, \cdots, \alpha_q^o)$ both unknown. We first set our guess of the ARCH order to be h, which has a sufficiently large

positive integer [1] so that $h > q$. Since the initial values $\epsilon_0, \cdots, \epsilon_{-h+1}$ are not available, we use $\epsilon_1, \cdots, \epsilon_h$ as a presample, hence the effective sample size is $T - h$. Now, having the data, we formulate the negative log conditional quasi-likelihood function $L_T(\boldsymbol{\theta})$ as

$$L_T(\boldsymbol{\theta}) = \sum_{t=h+1}^{T} \ell_t(\boldsymbol{\theta}), \tag{3.3}$$

where

$$\ell_t(\boldsymbol{\theta}) = \frac{1}{2}\log\left(\sigma_t{}^2(\boldsymbol{\theta})\right) + \frac{\epsilon_t{}^2}{2\sigma_t{}^2(\boldsymbol{\theta})} + \frac{1}{2}\log(2\pi), \tag{3.4}$$

and

$$\sigma_t^2(\boldsymbol{\theta}) = \alpha_0 + \alpha_1\epsilon_{t-1}^2 + \cdots + \alpha_h\epsilon_{t-h}^2, \tag{3.5}$$

for $t = h+1, \cdots, T$ with $\boldsymbol{\theta} = [\alpha_0, \boldsymbol{\alpha}']'$ and $\boldsymbol{\alpha} = [\alpha_1, \cdots, \alpha_h]'$.

Definition (**The doubly adaptive positive LASSO**). The doubly adaptive positive LASSO estimator or PAC-weighted adaptive positive LASSO, $\hat{\boldsymbol{\theta}}_T^{\text{dapL}}$, is the penalized conditional quasi-maximum likelihood estimators defined as

$$\hat{\boldsymbol{\theta}}_T^{\text{dapL}} = \underset{\boldsymbol{\theta}\in\Theta}{\operatorname{argmin}} \left\{ L_T(\boldsymbol{\theta}) + \lambda_T \left(\hat{w}_{T,0}\alpha_0 + \sum_{j=1}^{h} \hat{w}_{T,j}\alpha_j \right) \right\}, \tag{3.6}$$

where $\alpha_0 > 0, \alpha_j \geq 0$,

$$\hat{w}_{T,0} = \begin{cases} 0 & \text{if intercept not to be penalized} \\ \frac{1}{\tilde{\alpha}_0^{\gamma_1}\left(\sum_{i=0}^{h}|\hat{\rho}_{ii}|^{\gamma_0}\right)^{\gamma_2}} & \text{if intercept to be penalized} \end{cases} \tag{3.7}$$

$$\hat{w}_{T,j} = \frac{1}{\tilde{\alpha}_j^{\gamma_1}\left(\sum_{i=j}^{h}|\hat{\rho}_{ii}|^{\gamma_0}\right)^{\gamma_2}} = \frac{1}{\tilde{\alpha}_j^{\gamma_1} A_j^{\gamma_2}}, \tag{3.8}$$

$$A_j = \sum_{i=j}^{h} |\hat{\rho}_{ii}|^{\gamma_0}, \tag{3.9}$$

for $j = 1, \cdots, h$, $\tilde{\theta}_j$ is any consistent estimate, for example, $\hat{\theta}_j^{\text{qml}}$, $\hat{\rho}_{ii}$ is the estimate for the ith-lag partial autocorrelation of $\{\epsilon_t^2\}_{t=1}^T$, and $\gamma_0 > 0$, $\gamma_1 \geq 0$, and $\gamma_2 \geq 0$ are some fixed constants.

Remark 1: Both the LASSO (Tibshirani, 1996) and the adaptive LASSO (Zou, 2006) are special cases of the doubly adaptive LASSO. When $\gamma_1 = \gamma_2 = 0$, then $w_j = 1$, and the doubly

[1] h is set to be quite large, for instance, $h = \kappa T^{\alpha}$, $0 \leq \alpha \leq 1$ for some constant κ.

adaptive LASSO reduces to the LASSO. When $\gamma_2 = 0$, then $w_j = \hat{\theta}_j^{-\gamma_1}$, and the doubly adaptive LASSO reduces to the adaptive LASSO.

Remark 2: In the ARCH(q) model, the intercept is required to be strictly positive, so we recommend not to penalize the intercept. However, we may have some data that lead us to fit a model with unduly large intercept and unduly small coefficients. In this situation, it might be better for us to penalize the intercept also.

Remark 3: In the doubly adaptive LASSO procedure the partial autocorrelation information and the quasi-maximum likelihood estimates for the ARCH model work in tandem to perform subset selection and parameter estimation simultaneously. The basic idea can be elucidated from the following points:

Firstly, the monotonically decreasing (with respect to j) A_j's impose monotonically increasing penalty on θ_j as j goes from 1 to h. Hence $w_{T,i} < w_{T,k}$ for lag values satisfying $i < k$. Also, because A_j is a function of the sample PAC, the serial correlations embedded in the data are factored into the adaptive positive LASSO procedure. As a consequence, depending on the structure of serial correlations, an ARCH term with smaller lag is more likely to be included in the model.

Secondly, a big bump of $\{A_j\}_{j=1}^h$ at $j = q$ relative to $j > q$ provides the cutoff lag corresponding to the true order of the ARCH process, since $|\hat{\rho}_{ii}| = O_P(1/\sqrt{T})$ for $i = q+1, q+2 \cdots, h$. This means that the A_j's for $j > q$ are relatively very small. If j goes from h backwards to q, it is expected that the $\{A_j\}_{j=1}^h$ will exhibit a sharp jump at $j = q$. Consequently, the ARCH terms with lags greater than q get so much penalties that they will be excluded from the model, and the true order of the ARCH process is thus identified.

Finally, $|\tilde{\theta}_j|^{\gamma_1}$ imposes a larger penalty on θ_j if the corresponding ARCH term is not significant, and smaller penalty on θ_j if the corresponding ARCH term is significant. This is obvious because for an ARCH term ϵ_{t-j}^2 that is not significant, the value of $\tilde{\theta}_j$ is close to zero, $|\tilde{\theta}_j|^{-\gamma_1}$ is close to ∞. Consequently, the insignificant ARCH terms get so much penalties that they will be excluded from the model whereas the significant ARCH terms will be included in the model.

Remark 4: Actually, we use the doubly adaptive LASSO to estimate the extended true parameter vector, $\boldsymbol{\theta}^*$, defined as

$$\boldsymbol{\theta}^* = (\alpha_0^*, \alpha_1^*, \cdots, \alpha_q^*, \alpha_{q+1}^*, \cdots, \alpha_h^*)' = (\alpha_0^o, \alpha_1^o, \cdots, \alpha_q^o, 0, \cdots, 0)' \tag{3.10}$$

It is clear that the ARCH(q) process with the fixed parameters $\boldsymbol{\alpha}^o = (\alpha_0^o, \alpha_1^o, \cdots, \alpha_q^o)$ and the ARCH(h) processes with the fixed parameters $\boldsymbol{\theta}^*$ are equivalent.

3.3.2 The adaptive positive LASSO when q is known

Suppose that we have the data $\epsilon_1, \epsilon_2, \cdots, \epsilon_T$, which is a realization of the ARCH(q) process defined by (3.1) and (3.2) with the true order q known and true parameters $\boldsymbol{\alpha}^o = (\alpha_0^o, \alpha_1^o, \cdots, \alpha_q^o)$ unknown. Since the initial values $\epsilon_0, \cdots, \epsilon_{-q+1}$ are not available, we use $\epsilon_1, \cdots, \epsilon_q$ as a presample, hence the effective sample size is $T - q$. We set $h = q$ and $\gamma_2 = 0$ in (3.7) and (3.8). The doubly adaptive LASSO reduces to the adaptive LASSO.

3.4 Asymptotic properties of the doubly adaptive positive LASSO

The adaptive positive LASSO and the doubly adaptive positive LASSO methods yield biased estimators. In this section, however, we show that with properly chosen values for weighting parameters γ_0, γ_1, and γ_2 in (2.13) and tuning parameter λ_T, the doubly adaptive positive LASSO enjoys desirable asymptotic properties. Let q be the true unknown order of the ARCH model. Let $\boldsymbol{\theta}^o = (\theta_1^o, \cdots, \theta_q^o)'$, where $\theta_j^o = 0$ for some $j < p$ and $\theta_q^o \neq 0$, be the true unknown parameters of the ARCH(q) model. We actually study the asymptotic properties of the doubly adaptive LASSO estimator for $\boldsymbol{\theta}^*$, the extended true parameter vector defined by (3.10).

First, we clarify notations. Let $\mathbb{S}$ be the set of the true nonzero coefficient, i.e. $\mathbb{S} = \{j : \theta_j^* \neq 0\} = \text{supp}(\boldsymbol{\theta}^*) \subset \{1, 2, \cdots, h\}$ with h being set large enough such that $h > q$. Let $\mathbb{S}^c = \{1, 2, \cdots, h\} \setminus \mathbb{S}$. Let $s = |\mathbb{S}|$ be the cardinality of the set $\mathbb{S}$. The assumption of the model sparsity implies that $s < q$. Let $\tilde{\theta}_j$ be any consistent estimate for the true θ_j^*, say the QML estimate. Let $\hat{\theta}_{T,j}^{dapL}$ be the doubly adaptive positive LASSO estimate for θ_j^*. Let $\hat{\mathbb{S}}_T = \{j : \hat{\theta}_{T,j}^{dapL} \neq 0\}$ and $\hat{\mathbb{S}}_T^c = \{1, 2, \cdots, h\} \setminus \hat{\mathbb{S}}_T$. Let $\boldsymbol{\theta}_{\mathbb{S}}^*$ be the s-dimensional vector for true underlying nonzero parameters,

and $\boldsymbol{\theta}^*_{\mathbb{S}^c}$ be the vector for true underlying null parameters, i.e. $\boldsymbol{\theta}^*_{\mathbb{S}} = \{\theta^*_j : j \in \mathbb{S}\}$ and $\boldsymbol{\theta}^*_{\mathbb{S}^c} = \{\theta^*_j : j \in \mathbb{S}^c\}$. Let $\hat{\boldsymbol{\theta}}^{dapL}_{T,\mathbb{S}}$ be the vector for the PAC-weighted adaptive positive LASSO estimate for $\boldsymbol{\theta}^*_{\mathbb{S}}$ and $\hat{\boldsymbol{\theta}}^{dapL}_{T,\mathbb{S}^c}$ the vector for the PAC-weighted adaptive positive LASSO estimate for null vector $\boldsymbol{\theta}^*_{\mathbb{S}^c}$, i.e. $\hat{\boldsymbol{\theta}}^{daL}_{T,\mathbb{S}} = \{\hat{\theta}^{daL}_{T,j} : j \in \mathbb{S}\}$ and $\hat{\boldsymbol{\theta}}^{daL}_{T,\mathbb{S}^c} = \{\hat{\theta}^{daL}_{T,j} : j \in \mathbb{S}^c\}$. Let $\hat{\boldsymbol{\theta}}^{dapL}_{\hat{\mathbb{S}}_T}$ be the vector for nonzero estimates from the doubly adaptive positive LASSO and $\hat{\boldsymbol{\theta}}^{dapL}_{\hat{\mathbb{S}}^c_T}$ the vector for null estimates, i.e. $\hat{\boldsymbol{\theta}}^{dapL}_{\hat{\mathbb{S}}_T} = \{\hat{\theta}^{dapL}_{T,j} : j \in \hat{\mathbb{S}}_T\}$ and $\hat{\boldsymbol{\theta}}^{dapL}_{\hat{\mathbb{S}}^c_T} = \{\hat{\theta}^{dapL}_{T,j} : j \in \hat{\mathbb{S}}^c_T\}$.

Theorem 3.4.1 *(**Second-order stationarity** (Bollerslev, 1986)). The necessary and sufficient condition for the second-order stationarity of the pure ARCH(q) process defined by (3.1) and (3.2) is that* $\sum_{i=1}^{q} \alpha_i < 1$.

Let B_t be a random matrix defined as

$$B_t = \begin{pmatrix} \alpha_1\eta_t^2 & \alpha_2\eta_t^2 & \cdots & \alpha_{h-1}\eta_t^2 & \alpha_h\eta_t^2 \\ 1 & 0 & \cdots & 0 & 0 \\ \vdots & \vdots & & \vdots & \vdots \\ 0 & 0 & \cdots & 1 & 0 \end{pmatrix}. \tag{3.11}$$

Definition (**The top Lyapunov exponent**). Let $\boldsymbol{B}_t$ be the sequence of random matrices $\{B_t\}$ with B_t defined as (3.11).The top Lyapunov exponent is defined as

$$\gamma(\boldsymbol{B}_t) \equiv \inf_{t\in\mathbb{N}^*} \frac{1}{t} E\left(\log\|B_t \cdots B_1\|\right) \overset{a.s.}{=} \lim_{t\to\infty} \frac{1}{t} \log\|B_t B_{t-1} \cdots B_1\|. \tag{3.12}$$

Theorem 3.4.2 *(**Stationarity and ergodicity** (Bougerol and Picard, 1992)). The necessary and sufficient condition for the strict stationarity and ergodicity of the pure ARCH(q) process defined by (3.1) and (3.2) is that the top Lyapunov exponent is strictly negative , i.e.* $\gamma(\boldsymbol{B}_0) < 0$.

Let $B^{\otimes m} = B \otimes B \otimes \cdots B$ with m factors, where $\otimes$ denote the tensor product, or Kronecker product.

Theorem 3.4.3 *(**Even-order moments** (Ling and McAleer, 2002)). The necessary and sufficient condition for* $E\left[\epsilon_t^{2m}\right] < \infty$, *where* ϵ_t *is the pure ARCH(q) process defined by (3.1) and (3.2), is that* $\rho\left(E\left[B_0^{\otimes m}\right]\right) < 0$, *where* ρ *denotes the spectral radius of a matrix.*

Let

$$\boldsymbol{J} := E_{\boldsymbol{\theta}^*}\left[\frac{\partial^2 \ell_t(\boldsymbol{\theta}^*)}{\partial\boldsymbol{\theta}\partial\boldsymbol{\theta}'}\right] = E_{\boldsymbol{\theta}^*}\left[\frac{1}{\sigma_t^4(\boldsymbol{\theta}^*)} \frac{\partial\sigma_t^2(\boldsymbol{\theta}^*)}{\partial^2\boldsymbol{\theta}} \frac{\partial\sigma_t^2(\boldsymbol{\theta}^*)}{\partial^2\boldsymbol{\theta}'}\right]. \tag{3.13}$$

We can partition $\boldsymbol{J}$ as follows

$$\boldsymbol{J} = \begin{pmatrix} \boldsymbol{J}_{\mathbb{S}\mathbb{S}} & \boldsymbol{J}_{\mathbb{S}\mathbb{S}^c} \\ \boldsymbol{J}_{\mathbb{S}^c\mathbb{S}} & \boldsymbol{J}_{\mathbb{S}^c\mathbb{S}^c} \end{pmatrix},$$

where we retain the ordering according to the lag index of $\boldsymbol{\epsilon}_t$ within each partition.

Assumptions:

A1: $\boldsymbol{\theta}^* \in (0,1) \times [0,1)^{q-1} \times (0,1) \times [0,1)^{h-q} \subset \Theta$ and Θ is a compact set;

A2: η_t has a nondegenerate distribution with $E[\eta_t] = 0$ and $E[\eta_t^2] = 1$;

A3: $\kappa_\eta = E[\eta_t^4] < \infty$;

A4: $\gamma(\boldsymbol{B}_0) < 0$;

A5: $\rho\left(E\left[B_0^{\otimes 3}\right]\right) < 0$;

Remarks on assumptions:

1) Compactness in **A1** is always assumed.

2) Some of the parameters in the ARCH(h) model are on the boundary. When we talk about derivatives with respect to parameters on the boundary, i.e. $\boldsymbol{\theta}^*_{\mathbb{S}^c}$, we always mean the right derivatives.

3) **A4** ensures that $\{\epsilon_t\}$ is ergodic stationary.

4) **A5** ensures the existence of sixth moments of $\{\epsilon_t\}$.

Lemma 3.4.4 *Under **A1** – **A5**, we have*

(i) $E_{\boldsymbol{\theta}^*}\left\|\frac{\partial \ell_t(\boldsymbol{\theta}^*)}{\partial \boldsymbol{\theta}} \frac{\partial \ell_t(\boldsymbol{\theta}^*)}{\partial \boldsymbol{\theta}'}\right\| < \infty$;

(ii) $E_{\boldsymbol{\theta}^*}\left\|\frac{\partial^2 \ell_t(\boldsymbol{\theta}^*)}{\partial \boldsymbol{\theta} \partial \boldsymbol{\theta}'}\right\| < \infty$;

(iii) There exists a neighbourhood $\Upsilon(\boldsymbol{\theta}^*)$ of $\boldsymbol{\theta}^*$ such that

$$E_{\boldsymbol{\theta}^*} \sup_{\boldsymbol{\theta} \in \Upsilon(\boldsymbol{\theta}^*)} \left| \frac{\partial^3 \ell_t(\boldsymbol{\theta})}{\partial \theta_i \partial \theta_j \partial \theta_k} \right| < \infty.$$

Lemma 3.4.4 can be proved using the arguments similar to Francq and Zakoian (2010, p.159 - 168).

Lemma 3.4.5 *Under **A1 – A5**, the matrix $\boldsymbol{J}_{\mathbb{SS}}$ is positive definite and invertible.*

The submatrix $\boldsymbol{J}_{\mathbb{SS}}$ corresponds to the parameters in the interior of parameter space, i.e. $\boldsymbol{\theta}^*_{\mathbb{S}}$. So Lemma 3.4.5 can be proved using the arguments similar to Francq and Zakoian (2010, p.159 - 168).

Lemma 3.4.6 *Under **A1 – A5**, we have*

(i) $\frac{1}{\sqrt{T-h}} \sum_{t=h+1}^{T} \left(\frac{\partial \ell_t(\boldsymbol{\theta}^*_{\mathbb{S}})}{\partial \boldsymbol{\theta}'_{\mathbb{S}}} \right) \xrightarrow{D} N\left(\mathbf{0},\ (\kappa_\eta - 1)\boldsymbol{J}_{\mathbb{SS}}\right)$;

(ii) $\frac{1}{T-h} \sum_{t=h+1}^{T} \frac{\partial^2 \ell_t(\boldsymbol{\theta}^*)}{\partial \boldsymbol{\theta} \partial \boldsymbol{\theta}'} \xrightarrow{P} \boldsymbol{J}$.

Lemma 3.4.5 can be proved using the arguments similar to Francq and Zakoian (2010, p.159 - 168).

Francq and Zakoïan (2007) studied the asymptotic distribution of the QML estimator when the true parameter may have zero coefficients using their projection method. It is interesting enough to see that their results bear similarities to the results from the doubly adaptive LASSO.

Definition (**Estimation consistency**). The PAC-weighted adaptive positive LASSO estimator $\hat{\boldsymbol{\theta}}_T^{dapL}$ is said to be estimation consistent if $\|\hat{\boldsymbol{\theta}}_T^{dapL} - \boldsymbol{\theta}^*\| \xrightarrow{P} 0$ as $T \to \infty$.

Theorem 3.4.7 *(**Estimation Consistency of $\hat{\boldsymbol{\theta}}_T^{dapL}$**). Let $a_T = \sqrt{T-h} \min_{j \in \mathbb{S}} \left(|\tilde{\theta}_j|^{\gamma_1} A_j^{\gamma_2} \right)$. If $\lambda_T = o_p(a_T)$, then under **A1 – A5**, we have*

$$\|\hat{\boldsymbol{\theta}}_T^{dapL} - \boldsymbol{\theta}^*\| = O_p\left((T-h)^{-1/2}\right) \ \ as \ \ T \to \infty.$$

Proof Let $\Psi_T(\boldsymbol{\theta})$ be defined as

$$\Psi_T(\boldsymbol{\theta}) = \sum_{t=h+1}^{T} \ell_t(\boldsymbol{\theta}) + \lambda_T \sum_{j=1}^{h} \hat{w}_{T,j} |\theta_j|.$$

Following Fan and Li (2001), we show that for every $\epsilon > 0$ there exists a sufficiently large C such that

$$\mathbb{P}\left(\inf_{\|\boldsymbol{u}\| \geq C} \Psi_T\left(\boldsymbol{\theta}^* + \boldsymbol{u}/\sqrt{T-h}\right) > \Psi_T(\boldsymbol{\theta}^*) \right) > 1 - \epsilon,$$

which implies that with probability at least $1-\epsilon$ that there exists a local minimum in the ball $\{\boldsymbol{\theta}^* + \boldsymbol{u}/\sqrt{T-h} : \|\boldsymbol{u}\| \leq C\}$. Hence there exists a local minimizer such that $\| \hat{\boldsymbol{\theta}}_T^{dapL} - \boldsymbol{\theta}^* \| = O_p(T^{-1/2})$. Observe that

$$\begin{aligned}\Psi_T\left(\boldsymbol{\theta}^* + \boldsymbol{u}/\sqrt{T-h}\right) - \Psi_T(\boldsymbol{\theta}^*) &= \sum_{t=h+1}^{T} \ell_t\left(\boldsymbol{\theta}^* + \boldsymbol{u}/\sqrt{T-h}\right) - \sum_{t=h+1}^{T} \ell_t(\boldsymbol{\theta}^*) + \lambda_T \sum_{j=1}^{h} \hat{w}_{T,j}\left(\left|\theta_j^* + \frac{u_j}{\sqrt{T-h}}\right| - \left|\theta_j^*\right|\right) \\ &= A_{T,1} + A_{T,2} + A_{T,3} + A_{T,4},\end{aligned}$$

where

$$A_{T,1} = \frac{1}{2}\boldsymbol{u}'\left(\frac{1}{T-h}\sum_{t=h+1}^{T} \frac{\partial^2 \ell_t(\boldsymbol{\theta}^*)}{\partial\boldsymbol{\theta}\partial\boldsymbol{\theta}'}\right)\boldsymbol{u},$$

$$A_{T,2} = \frac{1}{\sqrt{T-h}}\sum_{t=h+1}^{T} \boldsymbol{u}'\left(\frac{\partial \ell_t(\boldsymbol{\theta}^*)}{\partial\boldsymbol{\theta}'}\right),$$

$$A_{T,3} = \frac{1}{6\sqrt{T-h}}\sum_{t=h+1}^{T}\frac{1}{T-h}\sum_{i=1}^{h}\sum_{j=1}^{h}\sum_{k=1}^{h} \frac{\partial^3 \ell_t(\boldsymbol{\theta})}{\partial\theta_i\partial\theta_j\partial\theta_k} u_i u_j u_k,$$

$$A_{T,4} = \lambda_T \sum_{j=1}^{h} \hat{w}_{T,j}\left\{\left|\theta_j^* + \frac{u_j}{\sqrt{T-h}}\right| - \theta_j^*\right\}.$$

For $A_{T,4}$, observe that

$$\begin{aligned}A_{T,4} &= \lambda_T \sum_{j\in\mathbb{S}} \hat{w}_{T,j}\left(\left|\theta_j^* + \frac{u_j}{\sqrt{T-h}}\right| - \left|\theta_j^*\right|\right) + \lambda_T \sum_{j\notin\mathbb{S}} \hat{w}_{T,j}\frac{|u_j|}{\sqrt{T-h}} \\ &\geq \lambda_T \sum_{j\in\mathbb{S}} \hat{w}_{T,j}\left(\left|\theta_j^* + \frac{u_j}{\sqrt{T-h}}\right| - \left|\theta_j^*\right|\right) \\ &\geq -\lambda_T \sum_{j\in\mathbb{S}} \hat{w}_{T,j}\frac{|u_j|}{\sqrt{T-h}},\end{aligned}$$

and

$$\begin{aligned}\lambda_T \sum_{j=1}^{h} \hat{w}_{T,j}\frac{|u_j|}{\sqrt{T-h}} &= \lambda_T \sum_{j\in\mathbb{S}} \left|\tilde{\theta}_j\right|^{-\gamma_1} A_j^{-\gamma_2}\frac{|u_j|}{\sqrt{T-h}} \\ &\leq \frac{\lambda_T}{\sqrt{T-h}}\left(\min_{j\in\mathbb{S}}\left(|\tilde{\theta}_j|^{\gamma_1} A_j^{\gamma_2}\right)\right)^{-1}\|\boldsymbol{u}\| \\ &= \frac{\lambda_T}{a_T}\|\boldsymbol{u}\| = o_p(1)\|\boldsymbol{u}\|,\end{aligned}$$

so that $A_{T,4} > -o_p(1)\|\boldsymbol{u}\|$. For $A_{T,3}$, by virtue of Lemma 3.4.4(iii), we have

$$\frac{1}{T-h}\sum_{t=h+1}^{T}\sum_{i=1}^{h}\sum_{j=1}^{h}\sum_{k=1}^{h} \frac{\partial^3 \ell_t(\boldsymbol{\theta})}{\partial\theta_i\partial\theta_j\partial\theta_k} u_i u_j u_k \xrightarrow{P} E\left[M(\epsilon_t^2)|\boldsymbol{u}|^3\right] < \infty.$$

Thus, $A_{T,3} \xrightarrow{P} 0$. For $A_{T,2}$, in light of Lemma 3.4.6 (i), we have $A_{T,2} \xrightarrow{D} \boldsymbol{u}'\boldsymbol{w} = \boldsymbol{u}'N(\boldsymbol{0},\ (\kappa_\eta - 1)\boldsymbol{J})$, hence $A_{T,2} = \boldsymbol{u}' o_p(\boldsymbol{1}) > -o_p(1)\|\boldsymbol{u}\|$. For $A_{T,1}$, in light of Lemma 3.4.6 (ii), we have$A_{T,1} \xrightarrow{P} \frac{1}{2}\boldsymbol{u}'\boldsymbol{J}\boldsymbol{u}$.

It follows that in probability

$$\Psi_T\left(\boldsymbol{\theta}^* + \boldsymbol{u}/\sqrt{T-h}\right) - \Psi_T(\boldsymbol{\theta}^*) \geq \frac{1}{2}\boldsymbol{u}'\boldsymbol{J}\boldsymbol{u} - 2o_p(1)\|\boldsymbol{u}\|,$$

as $T \to \infty$. The first term $\frac{1}{2}\boldsymbol{u}'\boldsymbol{J}\boldsymbol{u}$ is a quadratic form in $\boldsymbol{u}$. For any $\epsilon > 0$, there exists a sufficiently large C such that the term of quadratic term dominates the other terms with probability $\geq 1-\epsilon$. ∎

Proposition 3.4.8 *Let* $a_T = \sqrt{T-h}\min_{j\in\mathbb{S}}\left(|\tilde{\theta}_j|^{\gamma_1}A_j^{\gamma_2}\right)$, *and* $b_T = \sqrt{T-h}\max_{j\in\mathbb{S}^c}\left(|\tilde{\theta}_j|^{\gamma_1}A_j^{\gamma_2}\right)$. *If* $\lambda_T = o_p(a_T)$ *and* $\lambda_T/b_T \xrightarrow{P} \infty$, *then under* ***A1*** $-$ ***A5***, *we have*

$$\begin{cases} \sqrt{T-h}\left(\hat{\boldsymbol{\theta}}^{dapL}_{T,\mathbb{S}} - \boldsymbol{\theta}^*_{\mathbb{S}}\right) \xrightarrow{D} N\left(\mathbf{0},\ (\kappa_\eta - 1)(\boldsymbol{J}_{\mathbb{S}\mathbb{S}})^{-1}\right) \\ \sqrt{T-h}\left(\hat{\boldsymbol{\theta}}^{dapL}_{T,\mathbb{S}^c} - \boldsymbol{\theta}^*_{\mathbb{S}^c}\right) \xrightarrow{D} \mathbf{0} \end{cases}.$$

Proof We follow the methodology of Knight and Fu (2000) and Zou (2006).

Let $\boldsymbol{\theta} = \boldsymbol{\theta}^* + \boldsymbol{u}/\sqrt{T-h}$ and define

$$\Psi_T(\boldsymbol{u}) = L\left(\boldsymbol{\theta}^* + \frac{\boldsymbol{u}}{\sqrt{T-h}}\right) + \lambda_T \sum_{j=1}^{h} \hat{w}_{T,j}\left|\theta_j^* + \frac{u_j}{\sqrt{T-h}}\right|.$$

Let $V_T(\boldsymbol{u}) = \Psi_T(\boldsymbol{u}) - \Psi_T(\mathbf{0})$. Then the minimizing objective is equivalent to minimizing $V_T(\boldsymbol{u})$ with respect to $\boldsymbol{u}$. Let $\hat{\boldsymbol{u}}_T = \arg\min \Psi_T(\boldsymbol{u})$, then

$$\hat{\boldsymbol{\theta}}^{dapL}_T = \boldsymbol{\theta}^* + \hat{\boldsymbol{u}}_T/\sqrt{T-h},$$

or

$$\hat{\boldsymbol{u}}_T = \sqrt{T-h}\left(\hat{\boldsymbol{\theta}}^{dapL}_T - \boldsymbol{\theta}^*\right).$$

Observe that

$$\begin{aligned} V_T(\boldsymbol{u}) &= \sum_{t=h+1}^{T}\left\{\ell_t\left(\boldsymbol{\theta}^* + \frac{\boldsymbol{u}}{\sqrt{T-h}}\right) - \ell_t(\boldsymbol{\theta}^*)\right\} + \lambda_T \sum_{j=1}^{h}\hat{w}_{T,j}\left\{\left|\theta_j^* + \frac{u_j}{\sqrt{T-h}}\right| - \theta_j^*\right\} \\ &= A_{T,1} + A_{T,2} + A_{T,3} + A_{T,4}, \end{aligned}$$

where

$$A_{T,1} = \frac{1}{2}\boldsymbol{u}'\left(\frac{1}{T-h}\sum_{t=h+1}^{T}\frac{\partial^2\ell_t(\boldsymbol{\theta}^*)}{\partial\boldsymbol{\theta}\partial\boldsymbol{\theta}'}\right)\boldsymbol{u},$$

$$A_{T,2} = \frac{1}{\sqrt{T-h}}\sum_{t=h+1}^{T}\boldsymbol{u}'\left(\frac{\partial\ell_t(\boldsymbol{\theta}^*)}{\partial\boldsymbol{\theta}'}\right),$$

$$A_{T,3} = \frac{1}{6\sqrt{T-h}} \sum_{t=h+1}^{T} \frac{1}{T-h} \sum_{i=1}^{h}\sum_{j=1}^{h}\sum_{k=1}^{h} \frac{\partial^3 \ell_t(\boldsymbol{\theta})}{\partial\theta_i \partial\theta_j \partial\theta_k} u_i u_j u_k,$$

$$A_{T,4} = \lambda_T \sum_{j=1}^{h} \hat{w}_{T,j} \left\{ \left| \theta_j^* + \frac{u_j}{\sqrt{T-h}} \right| - \theta_j^* \right\}.$$

In light of Lemma 3.4.6, we have$A_{T,1} \xrightarrow{P} \frac{1}{2}\boldsymbol{u}'\boldsymbol{J}\boldsymbol{u}$, and $A_{T,2} \xrightarrow{D} \boldsymbol{u}'\boldsymbol{w} = \boldsymbol{u}' N(\boldsymbol{0}, \ (\kappa_\eta - 1)\boldsymbol{J})$. By virtue of Lemma 3.4.4(iii), we have

$$\frac{1}{T-h} \sum_{t=h+1}^{T} \sum_{i=1}^{h}\sum_{j=1}^{h}\sum_{k=1}^{h} \frac{\partial^3 \ell_t(\boldsymbol{\theta})}{\partial\theta_i \partial\theta_j \partial\theta_k} u_i u_j u_k \xrightarrow{P} E\left[M(\epsilon_t^2)|\boldsymbol{u}|^3\right] < \infty.$$

Thus, $A_{T,3} \xrightarrow{P} 0$. Now, consider the limiting behaviour of $A_{T,4}$. First, by the conditions required in the theorem, we have $\lambda_T \hat{w}_{T,j} / \sqrt{T-h} \leq \lambda_T / \left(\sqrt{T-h} \min_{j\in\mathbb{S}} \left(\left|\tilde{\theta}_j\right|^{\gamma_1} A_j^{\gamma_2} \right) \right) = \lambda_T / a_T \xrightarrow{P} 0$ for $j \in \mathbb{S}$ and $\frac{\lambda_T}{\sqrt{T-h}} w_{T,j} = \frac{\lambda_T}{\sqrt{T-h}} |\tilde{\theta}_j|^{-\gamma_1} A_j^{-\gamma_2} \geq \lambda_T / \left(\sqrt{T-h} \max_{j\notin\mathbb{S}} \left(\left|\tilde{\theta}_j\right|^{\gamma_1} A_j^{\gamma_2} \right) \right) = \lambda_T / b_T \xrightarrow{P} \infty$ for $j \notin \mathbb{S}$. In summary, we have

$$\frac{\lambda_T}{\sqrt{T-h}} \hat{w}_{T,j} = \frac{\lambda_T}{\sqrt{T-h}\left|\tilde{\theta}_j\right|^{\gamma_1} A_j^{\gamma_2}} \xrightarrow{P} \begin{cases} 0 & \text{if } \ j \in \mathbb{S} \\ \infty & \text{if } \ j \notin \mathbb{S} \end{cases}.$$

Secondly, we have

$$\sqrt{T-h}\left(\left| \theta_j^* + \frac{u_j}{\sqrt{T-h}} \right| - \theta_j^* \right) \ \to \begin{cases} u_j \text{sgn}(\theta_j^*) & \text{if } \ j \in \mathbb{S} \ \ (\theta_j^* = 0) \\ |u_j| & \text{if } \ j \notin \mathbb{S} \ \ (\theta_j^* \neq 0) \end{cases}.$$

By Slutsky's theorem, we have the following limiting behaviour of the third term

$$\frac{\lambda_T}{\sqrt{T-h}} \hat{w}_{T,j} \sqrt{T-h}\left(\left| \theta_j^* + \frac{u_j}{\sqrt{T-h}} \right| - \left|\theta_j^*\right| \right) \ \xrightarrow{P} \ \begin{cases} 0 & \text{if } \forall\, j \in \mathbb{S} \\ 0 & \text{if } u_j = 0, \ \ \forall\, j \notin \mathbb{S} \\ \infty & \text{otherwise} \end{cases}.$$

Thus, we have $V_T(\boldsymbol{u}) \to V(\boldsymbol{u})$ for every $\boldsymbol{u}$, where

$$\begin{aligned} V(\boldsymbol{u}) \ &= \ \frac{1}{2}\begin{pmatrix} \boldsymbol{u}'_{\mathbb{S}} & \boldsymbol{u}'_{\mathbb{S}^c} \end{pmatrix} \begin{pmatrix} \boldsymbol{J}_{\mathbb{S}\mathbb{S}} & \boldsymbol{J}_{\mathbb{S}\mathbb{S}^c} \\ \boldsymbol{J}_{\mathbb{S}^c\mathbb{S}} & \boldsymbol{J}_{\mathbb{S}^c\mathbb{S}^c} \end{pmatrix} \begin{pmatrix} \boldsymbol{u}_{\mathbb{S}} \\ \boldsymbol{u}_{\mathbb{S}^c} \end{pmatrix} + \begin{pmatrix} \boldsymbol{u}'_{\mathbb{S}} & \boldsymbol{u}'_{\mathbb{S}^c} \end{pmatrix} \begin{pmatrix} \boldsymbol{w}_{\mathbb{S}} \\ \boldsymbol{w}_{\mathbb{S}^c} \end{pmatrix} \\ &\quad + \sum_{j\in\mathbb{S}^c} \frac{\lambda_T}{\sqrt{T-h}} \hat{w}_{T,j} \sqrt{T-h}\left(\left| \theta_j^* + \frac{u_j}{\sqrt{T-h}} \right| - \left|\theta_j^*\right| \right) \\ &= \ \begin{cases} \frac{1}{2}\boldsymbol{u}'_{\mathbb{S}}\boldsymbol{J}_{\mathbb{S}\mathbb{S}}\boldsymbol{u}_{\mathbb{S}} + \boldsymbol{u}'_{\mathbb{S}}\boldsymbol{w}_{\mathbb{S}} & \text{if } \ \boldsymbol{u}_{\mathbb{S}^c} = \boldsymbol{0} \\ \infty & \text{otherwise} \end{cases}. \end{aligned}$$

where $\boldsymbol{w} \sim N(\mathbf{0},\ (\kappa_\eta - 1)\boldsymbol{J})$, and $\boldsymbol{w}_{\mathbb{S}} \sim N(\mathbf{0},\ (\kappa_\eta - 1)\boldsymbol{J}_{\mathbb{SS}})$. $V(\boldsymbol{u})$ is convex with the unique minimum $(-(\boldsymbol{J}_{\mathbb{SS}})^{-1}\boldsymbol{w}_{\mathbb{S}}, \mathbf{0})^T$. Following the epi-convergence results of Geyer (1994) and Knight-Fu (2000), $\operatorname{argmin}_{\boldsymbol{u}} V_T(\boldsymbol{u}) \xrightarrow{D} \operatorname{argmin}_{\boldsymbol{u}} V(\boldsymbol{u})$, we have

$$\begin{cases} \hat{\boldsymbol{u}}_{\mathbb{S}} \xrightarrow{D} -(\boldsymbol{J}_{\mathbb{SS}})^{-1}\boldsymbol{w}_{\mathbb{S}} \\ \hat{\boldsymbol{u}}_{\mathbb{S}^C} \xrightarrow{D} \mathbf{0} \end{cases},$$

or

$$\begin{cases} \sqrt{T-h}\left(\hat{\boldsymbol{\theta}}^{dapL}_{T,\mathbb{S}^c} - \boldsymbol{\theta}^*_{\mathbb{S}^c}\right) \xrightarrow{D} \mathbf{0} \\ \sqrt{T-h}\left(\hat{\boldsymbol{\theta}}^{dapL}_{T,\mathbb{S}} - \boldsymbol{\theta}^*_{\mathbb{S}}\right) \xrightarrow{D} N\left(\mathbf{0}\ \ (\kappa_\eta - 1)(\boldsymbol{J}_{\mathbb{SS}})^{-1}\right) \end{cases}.$$

∎

Corollary 3.4.9 *Let* $a_T = \sqrt{T-h}\min_{j\in\mathbb{S}}\left(|\tilde{\theta}_j|^{\gamma_1}A_j^{\gamma_2}\right)$, *and* $b_T = \sqrt{T-h}\max_{j\in\mathbb{S}^c}\left(|\tilde{\theta}_j|^{\gamma_1}A_j^{\gamma_2}\right)$. *If* $\lambda_T = o_p(a_T)$ *and* $\lambda_T/b_T \xrightarrow{P} \infty$, *then under* ***A1*** $-$ ***A5***, *we have that*

$$\mathbb{P}\left(j \in \hat{\mathbb{S}}_T\right) \to 1 \ \ \textit{if} \ \ j \in \mathbb{S},$$

as $T \to \infty$.

Proof By Theorem A.5.1, the $\sqrt{T-h}$-normality of $\hat{\boldsymbol{\theta}}^{dapL}_{T,\mathbb{S}}$ in Proposition 3.4.8 implies that $\|\hat{\boldsymbol{\theta}}^{dapL}_{T,\mathbb{S}} - \boldsymbol{\theta}^*_{\mathbb{S}}\| = O_p\left(1/\sqrt{T-h}\right)$. Thus, $\hat{\boldsymbol{\theta}}^{dapL}_{T,\mathbb{S}} \xrightarrow{P} \boldsymbol{\theta}^*_{\mathbb{S}}$, which implies that $\forall\, j \in \mathbb{S}$, we have $\mathbb{P}\left(j \in \hat{\mathbb{S}}_T\right) \to 1$, as $T \to \infty$. ∎

We extend the concept of oracle properties of an estimator discussed by Fan and Li (2001) to the context of time series analysis.

Definition (**Oracle properties**) . The doubly adaptive positive LASSO estimator $\hat{\boldsymbol{\theta}}^{dapL}_T$ for $\boldsymbol{\theta}^*$ is said to have the oracle properties if, with probability tending to 1, it could (i) identify the true sparsity pattern, i.e. $\lim P(\hat{\mathbb{S}}_T = \mathbb{S}) = 1$, (ii) identify the true lag order of the VAR process, i,e, $\lim P(\hat{q}^{dapL}_T = q) = 1$, and (iii) have an optimal estimation rate of the coefficients as $T \to \infty$.

The following theorem says that the doubly adaptive positive LASSO procedure is an oracle procedure.

Theorem 3.4.10 *(**Oracle properties of** $\hat{\boldsymbol{\theta}}_T^{dapL}$). Let* $a_T = \sqrt{T-h}\min_{j\in\mathbb{S}}\left(|\tilde{\theta}_j|^{\gamma_1}A_j^{\gamma_2}\right)$, *and* $b_T = \sqrt{T-h}\max_{j\in\mathbb{S}^c}\left(|\tilde{\theta}_j|^{\gamma_1}A_j^{\gamma_2}\right)$. *If* $\lambda_T = o_p(a_T)$ *and* $\lambda_T/b_T \xrightarrow{P} \infty$, *then under* ***A1*** $-$ ***A5***, $\hat{\boldsymbol{\theta}}_T^{dapL}$ *must satisfy:*

i) Selection Consistency: $\mathbb{P}\left(\hat{\mathbb{S}}_T = \mathbb{S}\right) \to 1$,

ii) Identification consistency: $\mathbb{P}\left(\hat{q}_T^{dapL} = q\right) \to 1$, *and*

iii) Asymptotic Normality: $\sqrt{T-h}\left(\hat{\boldsymbol{\theta}}_{\hat{\mathbb{S}}_T}^{dapL} - \boldsymbol{\theta}_{\mathbb{S}}^*\right) \xrightarrow{D} N\left(\mathbf{0},\ (\kappa_\eta - 1)(\boldsymbol{J}_{\mathbb{S}\mathbb{S}})^{-1}\right)$

as $T \to \infty$.

Proof (i) In view of Corollary 3.4.9, we know that $\forall\, j \in \mathbb{S}$, $P(j \in \hat{\mathbb{S}}_T) \to 1$. So it suffices to show that $\forall k \notin \mathbb{S}$, $P(k \in \hat{\mathbb{S}}_T) \to 0$. Now, we follow the methodology of Zou (2006).

Consider the event $\{k \in \hat{\mathbb{S}}_T\}$, where $k \notin \mathbb{S}$. The event $\{k \in \hat{\mathbb{S}}_T\}$ entails the KKT conditions for optimality, which requires that

$$\sum_{t=h+1}^{T} \frac{\partial \ell_t(\hat{\boldsymbol{\theta}}_T^{dapL})}{\partial \theta_k} + \lambda_T \hat{w}_{T,k} = 0.$$

Thus,

$$P(k \in \hat{\mathbb{S}}_T) \leq P\left(\frac{1}{\sqrt{T-h}}\sum_{t=h+1}^{T} \frac{\partial \ell_t(\hat{\boldsymbol{\theta}}_T^{dapL})}{\partial \theta_k} + \frac{\lambda_T}{\sqrt{T-h}}\hat{w}_{T,k} = 0\right).$$

By Taylor series expansion of $\frac{\partial \ell_t(\hat{\boldsymbol{\theta}}_T^{dapL})}{\partial \theta_k}$ around $\theta_k^* = 0$, we have

$$\frac{1}{\sqrt{T-h}}\sum_{t=h+1}^{T} \frac{\partial \ell_t(\hat{\boldsymbol{\theta}}_T^{dapL})}{\partial \theta_k} = B_{T,1} + B_{T,2} + B_{T,3},$$

where

$$B_{T,1} = \frac{1}{\sqrt{T-h}}\sum_{t=h+1}^{T} \frac{\partial \ell_t(\boldsymbol{\theta}^*)}{\partial \theta_k},$$

$$B_{T,2} = \frac{1}{\sqrt{T-h}}\sum_{t=h+1}^{T} \frac{\partial^2 \ell_t(\boldsymbol{\theta}^*)}{\partial^2 \theta_k}\hat{\theta}_k^{dapL},$$

$$B_{T,3} = \frac{1}{2\sqrt{T-h}}\sum_{t=h+1}^{T} \frac{\partial^3 \ell_t(\tilde{\boldsymbol{\theta}})}{\partial^3 \theta_k}(\hat{\theta}_k^{dapL})^2,$$

with $\tilde{\boldsymbol{\theta}}$ between $\boldsymbol{\theta}^*$ and $\hat{\boldsymbol{\theta}}_T^{dapL}$.

From Theorem 3.4.6, we have $B_{T,1} = \frac{1}{\sqrt{T-h}}\sum_{t=h+1}^{T} \frac{\partial \ell_t(\boldsymbol{\theta}^*)}{\partial \theta_k} \xrightarrow{P} N(0,\ \ (\kappa_\eta - 1)\boldsymbol{J}_{(k,k)})$, where $\boldsymbol{J}_{(k,k)}$ denotes the (k,k)-entry of the matrix $\boldsymbol{J}$. Thus, $B_{T,1} = O_p(1/\sqrt{T-h})$. From Lemma 3.4.6,

we also have $\frac{1}{T-h}\sum_{t=h+1}^{T}\frac{\partial^2\ell_t(\boldsymbol{\theta}^*)}{\partial^2\theta_k} \xrightarrow{P} \boldsymbol{J}_{(k,k)}$, where $\boldsymbol{J}_{(k,k)}$ denotes the (k,k)-entry of the matrix $\boldsymbol{J}$. In addition, Proposition 3.4.8 implies that $\hat{\theta}_k^{dapL} \xrightarrow{P} 0$. Thus, by the Slutsky's theorem, we have $B_{T,2} = O_p(1/\sqrt{T-h})$. Likewise, from Lemma 3.4.6 and Proposition 3.4.8, we have $B_{T,3} = O_p(1/\sqrt{T-h})$. Hence,

$$B_{T,1} + B_{T,2} + B_{T,3} = O_p(1/\sqrt{T-h}),$$

whereas by the condition of the theorem,

$$\frac{\lambda_T}{\sqrt{T-h}}\hat{w}_{T,k} = \frac{\lambda_T}{\sqrt{T-h}}\frac{1}{|\hat{\theta}_k|^{\gamma_1}A_{j'}^{\gamma_2}} \geq \frac{\lambda_T}{b_T} \xrightarrow{P} \infty.$$

Therefore,

$$\mathbb{P}(k \in \hat{\mathbb{S}}_T) \leq P\left(B_{T,1} + B_{T,2} + B_{T,3} + \frac{\lambda_T}{\sqrt{T-h}}\hat{w}_{T,k} = 0\right) \to 0,$$

and the property of selection consistency holds.

(ii) The ARCH order estimated by the doubly adaptive LASSO is

$$\hat{q}_T^{dapL} = \min\{j : \hat{\theta}_{T,k}^{dapL} = 0, \forall k = j+1, j+2, \cdots, h\},$$

or equivalently,

$$\hat{q}_T^{dapL} = \min\{j : k \in \hat{\mathbb{S}}_T^c, \forall k = j+1, j+2, \cdots h\}. \tag{3.14}$$

The true order q of the ARCH model is

$$q = \min\{j : k \in \mathbb{S}^c, \forall k = j+1, j+2, \cdots, h\}. \tag{3.15}$$

We have from (i) that $\hat{\mathbb{S}}_T^c \to \mathbb{S}^c$ in probability, so the RHS of (3.14) and (3.15) are equal in probability. Therefore $\hat{q}_T^{dapL} = q$ in probability.

(iii) From (i), we have that $\mathbb{P}\left(\hat{\boldsymbol{\theta}}_{\hat{\mathbb{S}}_T}^{dapL} = \hat{\boldsymbol{\theta}}_{T,\mathbb{S}}^{dapL}\right) \to 1$. Then, from Proposition 3.4.8, the asymptotic normality of $\hat{\boldsymbol{\theta}}_{\hat{\mathbb{S}}_T}^{dapL}$ follows. ∎

Remarks:

(**1**) Although the asymptotic distributions of $\hat{\boldsymbol{\theta}}_{T,\mathbb{S}}^{daL}$ and $\hat{\boldsymbol{\theta}}_{\hat{\mathbb{S}}_T}^{daL}$ are identical, $\hat{\boldsymbol{\theta}}_{T,\mathbb{S}}^{daL}$ and $\hat{\boldsymbol{\theta}}_{\hat{\mathbb{S}}_T}^{daL}$ represent different identities; $\hat{\boldsymbol{\theta}}_{T,\mathbb{S}}^{daL}$ is the daLASSO estimator for the vector of the true non-zero

parameters unknown in advance whereas $\hat{\boldsymbol{\theta}}^{daL}_{\hat{\mathbb{S}}_T}$ is the vector for non-zeros estimated by the daLASSO.

(**2**) Proposition 3.4.8 concerns $\hat{\boldsymbol{\theta}}^{daL}_{T,\mathbb{S}}$, the daLASSO estimators for the true non-zero parameters that are unknown in advance whereas Theorem 3.4.10 concerns $\hat{\boldsymbol{\theta}}^{daL}_{\hat{\mathbb{S}}_T}$, the non-zeros estimated by the daLASSO.

(**3**) Estimation consistency is necessary for oracle properties whereas oracle properties are sufficient for the former.

(**4**) Under the same asymptotic condition for tuning parameter λ_T (and other regularity conditions), the LASSO, the aLASSO and the daLASSO all have estimation consistency property.

(**5**) Under the same asymptotic condition for tuning parameter λ_T (and other regularity conditions), the aLASSO and the daLASSO both have oracle properties.

(**6**) The LASSO, the aLASSO and the daLASSO estimator might behaviour quite differently when finite samples are used. We need to investigate and compare their finite sample properties.

3.5 Computation algorithm for the doubly adaptive positive LASSO

We will modify the shooting algorithm described in Section 1.2.2 for the doubly adaptive LASSO. This requires quadratic approximation to the negative log quasi likelihood. The idea of quadratic approximation is not new. Chernoff (1954) implemented the idea of approximating the likelihood function by a quadratic function to establish the asymptotic properties of likelihood ratio tests. Tibshirani (1996) suggested the algorithm of iteratively reweighted least squares (IRLS) that would make use of quadratic approximation to a likelihood function. Andrews (1999) used this approach for estimation of a parameter on the boundary. Fan and Li (2001) proposed an unified algorithm for penalized likelihood based on the quadratic approximation of the log likelihood function. Francq and Zakoïan (2007) approximated the quasi-likelihood by quadratic function when they studied asymptotic distribution of the QML

estimator for ARCH processes when the true parameter may have zero coefficients. Wang and Leng (2007) proposed unified LASSO estimation via quadratic approximation.

3.5.1 Quadratic approximation to the negative log quasi likelihood

Let $(\epsilon_1, \cdots, \epsilon_T)$ be a realization of the ARCH(q) process defined by (3.1) and (3.2). Use $(\epsilon_h, \epsilon_{h-1}, \cdots, \epsilon_1)$ as a presample. The negative log of the Gaussian quasi-likelihood $L_T(\boldsymbol{\theta})$ is given by

$$L_T(\boldsymbol{\theta}) = \sum_{t=h+1}^{T} \ell_t(\boldsymbol{\theta}),$$

$$\ell_t(\boldsymbol{\theta}) = \frac{1}{2}\log {\sigma_t}^2 + \frac{{\epsilon_t}^2}{2{\sigma_t}^2} + \frac{1}{2} log(2\pi), \tag{3.16}$$

where $\boldsymbol{\theta} = [\alpha_0, \boldsymbol{\alpha}']'$ with $\boldsymbol{\alpha} = [\alpha_1, \cdots, \alpha_h]'$. Let

$$\boldsymbol{x}_{t-1} = \left(\epsilon_{t-1}^2, \ \epsilon_{t-2}^2, \ \cdots \ \epsilon_{t-h}^2\right)'$$

we express the conditional variance σ_t as

$$\sigma_t^2 = \alpha_0 + \boldsymbol{x}_{t-1}'\boldsymbol{\alpha}.$$

We approximate the negative likelihood $L_T(\boldsymbol{\theta})$ by second-order Taylor polynomial as follows.

$$\begin{aligned} L_T(\boldsymbol{\theta}) &\approx L_T(\boldsymbol{\theta}^*) + \boldsymbol{S}_T(\boldsymbol{\theta}^*)'(\boldsymbol{\theta} - \boldsymbol{\theta}^*) + \frac{1}{2}(\boldsymbol{\theta} - \boldsymbol{\theta}^*)'\boldsymbol{J}_T(\boldsymbol{\theta}^*)(\boldsymbol{\theta} - \boldsymbol{\theta}^*) \\ &= \frac{1}{2}\boldsymbol{\theta}'\boldsymbol{J}_T(\boldsymbol{\theta}^*)\boldsymbol{\theta} - [\boldsymbol{J}_T(\boldsymbol{\theta}^*)\boldsymbol{\theta}^* - \boldsymbol{S}_T(\boldsymbol{\theta}^*)]'\boldsymbol{\theta} + c_T(\boldsymbol{\theta}^*), \end{aligned} \tag{3.17}$$

where $\boldsymbol{\theta}^*$ is the unknown true parameter vector, and $c_T(\boldsymbol{\theta}^*) = \frac{1}{2}\boldsymbol{\theta}^{*\prime}\boldsymbol{J}_T(\boldsymbol{\theta}^*)\boldsymbol{\theta}^* - \boldsymbol{S}_T(\boldsymbol{\theta}^*)'\boldsymbol{\theta}^* + L_T(\boldsymbol{\theta}^*)$, the negative score vector $\boldsymbol{S}_T(\boldsymbol{\theta})$ is

$$\boldsymbol{S}_T(\boldsymbol{\theta}) = \frac{\partial L_T(\boldsymbol{\theta})}{\partial \boldsymbol{\theta}} = \sum_{t=h+1}^{T} \boldsymbol{s}_t(\boldsymbol{\theta}) = \sum_{t=h+1}^{T} \frac{1}{2\sigma_t^2}\left(1 - \frac{\epsilon_t^2}{\sigma_t^2}\right)\frac{\partial \sigma_t^2}{\partial \boldsymbol{\theta}},$$

and the negative Hessian matrix $\boldsymbol{J}_T(\boldsymbol{\theta})$ is

$$\boldsymbol{J}_T(\boldsymbol{\theta}) = \frac{\partial \boldsymbol{S}_T(\boldsymbol{\theta})}{\partial \boldsymbol{\theta}} = \sum_{t=h+1}^{T} \frac{\partial \boldsymbol{s}_t(\boldsymbol{\theta})}{\partial \boldsymbol{\theta}'} = \sum_{t=h+1}^{T} \frac{1}{2\sigma_t^4}\left(\frac{2\epsilon_t^2}{\sigma_t^2} - 1\right)\frac{\partial \sigma_t^2}{\partial \boldsymbol{\theta}}\frac{\partial \sigma_t^2}{\partial \boldsymbol{\theta}'}.$$

Since

$$\frac{\partial \sigma_t^2}{\partial \boldsymbol{\theta}} = \begin{pmatrix} 1 \\ \boldsymbol{x}_{t-1} \end{pmatrix},$$

we have

$$\boldsymbol{S}_T(\boldsymbol{\theta}) = \sum_{t=h+1}^{T} \frac{1}{2(\alpha_0 + \boldsymbol{x}_{t-1}'\boldsymbol{\alpha})} \left(1 - \frac{\epsilon_t^2}{\alpha_0 + \boldsymbol{x}_{t-1}'\boldsymbol{\alpha}}\right) \begin{pmatrix} 1 \\ \boldsymbol{x}_{t-1} \end{pmatrix},$$

and

$$\boldsymbol{J}_T(\boldsymbol{\theta}) = \sum_{t=h+1}^{T} \frac{1}{2(\alpha_0 + \boldsymbol{x}_{t-1}'\boldsymbol{\alpha})^2} \left(\frac{2\epsilon_t^2}{\alpha_0 + \boldsymbol{x}_{t-1}'\boldsymbol{\alpha}} - 1\right) \begin{pmatrix} 1 & \boldsymbol{x}_{t-1}' \\ \boldsymbol{x}_{t-1} & \boldsymbol{x}_{t-1}\boldsymbol{x}_{t-1}' \end{pmatrix}.$$

Now, we need to transform (3.17) into least squares and then iteratively minimize the penalized least squares, which will involve the decomposition of negative Hessian $\boldsymbol{J}_T(\boldsymbol{\theta})$. However, in each iteration step, say the *k-th* step, the Hessian evaluated at the estimated value $\boldsymbol{\theta}^{[k]}$ may not be positive definite, which precludes the Cholesky or LU decomposition. We may try the spectral decomposition instead. Since it is symmetric, the matrix $\boldsymbol{J}_T(\boldsymbol{\theta})$ has a spectral decomposition

$$\boldsymbol{J}_T(\boldsymbol{\theta}) = \boldsymbol{Q}(\boldsymbol{\theta})\boldsymbol{\Lambda}(\boldsymbol{\theta})\boldsymbol{Q}(\boldsymbol{\theta})',$$

where $\boldsymbol{\Lambda}(\boldsymbol{\theta})$ is a diagonal matrix with its diagonal elements being the eigenvalues of $\boldsymbol{J}_T(\boldsymbol{\theta})$, and $\boldsymbol{Q}(\boldsymbol{\theta})$ some orthogonal matrix. In order to use least-squares method, square-rooting the matrix $\boldsymbol{J}_T(\boldsymbol{\theta})$ is required. Unfortunately, we may not be able to calculate the square-root of diagonal matrix $\boldsymbol{\Lambda}(\boldsymbol{\theta})$ because some of the eigenvalues are negative. To bypass this problem, we define a surrogate for the Hessian matrix.

3.5.2 The surrogate of the quadratic approximation of likelihood

The surrogate for the Hessian matrix $\boldsymbol{J}_T(\boldsymbol{\theta})$, denoted by $\widetilde{\boldsymbol{J}}_T(\boldsymbol{\theta})$, is defined as

$$\widetilde{\boldsymbol{J}}_T(\boldsymbol{\theta}) = \boldsymbol{\Gamma}(\boldsymbol{\theta})|\boldsymbol{\Lambda}(\boldsymbol{\theta})|\boldsymbol{\Gamma}(\boldsymbol{\theta})',$$

where $|\boldsymbol{\Lambda}(\boldsymbol{\theta})|$ is a diagonal matrix with its diagonal elements being the absolute eigenvalues of $\boldsymbol{J}_T(\boldsymbol{\theta})$, and $\boldsymbol{\Gamma}(\boldsymbol{\theta})$ some orthogonal matrix. Accordingly, the surrogate for the quadratic approximation of likelihood $L_T(\boldsymbol{\theta})$ in (3.17), denoted by $\mathscr{S}_T(\boldsymbol{\theta})$, is defined as

$$\mathscr{S}_T(\boldsymbol{\theta}) = \frac{1}{2}\boldsymbol{\theta}'\boldsymbol{\Gamma}(\boldsymbol{\theta}^*)\left|\boldsymbol{\Lambda}(\boldsymbol{\theta}^*)\right|\boldsymbol{\Gamma}(\boldsymbol{\theta}^*)'\boldsymbol{\theta} - \boldsymbol{\theta}'\left[\boldsymbol{J}_T(\boldsymbol{\theta}^*)\boldsymbol{\theta}^* - \boldsymbol{S}_T(\boldsymbol{\theta}^*)\right] + c_T(\boldsymbol{\theta}^*).$$

Now, define and use the matrix

$$\widetilde{\boldsymbol{X}}(\boldsymbol{\theta}^*) = |\boldsymbol{\Lambda}(\boldsymbol{\theta}^*)|^{1/2}\boldsymbol{\Gamma}(\boldsymbol{\theta}^*)', \tag{3.18}$$

and the vector

$$\widetilde{y}(\boldsymbol{\theta}^*) = |\boldsymbol{\Lambda}(\boldsymbol{\theta}^*)|^{-1/2}\boldsymbol{\Gamma}(\boldsymbol{\theta}^*)'\left(\boldsymbol{J}_T(\boldsymbol{\theta}^*)\boldsymbol{\theta}^* - \boldsymbol{S}_T(\boldsymbol{\theta}^*)'\right). \tag{3.19}$$

A bit of manipulation yields the least squares form of the surrogate $\mathscr{S}_T(\boldsymbol{\theta})$ as follows

$$\mathscr{S}_T(\boldsymbol{\theta}) = \frac{1}{2}\left(\widetilde{\boldsymbol{y}}(\boldsymbol{\theta}^*) - \widetilde{\boldsymbol{X}}(\boldsymbol{\theta}^*)\boldsymbol{\theta}\right)'\left(\widetilde{\boldsymbol{y}}(\boldsymbol{\theta}^*) - \widetilde{\boldsymbol{X}}(\boldsymbol{\theta}^*)\boldsymbol{\theta}\right) + d_T(\boldsymbol{\theta}^*).$$

3.5.3 The modified shooting algorithm

The least squares form of the surrogate $\mathscr{S}_T(\boldsymbol{\theta})$ suggests an iterative algorithm for estimation. Suppose we get the estimates $\hat{\boldsymbol{\theta}}^{[k]}$ and $\tilde{\boldsymbol{\theta}}^{[k]}$ after the kth step, then at the $(k+1)$st step, we simply minimize the following least squares objective function

$$\left(\widetilde{\boldsymbol{y}}(\hat{\boldsymbol{\theta}}^{[k]}) - \widetilde{\boldsymbol{X}}(\hat{\boldsymbol{\theta}}^{[k]})\boldsymbol{\theta}\right)'\left(\widetilde{\boldsymbol{y}}(\hat{\boldsymbol{\theta}}^{[k]}) - \widetilde{\boldsymbol{X}}(\hat{\boldsymbol{\theta}}^{[k]})\boldsymbol{\theta}\right) + \lambda_T \sum_{j=1}^{h} \hat{w}_{T,j}(\tilde{\theta}_j^{[k]})\theta_j, \tag{3.20}$$

where $\widetilde{\boldsymbol{X}}$ and $\widetilde{y}$ are defined as in (3.18) and (3.19), respectively, and $\hat{w}_{T,1}(\tilde{\theta}_1^{[k]})$ corresponds to (3.7),

$$\hat{w}_{T,1}(\tilde{\theta}_1^{[k]}) = \begin{cases} 0 & \text{if intercept not to be penalized} \\ \frac{1}{(\tilde{\alpha}_0^{[k]})^{\gamma_1}\left(\sum_{i=0}^{h}|\hat{\rho}_{ii}|^{\gamma_0}\right)^{\gamma_2}} & \text{if intercept to be penalized} \end{cases} \tag{3.21}$$

and $\hat{w}_{T,j}(\tilde{\theta}_j^{[k]})$ for $j = 2, \cdots, h$ corresponds to (3.8),

$$\hat{w}_{T,j}(\tilde{\theta}_j^{[k]}) = \frac{1}{(\tilde{\alpha}_{j-1}^{[k]})^{\gamma_1}\left(\sum_{i=j-1}^{h}|\hat{\rho}_{ii}|^{\gamma_0}\right)^{\gamma_2}}, \quad j = 2, \cdots, h+1 \tag{3.22}$$

.

Applying the first optimization necessary condition to (5.22) with respect to $\boldsymbol{\theta}$ yields $q+1$ equations. Now, with reference to Section 1.2.2, we define

$$S_{0,j}^{[k]} = S_0\left(0,\ \boldsymbol{\theta}^{(-j)},\ \widetilde{\boldsymbol{X}}(\hat{\boldsymbol{\theta}}^{[k]}),\ \widetilde{\boldsymbol{y}}(\hat{\boldsymbol{\theta}}^{[k]})\right) = 2\sum_{i\neq j}\left(\widetilde{\boldsymbol{x}}(\hat{\boldsymbol{\theta}}^{[k]})^j\right)'\widetilde{\boldsymbol{x}}(\hat{\boldsymbol{\theta}}^{[k]})^j\theta_i - 2\left(\widetilde{\boldsymbol{x}}(\hat{\boldsymbol{\theta}}^{[k]})^j\right)'\widetilde{\boldsymbol{y}}(\hat{\boldsymbol{\theta}}^{[k]}), \tag{3.23}$$

$$S_j^{[k]} = S_j\left(\boldsymbol{\theta},\ \widetilde{\boldsymbol{X}}(\hat{\boldsymbol{\theta}}^{[k]}),\ \widetilde{\boldsymbol{y}}(\hat{\boldsymbol{\theta}}^{[k]})\right) = 2\left(\widetilde{\boldsymbol{x}}(\hat{\boldsymbol{\theta}}^{[k]})^j\right)'\widetilde{\boldsymbol{x}}(\hat{\boldsymbol{\theta}}^{[k]})^j\theta_j + S_{0,j}^{[k]},$$

and

$$\lambda_j^{[k]} = \lambda_T \hat{w}_{T,j}^{[k]} = \lambda_T \hat{w}_{T,j}(\tilde{\theta}_j^{[k]}),$$

where $\widetilde{\boldsymbol{x}}(\hat{\boldsymbol{\theta}}^{[k]})^j$ represents the jth column of $\widetilde{\boldsymbol{X}}(\hat{\boldsymbol{\theta}}^{[k]})$, and $\hat{w}_{T,j}(\tilde{\theta}_j^{[k]})$ is defined by (3.21) and (3.22).

Now, with aid of Figure 3.1, the (k+1)st step estimates for θ_j can be obtained using

$$\hat{\theta}_j^{[k+1]} = \begin{cases} \frac{-\lambda_j^{[k]} - S_{0,j}^{[k]}}{2\left(\widetilde{\boldsymbol{x}}(\hat{\boldsymbol{\theta}}^{[k]})^j\right)' \widetilde{\boldsymbol{x}}(\hat{\boldsymbol{\theta}}^{[k]})^j} & \text{if } S_{0,j}^{[k]} < -\lambda_j^{[k]}, \\ 0 & \text{otherwise.} \end{cases}$$

Note that superscripts $[k]$ are suppressed on Figure 3.1.

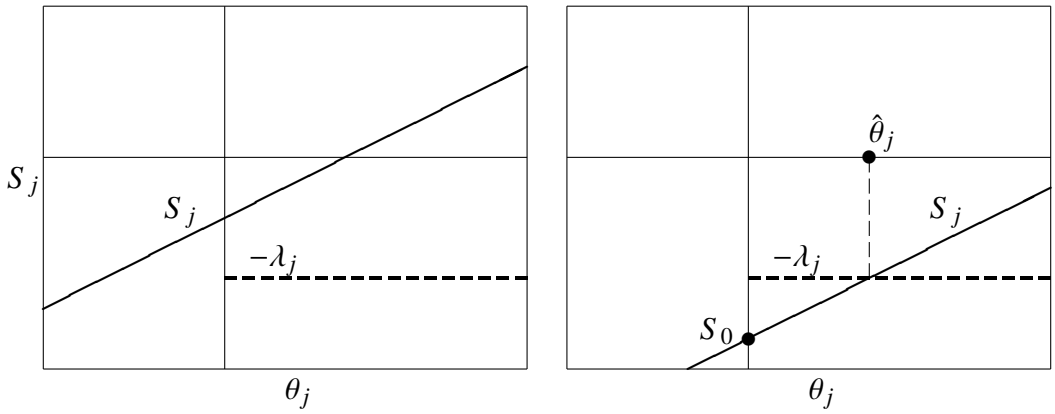


Figure 3.1: The modified shooting algorithm for the doubly adaptive positive LASSO. Left: Estimate for θ_j is 0. Right: $S_{0,j} < -\lambda_j$, the intersection of S_j and $-\lambda_j$ yields a positive estimate for θ_j.

Algorithm 5 shows computation steps in detail.

Algorithm 5: Modified shooting algorithm for the doubly adaptive positive LASSO given a value for the quadruple $(\lambda_T, \gamma_0, \gamma_1, \gamma_2)$

Input: Data $\epsilon_1, \cdots, \epsilon_T$, given values of $(\lambda_T, \gamma_0, \gamma_1, \gamma_2)$
Output: The $h+1$-dimensional vector estimate $\hat{\boldsymbol{\theta}}(\lambda_T, \gamma_0, \gamma_1, \gamma_2)$

Start: $k = 1$, initialize, say $\hat{\boldsymbol{\theta}}^{[k]} \leftarrow [0.0001, \cdots, 0.0001]$
Set stopping rule, $\|\hat{\boldsymbol{\theta}}^{[k+1]} - \hat{\boldsymbol{\theta}}^{[k]}\|_\infty < \zeta$, where ζ is a tiny number, say 0.00005
Iteration: Compute $\widetilde{\boldsymbol{X}}(\hat{\boldsymbol{\theta}}^{[k]})$ and $\widetilde{\boldsymbol{y}}(\hat{\boldsymbol{\theta}}^{[k]})$
Compute $\tilde{\boldsymbol{\theta}}^{[k]} \leftarrow \left(\widetilde{\boldsymbol{X}}(\hat{\boldsymbol{\theta}}^{[k]})'\widetilde{\boldsymbol{X}}(\hat{\boldsymbol{\theta}}^{[k]})\right)^{-1}\widetilde{\boldsymbol{y}}(\hat{\boldsymbol{\theta}}^{[k]})$
for $j \leftarrow 1$ ***to*** $h+1$ **do**
    $\lambda_j^{[k]} \leftarrow \lambda_T \hat{w}_{T,j}(\tilde{\theta}_j^{[k]})$ using (3.21) and (3.22)
    Compute $S_{0,j}^{[k]}$ using (3.23)
    if $S_{0,j}^{[k]} < -\lambda_j^{[k]}$ **then**
        $\hat{\theta}_j^{[k+1]} \leftarrow \left(-\lambda_j^{[k]} - S_{0,j}^{[k]}\right) / \left[2\left(\widetilde{\boldsymbol{x}}(\hat{\boldsymbol{\theta}}^{[k]})^j\right)'\widetilde{\boldsymbol{x}}(\hat{\boldsymbol{\theta}}^{[k]})^j\right]$
    else
        $\hat{\theta}_j^{[k+1]} \leftarrow 0$
if $\left\|\hat{\boldsymbol{\theta}}^{[k+1]} - \hat{\boldsymbol{\theta}}^{[k]}\right\|_\infty < \zeta$ **then**
    $\hat{\boldsymbol{\theta}}^{[k]} \leftarrow \hat{\boldsymbol{\theta}}^{[k+1]}$
    $k \leftarrow k+1$
    return *Iteration*
else
    Output: $\hat{\boldsymbol{\theta}} \leftarrow \hat{\boldsymbol{\theta}}^{[k+1]}$
End

The LASSO methodology yields a path of possible solutions defined by the continuum over tuning and weighting parameters. The choice of $\Lambda = (\lambda_T, \gamma_0, \gamma_1, \gamma_2)$ determines the tradeoff between model fit and model sparsity. We use the BIC criteria to select the optimal value for Λ. The BIC is defined as

$$BIC = 2L_T(\hat{\boldsymbol{\theta}}) + |\hat{\mathbb{S}}_T| \log(T-h),$$

where L_T is the negative log quasi-likelihood function defined in (3.3), $|\hat{\mathbb{S}}_T|$ is the cardinality of the set $\hat{\mathbb{S}}_T$. Define a 4-dimensional grid $\mathcal{G} = \lambda_T \times \gamma_0 \times \gamma_1 \times \gamma_2$ with a total number of G grid points. By using information criteria for LASSO, we have double penalization to be involved.

One is L_1 penalization by the LASSO, which yields the path solution of the LASSO,

$$\hat{\boldsymbol{\theta}}(\Lambda) = \arg\min_{\boldsymbol{\theta}} \mathscr{S}_T(\boldsymbol{\theta}) + \lambda_T \sum_{j=1}^{h+1} \hat{w}_{T,j}(\Lambda)\theta_j,$$

and the other is the L_0 penalization by the BIC, which yields

$$\Lambda^* = \arg\min_{\Lambda \in \mathcal{G}} \text{BIC}(\Lambda) = 2L_T(\hat{\boldsymbol{\theta}}(\Lambda)) + |\hat{\mathbb{S}}_T| \log(T-h).$$

Then the solution $\hat{\boldsymbol{\theta}}^{daL}$ is read off from the path against Λ^*. Algorithm 6 shows the complete computation steps.

Algorithm 6: Complete algorithm for the doubly adaptive positive LASSO

Input: Data: $\epsilon_1, \cdots, \epsilon_T$

Output: The doubly adaptive positive LASSO estimator $\hat{\boldsymbol{\phi}}_T^{dapL}$

Start: Set up a grid $\mathcal{G} = \lambda_T \times \gamma_0 \times \gamma_1 \times \gamma_2$ with $G = |\mathcal{G}|$
for $g \leftarrow 1$ ***to*** G **do**
    Apply Algorithm 5 to get $\hat{\boldsymbol{\theta}}(\Lambda^{(g)})$
    Calculate $\text{BIC}(\Lambda^{(g)}) = 2L_T(\hat{\boldsymbol{\theta}}(\Lambda^{(g)})) + |\hat{\mathbb{S}}_T^{(g)}| \log(T-h)$
Choose Λ^* such that $\text{BIC}(\hat{\boldsymbol{\theta}}(\Lambda^*)) = \min\{\text{BIC}(\Lambda^{(g)}) : \forall g = 1, \cdots, G\}$
Output $\hat{\boldsymbol{\theta}}_T^{daL} \leftarrow \hat{\boldsymbol{\theta}}(\Lambda^*)$
End

3.6 Monte Carlo study

We use Monte Carlo to empirically the performance of the adaptive positive LASSO estimator. The empirical minimum, maximum, mean, medium, mode (for ARCH lag order only), standard error, bias, MSE, MAD, and selection proportion were summarized. The definitions of empirical bias, MSE, and MAD are listed below for reference:

$$\widehat{Bias}(\hat{q}^{dapL}) = \hat{E}[\hat{q}^{dapL}] - q = \frac{1}{M}\sum_{m=1}^{M}(\hat{q}^{dapL})^{(m)} - q$$

$$\widehat{MSE}(\hat{q}^{dapL}) = \hat{E}[\hat{q}^{dapL} - q]^2 = \frac{1}{M}\sum_{m=1}^{M}((\hat{q}^{dapL})^{(m)} - q)^2$$

$$\widehat{MAD}(\hat{q}^{dapL}) = \hat{E}|\hat{q}^{dapL} - q| = \frac{1}{M}\sum_{m=1}^{M}|(\hat{q}^{dapL})^{(m)} - q|$$

$$\widehat{Bias}(\hat{\theta}_j^{dapL}) = \hat{E}[\hat{\theta}_j^{dapL}] - \theta_j^* = \frac{1}{M}\sum_{m=1}^{M}(\hat{\theta}_j^{dapL})^{(m)} - \theta_j^*$$

$$\widehat{MSE}(\hat{\theta}_j^{dapL}) = \hat{E}[\hat{\theta}_j^{dapL} - \theta_j^*]^2 = \frac{1}{M}\sum_{m=1}^{M}\left((\hat{\theta}_j^{dapL})^{(m)} - \theta_j^*\right)^2$$

$$\widehat{MAD}(\hat{\theta}_j^{dapL}) = \hat{E}|\hat{\theta}_j^{dapL} - \theta_j^*| = \frac{1}{M}\sum_{m=1}^{M}\left|(\hat{\theta}_j^{dapL})^{(m)} - \theta_j^*\right|$$

where M denotes the total number of MC runs.

We generated 764 data sets of sample size $T = 1000$ from the following sparse ARCH(12) model.

$$\begin{cases} \epsilon_t = \sqrt{\sigma_t}\eta_t, \\ \sigma_t^2 = 0.01 + 0.15\epsilon_{t-1}^2 + 0.3\epsilon_{t-4}^2 + 0.2\epsilon_{t-6}^2 + 0.15\epsilon_{t-10}^2 + 0.19\epsilon_{t-12}^2 \end{cases} \tag{3.24}$$

Pretending that we did not know the true lag order q, which is 12 in this case, of the underlying bivariate ARCH process, we set the maximum order $h = 50$. For the sake of simplicity we used $h = 50$ for all 764 models. To find an approximately optimal values for the quadruple $(\lambda_T, \gamma_0, \gamma_1, \gamma_2)$, we used grid-search method and the BIC criteria. Specifically, let $\mathcal{G} = \lambda_T \times \gamma_0 \times \gamma_1 \times \gamma_2 = [0.5, 1.7]_{\Delta=0.2} \times 2 \times [0, 1.75]_{\Delta=0.25} \times [0, 1.5]_{\Delta=0.25}$ [2]. For the sake of simplicity, the same 4-dimensional grid $\mathcal{G}$ was used for all 764 models. Algorithm 6 was applied to fit 764 models. Table 2.4 shows some empirical statistics such as Bias, MSE, and MAD of the ARCH order estimates. Empirical statistics were summarized in Table 3.1 and 3.2, from which a few points were observed.

Table 3.1: Empirical statistics of the doubly adaptive positive LASSO estimates for the ARCH order, based on 764 replications each of size T=1,000 generated from the model (3.24). The BIC was used to choose $(\lambda_T, \gamma_0, \gamma_1, \gamma_2)$

True	Minimum	Maximum	Mean	Median	Mode	SE	Bias	MSE	MAD
12	10	50	15	12	12	46.2	6.3	2.7	2.7

Observations:

(i) Order identification. Table 3.1 shows that the mode of 764 estimates for ARCH order is 12, suggesting that from a data set of moderate sample size the doubly adaptive positive

[2] Δ in the subscript represents the increment of the sequence.

Table 3.2: Empirical statistics of the doubly adaptive positive LASSO estimates for the ARCH coefficients, based on 764 replications each of size T=1,000 generated from the model (3.24). The BIC was used to choose $(\lambda_T, \gamma_0, \gamma_1, \gamma_2)$

Lag	TRUE	Minimum	Maximum	Mean	Median	SE	Bias	MSE	MAD	Proportion
0	0.01	0	0.0426	0.01206	0.011755	0.004333	0.002056	0.0000230	0.00360	0.995
1	0.15	0.0219	0.2597	0.13427	0.133238	0.037328	-0.015734	0.0016391	0.03277	1.000
2	0	0	0.0913	0.00254	0	0.010838	0.002540	0.0001238	0.00254	0.077
3	0	0	0.0721	0.00143	0	0.006229	0.001429	0.0000408	0.00143	0.073
4	0.3	0.0872	0.4691	0.27735	0.277100	0.052775	-0.022652	0.0032947	0.04563	1.000
5	0	0	0.0886	0.00169	0	0.008777	0.001691	0.0000798	0.00169	0.051
6	0.2	0	0.3618	0.17649	0.174962	0.054078	-0.023509	0.0034733	0.04684	0.993
7	0	0	0.0877	0.00100	0	0.006708	0.000997	0.0000459	0.00100	0.034
8	0	0	0.0925	0.00153	0	0.008291	0.001535	0.0000710	0.00153	0.052
9	0	0	0.0405	0.00072	0	0.003993	0.000716	0.0000164	0.00072	0.042
10	0.15	0	0.2762	0.11118	0.111969	0.049576	-0.038817	0.0039613	0.05128	0.959
11	0	0	0.0717	0.00102	0	0.006089	0.001021	0.0000381	0.00102	0.043
12	0.19	0	0.2949	0.14790	0.148414	0.047376	-0.042105	0.0040143	0.05183	0.993
13	0	0	0.0656	0.00043	0	0.003984	0.000433	0.0000160	0.00043	0.020
14	0	0	0.0627	0.00064	0	0.004714	0.000641	0.0000226	0.00064	0.026
15	0	0	0.0811	0.00031	0	0.003531	0.000309	0.0000125	0.00031	0.016
16	0	0	0.0791	0.00079	0	0.005920	0.000789	0.0000356	0.00079	0.027
17	0	0	0.0538	0.00053	0	0.004225	0.000527	0.0000181	0.00053	0.021
18	0	0	0.0584	0.00034	0	0.003390	0.000338	0.0000116	0.00034	0.013
19	0	0	0.0274	0.00020	0	0.001909	0.000196	0.0000037	0.00020	0.014
20	0	0	0.0296	0.00022	0	0.002173	0.000216	0.0000048	0.00022	0.012
21	0	0	0.0357	0.00035	0	0.002796	0.000351	0.0000079	0.00035	0.021
22	0	0	0.0460	0.00025	0	0.003003	0.000252	0.0000091	0.00025	0.009
23	0	0	0.0283	0.00015	0	0.001544	0.000147	0.0000024	0.00015	0.017
24	0	0	0.0342	0.00032	0	0.002883	0.000318	0.0000084	0.00032	0.016
25	0	0	0.0233	0.00013	0	0.001552	0.000133	0.0000024	0.00013	0.009
26	0	0	0.0279	0.00009	0	0.001268	0.000090	0.0000016	0.00009	0.007
27	0	0	0.0325	0.00023	0	0.002102	0.000230	0.0000045	0.00023	0.014
28	0	0	0.0183	0.00011	0	0.001307	0.000113	0.0000017	0.00011	0.009
29	0	0	0.0375	0.00012	0	0.001920	0.000117	0.0000037	0.00012	0.004
30	0	0	0.0141	0.00004	0	0.000636	0.000041	0.0000004	0.00004	0.005
31	0	0	0.0137	0.00004	0	0.000681	0.000040	0.0000005	0.00004	0.005
32	0	0	0.0197	0.00011	0	0.001341	0.000114	0.0000018	0.00011	0.010
33	0	0	0.0284	0.00009	0	0.001294	0.000087	0.0000017	0.00009	0.007
34	0	0	0.0128	0.00003	0	0.000581	0.000029	0.0000003	0.00003	0.003
35	0	0	0.0152	0.00002	0	0.000550	0.000021	0.0000003	0.00002	0.003
36	0	0	0.0235	0.00013	0	0.001569	0.000131	0.0000025	0.00013	0.009
37	0	0	0	0	0	0	0	0	0	0
38	0	0	0.0175	0.000038	0	0.000697	0.000038	0.0000005	0.00004	0.004
39	0	0	0.0126	0.000016	0	0.000455	0.000016	0.0000002	0.00002	0.001
40	0	0	0.0086	0.000011	0	0.000309	0.000011	0.0000001	0.00001	0.001
41	0	0	0.0074	0.000010	0	0.000266	0.000010	0.0000001	0.00001	0.001
42	0	0	0.0130	0.000021	0	0.000481	0.000021	0.0000002	0.00002	0.003
43	0	0	0.0079	0.000010	0	0.000285	0.000010	0.0000001	0.00001	0.001
44	0	0	0.0228	0.000065	0	0.001088	0.000065	0.0000012	0.00007	0.004
45	0	0	0	0	0	0	0	0	0	0
46	0	0	0	0	0	0	0	0	0	0
47	0	0	0.0175	0.0000229	0	0.000633	0.000023	0.0000004	0.00002	0.001
48	0	0	0.0045	0.0000059	0	0.000163	0.000006	0.0000000	0.00001	0.001
49	0	0	0	0	0	0	0	0	0	0
50	0	0	0.0019	0.0000024	0	0.000067	0.000002	0.0000000	0.00000	0.001

LASSO estimator is able to choose the true ARCH order most frequently . This is evident also from Table 3.2: The selection probabilities of ARCH(12) coefficients beyond the true order 12 is very little. The mean and median of estimates for ARCH order are 15 and 12, respectively, indicating that the distribution of ARCH order estimates is skewed to the left, which is not surprising since the LASSO methodology is conservative, as often observed in practice.

(ii) Variable selection. Table 3.2 shows that ϵ_{t-1}, ϵ_{t-4}, ϵ_{t-6}, ϵ_{t-10}, and ϵ_{t-12} are almost always selected by the doubly adaptive LASSO.

(iii) Coefficients Estimation. Table 3.2 shows that the bias, MSE, and MAD are very small on average, indicating that the estimation consistency is valid even for the moderate sample size.

The numerical example shows promising results for the doubly adaptive LASSO for ARCH models. It is consistent with the asymptotic properties, that is, with the values of γ_0, γ_1, and γ_2 properly chosen, the proposed doubly adaptive positive LASSO can achieve identification consistency, variable selection consistency, and variable estimation consistency.

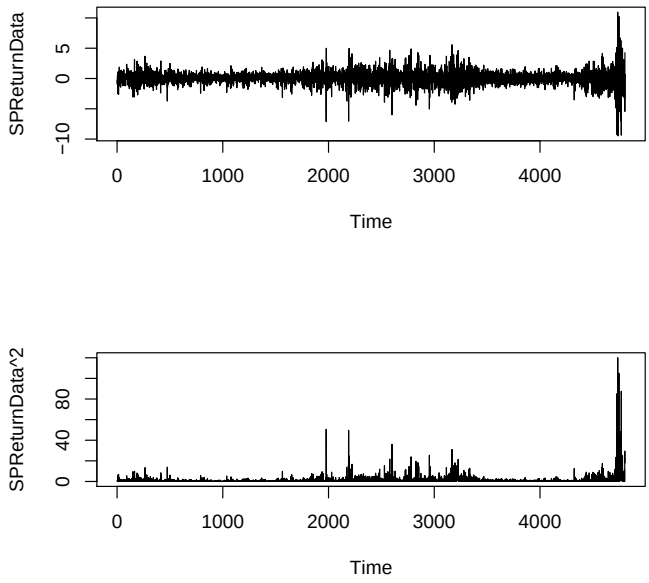
3.7 Real data analysis examples: models for stock indices

3.7.1 The US S&P500 Return Data

We collected 4804 observations of the S&P500 index that cover the period from January 2, 1990 to January 22, 2009 from the website of Yahoo Finance and the log returns were calculated. Some of the stylized facts are evident from Figure 3.2 and Figure 3.3, as we discussed in Section 3.1, which justify the use of ARCH models to capture those characteristics.

We set the maximum lag order $h = 70$, and used the minimum BIC criterion to select the optimal combination of values for $\lambda_T, \gamma_0, \gamma_1, \gamma_2$. The doubly adaptive positive LASSO yields a sparse ARCH(61) model with 16 ARCH terms 1,2,3,4,5,6,8,10,11,19,25,33,38,39,46 and 61:

Figure 3.2: The S&P500 Daily Returns and Squared Daily Returns from January 2, 1990 to January 22, 2009. Data source: Yahoo Finance



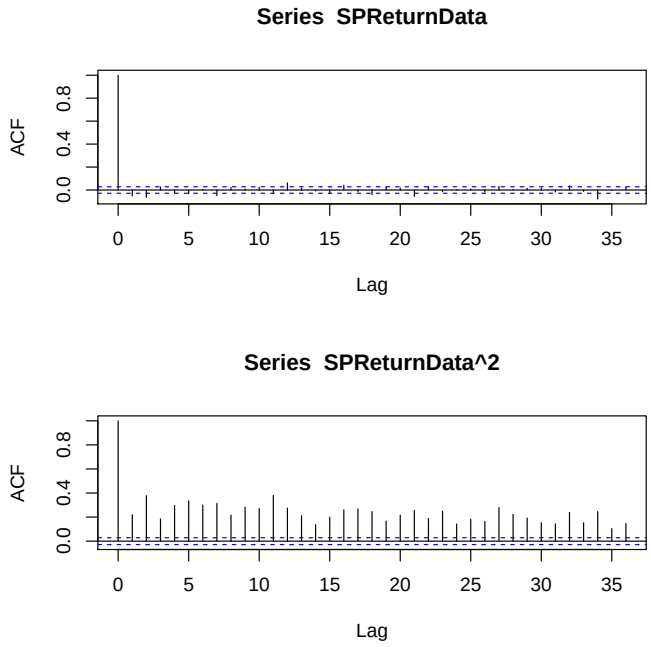
$$
\begin{aligned}
\hat{\sigma}_t^2 &= 0.1354 + 0.0233\epsilon_{t-1} + 0.1209\epsilon_{t-2} + 0.0506\epsilon_{t-3} + 0.1013\epsilon_{t-4} \\
&+ 0.083\epsilon_{t-5} + 0.0295\epsilon_{t-6} + 0.0533\epsilon_{t-8} + 0.0745\epsilon_{t-10} + 0.0506\epsilon_{t-11} \\
&+ 0.1013\epsilon_{t-19} + 0.083\epsilon_{t-25} + 0.0295\epsilon_{t-33} + 0.0506\epsilon_{t-38} + 0.1013\epsilon_{t-39} \\
&+ 0.083\epsilon_{t-46} + 0.0295\epsilon_{t-61}
\end{aligned}
$$

3.7.2 The Japan Nikkei Return Data

We collected 4804 observations of the the Japanese Nikkei index that cover the period from January 2, 1990 to January 22, 2009. Some of the stylized facts are evident from Figure 3.4 and Figure 3.5, as we discussed in Section 3.1, which justify the use of ARCH models to capture those characteristics.

We set the maximum lag order $h = 70$, and use the minimum BIC criterion to select the optimal combination of values for $\lambda_T, \gamma_0, \gamma_1, \gamma_2$. The Adaptive Positive LASSO yields a sparse

Figure 3.3: The ACF of S&P500 Daily Returns and Squared Daily Returns from January 2, 1990 to January 22, 2009



ARCH(21) model with 10 ARCH terms 1, 2, 3, 4, 5, 6, 7, 9, 15, and 21:

$$\begin{aligned}\hat{\sigma}_t^2 &= 0.642 + 0.0482\epsilon_{t-1} + 0.1082\epsilon_{t-2} + 0.1069\epsilon_{t-3} + 0.0893\epsilon_{t-4} + 0.1053\epsilon_{t-5} \\ &+ 0.066\epsilon_{t-6} + 0.0832\epsilon_{t-7} + 0.0198\epsilon_{t-9} + 0.049\epsilon_{t-15} + 0.0386\epsilon_{t-21}\end{aligned}$$

Figure 3.4: The Nikkei Daily Returns and Squared Daily Returns from January 2, 1990 to January 22, 2009. Data source: Yahoo Finance

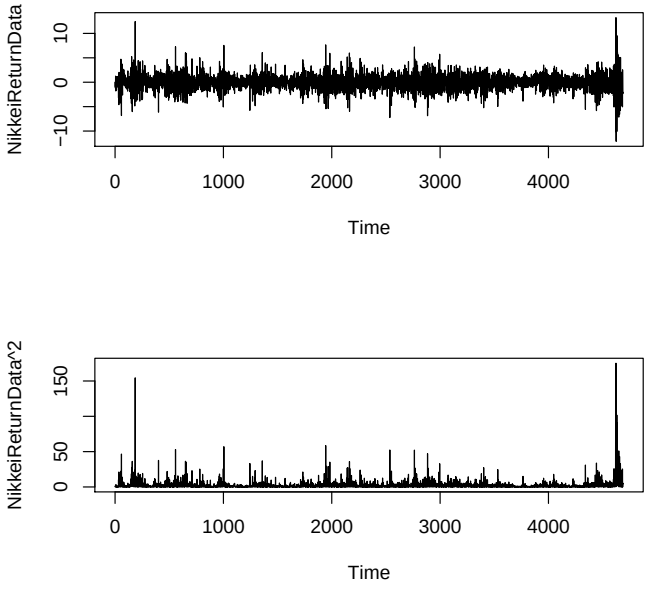
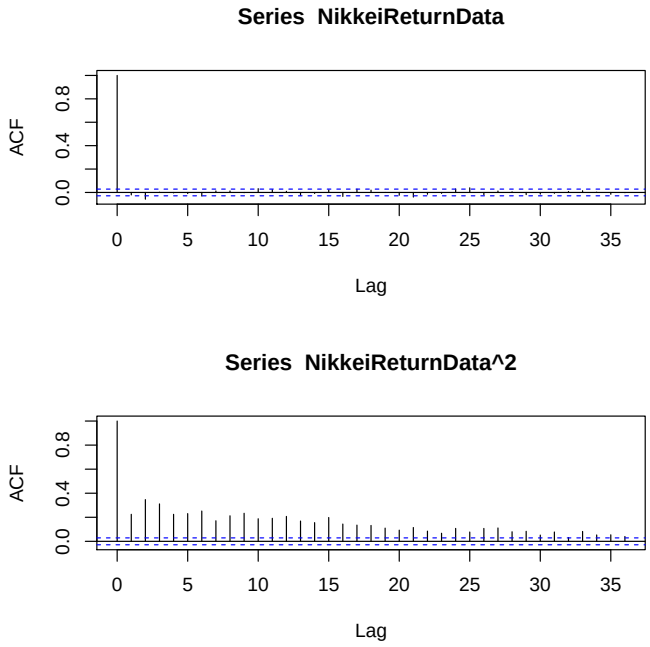


Figure 3.5: The ACF of Nikkei Daily Returns and Squared Daily Returns from January 2, 1990 to January 22, 2009. Data source: Yahoo Finance



Chapter 4

The Doubly Adaptive LASSO for Multivariate AR(p) Models

4.1 Introduction

The multivariate or vector autoregressive (VAR) model is a generalization of univariate AR process that can be used to model the dynamics of vector stationary time series. Recall that Wold's decomposition theorem tells us that any purely undeterministic multivariate stationary process with constant mean vector can be represented as the output of a causal linear filter with multivariate white noise input, and can be approximated well by a VAR(p) process, where the order p is finite, under quite general condition of absolute summability of the coefficients of the linear filter (see Lütkepohl, 2006 p.25). Naturally, we desire sparse VAR models since sparse ones may yield better forecasts compared to full models and may be easier to interpret. Because the number of VAR coefficients can be prohibitively large for even moderate dimensions, it is computationally infeasible to employ classical approaches such as all subsets selection to fitting a sparse VAR model quickly. Due to ample applications of the LASSO methodology to model selection, we naturally consider to apply the LASSO methodology to VAR modeling. There are quite a few results in the literature that applied the LASSO methodology to building VAR models, as we reviewed in Section 1.3.

In Chapter 2, we proposed the doubly adaptive LASSO for univariate AR models. The doubly adaptive LASSO integrates the temporal partial autocorrelations of a time series with the OLS or Yule-Walker estimates into the adaptive weights. This chapter inherits the same spirit from Chapter 2. We propose the doubly adaptive LASSO for modelling a VAR(p) pro-

cess, which integrates the norms of the partial lag autocorrelation matrices (Heyse, 1985) of a vector time series with the OLS or Yule-Walker estimates into the adaptive weights.

We start with a review on some basic concepts regarding the VAR(p) process, and standard procedure for building a VAR(p) model. In particular, we discuss the notion of partial lag autocorrelation (PLAC) matrix function (Heyse, 1985). In Section 4.3 we review the adaptive LASSO (Zou, 2006) for VAR(p) models when the lag order is known, and we propose the doubly adaptive LASSO for VAR models with the lag order is unknown a priori, as is the usual case. In Section 4.4 we study the asymptotic properties of the doubly adaptive LASSO estimators. The algorithmic implementation is discussed in Section 4.5. Results from simulation study are summarized in Section 4.6.

4.2 The VAR(p) process and standard modelling procedure

The content of this section can be found in advanced textbooks on multivariate time series analysis (e.g. Brockwell and Davis, 1991, p.401 - 420; Hannan, 1970, p.8 - 31, 1970; Hamilton, 1994, p.257 - 279; Lütkepohl, 2006, p.13 - 87, p.146 - 153; Wei, 2005, p.408 -412).

Definition (**The VAR(p) process**). The K-variate time series $\{\boldsymbol{y}_t\}, t \in \mathbb{Z} = \{0, \pm 1, \pm 2, \cdots\}$ is said to be a VAR(p) process if it is stationary, and it is the solution of the specification

$$\boldsymbol{y}_t = \boldsymbol{\Phi}_1 \boldsymbol{y}_{t-1} + \cdots + \boldsymbol{\Phi}_p \boldsymbol{y}_{t-p} + \boldsymbol{\epsilon}_t, \ t \in \mathbb{Z}, \tag{4.1}$$

where $\boldsymbol{\Phi}_i$'s are fixed $K \times K$ coefficient matrices, and the innovation process $\boldsymbol{\epsilon}_t \sim \mathrm{WN}_K(\boldsymbol{0}, \Sigma_\epsilon)$. We say that $\{\boldsymbol{y}_t\}$ is an VAR(p) process with mean $\boldsymbol{\mu}$ if $\{\boldsymbol{y}_t - \boldsymbol{\mu}\}$ is an VAR(p) process.

In this thesis, for convenience and without loss of generality, we deal with only the demeaned VAR(p) process.

Estimation of the VAR(p) model

Given the VAR order p there are a variety of approaches to estimating the parameters (see, for example, Lütkepohl (2006) p.69 - 102). If the distribution of the innovation process is known, we can get MLE by maximizing the log-likelihood function. Through the Yule-Walker

equations we can obtain the method-of-moments estimator. Maximizing the Gaussian quasi-likelihood yields QMLE if the normal distribution is used as a proxy for the unknown innovation distribution. A further possibility is to treat $\boldsymbol{y}_t = \boldsymbol{\Phi}_1 \boldsymbol{y}_{t-1} + \cdots + \boldsymbol{\Phi}_p \boldsymbol{y}_{t-p} + \boldsymbol{\epsilon}_t,\ t = 1, \cdots, T$ as multivariate regression equation and employ the ordinary least squares (OLS) method for estimation. As in the univariate case, the OLS estimator has downward bias (Jiostheim and Paulser, 1983; Nicholls and Pope 1988; Brannstrom, 1995). However, Hannan (1970) shows that the OLS estimator has nice asymptotic properties such as consistency and asymptotic normality under some regularity conditions.

Identification via information criteria

A sequence of VAR models are estimated with successively increasing orders $1, 2, ..., h$ with h sufficiently large. Then the model that minimizes some criterion is chosen. Some frequently used criteria include the final prediction error (FPE) (Akaike, 1969), the Akaike information criterion (AIC) (Akaike, 1974, 1978), the Bayesian information criterion (BIC) (Shwarz, 1978), and the HQ criteria (Hannan and Quinn, 1979).

The Partial lag autocorrelation matrix [1]

We may employ the Box-Jenkins methodology, starting with identification of the lag order. Then parameter estimation follows after the lag order identification. In extending the partial autocorrelation concept to vector time series, Heyse (1985) introduced the notion of the partial lag autocorrelation matrix function [2], which is the autocorrelation matrix between the elements of $\boldsymbol{y}_t$ and $\boldsymbol{y}_{t+s}$, after removing the linear dependence of each on the intervening vectors $\boldsymbol{y}_{t+1}, \cdots, \boldsymbol{y}_{t+s-1}$, which is defined as the ordinary correlation between the elements of residuals,

$$\boldsymbol{u}_{s-1,t+s} = \boldsymbol{y}_{t+s} - (\Psi_{s-1,1} \boldsymbol{y}_{t+s-1} + \cdots + \Psi_{s-1,s-1} \boldsymbol{y}_{t+1}), \tag{4.2}$$

and

$$\boldsymbol{v}_{s-1,t} = \boldsymbol{y}_t - (\Theta_{s-1,1} \boldsymbol{y}_{t+1} + \cdots + \Theta_{s-1,s-1} \boldsymbol{y}_{t+s-1}). \tag{4.3}$$

[1] For more detailed derivation and numerical computation, please go to Appendix C.

[2] De Jong (1976) extended the Durbin-Levinson recursive algorithm to vector case.

Definition (**Partial lag autocorrelation matrix** (Heyse, 1985)). The partial lag autocorrelation matrix function of lag s is defined as

$$\boldsymbol{P}(s) = D_{\boldsymbol{v}}(s)^{-1/2}\boldsymbol{V}_{\boldsymbol{vu}}(s)D_{\boldsymbol{u}}(s)^{-1/2}, \tag{4.4}$$

where

$$\begin{aligned}
\boldsymbol{V}_{\boldsymbol{u}}(s) &= \text{VAR}[\boldsymbol{u}_{s-1,t+s}], \\
\boldsymbol{V}_{\boldsymbol{v}}(s) &= \text{VAR}[\boldsymbol{v}_{s-1,t}], \\
\boldsymbol{V}_{\boldsymbol{vu}}(s) &= \text{Cov}(\boldsymbol{v}_{s-1,t}, \boldsymbol{u}_{s-1,t+s}),
\end{aligned}$$

and $D_{\boldsymbol{v}}(s)$ and $D_{\boldsymbol{u}}(s)$ are the diagonal matrices of $\boldsymbol{V}_{\boldsymbol{v}}(s)$ and $\boldsymbol{V}_{\boldsymbol{u}}(s)$, respectively.

The $K \times K$ matrix function of the lag s, $\boldsymbol{P}(s)$, is a vector extension of the partial autocorrelation function in the same manner as the autocorrelation matrix function is a vector extension of the autocorrelation function. It can be shown that for $s \geq 2$, we have

$$\boldsymbol{V}_{\boldsymbol{u}}(s) = \Gamma(0) - \sum_{k=1}^{s-1} \Psi_{s-1,k}\Gamma(k), \tag{4.5}$$

$$\boldsymbol{V}_{\boldsymbol{v}}(s) = \Gamma(0) - \sum_{k=1}^{s-1} \Theta_{s-1,k}\Gamma'(k), \tag{4.6}$$

$$\boldsymbol{V}_{\boldsymbol{vu}}(s) = \Gamma(s) - \sum_{k=1}^{s-1} \Gamma(s-k)\Psi'_{s-1,k}. \tag{4.7}$$

For the case $s = 1$ since there are no intervening vectors between $\boldsymbol{y}_t$ and $\boldsymbol{y}_{t+s}$ we have

$$\begin{aligned}
\boldsymbol{V}_{\boldsymbol{u}}(1) &= \text{VAR}(\boldsymbol{y}_{t+1}) = \Gamma(0), \\
\boldsymbol{V}_{\boldsymbol{v}}(1) &= \text{VAR}(\boldsymbol{y}_t) = \Gamma(0), \\
\boldsymbol{V}_{\boldsymbol{vu}}(1) &= \text{Cov}(\boldsymbol{y}_t, \boldsymbol{y}_{t+1}) = \Gamma(1),
\end{aligned}$$

and

$$\boldsymbol{P}(1) = D^{-1/2}\Gamma(1)D^{-1/2} = \boldsymbol{\rho}(1),$$

where D is the diagonal matrix of $\Gamma(0)$, and $\boldsymbol{\rho}(1)$ the regular autocorrelation matrix at lag 1.

It can be shown that for $K = 1$ the partial lag autocorrelation matrix function $\boldsymbol{P}(s)$ reduces to the partial autocorrelation function of a univariate autoregressive process.

Analogous to the partial autocorrelation function for the univariate case the partial lag autocorrelation matrix, $\boldsymbol{P}(s)$ has the cut-off property for autoregressive processes. So if $\{\boldsymbol{y}_t\}$ is a vector autoregressive process of order p then $\boldsymbol{P}(s)$ will be nonzero for $s = p$ and will equal 0 for $s > p$. This property makes $\boldsymbol{P}(s)$ a useful tool for identifying VAR processes.

Heyse (1985) also proposed a recursive procedure (See Algorithm 12) for computing $\boldsymbol{P}(s)$, which is a vector generalization of Durbin's (1960) recursive computational procedure for univariate partial autocorrelations. The algorithm requires that we first estimate the sample cross-covariance matrices. Given a realization an K-dimensional vector time serie $\boldsymbol{y}_1, \boldsymbol{y}_2, \cdots, \boldsymbol{y}_T$, the sample autocovariance matrix at lag s is computed by

$$\widehat{\Gamma}(s) = \frac{1}{T}\sum_{t=1}^{T-s}(\boldsymbol{y}_t - \bar{\boldsymbol{y}})(\boldsymbol{y}_t - \bar{\boldsymbol{y}})',$$

where $\bar{\boldsymbol{y}}$ is the vector of sample mean. The sample partial lag autocorrelation matrix, $\widehat{\boldsymbol{P}}(s)$, can be obtained by using $\widehat{\Gamma}(r)$ of $\Gamma(r)$ for $r = 0, \cdots, s-1$ in the recursive algorithm.

VAR order identification via sample PLAC matrix

Under the null hypothesis that $\{\boldsymbol{y}_t\}$ is a VAR(s-1) process, the two series of residuals $\{\boldsymbol{u}_{s-1,t+s}\}$ and $\{\boldsymbol{v}_{s-1,t}\}$ are uncorrelated, and each consists of K independent white noise series. Using Quenouille (1957, p.41) and Hannan(1970, p.400), the elements of $\widehat{\boldsymbol{P}}(s)$, denoted by $\widehat{P}_{ij}(s)$, are asymptotically $N(0, 1/T)$ distributed. Use Tiao and Box's notations "**+**" to indicate that $\widehat{P}_{ij}(s) > 2/\sqrt{T}$, "**−**" to indicate that $\widehat{P}_{ij}(s) < -2/\sqrt{T}$, and "**·**" to indicate that $-2/\sqrt{T} \le \widehat{P}_{ij}(s) \le 2/\sqrt{T}$. In addition, $T\left(\widehat{P}_{ij}(s)\right)^2 \sim \chi^2(1)$ asymptotically, which implies that asymptotically

$$X(s) = T\sum_{i=1}^{K}\sum_{i=1}^{K}\left(\widehat{P}_{ij}(s)\right)^2 \sim \chi^2(K^2). \tag{4.8}$$

$X(s)$ provides a diagnostic aid for determining the order of a vector autoregressive model.

4.3 The adaptive LASSO and doubly adaptive LASSO

In this section, we use the LASSO methodology to model the VAR(p) process. There are two situations. If the order is known in advance or has been identified already, we recommend

the adaptive LASSO of Zou (2006). If the order is not known in advance or difficult to identify, we propose the doubly adaptive LASSO, or partial lag autocorrelation or PLAC-weighted adaptive LASSO. By employing the PLAC-weighted adaptive LASSO we want to get order identification, subset selection and parameter estimation properly done in one go.

4.3.1 The doubly adaptive LASSO when p is unknown

Suppose that we observe a time series $\boldsymbol{y}_1, \boldsymbol{y}_2, \cdots, \boldsymbol{y}_T$, which is a realization of a stationary K-variate VAR(p) process with the true order p and true parameter matrix $\boldsymbol{\Phi}^o = (\boldsymbol{\Phi}_1^o, \cdots, \boldsymbol{\Phi}_p^o)$ unknown. We also denote the parameters in a vector form

$$\begin{aligned} \boldsymbol{\phi}^o &\equiv \left(\phi_1^o, \phi_2^o, \cdots, \phi_{pK^2}^o\right) \\ &= vec(\boldsymbol{\Phi}^o) = \left(vec(\boldsymbol{\Phi}_1^o)', vec(\boldsymbol{\Phi}_2^o)', \cdots, vec(\boldsymbol{\phi}_p^o)'\right)' \\ &= \left(\phi_{11,1}^o, \cdots, \phi_{KK,1}^o, \cdots, \phi_{11,p}^o, \cdots, \phi_{KK,p}^o\right)'. \end{aligned} \tag{4.9}$$

Because the true lag order p is not known a priori, we set the order to be h, which is sufficiently large such that $h > p$. Since the initial values $\boldsymbol{y}_0, \cdots, \boldsymbol{y}_{-h+1}$ are not available, we may use $\boldsymbol{y}_1, \cdots, \boldsymbol{y}_h$ as a presample. This will reduce the effective sample size from T to $T-h$. Now, having the data, we formulate the following VAR(h) model

$$\boldsymbol{y}_t = \boldsymbol{\Phi}_1 \boldsymbol{y}_{t-1} + \cdots + \boldsymbol{\Phi}_h \boldsymbol{y}_{t-h} + \boldsymbol{\epsilon}_t,\ t = h+1, \cdots T. \tag{4.10}$$

Let

$$\boldsymbol{\Phi} = (\boldsymbol{\Phi}_1, \boldsymbol{\Phi}_2, \cdots, \boldsymbol{\Phi}_h)_{K\times(hK)}, \tag{4.11}$$

$$\boldsymbol{x}_t = \left(\boldsymbol{y}_t', \boldsymbol{y}_{t-1}', \cdots, \boldsymbol{y}_{t-h+1}'\right)'_{(hK)\times 1}. \tag{4.12}$$

Then the model (4.10) can be written as

$$\boldsymbol{y}_t = \boldsymbol{\Phi} \boldsymbol{x}_{t-1},\ t = h+1, \cdots T.$$

If we define

$$\boldsymbol{Y} = (\boldsymbol{y}_{h+1}, \boldsymbol{y}_{h+2}, \cdots, \boldsymbol{y}_T)_{K\times(T-h)}, \tag{4.13}$$

$$\boldsymbol{X} = (\boldsymbol{x}_h, \boldsymbol{x}_{h+1}, \cdots, \boldsymbol{x}_{T-1})_{(hK)\times(T-h)}, \tag{4.14}$$

$$\boldsymbol{E} = (\boldsymbol{\epsilon}_{h+1}, \boldsymbol{\epsilon}_{h+2}, \cdots, \boldsymbol{\epsilon}_T)_{K\times(T-h)},$$

To estimate the model via the OLS method, we formulate compactly the multivariate-regression-type equations as

$$\boldsymbol{Y} = \boldsymbol{\Phi}\boldsymbol{X} + \boldsymbol{E}.$$

To see its structure, we expand the design matrix $\boldsymbol{X}$ as

$$\boldsymbol{X} = \begin{pmatrix} \boldsymbol{y}_h & \boldsymbol{y}_{h+1} & \cdots & \boldsymbol{y}_{T-1} \\ \boldsymbol{y}_{h-1} & \boldsymbol{y}_h & \cdots & \boldsymbol{y}_{T-2} \\ \vdots & \vdots & \vdots & \vdots \\ \boldsymbol{y}_1 & \boldsymbol{y}_2 & \cdots & \boldsymbol{y}_{T-h} \end{pmatrix}_{(hK)\times(T-h)}.$$

Equivalently, using *vec* operator and Kronecker product operator (see Appendix B for definitions of the two operators), we formulate the univariate-regression-type equations as

$$\boldsymbol{y} = (\boldsymbol{X}' \otimes \boldsymbol{I}_K)\boldsymbol{\phi} + \boldsymbol{\epsilon}, \tag{4.15}$$

where $\boldsymbol{y}$ and $\boldsymbol{\epsilon}$ are $K(T-h)\times 1$ vectors defined as

$$\boldsymbol{y} = vec(\boldsymbol{Y}) = \left(\boldsymbol{y}'_{h+1},\, \boldsymbol{y}'_{h+2},\, \cdots, \boldsymbol{y}'_T\right)', \tag{4.16}$$

$$\boldsymbol{e} = vec(\boldsymbol{E}) = \left(\boldsymbol{\epsilon}'_{h+1},\, \boldsymbol{\epsilon}'_{h+2},\, \cdots, \boldsymbol{\epsilon}'_T\right)', \tag{4.17}$$

and $\boldsymbol{\phi}$ is a $(hK^2)\times 1$ vector defined as

$$\begin{aligned} \boldsymbol{\phi} &= (\phi_1,\, \cdots,\, \phi_l,\, \cdots,\, \phi_{hK^2})' && (4.18) \\ &= vec(\boldsymbol{\Phi}) = (vec(\boldsymbol{\Phi}_1)',\, vec(\boldsymbol{\Phi}_2)',\, \cdots,\, vec(\boldsymbol{\Phi}_h)')' \\ &= \left(\phi_{11,1},\, \cdots,\, \phi_{KK,1}, \phi_{11,2},\, \cdots,\, \phi_{KK,2},\, \cdots, \phi_{ij,k},\, \cdots,\, \phi_{11,h},\, \cdots,\, \phi_{KK,h}\right)'. && (4.19) \end{aligned}$$

Note that the index l in (4.18) corresponds to the l-th element of the vector $\boldsymbol{\phi}$, and the index (ij,k) in (4.19) corresponds to the (i,j)-th element of the matrix $\boldsymbol{\Phi}_k$. The relation between (i,j,k) and l is bijective and defined by

$$l = f(i,j,k) = (k-1)K^2 + (j-1)K + i \tag{4.20}$$

where $l = 1,2,\, \cdots,\, (hK^2)$, $i,j = 1,2,\, \cdots,\, K$, and $k = 1,2,\, \cdots,\, h$.

We actually estimate the extended true parameter vector, $\boldsymbol{\Phi}^*$ or $\boldsymbol{\phi}^*$ defined as

$$\boldsymbol{\Phi}^* = (\boldsymbol{\Phi}_1^*, \cdots, \boldsymbol{\Phi}_p^*, \boldsymbol{\Phi}_{p+1}^*, \cdots, \boldsymbol{\Phi}_h^*)',$$

where

$$\mathbf{\Phi}_j^* = \begin{cases} \mathbf{\Phi}_j^o & \text{if } j \le p \\ \mathbf{0} & \text{if } p < j \le h \end{cases},$$

or

$$\begin{aligned} \boldsymbol{\phi}^* &\equiv \left(\phi_1^*,\, \phi_2^*,\, \cdots,\, \phi_{hK^2}^*\right) \\ &= vec(\mathbf{\Phi}^*) = \left(vec(\mathbf{\Phi}_1^*)',\, vec(\mathbf{\Phi}_2^*)',\, \cdots,\, vec(\mathbf{\Phi}_h^*)'\right)' \\ &= \left(\phi_{11,1}^*,\, \cdots,\, \phi_{KK,1}^*,\, \cdots,\, \phi_{11,p}^*,\, \cdots,\, \phi_{KK,p}^*,\, \cdots,\, \phi_{11,h}^*,\, \cdots,\, \phi_{KK,h}^*\right)' \\ &= \left(\phi_{11,1}^o,\, \cdots,\, \phi_{KK,1}^o,\, \cdots,\, \phi_{11,p}^o,\, \cdots,\, \phi_{KK,p}^o,\, 0,\, \cdots,\, 0\right)'. \end{aligned} \tag{4.21}$$

It is clear that under appropriate assumptions on the initial values for the VAR(p) and VAR(h) processes, the VAR(p) with the fixed true parameters $\mathbf{\Phi}^o$,

$$\boldsymbol{y}_t = \sum_{j=1}^{p} \mathbf{\Phi}_j^o \boldsymbol{y}_{t-j} + a_t, \;\; t = 1, \cdots, T,$$

and the AR(h) with the fixed extended true parameters $\mathbf{\Phi}^*$,

$$\boldsymbol{y}_t = \sum_{j=1}^{h} \mathbf{\Phi}_j^* \boldsymbol{y}_{t-j} + a_t, \;\; t = 1, \cdots, T$$

are equivalent.

Definition (**Entrywise norm**). For an $m \times n$ matrix A, its entrywise p-norm, denoted as $\|A\|_p$, is defined as

$$\|A\|_p = \|\text{vec}(A)\|_p = \left(\sum_{i=1}^{m} \sum_{j=1}^{n} |a_{ij}|^p\right)^{1/p}.$$

The Frobenius norm, which is the spacial case $p = 2$, is defined as

$$\|A\|_F = \sqrt{\sum_{i=1}^{m} \sum_{j=1}^{n} |a_{ij}|^2}$$

Definition (**The doubly adaptive LASSO**). The doubly adaptive LASSO or PLAC-weighted adaptive LASSO estimator $\hat{\boldsymbol{\phi}}_T^{daL}$ for $\boldsymbol{\phi}^*$ is defined as

$$\hat{\boldsymbol{\phi}}^{daL} = \arg\min_{\boldsymbol{\phi}} \left\{ \left\| \boldsymbol{y} - (\boldsymbol{X}' \otimes \boldsymbol{I}_K)\boldsymbol{\phi} \right\|^2 + \lambda_T \sum_{k=1}^{h} \sum_{i=1}^{K} \sum_{j=1}^{K} \hat{w}_{ij,k} \left| \phi_{ij,k} \right| \right\}, \tag{4.22}$$

where

$$\hat{w}_{ij,k} = \frac{1}{\left|\tilde{\phi}_{ij,k}\right|^{\gamma_1}\left(\sum_{s=k}^{h}\left\|\widehat{\boldsymbol{P}}(s)\right\|_{\gamma_0}^{\gamma_0}\right)^{\gamma_2}} = \frac{1}{\left|\tilde{\phi}_{ij,k}\right|^{\gamma_1} A_k^{\gamma_2}}, \tag{4.23}$$

$$A_k = \sum_{s=k}^{h}\left\|\widehat{\boldsymbol{P}}(s)\right\|_{\gamma_0}^{\gamma_0}, \tag{4.24}$$

$\tilde{\phi}_{ij,k}$ is the ordinary least squares estimate or any other consistent estimate for $\phi_{ij,k}$, $\left\|\widehat{\boldsymbol{P}}(s)\right\|_{\gamma_0} = \left(\sum_{i=1}^{K}\sum_{j=1}^{K}|\widehat{P}_{ij}(s)|^{\gamma_0}\right)^{1/\gamma_0}$ is the entrywise γ_0-norm of the sample partial lag autocorrelation matrix $\widehat{\boldsymbol{P}}(s)$ at lag s, and $\gamma_0 > 0$, $\gamma_1 \geq 0$, and $\gamma_2 \geq 0$ are some fixed constants, and h is the maximum lag we initially set.

Remarks:

(**1**) Both the LASSO (Tibshirani, 1996) and the adaptive LASSO (Zou, 2006) are special cases of the doubly adaptive LASSO. In former case, $\gamma_1 = \gamma_2 = 0$, and in latter case, $\gamma_2 = 0$.

(**2**) In the doubly adaptive LASSO procedure the PLAC information and the Y-W or OLS estimates of the VAR(h) model work in tandem to perform subset selection and parameter estimation simultaneously. The basic idea can be elucidated from the following points:

Firstly, note that $A_1 \geq \cdots \geq A_p \geq \cdots \geq A_h$. Hence, $w_{ij,k}$ is decreasing with increasing k. Therefore monotonically increasing penalties are imposed on $\phi_{ij,k}$'s as k increases from 1 to h. Consequently, depending on the structure of the PLAC, an VAR term with smaller lag is therefore more likely to be included in the model.

Secondly, due to the cutoff property of the PLAC, namely, the value of $\|\hat{\boldsymbol{P}}(s)\|$ for $s = p+1, p+2\cdots, h$ are relatively tiny, if k goes from h backwards to p, it is expected that the A_k will exhibit a sharp jump at $k = p$. Consequently, the VAR terms with lags greater than p get much more penalties compared to those with $k \leq p$. so that they are more likely to be excluded from the model, and the true order of the VAR process is thus automatically identified.

Finally, $|\tilde{\phi}_{ij,k}|^{\gamma_1}$ imposes larger penalty on $\phi_{ij,k}$ if the corresponding VAR term is not significant. This is obvious because if an VAR term is not important, the consistently estimated value

of the corresponding coefficient is close to zero, and the penalty is close to ∞. Consequently, the insignificant VAR terms get more penalties so that they are more likely to be excluded from the model whereas the significant VAR terms are more likely to be included in the model.

4.3.2 The adaptive LASSO when p is known

Suppose that we observe a time series $\boldsymbol{y}_1, \boldsymbol{y}_2, \cdots, \boldsymbol{y}_T$, which is a realization of a stationary K-VAR process with the true order p known or has been identified and true parameters $\boldsymbol{\Phi}^o = (\boldsymbol{\Phi}_1^o, \cdots, \boldsymbol{\Phi}_p^o)$ unknown. Since the initial values $\boldsymbol{y}_0, \cdots, \boldsymbol{y}_{-p+1}$ are not available, we may use $\boldsymbol{y}_1, \cdots, \boldsymbol{y}_p$ as a presample. This will reduce the effective sample size from T to $T-p$. We set $h = p$ and $\gamma_2 = 0$ in (4.23). The PLAC-weighted adaptive LASSO reduces to the adaptive LASSO.

4.4 The asymptotic properties of the doubly adaptive LASSO

The adaptive LASSO and the doubly adaptive LASSO methods yield biased estimators. In this section, however, we show that with properly chosen values for γ_0, γ_1, and γ_2 in (4.23), together with a proper choice of λ_T, the doubly adaptive LASSO enjoys desirable asymptotic properties. We actually study the asymptotic properties of the doubly adaptive LASSO estimator for the extended true parameter vector $\boldsymbol{\phi}^*$ in (4.21) instead of $\boldsymbol{\phi}^o$ in (4.9).

First, we clarify notations. Let $\mathbb{S}$ be the set of the true nonzero coefficient, i.e. $\mathbb{S} = \{l : \phi_l^* \neq 0\} = \text{supp}(\boldsymbol{\phi}^*) \subset \{1, 2, \cdots, hK^2\}$ with h being set large enough such that $h > p$. Let $\mathbb{S}^c = \{1, 2, \cdots, hK^2\} \setminus \mathbb{S}$. Let $s = |\mathbb{S}|$ be the cardinality of the set $\mathbb{S}$. The assumption of the model sparsity implies that $s < pK^2$. Let $\tilde{\phi}_l$ be any consistent estimate for the true ϕ_l^*, say the OLS or Yule-Walker estimate. Let $\hat{\phi}_{T,l}^{daL}$ be the doubly adaptive LASSO estimate for ϕ_l^*. Let $\hat{\mathbb{S}}_T = \{l : \hat{\phi}_{T,l}^{daL} \neq 0\}$ and $\hat{\mathbb{S}}_T^c = \{1, 2, \cdots, hK^2\} \setminus \hat{\mathbb{S}}_T$. Let $\boldsymbol{\phi}_{\mathbb{S}}^*$ be the s-dimensional vector for true underlying nonzero parameters, and $\boldsymbol{\phi}_{\mathbb{S}^c}^*$ be the vector for true underlying null parameters, i.e. $\boldsymbol{\phi}_{\mathbb{S}}^* = \{\phi_l^* : l \in \mathbb{S}\}$ and $\boldsymbol{\phi}_{\mathbb{S}^c}^* = \{\phi_l^* : l \in \mathbb{S}^c\}$. Let $\hat{\boldsymbol{\phi}}_{T,\mathbb{S}}^{daL}$ be the vector for the PAC-weighted adaptive LASSO estimate for $\boldsymbol{\phi}_{\mathbb{S}}^*$ and $\hat{\boldsymbol{\phi}}_{T,\mathbb{S}^c}^{daL}$ the vector for PAC-weighted adaptive LASSO estimate for null vector $\boldsymbol{\phi}_{\mathbb{S}^c}^*$, i.e. $\hat{\boldsymbol{\phi}}_{T,\mathbb{S}}^{daL} = \{\hat{\phi}_{T,l}^{daL} : l \in \mathbb{S}\}$ and $\hat{\boldsymbol{\phi}}_{T,\mathbb{S}^c}^{daL} = \{\hat{\phi}_{T,l}^{daL} : l \in \mathbb{S}^c\}$. Let $\hat{\boldsymbol{\phi}}_{\hat{\mathbb{S}}_T}^{daL}$ be the vector for nonzero estimates from

the doubly adaptive LASSO and $\hat{\boldsymbol{\phi}}^{daL}_{\hat{\mathbb{S}}^c_T}$ the vector for null estimates, i.e. $\hat{\boldsymbol{\phi}}^{daL}_{\hat{\mathbb{S}}_T} = \{\hat{\phi}^{daL}_{T,l} : l \in \hat{\mathbb{S}}_T\}$ and $\hat{\boldsymbol{\phi}}^{daL}_{\hat{\mathbb{S}}^c_T} = \{\hat{\phi}^{daL}_{T,l} : l \in \hat{\mathbb{S}}^c_T\}$.

Proposition 4.4.1 *(**The condition for the ergodic stationarity**). The VAR(p) process specified by (4.1) is ergodic stationary if and only if the corresponding characteristic equation satisfies the stability condition, namely,*

$$\det(I - \boldsymbol{\Phi}_1 z - \cdots - \boldsymbol{\Phi}_p z^p) \neq 0$$

for $|z| \leq 1$.

See Lütkepohl (2006) p.14-16 for proof.

Let $\boldsymbol{\Gamma}$ be the covariance matrix of $\boldsymbol{x}_t$ in (4.12), namely,

$$\boldsymbol{\Gamma} = E[\boldsymbol{x}_t \boldsymbol{x}_t'] = \begin{pmatrix} \Gamma(0) & \Gamma(-1) & \cdots & \Gamma(-h+1) \\ \Gamma(1) & \Gamma(0) & \cdots & \Gamma(-h+2) \\ \vdots & \vdots & & \vdots \\ \Gamma(h-1) & \Gamma(h-2) & \cdots & \Gamma(0) \end{pmatrix}_{(hK)\times(hK)},$$

where $\Gamma(s)$ is covariance matrix of $\boldsymbol{y_t}$. Note that $\boldsymbol{\Gamma}$ is symmetric whereas $\Gamma(s)$ is not symmetric. Instead, $\Gamma(s)' = \Gamma(-s)$. We can partition $\boldsymbol{\Gamma}$ as follows

$$\boldsymbol{\Gamma} = \begin{pmatrix} \boldsymbol{\Gamma}_{\mathbb{S}\mathbb{S}} & \boldsymbol{\Gamma}_{\mathbb{S}\mathbb{S}^c} \\ \boldsymbol{\Gamma}_{\mathbb{S}^c\mathbb{S}} & \boldsymbol{\Gamma}_{\mathbb{S}^c\mathbb{S}^c} \end{pmatrix},$$

where we retain the ordering according to the lag index of $\boldsymbol{x}_t$ within each partition.

Assumptions:

A0: The coefficients matrix $\boldsymbol{\Phi}$ defined in (4.11) belongs to a compact set.

A1: For all $\boldsymbol{\Phi}$, $\det(I - \boldsymbol{\Phi}_1 z - \cdots - \boldsymbol{\Phi}_h z^h) \neq 0$ for $|z| \leq 1$.

A2: $\boldsymbol{\epsilon}_t = (\epsilon_1, \cdots, \epsilon_K)'$ is a strong white noise process, i.e. $E[\boldsymbol{\epsilon}_t] = \mathbf{0}$, $E[\boldsymbol{\epsilon}_t \boldsymbol{\epsilon}_t'] = \Sigma_\epsilon \succ 0$, ϵ_t and ϵ_s are independent for $s \neq t$, and $E|\epsilon_{it}\epsilon_{jt}\epsilon_{kt}\epsilon_{lt}| < M < \infty$ for $i,\ j,\ k,\ l = 1, \cdots, K$.

A3: The submatrix $\boldsymbol{\Gamma}_{\mathbb{S}\mathbb{S}}$ is not singular and therefore invertible.

Remarks on assumptions:

1) **A0** is always assumed.

2) **A1** ensures that $\boldsymbol{x}_t'$) is ergodic stationary

3) **A2** requires the existence of finite fourth moments of $\{\boldsymbol{y}_t\}$.

4) No normality of $\boldsymbol{\epsilon}_t$ is assumed.

5) **A2** guarantees the existence of the covariance matrix $\boldsymbol{\Gamma}$.

Lemma 4.4.2 [3] *Under **A1** – **A2**, we have*

(i) $\frac{1}{T-h}\boldsymbol{X}\boldsymbol{X}' \xrightarrow{a.s.} \boldsymbol{\Gamma}$,

(ii) $\frac{1}{T-h}(\boldsymbol{X}\otimes I_K)\boldsymbol{e} \xrightarrow{a.s.} \boldsymbol{0}$, *and*

(iii) $\frac{1}{\sqrt{T-h}}(\boldsymbol{X}\otimes I_K)\boldsymbol{e} \xrightarrow{D} \boldsymbol{w} \sim N(\boldsymbol{0}, \boldsymbol{\Gamma}\otimes\Sigma_\epsilon)$,

where $\otimes$ denotes the Kronecker product.

Proof (i) It is easy to check that $\boldsymbol{X}\boldsymbol{X}' = \sum_{t=h}^{T-1} \boldsymbol{x}_t\boldsymbol{x}_t'$. By **A1**, $\boldsymbol{x}_t$ is ergodic stationary. By Theorem A.3.1 for ergodicity of functions, $\boldsymbol{x}_t\boldsymbol{x}_t'$ is also ergodic stationary. By Ergodic Theorem A.3.2, we have

$$\frac{1}{T-h}\boldsymbol{X}\boldsymbol{X}' \xrightarrow{a.s.} E[\boldsymbol{x}_t\boldsymbol{x}_t'] = \boldsymbol{\Gamma}.$$

(ii) It is not very hard to check that $(\boldsymbol{X}\otimes I_K)\boldsymbol{e} = \sum_{t=h+1}^{T}(\boldsymbol{x}_{t-1}\otimes I_K)\boldsymbol{\epsilon}_t$. Since $\boldsymbol{x}_t$ is ergodic stationary by **A1**, so is $(\boldsymbol{x}_{t-1}\otimes I_K)\boldsymbol{\epsilon}_t$ by Theorem A.3.1 for ergodicity of functions. By Ergodic Theorem A.3.2, we have

$$\frac{1}{T-h}(\boldsymbol{X}\otimes I_K)\boldsymbol{e} \xrightarrow{a.s.} E[(\boldsymbol{x}_{t-1}\otimes I_K)\boldsymbol{\epsilon}_t],$$

where $E[(\boldsymbol{x}_{t-1}\otimes I_K)\boldsymbol{\epsilon}_t] = E\left[[(\boldsymbol{x}_{t-1}\otimes I_K)\boldsymbol{\epsilon}_t|\mathcal{F}_{t-1}]\right] = (\boldsymbol{x}_{t-1}\otimes I_K)E[\boldsymbol{\epsilon}_t|\mathcal{F}_{t-1}] = \boldsymbol{0}$.

(iii) Let $\boldsymbol{\nu}_t = (\boldsymbol{x}_{t-1}\otimes I_K)\boldsymbol{\epsilon}_t$. Then $\{\boldsymbol{\nu}_t\}$ is a vector martingale difference because $E[\boldsymbol{\nu}_t|\mathcal{F}_{t-1}] = \boldsymbol{0}$. By **A1**, **A2**, and Theorem A.4.1, the CLT for the MDS (Billingsley, 1961), we have

$$\frac{1}{\sqrt{T-h}}\sum_{t=h+1}^{T}\boldsymbol{\nu}_t \xrightarrow{D} N(\boldsymbol{0},\ \Sigma_\nu),$$

where $\Sigma_\nu = \mathrm{Var}[\boldsymbol{\nu}_t] = \mathrm{Var}[(\boldsymbol{x}_{t-1}\otimes I_K)\boldsymbol{\epsilon}_t] = E[(\boldsymbol{x}_{t-1}\otimes I_K)\boldsymbol{\epsilon}_t\boldsymbol{\epsilon}_t'(\boldsymbol{x}_{t-1}'\otimes I_K)] = \boldsymbol{\Gamma}\otimes\Sigma_\epsilon$. ∎

Definition (**Estimation consistency**). The PLAC-weighted adaptive LASSO estimator $\hat{\boldsymbol{\phi}}_T^{daL}$ is said to be consistent for $\boldsymbol{\phi}^*$ if

$$\|\,\hat{\boldsymbol{\phi}}_T^{daL} - \boldsymbol{\phi}^*\,\| \xrightarrow{P} 0 \ \text{ as } \ T\to\infty.$$

[3] Lütkepohl (1996) p.73 states the lemma without proof.

Theorem 4.4.3 ***(Estimation Consistency of $\hat{\boldsymbol{\phi}}_T^{daL}$).*** *Let $a_T = \sqrt{T-h}\min_{l\in\mathbb{S}}\left(|\tilde{\phi}_l|^{\gamma_1}A_l^{\gamma_2}\right)$, where $\left(|\tilde{\phi}_l|^{\gamma_1}A_l^{\gamma_2}\right)$ corresponds to $\left(|\tilde{\phi}_{(i,j,k)}|^{\gamma_1}A_k^{\gamma_2}\right)$ by the bijective function (4.20). If $\lambda_T = o_p(a_T)$, then under **A0 – A2** we must satisfy:*

$$\| \hat{\boldsymbol{\phi}}_T^{daL} - \boldsymbol{\phi}^* \| \xrightarrow{P} 0 \ \ as \ \ T \to \infty,$$

as $T \to \infty$.

Proof Let $\Psi_T(\boldsymbol{\phi})$ be defined as

$$\Psi_T(\boldsymbol{\phi}) = \| \boldsymbol{y} - (\boldsymbol{X}' \otimes I_K)\boldsymbol{\phi}) \|^2 + \lambda_T \sum_{l=1}^{hK^2} \hat{w}_{T,l}|\phi_l|,$$

where $\boldsymbol{X}$ is defined in (4.14) and $\boldsymbol{y}$ in (4.16).Following Fan and Li (2001), we show that for every $\epsilon > 0$ there exists a sufficiently large C such that

$$\mathbb{P}\left(\inf_{\|\boldsymbol{u}\|\geq C} \Psi_T\left(\boldsymbol{\phi}^* + \boldsymbol{u}/\sqrt{T-h}\right) > \Psi_T(\boldsymbol{\phi}^*)\right) \geq 1-\epsilon,$$

which implies that with probability at least $1-\epsilon$ that there exists a minimum in the ball $\{\boldsymbol{\phi}^* + \boldsymbol{u}/\sqrt{T-h} : \|\boldsymbol{u}\| \leq C\}$. Hence there exists a local minimizer such that $\| \hat{\boldsymbol{\phi}}_T^{daL} - \boldsymbol{\phi}^* \| = O_p(T^{-1/2})$. Observe that

$$\begin{aligned}
&\Psi_T\left(\boldsymbol{\phi}^* + \boldsymbol{u}/\sqrt{T-h}\right) - \Psi_T(\boldsymbol{\phi}^*) \\
&= \left\|\boldsymbol{y} - (\boldsymbol{X}' \otimes I_K)\left(\boldsymbol{\phi}^* + \boldsymbol{u}/\sqrt{T-h}\right)\right\|^2 - \left\|\boldsymbol{y} - (\boldsymbol{X}' \otimes I_K)\boldsymbol{\phi}^*\right\|^2 + \lambda_T \sum_{l=1}^{hK^2} \hat{w}_{T,l}\left(\left|\phi_l^* + \frac{u_l}{\sqrt{T-h}}\right| - \left|\phi_l^*\right|\right) \\
&= \boldsymbol{u}'\left(\frac{1}{T-h}(\boldsymbol{X}\boldsymbol{X}' \otimes I_K)\right)\boldsymbol{u} - 2\boldsymbol{u}'\left(\frac{1}{\sqrt{T-h}}(\boldsymbol{X} \otimes I_K)\boldsymbol{e}\right) + \lambda_T \sum_{l=1}^{hK^2} \hat{w}_{T,l}\left(\left|\phi_l^* + \frac{u_l}{\sqrt{T-h}}\right| - \left|\phi_l^*\right|\right) \\
&= \boldsymbol{u}'\left(\frac{1}{T-h}(\boldsymbol{X}\boldsymbol{X}' \otimes I_K)\right)\boldsymbol{u} - 2\boldsymbol{u}'\left(\frac{1}{\sqrt{T-h}}(\boldsymbol{X} \otimes I_K)\boldsymbol{e}\right) + \lambda_T \sum_{l\in\mathbb{S}} \hat{w}_{T,l}\left(\left|\phi_l^* + \frac{u_l}{\sqrt{T-h}}\right| - \left|\phi_l^*\right|\right) + \lambda_T \sum_{l\notin\mathbb{S}} \hat{w}_{T,l}\frac{|u_l|}{\sqrt{T-h}} \\
&\geq \boldsymbol{u}'\left(\frac{1}{T-h}(\boldsymbol{X}\boldsymbol{X}' \otimes I_K)\right)\boldsymbol{u} - 2\boldsymbol{u}'\left(\frac{1}{\sqrt{T-h}}(\boldsymbol{X} \otimes I_K)\boldsymbol{e}\right) + \lambda_T \sum_{l\in\mathbb{S}} \hat{w}_{T,l}\left(\left|\phi_l^* + \frac{u_l}{\sqrt{T-h}}\right| - \left|\phi_l^*\right|\right) \\
&\geq \boldsymbol{u}'\left(\frac{1}{T-h}(\boldsymbol{X}\boldsymbol{X}' \otimes I_K)\right)\boldsymbol{u} - 2\boldsymbol{u}'\left(\frac{1}{\sqrt{T-h}}(\boldsymbol{X} \otimes I_K)\boldsymbol{e}\right) - \lambda_T \sum_{l\in\mathbb{S}} \hat{w}_{T,l}\frac{|u_l|}{\sqrt{T-h}} \ .
\end{aligned}$$

First, consider the third term, which can be expressed as

$$\begin{aligned}
\lambda_T \sum_{l=1}^{hK^2} \hat{w}_{T,l}\frac{|u_l|}{\sqrt{T-h}} &= \frac{\lambda_T}{\sqrt{T-h}} \sum_{l\in\mathbb{S}} \left|\tilde{\phi}_l\right|^{-\gamma_1} A_l^{-\gamma_2}|u_l| \\
&\leq \frac{\lambda_T}{\sqrt{T-h}}\left(\min_{l\in\mathbb{S}}\left(|\tilde{\phi}_l|^{\gamma_1}A_l^{\gamma_2}\right)\right)^{-1} \| \boldsymbol{u} \| \\
&= \frac{\lambda_T}{a_T} \| \boldsymbol{u} \| = o_p(1) \| \boldsymbol{u} \| .
\end{aligned}$$

For the second term, by Lemma (4.4.2) (iii), we have

$$\boldsymbol{u}'\left(\frac{1}{\sqrt{T-h}}\right)(\boldsymbol{X}'\otimes I_K)'\boldsymbol{e} = \boldsymbol{u}' o_P(\mathbf{1}) \leq o_p(1)\parallel \boldsymbol{u}\parallel .$$

Fir the first term, by Lemma (4.4.2) (i), we have

$$\left(\frac{1}{T-h}(\boldsymbol{X}\boldsymbol{X}'\otimes I_K)\right)\rightarrow(\boldsymbol{\Gamma}\otimes I_K) \text{ a.s..}$$

So the first term is a quadratic form in $\boldsymbol{u}$.

Then it follows that in probability

$$\Psi_T\left(\boldsymbol{\phi}^*+\boldsymbol{u}/\sqrt{T-h}\right)-\Psi_T(\boldsymbol{\phi}^*)\geq \boldsymbol{u}^T(\boldsymbol{\Gamma}\otimes I_K)\boldsymbol{u}-2o_p(1)\parallel\boldsymbol{u}\parallel,$$

as $T\rightarrow\infty$. Therefore, for any $\epsilon>0$, there exists a sufficiently large C such that the term of quadratic term dominates the other terms with probability $\geq 1-\epsilon$. ∎

Proposition 4.4.4 *Let* $a_T=\sqrt{T-h}\min_{l\in\mathbb{S}}\left(|\tilde{\phi}_l|^{\gamma_1}A_l^{\gamma_2}\right)$, *and* $b_T=\sqrt{T-h}\max_{l\in\mathbb{S}^c}\left(|\tilde{\phi}_l|^{\gamma_1}A_l^{\gamma_2}\right)$, *where* $\left(|\tilde{\phi}_l|^{\gamma_1}A_l^{\gamma_2}\right)$ *corresponds to* $\left(|\tilde{\phi}_{(ij,k)}|^{\gamma_1}A_k^{\gamma_2}\right)$ *by the bijective function (4.20). If* $\lambda_T=o_p(a_T)$ *and* $\lambda_T/b_T\xrightarrow{P}\infty$, *then under* ***A0*** *–* ***A3****, we have*

$$\begin{cases}\sqrt{T-h}\left(\hat{\boldsymbol{\phi}}^{daL}_{T,\mathbb{S}}-\boldsymbol{\phi}^*_{\mathbb{S}}\right)\xrightarrow{D} \quad N\left(\mathbf{0},\ (\boldsymbol{\Gamma}_{\mathbb{S}\mathbb{S}})^{-1}\otimes\Sigma_\epsilon\right),\\ \sqrt{T-h}\left(\hat{\boldsymbol{\phi}}^{daL}_{T,\mathbb{S}^c}-\boldsymbol{\phi}^*_{\mathbb{S}^c}\right)\xrightarrow{D}\mathbf{0}.\end{cases}$$

Proof We follow the methodology of Knight and Fu (2000) and Zou (2006).

Let $\boldsymbol{\phi}=\boldsymbol{\phi}^*+\boldsymbol{u}/\sqrt{T-h}$ and define

$$\Psi_T(\boldsymbol{u})=\left\|\boldsymbol{y}-(\boldsymbol{X}'\otimes I_K)\left(\boldsymbol{\phi}^*+\frac{\boldsymbol{u}}{\sqrt{T-h}}\right)\right\|^2+\lambda_T\sum_{j=1}^{h}\hat{w}_{T,j}\left|\phi_j^*+\frac{u_j}{\sqrt{T-h}}\right|,$$

where $\boldsymbol{X}$ is defined by (4.14) and $\boldsymbol{y}$ by (4.16). Define the reparameterized objective function as

$$V_T(\boldsymbol{u})=\Psi_T(\boldsymbol{u})-\Psi_T(\mathbf{0}).$$

Then the minimizing objective is equivalent to minimizing $V_T(\boldsymbol{u})$ with respect to $\boldsymbol{u}$. Let $\hat{\boldsymbol{u}}_T=\arg\min V_T(\boldsymbol{u})$, then

$$\hat{\boldsymbol{\phi}}^{daL}_T=\boldsymbol{\phi}^*+\hat{\boldsymbol{u}}_T/\sqrt{T-h},$$

or

$$\hat{\boldsymbol{u}}_T = \sqrt{T-h}\left(\hat{\boldsymbol{\phi}}_T^{daL} - \boldsymbol{\phi}^*\right).$$

Observe that

$$V_T(\boldsymbol{u}) = \boldsymbol{u}'\left(\frac{1}{T-h}(\boldsymbol{X}\boldsymbol{X}'\otimes I_K)\right)\boldsymbol{u} - 2\boldsymbol{u}'\left(\frac{1}{\sqrt{T-h}}(\boldsymbol{X}\otimes I_K)\boldsymbol{e}\right) + \frac{\lambda_T}{\sqrt{T-h}}\sum_{l=1}^{hK^2}\hat{w}_{T,l}\sqrt{T-h}\left(\left|\phi_l^* + \frac{u_l}{\sqrt{T-h}}\right| - \left|\phi_l^*\right|\right).$$

By Lemma (4.4.2) we have $\left(\frac{1}{T-h}(\boldsymbol{X}\boldsymbol{X}'\otimes I_K)\right) \xrightarrow{a.s.} (\boldsymbol{\Gamma}\otimes I_K)$,and $\frac{1}{\sqrt{T-h}}(\boldsymbol{X}\otimes I_K)\boldsymbol{e} \xrightarrow{D} \boldsymbol{w} \sim N(\boldsymbol{0}, \boldsymbol{\Gamma}\otimes\Sigma_\epsilon)$. Consider the limiting behaviour of the third term. First, by the conditions required in the theorem, we have $\lambda_T\hat{w}_{T,l}/\sqrt{T-h} \le \lambda_T/\left(\sqrt{T-h}\min_{l\in\mathbb{S}}\left(\left|\tilde{\phi}_l\right|^{\gamma_1}A_l^{\gamma_2}\right)\right) = \lambda_T/a_T \xrightarrow{P} 0$ for $l\in\mathbb{S}$ and $\frac{\lambda_T}{\sqrt{T-h}}w_{T,l} = \frac{\lambda_T}{\sqrt{T-h}}|\tilde{\phi}_l|^{-\gamma_1}A_l^{-\gamma_2} \ge \lambda_T/\left(\sqrt{T-h}\max_{l\notin\mathbb{S}}\left(\left|\tilde{\phi}_l\right|^{\gamma_1}A_l^{\gamma_2}\right)\right) = \lambda_T/b_T \xrightarrow{P} \infty$ for $l\notin\mathbb{S}$. In summary, we have

$$\frac{\lambda_T}{\sqrt{T-h}}\hat{w}_{T,l} = \frac{\lambda_T}{\sqrt{T-h}\left|\tilde{\phi}_l\right|^{\gamma_1}A_l^{\gamma_2}} \xrightarrow{P} \begin{cases} 0 & \text{if } l\in\mathbb{S} \\ \infty & \text{if } l\notin\mathbb{S} \end{cases}.$$

Secondly, we have

$$\sqrt{T-h}\left(\left|\phi_l^* + \frac{u_l}{\sqrt{T-h}}\right| - \phi_l^*\right) \to \begin{cases} u_l\text{sgn}(\phi_l^*) & \text{if } l\in\mathbb{S}\ \ (\phi_l^*=0) \\ |u_l| & \text{if } l\notin\mathbb{S}\ \ (\phi_l^*\neq 0) \end{cases}.$$

By Slutsky's theorem, we have the following limiting behaviour of the third term

$$\frac{\lambda_T}{\sqrt{T-h}}\hat{w}_{T,l}\sqrt{T-h}\left(\left|\phi_l^* + \frac{u_l}{\sqrt{T-h}}\right| - \left|\phi_l^*\right|\right) \xrightarrow{P} \begin{cases} 0 & \text{if } \forall l\in\mathbb{S} \\ 0 & \text{if } u_l = 0,\ \ \forall l\notin\mathbb{S} \\ \infty & \text{otherwise} \end{cases}.$$

Thus, we have $V_T(\boldsymbol{u}) \to V(\boldsymbol{u})$ for every $\boldsymbol{u}$, where

$$\begin{aligned} V(\boldsymbol{u}) &= \begin{pmatrix}\boldsymbol{u}_{\mathbb{S}}' & \boldsymbol{u}_{\mathbb{S}^c}'\end{pmatrix}\begin{pmatrix}(\boldsymbol{\Gamma}\otimes I_K)_{\mathbb{S}\mathbb{S}} & (\boldsymbol{\Gamma}\otimes I_K)_{\mathbb{S}\mathbb{S}^c} \\ (\boldsymbol{\Gamma}\otimes I_K)_{\mathbb{S}^c\mathbb{S}} & (\boldsymbol{\Gamma}\otimes I_K)_{\mathbb{S}^c\mathbb{S}^c}\end{pmatrix}\begin{pmatrix}\boldsymbol{u}_{\mathbb{S}} \\ \boldsymbol{u}_{\mathbb{S}^c}\end{pmatrix} - 2\begin{pmatrix}\boldsymbol{u}_{\mathbb{S}}' & \boldsymbol{u}_{\mathbb{S}^c}'\end{pmatrix}\begin{pmatrix}\boldsymbol{w}_{\mathbb{S}} \\ \boldsymbol{w}_{\mathbb{S}^c}\end{pmatrix} \\ &\quad + \sum_{l\in\mathbb{S}^c}\frac{\lambda_T}{\sqrt{T-h}}\hat{w}_{T,l}\sqrt{T-h}\left(\left|\phi_l^* + \frac{u_l}{\sqrt{T-h}}\right| - \left|\phi_l^*\right|\right) \\ &= \begin{cases}\boldsymbol{u}_{\mathbb{S}}'(\boldsymbol{\Gamma}_{\mathbb{S}\mathbb{S}}\otimes I_K)\boldsymbol{u}_{\mathbb{S}} - 2\boldsymbol{u}_{\mathbb{S}}'\boldsymbol{w}_{\mathbb{S}} & \text{if } \boldsymbol{u}_{\mathbb{S}^c} = \boldsymbol{0} \\ \infty & \text{otherwise}\end{cases}. \end{aligned}$$

$V_T(\boldsymbol{u})$ is convex with the unique minimum $\left(((\boldsymbol{\Gamma}_{\mathbb{S}\mathbb{S}})^{-1}\otimes I_K)\boldsymbol{w}_{\mathbb{S}},\ \ \boldsymbol{0}\right)'$. Following the epi-convergence results of Geyer (1994) and Knight and Fu (2000), $\text{argmin}_{\boldsymbol{u}}\, V_T(\boldsymbol{u}) \xrightarrow{D} \text{argmin}_{\boldsymbol{u}}\, V(\boldsymbol{u})$, we have

$$\begin{cases}\hat{\boldsymbol{u}}_{\mathbb{S}} \xrightarrow{D} \left((\boldsymbol{\Gamma}_{\mathbb{S}\mathbb{S}})^{-1}\otimes I_K\right)\boldsymbol{w}_{\mathbb{S}} \\ \hat{\boldsymbol{u}}_{\mathbb{S}^C} \xrightarrow{D} \boldsymbol{0}\end{cases},$$

or

$$\begin{cases} \sqrt{T-h}\left(\hat{\boldsymbol{\phi}}_{T,\mathbb{S}}^{daL} - \boldsymbol{\phi}_{\mathbb{S}}^{*}\right) \xrightarrow{D} N\left(\mathbf{0},\ (\boldsymbol{\Gamma}_{\mathbb{S}\mathbb{S}})^{-1} \otimes \Sigma_{\epsilon}\right) \\ \sqrt{T-h}\left(\hat{\boldsymbol{\phi}}_{T,\mathbb{S}^c}^{daL} - \boldsymbol{\phi}_{\mathbb{S}^c}^{*}\right) \xrightarrow{D} \mathbf{0} \end{cases}.$$

Corollary 4.4.5 *Let* $a_T = \sqrt{T-h}\min_{l\in\mathbb{S}}\left(|\tilde{\phi}_l|^{\gamma_1}A_l^{\gamma_2}\right)$, *and* $b_T = \sqrt{T-h}\max_{l\in\mathbb{S}^c}\left(|\tilde{\phi}_l|^{\gamma_1}A_l^{\gamma_2}\right)$, *where* $\left(|\tilde{\phi}_l|^{\gamma_1}A_l^{\gamma_2}\right)$ *corresponds to* $\left(|\tilde{\phi}_{(ijk)}|^{\gamma_1}A_k^{\gamma_2}\right)$ *by the bijective function (4.20). If* $\lambda_T = o_p(a_T)$ *and* $\lambda_T/b_T \xrightarrow{P} \infty$, *then under* ***A0*** $-$ ***A3***, *we have that*

$$\mathbb{P}\left(l \in \hat{\mathbb{S}}_T\right) \to 1 \ \text{ if } \ l \in \mathbb{S},$$

as $T \to \infty$.

Proof By Theorem A.5.1, the $\sqrt{T-h}$-normality of $\hat{\boldsymbol{\phi}}_{T,\mathbb{S}}^{daL}$ in Proposition 4.4.4 implies that $\|\hat{\boldsymbol{\phi}}_{T,\mathbb{S}}^{daL} - \boldsymbol{\phi}_{\mathbb{S}}^{*}\| = O_p\left(1/\sqrt{T-h}\right)$. Thus, $\hat{\boldsymbol{\phi}}_{T,\mathbb{S}}^{daL} \xrightarrow{P} \boldsymbol{\phi}_{\mathbb{S}}^{*}$, which implies that $\forall l \in \mathbb{S}$, we have $\mathbb{P}\left(l \in \hat{\mathbb{S}}_T\right) \to 1$, as $T \to \infty$. ∎

Fan and Li (2001) specified the oracle properties of a sparse estimator in the language of Donoho and Johnstone (1994). Heuristically, an oracle procedure can perform as well asymptotically as if the true submodel were known in advance. We extend the notion of the oracle properties of an estimator to the context of VAR times series models.

Definition (**Oracle properties**) . The doubly adaptive positive LASSO estimator $\hat{\boldsymbol{\phi}}_T^{daL}$ for $\boldsymbol{\phi}^*$ is said to have the oracle properties if, with probability tending to 1, it could (i) identify the true sparsity pattern, i.e. $\lim P(\hat{\mathbb{S}}_T = \mathbb{S}) = 1$, (ii) identify the true lag order of the VAR process, i,e, $\lim P(\hat{p}_T^{daL} = p) = 1$, and (iii) have an optimal estimation rate of the coefficients as $T \to \infty$.

The following theorem says that the doubly adaptive LASSO procedure is an oracle procedure.

Theorem 4.4.6 ***(Oracle properties of*** $\hat{\boldsymbol{\phi}}_T^{daL}$***).*** *Let* $a_T = \sqrt{T-h}\min_{l\in\mathbb{S}}\left(|\tilde{\phi}_l|^{\gamma_1}A_l^{\gamma_2}\right)$, *and* $b_T = \sqrt{T-h}\max_{l\in\mathbb{S}^c}\left(|\tilde{\phi}_l|^{\gamma_1}A_l^{\gamma_2}\right)$, *where* $\left(|\tilde{\phi}_l|^{\gamma_1}A_l^{\gamma_2}\right)$ *corresponds to* $\left(|\tilde{\phi}_{(ijk)}|^{\gamma_1}A_k^{\gamma_2}\right)$ *by the bijective function (4.20) If* $\lambda_T = o_p(a_T)$ *and* $\lambda_T/b_T \xrightarrow{P} \infty$, *then under* ***A0*** $-$ ***A3***, $\hat{\boldsymbol{\phi}}_T^{daL}$ *must satisfy:*

(i) Selection Consistency: $\mathbb{P}\left(\hat{\mathbb{S}}_T = \mathbb{S}\right) \to 1$,

(ii) Identification consistency: $\mathbb{P}\left(\hat{p}_T^{daL} = p\right) \to 1$, *and*

(iii) Asymptotic Normality: $\sqrt{T-h}\left(\hat{\boldsymbol{\phi}}_{\hat{\mathbb{S}}_T}^{daL} - \boldsymbol{\phi}_{\mathbb{S}}^*\right) \xrightarrow{D} N\left(\mathbf{0},\ (\boldsymbol{\Gamma}_{\mathbb{S}\mathbb{S}})^{-1} \otimes \Sigma_\epsilon\right)$,

as $T \to \infty$.

Proof (i) In view of Corollary 4.4.5, we know that $\forall\, j \in \mathbb{S},\ \ P(j \in \hat{\mathbb{S}}_T) \to 1$. So it suffices to show that $\forall m \notin \mathbb{S}$, $P(m \in \hat{\mathbb{S}}_T) \to 0$. Now, we follow the methodology of Zou (2006).

Consider the event $\{m \in \hat{\mathbb{S}}_T\}$. The KKT conditions entail that

$$2(\boldsymbol{X} \otimes I_K)_{(m,\cdot)}\left(\boldsymbol{y} - (\boldsymbol{X}' \otimes I_K)\hat{\boldsymbol{\phi}}_T^{daL}\right) = \lambda_T \hat{w}_{T,m} \operatorname{sgn}\left(\hat{\phi}_{T,m}^{daL}\right),$$

where the subscript $(m,\ \cdot)$ denotes the m-th row of a matrix, so $(\boldsymbol{X} \otimes I_K)_{(m,\cdot)}$ is the m-th row of $(T-h)K \times hK^2$ matrix $(\boldsymbol{X} \otimes I_K)$. If $\lambda_T / b_T \xrightarrow{P} \infty$, we have

$$\frac{\lambda_T}{\sqrt{T}} \hat{w}_{T,m} = \frac{\lambda_T}{\sqrt{T}} \frac{1}{|\tilde{\phi}_m|^{\gamma_1} A_m^{\gamma_2}} \geq \frac{\lambda_T}{b_T} \xrightarrow{P} \infty,$$

whereas

$$\frac{(\boldsymbol{X} \otimes I_K)_{(m,\cdot)}\left(\boldsymbol{y} - (\boldsymbol{X}' \otimes I_K)\hat{\boldsymbol{\phi}}_T^{daL}\right)}{\sqrt{T}} = \left(\frac{(\boldsymbol{X} \otimes I_K)_{(m,\cdot)}(\boldsymbol{X}' \otimes I_K)}{T}\right) \sqrt{T}\left(\boldsymbol{\phi}^* - \hat{\boldsymbol{\phi}}_T^{daL}\right) + \frac{(\boldsymbol{X} \otimes I_K)_{(m,\cdot)}\boldsymbol{e}}{\sqrt{T}}.$$

Note that $(\boldsymbol{X} \otimes I_K)_{(m,\cdot)}\boldsymbol{e}$ is the m-th element of the vector $(\boldsymbol{X} \otimes I_K)\boldsymbol{e}$, denoted by $((\boldsymbol{X} \otimes I_K)\boldsymbol{e})_m$. By Lemma (4.4.2), we have

$$\frac{1}{\sqrt{T}}((\boldsymbol{X} \otimes I_K)\boldsymbol{e})_m \xrightarrow{D} N\left(0,\ (\boldsymbol{\Gamma} \otimes \Sigma_\epsilon)_{(m,m)}\right),$$

where $(\boldsymbol{\Gamma} \otimes \Sigma_\epsilon)_{(m,m)}$ is the m-th diagonal element of $(\boldsymbol{\Gamma} \otimes \Sigma_\epsilon)$. Note also that $(\boldsymbol{X} \otimes I_K)_{(m,\cdot)}(\boldsymbol{X}' \otimes I_K)$ is the m-th row of the matrix $(\boldsymbol{X}\boldsymbol{X}' \otimes I_K)$, denoted by $(\boldsymbol{X}\boldsymbol{X}' \otimes I_K)_{(m,\cdot)}$. By Lemma (4.4.2), we have

$$\frac{1}{T}(\boldsymbol{X}\boldsymbol{X}' \otimes I_K)_{(m,\cdot)} \xrightarrow{a.s.} (\boldsymbol{\Gamma} \otimes I_K)_{(m,\cdot)}\ .$$

By Slutsky's theorem and the results of (i), we see that

$$\frac{1}{T}(\boldsymbol{X} \otimes I_K)_{(m,\cdot)}(\boldsymbol{X}' \otimes I_K)\sqrt{T}\left(\boldsymbol{\phi}^* - \hat{\boldsymbol{\phi}}_T^{daL}\right) \xrightarrow{D} (\boldsymbol{\Gamma} \otimes I_K)_{(m,\cdot)}\boldsymbol{z}\ ,$$

where $\boldsymbol{z}$ is a normally-distributed vector, and thus $(\boldsymbol{\Gamma} \otimes I_K)_{(m,\cdot)}\boldsymbol{z}$ a normally-distributed scalar variable. Therefore,

$$P(m \in \hat{\mathbb{S}}_T) \leq P\left(2(\boldsymbol{X} \otimes I_K)_{(m,\cdot)}\left(\boldsymbol{y} - (\boldsymbol{X}' \otimes I_K)\hat{\boldsymbol{\phi}}_T^{daL}\right) = \lambda_T \hat{w}_m \operatorname{sgn}\left(\hat{\phi}_{T,m}^{daL}\right)\right) \to 0.$$

(ii) The VAR order estimated by the doubly adaptive LASSO is

$$\hat{p}_T^{daL} = \min\left\{s : \hat{\phi}_{i,j,k}^{daL} = 0, \forall k = s+1, s+2, \cdots, h, \text{ and } i, j = 1, \cdots, K\right\},$$

or equivalently, in light of the bijective function (4.20),

$$\hat{p}_T^{daL} = \min\left\{s : (k-1)K^2 + (i-1)K + j \in \hat{\mathbb{S}}_T^c, \forall k = s+1, s+2, \cdots, h, \text{ and } i, j = 1, \cdots, K\right\}. \tag{4.25}$$

The true order p of the VAR model is

$$p = \min\left\{s : (k-1)K^2 + (i-1)K + j \in \mathbb{S}^c, \forall k = s+1, s+2, \cdots, h, \text{ and } i, j = 1, \cdots, K\right\}. \tag{4.26}$$

We have from (i) that $\hat{\mathbb{S}}_T^c \to \mathbb{S}^c$ in probability, so the RHS of (4.25) and (4.26) are equal in probability. Therefore, $\lim P(\hat{p}_T^{daL} = p) = 1$.

(iii) From (i), we have that $\lim \mathbb{P}\left(\hat{\boldsymbol{\phi}}_{\hat{\mathbb{S}}_T}^{daL} = \hat{\boldsymbol{\phi}}_{T,\mathbb{S}}^{daL}\right) \to 1$. Then, from Proposition 4.4.4, the asymptotic normality of $\hat{\boldsymbol{\phi}}_{\hat{\mathbb{S}}_T}^{daL}$ follows. ∎

Remarks:

(**1**) Although the asymptotic distributions of $\hat{\boldsymbol{\phi}}_{T,\mathbb{S}}^{daL}$ and $\hat{\boldsymbol{\phi}}_{\hat{\mathbb{S}}_T}^{daL}$ are identical, $\hat{\boldsymbol{\phi}}_{T,\mathbb{S}}^{daL}$ and $\hat{\boldsymbol{\phi}}_{\hat{\mathbb{S}}_T}^{daL}$ represent different identities; $\hat{\boldsymbol{\phi}}_{T,\mathbb{S}}^{daL}$ is the daLASSO estimator for the true non-zero parameter vector unknown in advance whereas $\hat{\boldsymbol{\phi}}_{\hat{\mathbb{S}}_T}^{daL}$ is the vector for non-zeros estimated by the daLASSO.

(**2**) The oracle properties we discuss here concern $\hat{\boldsymbol{\phi}}_{\hat{\mathbb{S}}_T}^{daL}$ rather than $\hat{\boldsymbol{\phi}}_{T,\mathbb{S}}^{daL}$.

(**3**) Proposition 4.4.4 concerns $\hat{\boldsymbol{\phi}}_{T,\mathbb{S}}^{daL}$, the daLASSO estimators for the true non-zero parameters, which are unknown in advance whereas Theorem 4.4.6 concerns $\hat{\boldsymbol{\phi}}_{\hat{\mathbb{S}}_T}^{daL}$, the non-zeros estimated by the doubly adaptive LASSO.

(**4**) Estimation consistency is necessary for oracle properties whereas oracle properties are sufficient for the former.

(**5**) Under the same asymptotic condition for tuning parameter λ_T (and other regularity conditions), the LASSO, the aLASSO and the daLASSO all have estimation consistency property.

(**6**) Under the same asymptotic condition for tuning parameter λ_T (and other regularity conditions), the aLASSO and the daLASSO both have oracle properties.

(**7**) The LASSO, the aLASSO and the daLASSO estimator might behaviour quite differently when finite samples are used. We need to investigate and compare their finite sample properties.

4.5 Computation algorithm for the doubly adaptive LASSO

Given values of γ_0, γ_1, and γ_2, the PLAC-weighted adaptive LASSO procedure is implemented via the *lars* developed by Efron et al (2004). The *lars* algorithm is very efficient, requiring the same order of computational cost as that of a single least squares fit. The LASSO methodology yields a path of possible solutions defined by the continuum over tuning and weighting parameters. The choice of these parameters determines the tradeoff between model fit and model sparsity. We use the BIC criteria to select the optimal value for Λ. The BIC is defined as

$$BIC = \log(\det\hat{\Sigma}_\epsilon) + |\hat{\mathbb{S}}_T|\log(T-h), \tag{4.27}$$

where

$$\hat{\Sigma}_\epsilon = \frac{1}{T-h}(\boldsymbol{Y} - \hat{\boldsymbol{\Phi}}^{daL}\boldsymbol{Y})(\boldsymbol{Y} - \hat{\boldsymbol{\Phi}}^{daL}\boldsymbol{X})', \tag{4.28}$$

$|\hat{\mathbb{S}}_T|$ is the cardinality of the set $\hat{\mathbb{S}}_T$, $\hat{\boldsymbol{\Phi}}$ being the estimates for (4.11), $\boldsymbol{Y}$ is (4.13), and $\boldsymbol{X}$ is (4.14). Algorithm 7 is the detailed computational procedure for the doubly adaptive LASSO given the value of the triple $(\gamma_0, \gamma_1, \gamma_2)$. Algorithm 8 shows the complete computation steps.

Algorithm 7: The *lars* algorithm for the doubly adaptive LASSO given $(\gamma_0, \gamma_1, \gamma_2)$.

Input: Data $\boldsymbol{y}_t, t = 1, \cdots, T$, and a specific value for $(\gamma_0, \gamma_1, \gamma_2)$.

Output: $\widehat{\boldsymbol{\Phi}}_T^{daL}$ for specific $(\gamma_0, \gamma_1, \gamma_2)$.

1 START

2 Compute $\hat{w}_{ij,k}$ defined by (4.23) and transform to $\hat{w}_{T,l}$ according to (4.20).

3 Compute $\boldsymbol{X}^* = \boldsymbol{X}\boldsymbol{W}^{-1}$, where $\boldsymbol{W} = \text{diag}[\hat{w}_1, \cdots, \hat{w}_{hK^2}]$, i.e. $\boldsymbol{x}_l^* = \boldsymbol{x}_l/\hat{w}_l, l = 1, \cdots, hK^2$.

4 Apply *lars* to obtain $\hat{\boldsymbol{\phi}}(\lambda_T) = \text{argmin}_{\boldsymbol{\phi}}\left\{(\boldsymbol{y} - \boldsymbol{X}^*\boldsymbol{\phi})^T(\boldsymbol{y} - \boldsymbol{X}^*\boldsymbol{\phi}) + \lambda_T \sum_{j=1}^{hK^2}|\phi_j|\right\}$.

5 Compute $\hat{\boldsymbol{\phi}}_T^{daL}(\lambda_T) = \boldsymbol{W}^{-1}\hat{\boldsymbol{\phi}}$.

6 Compute BIC(λ_T) according to (4.27) for the whole path.

7 Output $\widehat{\boldsymbol{\Phi}}_T^{daL}(\lambda_T^*)$ where λ_T^* is such that BIC$(\lambda_T^*) \leq$ BIC(λ_T).

8 END

Algorithm 8: Complete algorithm for the doubly adaptive positive LASSO

Input: Data: $\boldsymbol{y}_t, t = 1, \cdots, T$

Output: The doubly adaptive positive LASSO estimator $\widehat{\boldsymbol{\Phi}}_T^{daL}$

Start: Set up a grid $\mathcal{G} = \gamma_0 \times \gamma_1 \times \gamma_2$ with $G = |\mathcal{G}|$.
for $g \leftarrow 1$ ***to*** G **do**
    Apply Algorithm 7 to get $\widehat{\boldsymbol{\Phi}}_T\left(\gamma_0^{(g)}, \gamma_1^{(g)}, \gamma_2^{(g)}\right)$.
    Calculate $\text{BIC}(\gamma_0^{(g)}, \gamma_1^{(g)}, \gamma_2^{(g)})$.
Choose $(\gamma_0^*, \gamma_1^*, \gamma_2^*)$ such that $\text{BIC}(\gamma_0^*, \gamma_1^*, \gamma_2^*) = \min\{\text{BIC}(\gamma_0^{(g)}, \gamma_1^{(g)}, \gamma_2^{(g)}) : \forall g = 1, \cdots, G\}$.
Output $\widehat{\boldsymbol{\Phi}}_T^{daL} \leftarrow \widehat{\boldsymbol{\Phi}}_T(\gamma_0^*, \gamma_1^*, \gamma_2^*)$.
End

4.6 Monte Carlo study

We use Monte Carlo to investigate the sampling properties of the PLAC-weighted adaptive LASSO estimator for VAR models. Specifically, we would like to assess its performance in terms of order identification, the parameter estimation, and subset selection. The empirical statistics such as minimum, maximum, mean, medium, mode (for VAR lag order only), standard error, bias, MSE, MAD, and selection proportion were summarized based on 1000 replications. The definitions of empirical bias, MSE, and MAD are listed below for reference (and the rest omitted):

$$\widehat{Bias}(\hat{p}^{daL}) = \hat{E}[\hat{p}^{daL}] - p = \frac{1}{M}\sum_{m=1}^{M}(\hat{p}^{daL})^{(m)} - p$$

$$\widehat{MSE}(\hat{p}^{daL}) = \hat{E}[\hat{p}^{daL} - p]^2 = \frac{1}{M}\sum_{m=1}^{M}((\hat{p}^{daL})^{(m)} - p)^2$$

$$\widehat{MAD}(\hat{p}^{daL}) = \hat{E}|\hat{p}^{daL} - p| = \frac{1}{M}\sum_{m=1}^{M}|(\hat{p}^{daL})^{(m)} - p|$$

$$\widehat{Bias}(\hat{\phi}_j^{daL}) = \hat{E}[\hat{\phi}_j^{daL}] - \phi_j^* = \frac{1}{M}\sum_{m=1}^{M}(\hat{\phi}_j^{daL})^{(m)} - \phi_j^*$$

$$\widehat{MSE}(\hat{\phi}_j^{daL}) = \hat{E}[\hat{\phi}_j^{daL} - \phi_j^*]^2 = \frac{1}{M}\sum_{m=1}^{M}\left((\hat{\phi}_j^{daL})^{(m)} - \phi_j^*\right)^2$$

$$\widehat{MAD}(\hat{\phi}_j^{daL}) = \hat{E}|\hat{\phi}_j^{daL} - \phi_j^*| = \frac{1}{M}\sum_{m=1}^{M}\left|(\hat{\phi}_j^{daL})^{(m)} - \phi_j^*\right|$$

where M denotes the total number of MC runs.

4.6.1 A bivariate VAR(5) process

We use R function of `mAr.sim` implemented in the R package `mAR` (Barbosa, 2009) to generate $1,000$ data sets, denoted as $\mathscr{D}^{(m)}, m = 1, \cdots, 1000$, of sample size $T = 2000$ from the following stationary and stable bivariate VAR(5) process defined by (4.29) and (4.30).

$$\boldsymbol{y}_t = \Phi_1 \boldsymbol{y}_{t-1} + \Phi_2 \boldsymbol{y}_{t-2} + \Phi_4 \boldsymbol{y}_{t-4} + \Phi_5 \boldsymbol{y}_{t-5} + \boldsymbol{e}_t, \tag{4.29}$$

where

$$\Phi_1 = \begin{pmatrix} 0.4 & 1.2 \\ 0.3 & 0.0 \end{pmatrix}, \Phi_2 = \begin{pmatrix} 0.35 & -0.3 \\ 0.0 & -0.5 \end{pmatrix}, \Phi_4 = \begin{pmatrix} 0.0 & -0.5 \\ 0.4 & 0.0 \end{pmatrix}, \Phi_5 = \begin{pmatrix} 0.0 & 0.0 \\ 0.4 & -0.3 \end{pmatrix}, \tag{4.30}$$

and $\boldsymbol{e}_t$ is a Gaussian white noise with positive definite covariance matrix

$$\Sigma = \begin{pmatrix} 1.0 & -0.6 \\ 0.0 & 2.5 \end{pmatrix}.$$

The PLAC-weighted adaptive LASSO procedure was applied to fit 1,000 bivariate VAR models to $\mathscr{D}^{(m)}, m = 1, \cdots, 1000$. Pretending that we do not know the true lag order p, which is 5 in this case, of the underlying bivariate VAR process , we set the maximum order h to be 10. For the sake of simplicity h = 10 for all 1000 models, which we believe to be large enough in this example. To find an approximately optimal combination of γ_0, γ_1, and γ_2, we use grid-search method and the BIC criteria. Specifically, let $\mathcal{G} = \gamma_0 \times \gamma_1 \times \gamma_2 = [2.0, 4.0]_{\Delta=0.25} \times [1.5, 8.0]_{\Delta=0.25} \times [1.5, 8.0]_{\Delta=0.25}$. [4] For the sake of simplicity, the same 3-dimensional grid $\mathcal{G}$ is used for all 1000 models. Algorithm 9 describes the computational procedure for simulation study.

[4] Δ in the subscript represents the increment of the sequence.

Algorithm 9: Algorithm for Monte Carlo

Input: Data $\mathscr{D}^{(m)}, m = 1, \cdots, 1{,}000 = M$ and Grid $\mathcal{G}$.

Output: The LASSO estimate $\widehat{\mathbf{\Phi}}^{daL(m)}, m = 1, \cdots, M$.

```
Start
for m ← 1 to M do
    Apply Algorithm 8 to get Φ̂^{daL(m)}.
Compute empirical statistics.
End
```

Table 4.1 shows some empirical statistics such as Bias, MSE, and MAD of the VAR order estimates. Table 4.2 shows the distribution of the VAR order estimates. Table 4.3 shows empirical statistics for VAR coefficients. We summarize a few observations as follows:

(1) VAR lag order identification. Table 4.1 shows that the mode of 1,000 bivariate VAR order estimates is 5, the true lag order. Table 4.2 shows that almost 86% the fitted models have the order 5. The last column in Table 4.3 shows that autoregressors $\boldsymbol{y}_{t-k}$ for $k > 5$ have very slight chance to be included in models. Table 4.1 shows the mean and median of VAR order estimates are 5.234 and 5, respectively, indicating that the distribution of VAR order estimates is slight skewed to the right with a right tail in distribution as evident in Table 4.2. This example confirms that the doubly adaptive LASSO procedure is very excellent in identifying the order of a vector AR process.

(2) VAR subset selection. The last column in Table 4.3 shows that the non-zero coefficients were selected into the model 100% of time. On the other hand, some variables that are not included in the true bivariate VAR(5) process are also selected with quite high false inclusion rate. For example, $\Phi_3^* = \mathbf{0}$, but $20\% - 47\%$ of time it was falsely estimated as non-sero. The variables corresponding to the coefficients $\phi_{22,1}$, $\phi_{21,2}$, and $\phi_{22,4}$ are falsely included in the models 30%, 41%, and 40% of time, respectively. This confirms the suggestion that the doubly adaptive LASSO procedure have large power and be conservative in terms of subset selection.

(3) VAR coefficients estimation. The Mean, Median, SE, BIAS, and MSE columns in Table 4.3 suggests that the parameters are consistently estimated. In addition, the minimum and maximum columns in Table 4.3 shows that the signs of parameters are identified correctly

almost 100% of times: if the true value of a parameter is positive, the minimum of estimates never falls below 0; if the true value of a parameter is negative, the maximum of estimates never goes beyond 0. This example confirms the suggestion that doubly adaptive LASSO procedure estimate the parameters consistently.

Table 4.1: Empirical statistics of the doubly adaptive LASSO estimates for the bivariate AR order based on 1,000 replications each of size T=2,000, generated from bivariate AR(5) model with coefficients defined in (4.30). Set h=10. Use the BIC to choose λ, γ_0, γ_1, and γ_2.

True	Minimum	Maximum	Mean	Median	Mode	SE	Bias	MSE	MAD
5	5	10	5.234	5	5	0.682	0.234	0.52	0.234

Table 4.2: Empirical distribution of the doubly adaptive LASSO estimates for the bivariate AR order based on 1,000 replications each of size T=2,000, generated from the bivariate AR(5) model with coefficients defined in (4.30). Set h=10. Use the BIC to choose λ, γ_0, γ_1, and γ_2.

Lag Order	5	6	7	8	9	10
Percentage	86.7%	6.2%	5.0%	1.5%	0.3%	0.3%

4.6.2 A trivariate VAR(5) process

We also conduct another simulation study on a sparse trivariate VAR(5) process. We use R function of `mAr.sim` implemented in the R package `mAR` (Barbosa, 2009) to generate 1,000 data sets of sample size $T = 2000$ from the stationary process defined by (4.31) and (4.32). The doubly adaptive LASSO was applied to fit 1000 models. We use grid-search method and the BIC criteria to find an approximately optimal combination of γ_0, γ_1, and γ_2,. Specifically, let $\mathcal{G} = \gamma_0 \times \gamma_1 \times \gamma_2 = [2.0, 4.0]_{\Delta=0.25} \times [1.5, 8.0]_{\Delta=0.25} \times [1.5, 8.0]_{\Delta=0.25}$. For the sake of simplicity, the same 3-dimensional grid $\mathcal{G}$ is used for all 1000 models. Table 4.4 shows some empirical statistics such as Bias, MSE, and MAD of the VAR order estimates.

Table 4.5 shows the distribution of the VAR order estimates. Table 4.6 and 4.7 show empirical statistics for VAR coefficients. A few observations are summarized below, which confirm what we got previously:

Table 4.3: Empirical statistics of the doubly adaptive LASSO estimates for the bivariate AR coefficients $\mathbf{\Phi_1} - \mathbf{\Phi_5}$ based on 1,000 replications each of size T=2,000, generated from bivariate AR(5) model with coefficients defined in (4.30). Set h=10. Use the BIC to choose λ, γ_0, γ_1, and γ_2.

True	Min	Max	Mean	Median	Mode	SE	Bias	MSE	MAD	Prop
$\phi_{11,1}$	0.4	0.3350	0.4614	0.3995	0.3993	0.0166	-0.0005	0.0003	0.0132	1
$\phi_{21,1}$	0.3	0.1888	0.4163	0.2985	0.2986	0.0325	-0.0015	0.0011	0.0254	1
$\phi_{12,1}$	1.2	1.1569	1.2379	1.1994	1.1995	0.0110	-0.0006	0.0001	0.0087	1
$\phi_{22,1}$	0	-0.0621	0.0718	0.0002	0	0.0154	0.0002	0.0002	0.0075	0.304
$\phi_{11,2}$	0.35	0.3016	0.3985	0.3496	0.3495	0.0133	-0.0004	0.0002	0.0097	1
$\phi_{21,2}$	0	-0.1082	0.1059	0.0019	0	0.0284	0.0019	0.0008	0.0154	0.407
$\phi_{12,2}$	-0.3	-0.3734	-0.2041	-0.2999	-0.3006	0.0229	0.0001	0.0005	0.0181	1
$\phi_{22,2}$	-0.5	-0.6578	-0.3517	-0.4981	-0.4988	0.0433	0.0019	0.0019	0.0341	1
$\phi_{11,3}$	0	-0.0599	0.0550	0.0002	0	0.0102	0.0002	0.0001	0.0039	0.214
$\phi_{21,3}$	0	-0.1157	0.1357	0.0000	0	0.0254	0.0000	0.0006	0.0134	0.405
$\phi_{12,3}$	0	-0.0715	0.0709	-0.0003	0	0.0155	-0.0003	0.0002	0.0070	0.296
$\phi_{22,3}$	0	-0.1466	0.1469	-0.0018	0	0.0349	-0.0018	0.0012	0.0193	0.461
$\phi_{11,4}$	0	-0.0518	0.0485	-0.0002	0	0.0088	-0.0002	0.0001	0.0033	0.209
$\phi_{21,4}$	0.4	0.3100	0.4916	0.4002	0.4013	0.0275	0.0002	0.0008	0.0216	1
$\phi_{12,4}$	-0.5	-0.5807	-0.4372	-0.4995	-0.4996	0.0157	0.0005	0.0002	0.0117	1
$\phi_{22,4}$	0	-0.1210	0.1109	-0.0001	0	0.0285	-0.0001	0.0008	0.0148	0.403
$\phi_{11,5}$	0	-0.0305	0.0299	0.0001	0	0.0049	0.0001	0.0000	0.0014	0.123
$\phi_{21,5}$	0.4	0.3181	0.5042	0.3993	0.3993	0.0190	-0.0007	0.0004	0.0143	1
$\phi_{12,5}$	0	-0.0703	0.0672	0.0003	0	0.0119	0.0003	0.0001	0.0041	0.169
$\phi_{22,5}$	-0.3	-0.4167	-0.1504	-0.3006	-0.3011	0.0372	-0.0006	0.0014	0.0293	1
$\phi_{11,6}$	0	-0.0220	0.0393	0.0000	0	0.0016	0.0000	0.0000	0.0001	0.004
$\phi_{21,6}$	0	-0.0848	0.0812	-0.0002	0	0.0078	-0.0002	0.0001	0.0012	0.029
$\phi_{12,6}$	0	-0.0502	0.0423	0.0000	0	0.0030	0.0000	0.0000	0.0002	0.006
$\phi_{22,6}$	0	-0.1159	0.1270	-0.0001	0	0.0148	-0.0001	0.0002	0.0028	0.043
$\phi_{11,7}$	0	0	0	0	0	0	0	0	0	0
$\phi_{21,7}$	0	-0.0684	0.0743	0.0002	0	0.0054	0.0002	0.0000	0.0006	0.013
$\phi_{12,7}$	0	-0.0495	0.0236	0.0000	0	0.0021	0.0000	0.0000	0.0001	0.005
$\phi_{22,7}$	0	-0.0901	0.1083	0.0003	0	0.0109	0.0003	0.0001	0.0019	0.036
$\phi_{11,8}$	0	0	0	0	0	0	0	0	0	0
$\phi_{21,8}$	0	-0.0361	0	-0.0001	0	0.0015	-0.0001	0.0000	0.0001	0.002
$\phi_{12,8}$	0	0	0	0	0	0	0	0	0	0
$\phi_{22,8}$	0	-0.0953	0.0541	-0.0003	0	0.0055	-0.0003	0.0000	0.0005	0.013
$\phi_{11,9}$	0	0	0	0	0	0	0	0	0	0
$\phi_{21,9}$	0	0	0	0	0	0	0	0	0	0
$\phi_{12,9}$	0	0	0.0207	0.0000	0	0.0007	0.0000	0.0000	0.0000	0.001
$\phi_{22,9}$	0	-0.0366	0.0706	0.0000	0	0.0025	0.0000	0.0000	0.0001	0.002
$\phi_{11,10}$	0	0	0	0	0	0	0	0	0	0
$\phi_{21,10}$	0	0	0	0	0	0	0	0	0	0
$\phi_{12,10}$	0	0	0	0	0	0	0	0	0	0
$\phi_{22,10}$	0	-0.0431	0.0416	0.0000	0	0.0020	0.0000	0.0000	0.0001	0.003

$$y_t = \Phi_1 y_{t-1} + \Phi_2 y_{t-2} + \Phi_4 y_{t-4} + \Phi_5 y_{t-5} + e_t, \tag{4.31}$$

where

$$\begin{aligned} \Phi_1 &= \begin{pmatrix} 0.3 & 0.2 & 0.3 \\ 0.5 & 0.0 & 0.0 \\ 0.0 & 0.1 & -0.5 \end{pmatrix}, \Phi_2 = \begin{pmatrix} -0.3 & 0.0 & 0.0 \\ 0.0 & 0.1, & -0.5 \\ 0.7 & 0.2 & 0.0 \end{pmatrix}, \\ \Phi_4 &= \begin{pmatrix} 0.0 & 0.4 & -0.2 \\ 0.6 & 0.0 & 0.0 \\ 0.0 & -0.4, & 0.0 \end{pmatrix}, \Phi_5 = \begin{pmatrix} 0.2 & 0.0 & 0.0 \\ 0.0 & 0.0 & 0.4 \\ 0.0 & 0.3 & 0.3 \end{pmatrix}, \end{aligned} \tag{4.32}$$

and $\boldsymbol{e}_t$ is a Gaussian white noise with positive definite covariance matrix

$$\Sigma = \begin{pmatrix} 1.0 & -0.6 & 0.4 \\ 0.2 & 1.2 & 0.3 \\ -0.5 & 0.1 & 1.1 \end{pmatrix}.$$

(1) VAR lag order identification. Table 4.4 shows that the mode of 1,000 trivariate VAR order estimates is 5, the true lag order. Table 4.5 shows that almost 84% of 1000 models have the order 5, the true lag order; only around 16% models have lag orders greater than 5. The last column in Table 4.7 shows that autoregressors $\boldsymbol{y}_{t-k}$ for $k > 5$ have very slight chance to be included in models. Table 4.4 shows the mean and median of 1,000 VAR order estimates are 5.234 and 5, respectively, indicating that the distribution of VAR order estimates is slight skewed to the right with a right tail in distribution as evident in Table 4.5. This example again suggests that the doubly adaptive LASSO procedure be very excellent in identifying the order of a vector AR process.

(2) VAR subset selection. The last column in Table 4.6 shows that if the entries of a autoregressor vector are significant, then they are selected into the model 100% of time except that those corresponding to $\phi_{32,1}$ and $\phi_{22,2}$ have the inclusion rates being 99.9% and 99.6%, respectively. On the other hand, some variables that are not included in the true trivariate VAR(5) process are also falsely selected with quite high inclusion rate. For example, $\Phi_3 = \mathbf{0}$ in the underlying process 4.31, but the false inclusion rate of $\boldsymbol{y}_{t-3}$ in the model is somewhere between $19\% - 37\%$. This example also suggests that the doubly adaptive LASSO procedure have large power and be conservative in terms of subset selection.

(3) VAR coefficients estimation. The Mean, Median, SE, BIAS, and MSE columns in Table 4.6 suggests that the parameters are consistently estimated. In addition, the Min and Max columns in Table 4.6 shows that the signs of parameters are identified correctly 100% of times: if the true value of a parameter is positive, the Min of its estimates is never falls below 0; if the true value of a parameter is negative, the Max of its estimates is never goes beyond 0. This example suggests that doubly adaptive LASSO procedure estimate the parameters consistently.

Table 4.4: Empirical statistics of the doubly adaptive LASSO estimates for the trivariate AR order, based on 1,000 replications each of size T=2,000, generated from trivariate AR(5) model with coefficients defined in (4.32). Set h=10. Use the BIC to choose λ, γ_0, γ_1, and γ_2.

True	Minimum	Maximum	Mean	Median	Mode	SE	Bias	MSE	MAD
5	5	10	5.286	5	5	0.748	0.286	0.64	0.286

Table 4.5: Empirical distribution of the doubly adaptive LASSO estimates for the bivariate AR order based on 1,000 replications each of size T=2,000, generated from bivariate AR(5) model with coefficients defined in (4.32). Set h=10. Use the BIC to choose λ, γ_0, γ_1, and γ_2.

Lag Order	5	6	7	8	9	10
Percentage	83.6%	8.6%	4.4%	2.5%	0.8%	0.1%

4.7 Real data analysis

Figure 4.1 and 4.2 shows the data of quarterly West German investment, income, and consumption data (1960–1982) from Lütkepohl (2006, p. 77–79) and first differences of logarithms, respectively. Using the software Stata function **var** we fit a VAR(2) model with estimated coefficients shown in the following with the significant ones being bold-faced.

$$\hat{\Phi}_1 = \begin{pmatrix} \mathbf{-0.273} & \mathbf{0.337} & 0.652 \\ 0.043 & -0.123 & \mathbf{0.305} \\ 0.003 & \mathbf{0.289} & \mathbf{-0.285} \end{pmatrix}, \quad \hat{\Phi}_2 = \begin{pmatrix} -0.134 & 0.183 & 0.598 \\ \mathbf{0.062} & 0.021 & 0.049 \\ \mathbf{0.050} & \mathbf{0.366} & -0.116 \end{pmatrix}.$$

We use the PLAC-weight adaptive LASSO to fit a sparse VAR model. We set $h = 4$ and the grid $\mathcal{G} = \gamma_0 \times \gamma_1 \times \gamma_2 = [1.0, 4.0]_{\Delta=0.5} \times [1.0, 4.0]_{\Delta=0.25} \times [1.0, 5.0]_{\Delta=0.25}$. We use the BIC

Table 4.6: Empirical statistics of the doubly adaptive LASSO estimates for the bivariate AR coefficients $\mathbf{\Phi_1} - \mathbf{\Phi_5}$ based on 1,000 replications each of size T=2,000, generated from bivariate AR(5) model with coefficients defined in (4.32). Set h=10. Use the BIC to choose λ, γ_0, γ_1, and γ_2). See Table 4.7 for $\mathbf{\Phi_6} - \mathbf{\Phi}_{10}$.

Coeff	True	Min	Max	Mean	Median	SE	Bias	MSE	MAD	Prop
$\phi_{11,1}$	0.3	0.198	0.367	0.3003	0.301	0.022	0.0003	0.0005	0.0176	1
$\phi_{21,1}$	0.5	0.433	0.580	0.4993	0.500	0.023	-0.0007	0.0005	0.0183	1
$\phi_{31,1}$	0	-0.091	0.093	0.0001	0	0.022	0.0001	0.0005	0.0121	0.355
$\phi_{12,1}$	0.2	0.130	0.259	0.2003	0.201	0.019	0.0003	0.0004	0.0146	1
$\phi_{22,1}$	0	-0.057	0.058	0.0008	0	0.015	0.0008	0.0002	0.0072	0.303
$\phi_{32,1}$	0.1	0	0.166	0.0996	0.100	0.019	-0.0004	0.0003	0.0146	0.999
$\phi_{13,1}$	0.3	0.237	0.352	0.2994	0.299	0.018	-0.0006	0.0003	0.0140	1
$\phi_{23,1}$	0	-0.065	0.063	-0.0007	0	0.015	-0.0007	0.0002	0.0076	0.356
$\phi_{33,1}$	-0.5	-0.568	-0.431	-0.5005	-0.500	0.019	-0.0005	0.0003	0.0142	1
$\phi_{11,2}$	-0.3	-0.390	-0.222	-0.3013	-0.301	0.023	-0.0013	0.0005	0.0183	1
$\phi_{21,2}$	0	-0.086	0.100	0.0000	0	0.021	0.0000	0.0004	0.0110	0.364
$\phi_{31,2}$	0.7	0.612	0.774	0.7007	0.701	0.025	0.0007	0.0006	0.0193	1
$\phi_{12,2}$	0	-0.058	0.079	-0.0004	0	0.012	-0.0004	0.0001	0.0045	0.221
$\phi_{22,2}$	0.1	0	0.160	0.0992	0.100	0.021	-0.0008	0.0004	0.0158	0.996
$\phi_{32,2}$	0.2	0.149	0.273	0.2012	0.201	0.018	0.0012	0.0003	0.0142	1
$\phi_{13,2}$	0	-0.073	0.070	-0.0004	0	0.015	-0.0004	0.0002	0.0065	0.3
$\phi_{23,2}$	-0.5	-0.572	-0.419	-0.5002	-0.500	0.019	-0.0002	0.0004	0.0147	1
$\phi_{33,2}$	0	-0.107	0.071	-0.0012	0	0.017	-0.0012	0.0003	0.0081	0.311
$\phi_{11,3}$	0	-0.077	0.090	0.0004	0	0.019	0.0004	0.0004	0.0092	0.331
$\phi_{21,3}$	0	-0.090	0.089	0.0009	0	0.023	0.0009	0.0005	0.0120	0.366
$\phi_{31,3}$	0	-0.083	0.097	-0.0007	0	0.023	-0.0007	0.0005	0.0113	0.335
$\phi_{12,3}$	0	-0.078	0.049	-0.0001	0	0.011	-0.0001	0.0001	0.0041	0.197
$\phi_{22,3}$	0	-0.061	0.061	0.0001	0	0.012	0.0001	0.0002	0.0052	0.236
$\phi_{32,3}$	0	-0.066	0.063	0.0002	0	0.013	0.0002	0.0002	0.0048	0.192
$\phi_{13,3}$	0	-0.076	0.089	0.0000	0	0.014	0.0000	0.0002	0.0062	0.29
$\phi_{23,3}$	0	-0.079	0.089	-0.0001	0	0.016	-0.0001	0.0003	0.0076	0.327
$\phi_{33,3}$	0	-0.077	0.108	-0.0015	0	0.015	-0.0015	0.0002	0.0071	0.324
$\phi_{11,4}$	0	-0.087	0.089	0.0003	0	0.019	0.0003	0.0003	0.0085	0.294
$\phi_{21,4}$	0.6	0.497	0.682	0.5998	0.5991	0.024	-0.0002	0.0006	0.0190	1
$\phi_{31,4}$	0	-0.089	0.089	0.0010	0	0.019	0.0010	0.0004	0.0097	0.338
$\phi_{12,4}$	0.4	0.342	0.457	0.3994	0.3996	0.016	-0.0006	0.0003	0.0127	1
$\phi_{22,4}$	0	-0.061	0.052	-0.0006	0	0.012	-0.0006	0.0001	0.0047	0.23
$\phi_{32,4}$	-0.4	-0.460	-0.322	-0.3993	-0.3990	0.016	0.0007	0.0003	0.0124	1
$\phi_{13,4}$	-0.2	-0.265	-0.144	-0.1995	-0.1999	0.017	0.0005	0.0003	0.0138	1
$\phi_{23,4}$	0	-0.068	0.066	-0.0001	0	0.013	-0.0001	0.0002	0.0056	0.294
$\phi_{33,4}$	0	-0.057	0.069	0.0001	0	0.013	0.0001	0.0002	0.0055	0.254
$\phi_{11,5}$	0.2	0.114	0.279	0.1985	0.1982	0.024	-0.0015	0.0006	0.0192	1
$\phi_{21,5}$	0	-0.084	0.089	-0.0005	0	0.019	-0.0005	0.0004	0.0085	0.272
$\phi_{31,5}$	0	-0.088	0.100	-0.0001	0	0.019	-0.0001	0.0004	0.0080	0.226
$\phi_{12,5}$	0	-0.060	0.070	0.0001	0	0.010	0.0001	0.0001	0.0030	0.13
$\phi_{22,5}$	0	-0.075	0.063	-0.0004	0	0.012	-0.0004	0.0001	0.0043	0.159
$\phi_{32,5}$	0.3	0.225	0.371	0.3006	0.3011	0.019	0.0006	0.0004	0.0152	1
$\phi_{13,5}$	0	-0.062	0.043	0.0003	0	0.008	0.0003	0.0001	0.0030	0.216
$\phi_{23,5}$	0.4	0.335	0.460	0.3996	0.3993	0.016	-0.0004	0.0003	0.0125	1
$\phi_{33,5}$	0.3	0.250	0.356	0.2983	0.2977	0.018	-0.0017	0.0003	0.0140	1

Table 4.7: Empirical statistics of the doubly adaptive LASSO estimates for the VAR coefficients $\mathbf{\Phi_6} - \mathbf{\Phi}_{10}$ based on 1,000 replications each of size T=2,000, generated from VAR(5) model with coefficients defined in (4.32). Set h=10. Use the BIC to choose λ, γ_0, γ_1, and γ_2). See Table 4.6 for $\mathbf{\Phi_1} - \mathbf{\Phi}_5$.

Coeff	True	Min	Max	Mean	Median	SE	Bias	MSE	MAD	Prop
$\phi_{11,6}$	0	-0.073	0.068	-0.0001	0	0.007	-0.0001	0.0000	0.0009	0.021
$\phi_{21,6}$	0	-0.074	0.101	0.0008	0	0.010	0.0008	0.0001	0.0016	0.03
$\phi_{31,6}$	0	-0.070	0.076	0.0000	0	0.008	0.0000	0.0001	0.0012	0.026
$\phi_{12,6}$	0	0	0.052	0.0001	0	0.002	0.0001	0.0000	0.0001	0.003
$\phi_{22,6}$	0	-0.084	0.080	0.0001	0	0.005	0.0001	0.0000	0.0005	0.008
$\phi_{32,6}$	0	-0.039	0.061	0.0001	0	0.003	0.0001	0.0000	0.0002	0.004
$\phi_{13,6}$	0	-0.020	0.027	0.0000	0	0.001	0.0000	0.0000	0.0001	0.005
$\phi_{23,6}$	0	-0.063	0.068	-0.0001	0	0.004	-0.0001	0.0000	0.0004	0.012
$\phi_{33,6}$	0	0	0.048	0.0001	0	0.002	0.0001	0.0000	0.0001	0.003
$\phi_{11,7}$	0	-0.078	0.053	-0.0002	0	0.004	-0.0002	0.0000	0.0003	0.007
$\phi_{21,7}$	0	-0.086	0.064	-0.0005	0	0.006	-0.0005	0.0000	0.0008	0.017
$\phi_{31,7}$	0	-0.069	0.087	0.0003	0	0.006	0.0003	0.0000	0.0006	0.011
$\phi_{12,7}$	0	0	0	0	0	0	0	0	0	0
$\phi_{22,7}$	0	-0.066	0.012	-0.0001	0	0.003	-0.0001	0.0000	0.0001	0.003
$\phi_{32,7}$	0	0	0.070	0.0001	0	0.002	0.0001	0.0000	0.0001	0.001
$\phi_{13,7}$	0	0	0.052	0.0001	0	0.002	0.0001	0.0000	0.0001	0.002
$\phi_{23,7}$	0	-0.036	0.073	0.0002	0	0.004	0.0002	0.0000	0.0003	0.006
$\phi_{33,7}$	0	-0.045	0.027	-0.0001	0	0.002	-0.0001	0.0000	0.0001	0.004
$\phi_{11,8}$	0	0	0.047	0.0001	0	0.002	0.0001	0.0000	0.0001	0.003
$\phi_{21,8}$	0	-0.033	0.054	0.0002	0	0.004	0.0002	0.0000	0.0004	0.013
$\phi_{31,8}$	0	-0.045	0.041	0.0000	0	0.002	0.0000	0.0000	0.0001	0.004
$\phi_{12,8}$	0	0	0	0	0	0	0	0	0	0
$\phi_{22,8}$	0	0	0.051	0.0001	0	0.002	0.0001	0.0000	0.0001	0.001
$\phi_{32,8}$	0	-0.068	0	-0.0001	0	0.002	-0.0001	0.0000	0.0001	0.001
$\phi_{13,8}$	0	0	0	0	0	0	0	0	0	0
$\phi_{23,8}$	0	-0.080	0.038	-0.0001	0	0.003	-0.0001	0.0000	0.0002	0.004
$\phi_{33,8}$	0	-0.019	0.025	0.0000	0	0.001	0.0000	0.0000	0.0001	0.003
$\phi_{11,9}$	0	-0.044	0.027	0.0000	0	0.002	0.0000	0.0000	0.0001	0.002
$\phi_{21,9}$	0	-0.042	0.071	0.0001	0	0.003	0.0001	0.0000	0.0002	0.003
$\phi_{31,9}$	0	-0.039	0.022	0.0000	0	0.001	0.0000	0.0000	0.0001	0.002
$\phi_{12,9}$	0	-0.047	0	0.0000	0	0.001	0.0000	0.0000	0.0000	0.001
$\phi_{22,9}$	0	0	0	0	0	0	0	0	0	0
$\phi_{32,9}$	0	0	0	0	0	0	0	0	0	0
$\phi_{13,9}$	0	0	0.036	0.0000	0	0.001	0.0000	0.0000	0.0000	0.001
$\phi_{23,9}$	0	0	0	0	0	0	0	0	0	0
$\phi_{33,9}$	0	0	0	0	0	0	0	0	0	0
$\phi_{11,10}$	0	0	0	0	0	0	0	0	0	0
$\phi_{21,10}$	0	0	0	0	0	0	0	0	0	0
$\phi_{31,10}$	0	-0.065	0	-0.0001	0	0.002	-0.0001	0.0000	0.0001	0.001
$\phi_{12,10}$	0	0	0	0	0	0	0	0	0	0
$\phi_{22,10}$	0	0	0	0	0	0	0	0	0	0
$\phi_{32,10}$	0	0	0	0	0	0	0	0	0	0
$\phi_{13,10}$	0	0	0	0	0	0	0	0	0	0
$\phi_{23,10}$	0	0	0	0	0	0	0	0	0	0
$\phi_{33,10}$	0	0	0	0	0	0	0	0	0	0

to select the optimal value for tuning and weighting parameters. A VAR(4) sparse model with estimated coefficients as follows.

$$\hat{\Phi}_1^{daL} = \begin{pmatrix} -0.261 & 0.381 & 0.399 \\ 0.018 & 0 & 0.534 \\ 0 & 0.456 & -0.139 \end{pmatrix}, \quad \hat{\Phi}_2^{daL} = \begin{pmatrix} 0.399 & 0.030 & 0.426 \\ 0.534 & 0 & 0.378 \\ -0.139 & 0.536 & 0 \end{pmatrix}, \quad \hat{\Phi}_3^{daL} = \hat{\Phi}_4^{daL} = \mathbf{0}.$$

We observe that (i) all coefficient matrices beyond the lag 2 were shrank to zero, (ii) all significant coefficients were included in the model, (iii) all coefficients that were set to 0 are insignificant, and (iv) some insignificant coefficients were included in the model by the doubly adaptive LASSO procedure.

Figure 4.1: Quarterly West German investment, income, and consumption data (1960-1982) (Lütkepohl, 2006, p. 77 – 79)

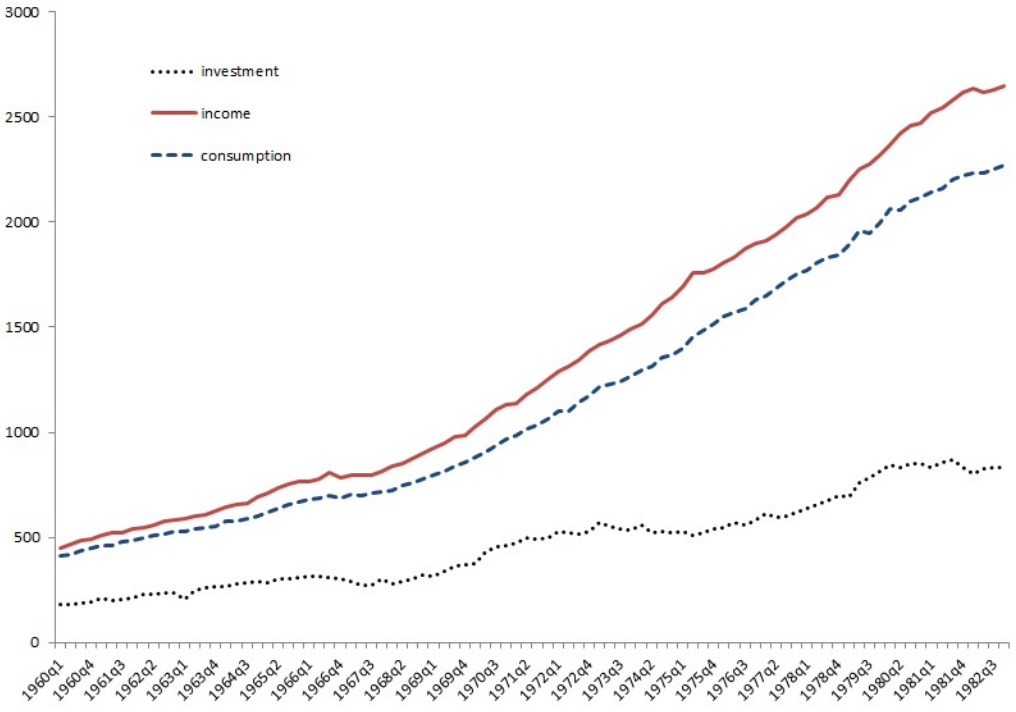
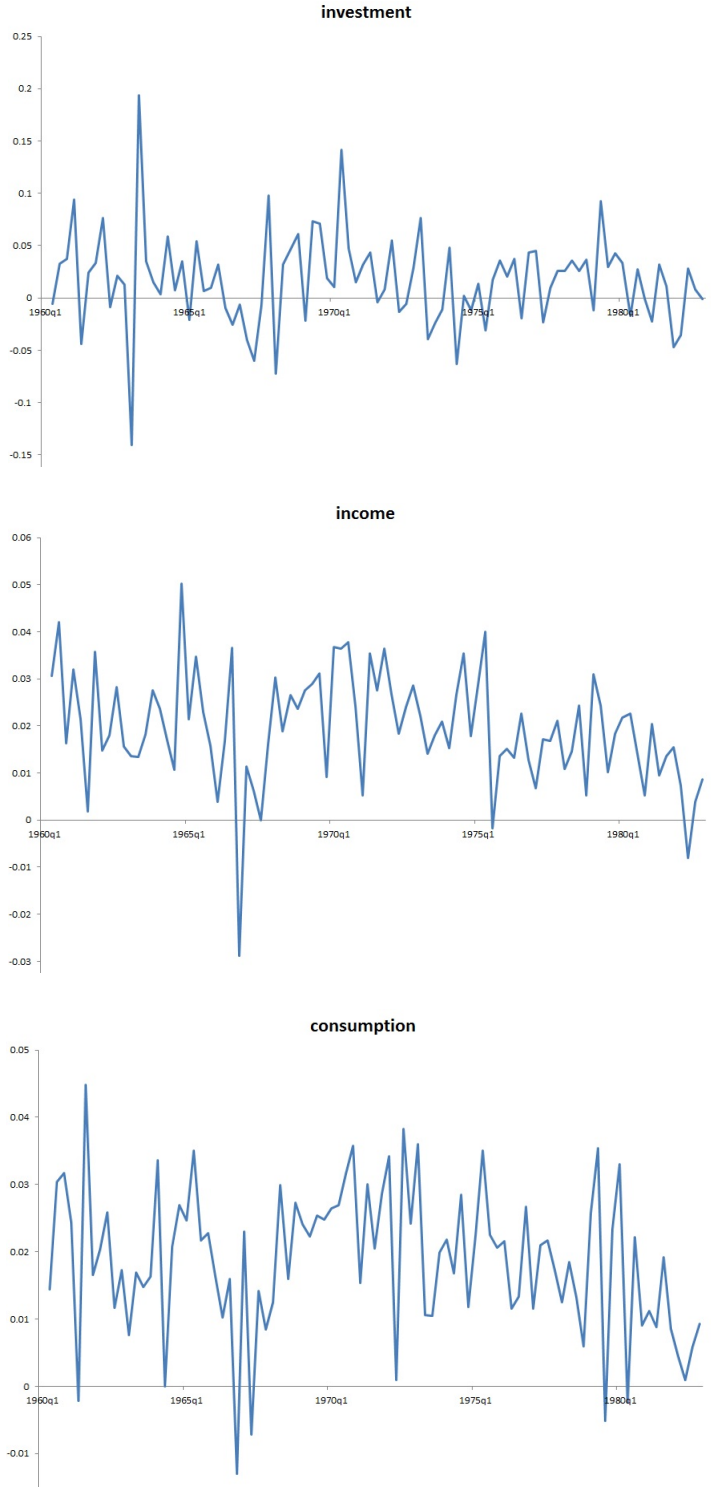


Figure 4.2: First differences of logarithms of quarterly West German investment, income, and consumption data (1960-1982) (Lütkepohl, 2006, p. 77 – 79)



Chapter 5

The Doubly Adaptive LASSO for BEKK Multivariate ARCH(q) models

5.1 Introduction

As we saw in Chapter 3, because it can capture some important stylized facts present in financial time series data, the ARCH(q) model has been widely used to model volatilities of financial assets. It is also of great practical importance to understand the comovements of several financial times series. For instance, asset pricing depends on the covariance of financial assets in a portfolio. Therefore, it is desirable to extend the univariate ARCH model to multivariate or vector ARCH (VARCH) model. A variety of multivariate models has been proposed in the literature. The Baba-Engle-Kroner-Kraft (BEKK) model (Engle and Kroner, 1995) is a well-known multivariate ARCH model. The BEKK model was constructed in such a way that the covariance matrices are guaranteed to be positive definite. This is an attractive property of the BEKK model.

Naturally, we desire sparse VARCH models since sparse ones may yield better forecasts compared to full models. Due to the successful examples of the LASSO in model selection, it is natural for us to consider the application of the LASSO methodology to VARCH modeling. Unfortunately, in the literature we have not yet found any results that applied the LASSO to modeling VARCH processes. The *curse of dimensionality* may be the major reason for the scarcity of examples. The number of parameters increases very rapidly as the dimension of vector process increases or as the lag order of the modes increases. This causes difficulties in

model estimation because numerical optimization will be time consuming and numerically unstable. In this chapter, we propose the doubly adaptive LASSO, the partial lag autocorrelation or PLAC-weighted adaptive LASSO, for modelling the sparse BEKK VARCH processes. By applying the doubly adaptive LASSO procedure we get identification, selection and estimation done all in one go.

We review the BEKK VARCH(q) models and standard modeling procedure in Section 5.2. We formulate the doubly adaptive positive LASSO tailored to ARCH processes in Section 5.3. Computation details are described in 5.4. Results from numerical experiments are contained in Section 5.5.

5.2 The BEKK VARCH(q) model and standard modelling procedure

In this section, we review the basic concepts of the BEKK VARCH(q) model and the standard modeling methods including order identification and quasi maximum likelihood estimation.

The BEKK VARCH(q) process

Let $\{\boldsymbol{y}_t\}, t = 0, \pm 1, \pm 2 \cdots, \pm\infty$ be a d-variate time series and $\mathcal{F}_t$ be the σ-field generated by past $\{\boldsymbol{y}_t\}$'s, i.e. $\mathcal{F}_t = \sigma(\boldsymbol{y}_t, \boldsymbol{y}_{t-1}, \cdots)$. Suppose that $\boldsymbol{y}_t$ is square-integrable and

$$\boldsymbol{y}_t = \mathbf{H}_t{}^{1/2}\eta_t \quad \text{with} \quad \boldsymbol{\eta}_t \sim iid(0, \boldsymbol{I}_d), \tag{5.1}$$

where $\boldsymbol{I}_d$ is the $d \times d$ identity matrix. The time series $\{\boldsymbol{y}_t\}$ is a martingale difference

$$\mathbb{E}[\boldsymbol{y}_t | \mathcal{F}_{t-1}] = \mathbf{0} \text{ a.s.}, \tag{5.2}$$

with time-varying conditional covariance matrix

$$\mathbb{E}[\boldsymbol{y}_t \boldsymbol{y}_t' | \mathcal{F}_{t-1}] = \boldsymbol{H}_t. \tag{5.3}$$

The BEKK(p, q, k) specification for $\boldsymbol{H}_t, t = 0, \pm 1, \pm 2 \cdots, \pm\infty$ (Engle and Kroner, 1995) is defined as

$$\boldsymbol{H}_t = \boldsymbol{C}\boldsymbol{C}' + \sum_{i=1}^{q}\left(\sum_{j=1}^{k} \boldsymbol{A}_{ij}\boldsymbol{y}_{t-i}\boldsymbol{y}_{t-i}'\boldsymbol{A}_{ij}'\right) + \sum_{i=1}^{p}\left(\sum_{j=1}^{k} \boldsymbol{B}_{ij}\boldsymbol{H}_{t-i}\boldsymbol{B}_{ij}'\right),$$

where $\boldsymbol{C}$ is $d \times d$ triangular matrix, $\boldsymbol{A}_{ij}$'s, and $\boldsymbol{B}_{ij}$'s are $d \times d$ matrices, and $k < d(d+1)/2$ determines the generality of the process. The advantage of the BEKK specification is that it guarantees the positive definiteness of $\boldsymbol{H}_t$.

We will consider a multivariate ARCH(q) volatility model, a special case of BEKK(p, q, k) specification in which $p = 0$ and $k = 1$:

$$\boldsymbol{H}_t = \boldsymbol{C}\boldsymbol{C}' + \sum_{j=1}^{q} \boldsymbol{A}_j \boldsymbol{y}_{t-j} \boldsymbol{y}'_{t-j} \boldsymbol{A}'_j = \boldsymbol{C}\boldsymbol{C}' + \boldsymbol{A}\boldsymbol{Y}_{t-1}\boldsymbol{A}', \tag{5.4}$$

where

$$\boldsymbol{A} = \left[\boldsymbol{A}_1 \vdots \boldsymbol{A}_2 \vdots \cdots \vdots \boldsymbol{A}_q\right], \tag{5.5}$$

and

$$\boldsymbol{Y}_{t-1} = \begin{pmatrix} \boldsymbol{y}_{t-1}\boldsymbol{y}'_{t-1} & \boldsymbol{0} & \cdots & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{y}_{t-2}\boldsymbol{y}'_{t-2} & \cdots & \boldsymbol{0} \\ \vdots & \vdots & \ddots & \vdots \\ \boldsymbol{0} & \boldsymbol{0} & \cdots & \boldsymbol{y}_{t-q}\boldsymbol{y}'_{t-q} \end{pmatrix}. \tag{5.6}$$

Let $\boldsymbol{h}_t = vec\boldsymbol{H}_t$, where the *vec* operator is defined in Appendix B. The model (5.4) can also be expressed in *vec* format as

$$\begin{aligned} \boldsymbol{h}_t &= vec(\boldsymbol{C}\boldsymbol{C}') + \sum_{j=1}^{q} vec(\boldsymbol{A}_j \boldsymbol{y}_{t-j} \boldsymbol{y}'_{t-j} \boldsymbol{A}'_j) \\ &= (\boldsymbol{C} \otimes \boldsymbol{C}) vec\boldsymbol{I}_d + \sum_{j=1}^{q} (\boldsymbol{A}_j \otimes \boldsymbol{A}_j)(\boldsymbol{y}_{t-j} \otimes \boldsymbol{y}_{t-j}). \end{aligned} \tag{5.7}$$

Identifiability of the BEKK VARCH Models

Identifiability of the BEKK vector ARCH(q) model (5.4) requires additional constraints. Indeed, the equivalent representation holds if $\boldsymbol{A}_j$ is replaced by $-\boldsymbol{A}_j$. For the identifiability of the parameters of the model (5.4), the diagonal entries of the constant matrix $\boldsymbol{C}$ are restricted to be positive, and the entries of the ARCH matrices $\boldsymbol{A}_j$'s nonnegative.

Identification of the BEKK VARCH Models

As in the case of univariate ARCH, the $vech(\boldsymbol{y}_t\boldsymbol{y}'_t)$, where the *vech* operator is defined in Appendix B, $t = 0, \pm 1, \pm 2 \cdots, \pm\infty$ process is the solution of a VAR(q) model. Indeed, define the

innovation process of $vech(\boldsymbol{y}_t\boldsymbol{y}_t')$ as

$$\boldsymbol{\nu}_t = vech(\boldsymbol{y}_t\boldsymbol{y}_t') - vech(\boldsymbol{H}_t),$$

and we have

$$vech(\boldsymbol{y}_t\boldsymbol{y}_t') = vech(\boldsymbol{C}\boldsymbol{C}') + \sum_{j=1}^{q} \boldsymbol{L}_d(\boldsymbol{A}_j \otimes \boldsymbol{A}_j)\boldsymbol{D}_d vech(\boldsymbol{y}_{t-j}\boldsymbol{y}_{t-j}') + \boldsymbol{\nu}_t,$$

where $\boldsymbol{D}_d$ is the $d^2 \times d(d+1)/2$ duplication matrix, and $\boldsymbol{L}_d$ is the $d(d+1)/2 \times d^2$ elimination matrix.

We then compute the partial lag autocorrelation matrix for the VAR process $vech(\boldsymbol{y}_t\boldsymbol{y}_t')$, $t = 1, \cdots, T$, thereby determining the order of $vech(\boldsymbol{y}_t\boldsymbol{y}_t')$, which is also the order of the vector ARCH process $\boldsymbol{y}_t$ defined by (5.4).

The quasi-maximum likelihood estimator

The classic approach to estimating the BEKK models is to minimize the negative quasi-maximum likelihood function. An estimator from this approach is called quasi-maximum likelihood estimator (QMLE). Suppose we have on a realization of size T d-variate time series $\boldsymbol{y}_t$, $t = 1, \cdots, T$. Using $\boldsymbol{y}_q, \cdots, \boldsymbol{y}_1$ as initial values with effective sample size reduced to $T - q$, the negative conditional quasi-likelihood function $L_T(\boldsymbol{\theta})$ of the BEKK VARCH(q) model is defined as

$$\begin{aligned} L_T(\boldsymbol{\theta}) &= \sum_{t=q+1}^{T} (-\ell_t(\boldsymbol{\theta})) \\ &= \frac{1}{2}dT\log(2\pi) + \frac{1}{2}\sum_{t=q+1}^{T} \log|\boldsymbol{H}_t(\boldsymbol{\theta})| + \frac{1}{2}\sum_{t=q+1}^{T} \boldsymbol{y}_t{}'\boldsymbol{H}_t(\boldsymbol{\theta})^{-1}\boldsymbol{y}_t, \end{aligned} \tag{5.8}$$

where parameter vector $\boldsymbol{\theta} = \left(vech(\boldsymbol{C})', vec(\boldsymbol{A}_1)', \cdots, vec(\boldsymbol{A}_q)'\right)'$.

The quasi-maximum likelihood estimator for $\boldsymbol{\theta}^*$ is defined as

$$\hat{\boldsymbol{\theta}}_T^{qml} = \underset{\boldsymbol{\theta}\in\Theta}{\operatorname{argmin}}\, L_T(\boldsymbol{\theta}).$$

5.3 The adaptive and doubly adaptive LASSO

We know that the LASSO could identify a subset of predictors by directly shrinking the coefficients corresponding to insignificant predictors to exact 0, and simultaneously yield estimates for non-zero coefficients. It is desirable to use the LASSO methodology for modelling BEKK vector ARCH processes because we like to get selection and estimation in one goal.

5.3.1 The adaptive LASSO when q is known

If the order q of BEKK vector ARCH model is known or has been identified a priori, then we apply the adaptive LASSO approach (Zou 2006) for a sparse estimator. The *adaptive LASSO* estimator, $\hat{\boldsymbol{\theta}}_T^{aL}$, is the adaptive LASSO-regularized quasi-maximum likelihood estimators for $\boldsymbol{\theta}^*$, which is defined as

$$\hat{\boldsymbol{\theta}}_T^{aL} = \underset{\boldsymbol{\theta}\in\Theta}{\operatorname{argmin}} \left\{ L_T(\boldsymbol{\theta}) + \lambda_T \left(\sum_{j=1}^{d'} \hat{w}_{T,j} \left|\theta_j\right| + \sum_{j=d'+1}^{q'} \hat{w}_{T,j} \left|\theta_j\right| \right) \right\}, \tag{5.9}$$

where $L_T(\boldsymbol{\theta})$ is defined by (5.8), $d' = d(d+1)/2$ the total number of parameters in the lower-triangular intercept matrix $\boldsymbol{C}$, $q' = d' + qd^2$ the total number of parameters in the vector $\boldsymbol{\theta}$,

$$\hat{w}_{T,j} = \begin{cases} \frac{1}{|\tilde{\theta}_j|^\gamma} & \text{if intercepts to be penalized} \\ 0 & \text{if intercepts not to be penalized} \end{cases} \tag{5.10}$$

for $j = 1, \cdots, d'$, and

$$\hat{w}_{T,j} = \frac{1}{\left|\tilde{\theta}_j\right|^\gamma} \tag{5.11}$$

for $j = d'+1, \cdots, q'$, where $\tilde{\theta}_j$ is any consistent estimate for θ_j, for instance, $\hat{\theta}_j^{qml}$

If the parameters in the coefficient matrices of the model are restricted to be nonnegative for identifiability, then following Efron (2004), we call the restricted adaptive LASSO estimator the *adaptive positive* LASSO estimator defined as

$$\hat{\boldsymbol{\theta}}_T^{apL} = \underset{\boldsymbol{\theta}\in\Theta}{\operatorname{argmin}} \left\{ L_T(\boldsymbol{\theta}) + \lambda_T \left(\sum_{j=1}^{d'} \hat{w}_{T,j} \left|\theta_j\right| + \sum_{j=d'+1}^{q'} \hat{w}_{T,j} \theta_j \right) \right\}, \tag{5.12}$$

where θ_j for $j = d'+1, \cdots, q'$ are restricted to be nonnegative and $\hat{w}_{T,j}$ still defined by (5.10) or (5.11).

5.3.2 The doubly adaptive LASSO when q is unknown

Usually, the order q of the BEKK vector ARCH model is unknown or difficult to be identified a priori. Let h [1] be our initial guess of the order. For this situation we propose the doubly adaptive LASSO or PLAC-weighted adaptive LASSO approach for a sparse estimator. Using $\boldsymbol{y}_h, \cdots, \boldsymbol{y}_1$ as initial values with effective sample size reduced to $T-h$, the negative conditional quasi-likelihood function $L_T(\boldsymbol{\theta})$ of the BEKK VARCH(h) model is

$$L_T(\boldsymbol{\theta}) = \frac{1}{2}dT\log(2\pi) + \frac{1}{2}\sum_{t=h+1}^{T}\log|\boldsymbol{H}_t(\boldsymbol{\theta})| + \frac{1}{2}\sum_{t=h+1}^{T}\boldsymbol{y}_t{}'\boldsymbol{H}_t(\boldsymbol{\theta})^{-1}\boldsymbol{y}_t, \tag{5.13}$$

where

$$\boldsymbol{H}_t = \boldsymbol{C}\boldsymbol{C}' + \sum_{j=1}^{h}\boldsymbol{A}_j\boldsymbol{y}_{t-j}\boldsymbol{y}_{t-j}'\boldsymbol{A}_j' \text{ for } t = h+1,\cdots,T, \tag{5.14}$$

$$\boldsymbol{A} = \left[\boldsymbol{A}_1 \vdots \boldsymbol{A}_2 \vdots \cdots \vdots \boldsymbol{A}_h\right],$$

and

$$\begin{aligned}\boldsymbol{\theta} &= (\theta_1, \cdots, \theta_l, \cdots, \theta_{d'+hd^2})' \\ &= (vech(\boldsymbol{C})', vec(\boldsymbol{A}_1)', \cdots, vec(\boldsymbol{A}_h)')' \\ &= \left(c_{11},\cdots,c_{d1},c_{22},\cdots,c_{d2},\cdots,c_{dd},a_{11,1},\cdots,a_{dd,1},\cdots,a_{ij,k},\cdots,a_{11,h},\cdots,a_{dd,h}\right)'\end{aligned}$$

with $d' = d(d+1)/2$. Note that the index l corresponds to the l-th element of the vector $\boldsymbol{\theta}$. The relation between (i,j), the subscripts of c_{ij}, and l is bijective and defined by

$$l = f(i,j) = (j-1)d + i - (j-1)j/2$$

for $l = 1,2,\cdots,d(d+1)/2$, and the relation between (i,j,k), the subscripts of $a_{ij,k}$, and l is bijective and defined by

$$l = f(i,j,k) = d' + (k-1)d^2 + (j-1)d + i$$

where $l = d'+1,\cdots,d'+hd^2$, $i,j = 1,2,\cdots,d$, and $k = 1,2,\cdots,h$.

[1] h is set to be quite large, for instance, $h = \kappa T^{\alpha}$, $0 \le \alpha \le 1$ for some constant κ.

The doubly adaptive LASSO or PLAC-weighted adaptive LASSO estimator for $\boldsymbol{\theta}^*$, denoted by $\hat{\boldsymbol{\theta}}_T^{daL}$, is defined as

$$\hat{\boldsymbol{\theta}}_T^{daL} = \underset{\boldsymbol{\theta}\in\Theta}{\operatorname{argmin}} \left\{ L_T(\boldsymbol{\theta}) + \lambda_T \left(\sum_{i=1}^{d} \sum_{i\geq j=1}^{d} \hat{w}_{ij,0} \left| c_{ij} \right| + \sum_{k=1}^{h} \sum_{i=1}^{d} \sum_{j=1}^{d} \hat{w}_{ij,k} \left| a_{ij,k} \right| \right) \right\}, \tag{5.15}$$

where $L_T(\boldsymbol{\theta})$ is defined by (5.13), c_{ij} the (i,j)th entry $(i > j)$ of the intercept matrix $\boldsymbol{C}$, $a_{ij,k}$ the (i,j)th entry of the coefficient matrix $\boldsymbol{A}_k$,

$$\hat{w}_{ij,0} == \begin{cases} \dfrac{1}{|\tilde{c}_{ij}|^{\gamma_1} \left(\sum\limits_{s=0}^{h} \left\| \widehat{\boldsymbol{P}}(s) \right\|_{\gamma_0}^{\gamma_0} \right)^{\gamma_2}} & \text{if intercepts to be adaptively penalized} \\ 0 & \text{if intercepts not to be penalized} \end{cases} \tag{5.16}$$

and

$$\hat{w}_{ij,k} = \frac{1}{\left| \tilde{a}_{ij,k} \right|^{\gamma_1} \left(\sum\limits_{s=k}^{h} \left\| \widehat{\boldsymbol{P}}(s) \right\|_{\gamma_0}^{\gamma_0} \right)^{\gamma_2}}, \tag{5.17}$$

where $\tilde{c}_{ij}$ and $\tilde{a}_{ij,k}$ are any consistent estimates for c_{ij} and $a_{ij,k}$, for instance, $\hat{c}_{ij}^{qml}$ and $\hat{a}_{ij,k}^{qml}$ respectively, $\widehat{\boldsymbol{P}}(s)$ [2] is the sample partial lag autocorrelation matrix $(d' \times d')$ of the $vech(\boldsymbol{y}_t \boldsymbol{y}_t')$ process [3], $\|\cdot\|_{\gamma_0}$ is the entrywise γ_0-norm so that

$$\left\| \widehat{\boldsymbol{P}}(s) \right\|_{\gamma_0} = \left(\sum\nolimits_{i=1}^{d'} \sum\nolimits_{j=1}^{d'} |\widehat{P}_{ij}(s)|^{\gamma_0} \right)^{1/\gamma_0}$$

is the entrywise γ_0-norm of $\widehat{\boldsymbol{P}}(s)$ at lag s, $\gamma_0 > 0$, $\gamma_1 \geq 0$, and $\gamma_2 \geq 0$ are some fixed constants.

First note that we suppress T from the subscripts of the weights for simplicity.

If the parameters in the coefficient matrices of the model are restricted to be nonnegative for identifiability, then following Efron, et al. (2004), we call the restricted doubly adaptive LASSO estimator the *doubly adaptive positive* LASSO estimator defined as

$$\hat{\boldsymbol{\theta}}_T^{dapL} = \underset{\boldsymbol{\theta}\in\Theta}{\operatorname{argmin}} \left\{ L_T(\boldsymbol{\theta}) + \lambda_T \left(\sum_{i=1}^{d} \sum_{i\geq j=1}^{d} \hat{w}_{ij,0} \left| c_{ij} \right| + \sum_{k=1}^{h} \sum_{i=1}^{d} \sum_{j=1}^{d} \hat{w}_{ij,k} a_{ij,k} \right) \right\}, \tag{5.18}$$

where $a_{ij,k}$ for $i, j = 1, \cdots, d, \text{and} k = 1, \cdots, h$ are restricted to be nonnegative and $\hat{w}_{ij,k}$ still defined by (5.16) or (5.17).

[2] See Appendix C for the definition and calculation of the the sample partial lag autocorrelation matrix.

[3] The VAR order of $vech(\boldsymbol{y}_t \boldsymbol{y}_t')$ suggests the VARCH order of $\boldsymbol{y}_t$. This is analogous to the univariate case where the ARCH order may also be suggested by the order of the squared process.(Shin and Kang, 2001; and Francq and Zakonïan, 2010, page 109.)

Remark 1: Both the LASSO (Tibshirani, 1996) and the adaptive LASSO (Zou, 2006) are the special cases of the doubly adaptive LASSO. In former case, $\gamma_1 = \gamma_2 = 0$, and in latter case, $\gamma_2 = 0$.

Remark 2: In the doubly adaptive LASSO procedure the partial lag autocorrelation information and the quasi-maximum likelihood estimates of the BEKK vector ARCH model work in tandem to perform subset selection and parameter estimation simultaneously. The basic idea can be elucidated from the following points:

Firstly, let $B_k = \sum_{s=k}^{h} \left\| \widehat{\boldsymbol{P}}(s) \right\|_{\gamma_0}^{\gamma_0}$, which is the tailed cumulative sum of the γ_0-norm of $\widehat{\boldsymbol{P}}(s)$ raised to the power γ_0 from kth-lag to the maximum hth lag, and note that $B_1 \geq \cdots \geq B_q \geq \cdots \geq B_h$. Hence, $w_{ij,k}$ is decreasing with increasing k. Consequently, depending on the structure of partial lag autocorrelation matrices, an ARCH term with smaller lag is more likely to be included in the model.

Secondly, the big bump of $\{B_k\}_{k=1}^{h}$ at $k = q$ relative to $k > q$ provides the cutoff at the true order of the vector ARCH process. This is because $\left\| \widehat{\boldsymbol{P}}(s) \right\|_{\gamma_0} = O_P(1/\sqrt{T})$ for $i = q+1, \cdots, h$, hence the B_j's for $j > q$ are relatively tiny. If j goes from h backwards to q, it is expected that the $\{B_j\}_{j=1}^{h}$ will exhibit a sharp jump at $j = q$. Consequently, the ARCH terms with lags greater than q get much more penalties so that they are more likely to be excluded from the model, and the true order of the ARCH process is thus identified.

Finally, $|\tilde{a}_{ij,k}|^{\gamma_1}$ imposes larger penalty on $a_{ij,k}$ if the corresponding ARCH term is not statistically significant. This is obvious because for an ARCH term is not important, the value of $\tilde{a}_{ij,k}$ is close to zero, $|\tilde{a}_{ij,k}|^{\gamma_1}$ is close to ∞. Consequently, the statistically insignificant ARCH terms get more penalties so that they are more likely to be excluded from the model whereas the statistically significant ARCH terms are more likely to be included in the model.

5.4 Computation algorithm for the doubly adaptive positive LASSO

We will modify the shooting algorithm described in Section 1.2.2 for the doubly adaptive LASSO for BEKK VARCH(q) model, as we did for univariate ARCH(q) model. We needs

quadratic approximation to the negative log quasi likelihood. The idea of quadratic approximation is not new, for theoretical analysis or for compuatation. Chernoff (1954), Tibshirani (1996), Andrews (1999), Fan and Li (2001), Francq and Zakoïan (2007), and Wang and Leng (2007) are examples to utilize quadratic approximation.

5.4.1 The quadratic approximation to the negative quasi-likelihood

Let $\boldsymbol{y}_1, \cdots, \boldsymbol{y}_T$ be a realization of d-variate time series generated by the BEKK VARCH model defined by (5.4). We approximate the the negative likelihood by second-order Taylor polynomial. This requires the derivation of the analytical score and analytical Hessian. The derivation is complicated and demanding and we put all the details in Appendix D and record the final result in the below. The quadratic approximation to the negative likelihood (5.13) is

$$\begin{aligned} L_T(\boldsymbol{\theta}) &\approx L_T(\boldsymbol{\theta}^*) + (\boldsymbol{\theta}-\boldsymbol{\theta}^*)' \boldsymbol{S}_T(\boldsymbol{\theta}^*)' + \frac{1}{2}(\boldsymbol{\theta}-\boldsymbol{\theta}^*)' \boldsymbol{J}_T(\boldsymbol{\theta}^*)(\boldsymbol{\theta}-\boldsymbol{\theta}^*) \\ &= \frac{1}{2}\boldsymbol{\theta}' \boldsymbol{J}_T(\boldsymbol{\theta}^*)\boldsymbol{\theta} - \boldsymbol{\theta}' (\boldsymbol{J}_T(\boldsymbol{\theta}^*)\boldsymbol{\theta}^* - \boldsymbol{S}_T(\boldsymbol{\theta}^*)') + c_T(\boldsymbol{\theta}^*), \end{aligned} \tag{5.19}$$

where $\boldsymbol{\theta}^*$ is the unknown true parameter vector,

$$\begin{aligned} \boldsymbol{J}_T(\boldsymbol{\theta}^*) &= \sum_{t=1}^{T} \left\{ \frac{\partial vec(\boldsymbol{R}_{t-1})'}{\partial \boldsymbol{\theta}} (\boldsymbol{I}_{h'} \otimes \boldsymbol{N}_d \boldsymbol{Q}_t(\boldsymbol{\theta}^*)') + \frac{\partial \boldsymbol{Q}_t(\boldsymbol{\theta}^*)}{\partial \boldsymbol{\theta}} \boldsymbol{N}_d \boldsymbol{R}_{t-1}(\boldsymbol{\theta}^*) \right\}, \\ \boldsymbol{S}_T(\boldsymbol{\theta}^*) &= \sum_{t=1}^{T} \boldsymbol{Q}_t(\boldsymbol{\theta}^*) \boldsymbol{N}_d \boldsymbol{R}_{t-1}(\boldsymbol{\theta}^*), \\ c_T(\boldsymbol{\theta}^*) &= \frac{1}{2}\boldsymbol{\theta}^{*\prime} \boldsymbol{J}_T(\boldsymbol{\theta}^*)\boldsymbol{\theta}^* - \boldsymbol{\theta}^{*\prime} \boldsymbol{S}_T(\boldsymbol{\theta}^*)' + L_T(\boldsymbol{\theta}^*), \text{and} \\ L_T(\boldsymbol{\theta}^*) &= \frac{1}{2} dT \log(2\pi) + \frac{1}{2}\sum_{t=1}^{T} \log \left| \boldsymbol{H}_t(\boldsymbol{\theta}^*) \right| + \frac{1}{2}\sum_{t=1}^{T} \boldsymbol{y}_t' \boldsymbol{H}_t^{-1}(\boldsymbol{\theta}^*) \boldsymbol{y}_t. \end{aligned}$$

where $\boldsymbol{R}_{t-1}$, $\boldsymbol{N}_d$, and $\boldsymbol{Q}_{t-1}$ are defined in Appendix D. Pay attention to the dimensions of matrices. The order of BEKK VARCH in Appendix D is q but here is h. Here $h' = d(d+1)/2 + hd^2$ and $\bar{\boldsymbol{R}}_{t-1}$ is $d^2 \times h'$ whereas in Appendix D $q' = d(d+1)/2 + qd^2$ and $\bar{\boldsymbol{R}}_{t-1}$ is $d^2 \times q'$.

As discussed in Chapter 3, iterative least-squares methods can be applied to estimation of BEKK VARCH models, which will involve the decomposition of the Hessian matrix $\boldsymbol{J}_T(\boldsymbol{\theta})$. However, at each iteration step, say the r-th step, the matrix $\boldsymbol{J}_T(\boldsymbol{\theta}^{[r]})$, the Hessian evaluated

at the estimated value $\boldsymbol{\theta}^{[r]}$ may not be positive definite, in which case the Cholesky or LU decomposition is not applicable. We may use the spectral decomposition instead. Since it is symmetric, the matrix $\boldsymbol{J}_T(\boldsymbol{\theta}^{[r]})$ has a spectral decomposition $\boldsymbol{J}_T(\boldsymbol{\theta}^{[r]}) = \boldsymbol{\Gamma}(\boldsymbol{\theta}^{[r]})\boldsymbol{\Lambda}(\boldsymbol{\theta}^{[r]})\boldsymbol{\Gamma}(\boldsymbol{\theta}^{[r]})'$, where $\boldsymbol{\Lambda}(\boldsymbol{\theta}^{[r]})$ is a diagonal matrix with its diagonal elements being the eigenvalues of $\boldsymbol{J}_T(\boldsymbol{\theta}^{[r]})$, and $\boldsymbol{\Gamma}(\boldsymbol{\theta}^{[r]})$ some orthogonal matrix. In order to use least-squares method, square-rooting the matrix $\boldsymbol{J}_T(\boldsymbol{\theta}^{[r]})$ is required. Unfortunately, $\boldsymbol{J}_T(\boldsymbol{\theta}^{[r]})$ may not be positive definite, in which case we cannot calculate the square-root of diagonal matrix because some of the eigenvalues are negative. To bypass this problem, we approximate the Hessian $\boldsymbol{J}_T(\boldsymbol{\theta}^{[r]})$ by replacing $\boldsymbol{\Lambda}(\boldsymbol{\theta}^{[r]})$ with its absolute value $|\boldsymbol{\Lambda}(\boldsymbol{\theta}^{[r]})|$.

5.4.2 The surrogate of the quadratic approximation of likelihood

The surrogate for the Hessian matrix $\boldsymbol{J}_T(\boldsymbol{\theta})$, denoted by $\widetilde{\boldsymbol{J}}_T(\boldsymbol{\theta})$, is defined as

$$\widetilde{\boldsymbol{J}}_T(\boldsymbol{\theta}) = \boldsymbol{\Gamma}(\boldsymbol{\theta})|\boldsymbol{\Lambda}(\boldsymbol{\theta})|\boldsymbol{\Gamma}(\boldsymbol{\theta})',$$

where $\boldsymbol{\Lambda}(\boldsymbol{\theta})$ is a diagonal matrix with its diagonal elements being the eigenvalues of $\boldsymbol{J}_T(\boldsymbol{\theta})$, and $\boldsymbol{\Gamma}(\boldsymbol{\theta})$ some orthogonal matrix. Accordingly, the surrogate for the quadratic approximation of likelihood $L_T(\boldsymbol{\theta})$ in (3.17), denoted by $\mathscr{S}_T(\boldsymbol{\theta})$, is defined as

$$\mathscr{S}_T(\boldsymbol{\theta}) \;=\; \frac{1}{2}\boldsymbol{\theta}'\boldsymbol{\Gamma}(\boldsymbol{\theta}^*)\left|\boldsymbol{\Lambda}(\boldsymbol{\theta}^*)\right|\boldsymbol{\Gamma}(\boldsymbol{\theta}^*)'\boldsymbol{\theta} - \boldsymbol{\theta}'\left[\boldsymbol{J}_T(\boldsymbol{\theta}^*)\boldsymbol{\theta}^* - \boldsymbol{S}_T(\boldsymbol{\theta}^*)\right] + c_T(\boldsymbol{\theta}^*).$$

Now, define and use the matrix

$$\widetilde{\boldsymbol{X}}(\boldsymbol{\theta}^*) = |\boldsymbol{\Lambda}(\boldsymbol{\theta}^*)|^{1/2}\boldsymbol{\Gamma}(\boldsymbol{\theta}^*)', \tag{5.20}$$

and the vector

$$\widetilde{y}(\boldsymbol{\theta}^*) = |\boldsymbol{\Lambda}(\boldsymbol{\theta}^*)|^{-1/2}\boldsymbol{\Gamma}(\boldsymbol{\theta}^*)'\left(\boldsymbol{J}_T(\boldsymbol{\theta}^*)\boldsymbol{\theta}^* - \boldsymbol{S}_T(\boldsymbol{\theta}^*)'\right). \tag{5.21}$$

A bit of manipulation yields the least squares form of the surrogate $\mathscr{S}_T(\boldsymbol{\theta})$ as follows

$$\mathscr{S}_T(\boldsymbol{\theta}) = \frac{1}{2}\left(\widetilde{\boldsymbol{y}}(\boldsymbol{\theta}^*) - \widetilde{\boldsymbol{X}}(\boldsymbol{\theta}^*)\boldsymbol{\theta}\right)'\left(\widetilde{\boldsymbol{y}}(\boldsymbol{\theta}^*) - \widetilde{\boldsymbol{X}}(\boldsymbol{\theta}^*)\boldsymbol{\theta}\right) + d_T(\boldsymbol{\theta}^*).$$

5.4.3 The modified shooting algorithm

The least squares form of the surrogate $\mathscr{S}_T(\boldsymbol{\theta})$ allows us to estimate iteratively. Suppose we get the estimates $\hat{\boldsymbol{\theta}}^{[r]}$ and $\tilde{\boldsymbol{\theta}}^{[r]}$ after the r-th step, then at the (r+1)st step, we simply minimize the following least squares objective function

$$\left(\widetilde{\boldsymbol{y}}(\hat{\boldsymbol{\theta}}^{[r]}) - \widetilde{\boldsymbol{X}}(\hat{\boldsymbol{\theta}}^{[r]})\boldsymbol{\theta}\right)' \left(\widetilde{\boldsymbol{y}}(\hat{\boldsymbol{\theta}}^{[r]}) - \widetilde{\boldsymbol{X}}(\hat{\boldsymbol{\theta}}^{[r]})\boldsymbol{\theta}\right) + \lambda_T \sum_{k=1}^{h}\sum_{i=1}^{d}\sum_{j=1}^{d} \hat{w}_{ij,k}(\tilde{\theta}_l^{[r]})\theta_l, \tag{5.22}$$

where $\widetilde{\boldsymbol{X}}$ and $\widetilde{y}$ are defined as in (5.20) and (5.21), respectively, and $\hat{w}_{ij,k}(\tilde{\theta}_l^{[r]})$ should be computed accordingly using (5.16) and (5.17). In particular, the relationship between the subscripts (ij,k) of $\hat{w}$ and the subscript l of $\tilde{\theta}$ are bijective. Now, with reference to Section 1.2.2 and Section 3.5.3, we define

$$S_{0,l}^{[r]} = S_0\left(0,\ \boldsymbol{\theta}^{(-l)},\ \widetilde{\boldsymbol{X}}(\hat{\boldsymbol{\theta}}^{[r]}),\ \widetilde{\boldsymbol{y}}(\hat{\boldsymbol{\theta}}^{[r]})\right) = 2\sum_{i\neq l}\left(\widetilde{\boldsymbol{x}}(\hat{\boldsymbol{\theta}}^{[r]})^l\right)'\widetilde{\boldsymbol{x}}(\hat{\boldsymbol{\theta}}^{[r]})^l\theta_i - 2\left(\widetilde{\boldsymbol{x}}(\hat{\boldsymbol{\theta}}^{[r]})^l\right)'\widetilde{\boldsymbol{y}}(\hat{\boldsymbol{\theta}}^{[r]}), \tag{5.23}$$

$$S_{l}^{[r]} = S_l\left(\boldsymbol{\theta},\ \widetilde{\boldsymbol{X}}(\hat{\boldsymbol{\theta}}^{[r]}),\ \widetilde{\boldsymbol{y}}(\hat{\boldsymbol{\theta}}^{[r]})\right) = 2\left(\widetilde{\boldsymbol{x}}(\hat{\boldsymbol{\theta}}^{[r]})^l\right)'\widetilde{\boldsymbol{x}}(\hat{\boldsymbol{\theta}}^{[r]})^l\theta_l + S_{0,l}^{[r]},$$

and

$$\lambda_l^{[r]} = \lambda_T \hat{w}_{ij,k}(\tilde{\theta}_l^{[r]}),$$

where $\widetilde{\boldsymbol{x}}(\hat{\boldsymbol{\theta}}^{[r]})^l$ represents the lth column of $\widetilde{\boldsymbol{X}}(\hat{\boldsymbol{\theta}}^{[r]})$, and $\hat{w}_{ij,k}(\tilde{\theta}_l^{[r]})$ is defined by (5.16) and (5.17). Now, with aid of Figure 1.2, the (r+1)st step estimates for θ_l can be obtained using

$$\hat{\theta}_l^{[r+1]} = \begin{cases} \dfrac{\lambda_l^{[r]} - S_{0,l}^{[r]}}{2\left(\widetilde{\boldsymbol{x}}(\hat{\boldsymbol{\theta}}^{[r]})^l\right)'\widetilde{\boldsymbol{x}}(\hat{\boldsymbol{\theta}}^{[r]})^l} & \text{if } S_{0,l}^{[r]} > \lambda_l^{[r]}, \\ 0 & \text{if } |S_{0,l}^{[r]}| < \lambda_l^{[r]}, \\ \dfrac{-\lambda_l^{[r]} - S_{0,l}^{[r]}}{2\left(\widetilde{\boldsymbol{x}}(\hat{\boldsymbol{\theta}}^{[r]})^l\right)'\widetilde{\boldsymbol{x}}(\hat{\boldsymbol{\theta}}^{[r]})^l} & \text{if } S_{0,l}^{[r]} < -\lambda_l^{[r]}, \end{cases}$$

Algorithm 10 shows the computation steps in detail.

Algorithm 10: Modified shooting algorithm for the doubly adaptive positive LASSO given a value for the quadruple $(\lambda_T, \gamma_0, \gamma_1, \gamma_2)$

Input: Data $\boldsymbol{y}_1, \cdots, \boldsymbol{y}_T$, given values of $(\lambda_T, \gamma_0, \gamma_1, \gamma_2)$
Output: The $(d' + hd^2)$-dimensional vector estimate $\hat{\boldsymbol{\theta}}(\lambda_T, \gamma_0, \gamma_1, \gamma_2)$

Start: $k = 1$, initialize, say $\hat{\boldsymbol{\theta}}^{[r]} \leftarrow [0.0001, \cdots, 0.0001]$
Set stopping rule, $\|\hat{\boldsymbol{\theta}}^{[r+1]} - \hat{\boldsymbol{\theta}}^{[r]}\|_\infty < \zeta$, where ζ is a tiny number, say 0.00005
Iteration: Compute $\widetilde{\boldsymbol{X}}(\hat{\boldsymbol{\theta}}^{[r]})$ and $\widetilde{\boldsymbol{y}}(\hat{\boldsymbol{\theta}}^{[r]})$
Compute $\tilde{\boldsymbol{\theta}}^{[r]} \leftarrow \left(\widetilde{\boldsymbol{X}}(\hat{\boldsymbol{\theta}}^{[r]})'\widetilde{\boldsymbol{X}}(\hat{\boldsymbol{\theta}}^{[r]})\right)^{-1}\widetilde{\boldsymbol{y}}(\hat{\boldsymbol{\theta}}^{[r]})$
for $l \leftarrow 1$ ***to*** $d' + hd^2$ **do**
    $\lambda_l^{[r]} \leftarrow \lambda_T \hat{w}_{T,l}(\tilde{\theta}_l^{[r]})$ using (5.16) and (5.17)
    Compute $S_{0,l}^{[r]}$ using (5.23)
    if $S_{0,l}^{[r]} > \lambda_l^{[r]}$ **then**
        $\hat{\theta}_l^{[r+1]} \leftarrow \left(\lambda_l^{[r]} - S_{0,l}^{[r]}\right) / \left[2\left(\widetilde{\boldsymbol{x}}(\hat{\boldsymbol{\theta}}^{[r]})^l\right)'\widetilde{\boldsymbol{x}}(\hat{\boldsymbol{\theta}}^{[r]})^l\right]$
    if $S_{0,l}^{[r]} < -\lambda_l^{[r]}$ **then**
        $\hat{\theta}_l^{[r+1]} \leftarrow \left(-\lambda_l^{[r]} - S_{0,l}^{[r]}\right) / \left[2\left(\widetilde{\boldsymbol{x}}(\hat{\boldsymbol{\theta}}^{[r]})^l\right)'\widetilde{\boldsymbol{x}}(\hat{\boldsymbol{\theta}}^{[r]})^l\right]$
    else
        $\hat{\theta}_l^{[r+1]} \leftarrow 0$
if $\left\|\hat{\boldsymbol{\theta}}^{[r+1]} - \hat{\boldsymbol{\theta}}^{[r]}\right\|_\infty < \zeta$ **then**
    $\hat{\boldsymbol{\theta}}^{[r]} \leftarrow \hat{\boldsymbol{\theta}}^{[r+1]}$
    $r \leftarrow r + 1$
    return *Iteration*
else
    Output: $\hat{\boldsymbol{\theta}} \leftarrow \hat{\boldsymbol{\theta}}^{[r+1]}$
End

We may also restrict all the parameters to be nonnegative. In this case, we apply the doubly adaptive positive LASSO as follow.

$$\hat{\theta}_l^{[r+1]} = \begin{cases} \frac{-\lambda_l^{[r]} - S_{0,l}^{[r]}}{2\left(\widetilde{\boldsymbol{x}}(\hat{\boldsymbol{\theta}}^{[r]})^l\right)'\widetilde{\boldsymbol{x}}(\hat{\boldsymbol{\theta}}^{[r]})^l} & \text{if } S_{0,l}^{[r]} < -\lambda_l^{[r]}, \\ 0 & \text{otherwise.} \end{cases}$$

The computational details are the same as Algorithm 10 except that the second **if** is removed from the algorithm.

We use the BIC criteria to select the optimal value for $\Lambda = (\lambda_T, \gamma_0, \gamma_1, \gamma_2)$. The BIC is defined as

$$BIC = 2L_T(\hat{\boldsymbol{\theta}}) + |\hat{\mathbb{S}}_T| \log(T-h),$$

where L_T is the negative log quasi-likelihood function defined in (5.13), $|\hat{\mathbb{S}}_T|$ is the cardinality of the set $\hat{\mathbb{S}}_T$. Define a 4-dimensional grid $\mathcal{G} = \lambda_T \times \gamma_0 \times \gamma_1 \times \gamma_2$ with a total number of G grid points. By using information criteria for LASSO, we have double penalization to be involved. One is L_1 penalization by the LASSO, which yields the path solution of the LASSO,

$$\hat{\boldsymbol{\theta}}(\Lambda) = \arg\min_{\boldsymbol{\theta}} \mathscr{S}_T(\boldsymbol{\theta}) + \lambda_T \sum_{l=1}^{d'+hd^2+1} \hat{w}_{T,l}(\Lambda)\theta_l,$$

and the other is the L_0 penalization by the BIC, which yields

$$\Lambda^* = \arg\min_{\Lambda \in \mathcal{G}} \mathrm{BIC}(\Lambda) = 2L_T(\hat{\boldsymbol{\theta}}(\Lambda)) + |\hat{\mathbb{S}}_T| \log(T-h).$$

Then the solution $\hat{\boldsymbol{\theta}}^{daL}$ is read off from the path against Λ^*. Algorithm 11 shows the complete computation steps.

Algorithm 11: Complete algorithm for the doubly adaptive LASSO

Input: Data: $\boldsymbol{y}_1, \cdots, \boldsymbol{y}_T$

Output: The doubly adaptive LASSO estimator $\hat{\boldsymbol{\theta}}_T^{daL}$

Start: Set up a grid $\mathcal{G} = \lambda_T \times \gamma_0 \times \gamma_1 \times \gamma_2$ with $G = |\mathcal{G}|$

for $g \leftarrow 1$ ***to*** G **do**

  Apply Algorithm 10 to get $\hat{\boldsymbol{\theta}}(\Lambda^{(g)})$

  Calculate $\mathrm{BIC}(\Lambda^{(g)}) = 2L_T(\hat{\boldsymbol{\theta}}(\Lambda^{(g)})) + |\hat{\mathbb{S}}_T^{(g)}| \log(T-h)$

Choose Λ^* such that $\mathrm{BIC}(\hat{\boldsymbol{\theta}}(\Lambda^*)) = \min\{\mathrm{BIC}(\Lambda^{(g)}) : \forall g = 1, \cdots, G\}$

Output $\hat{\boldsymbol{\theta}}_T^{daL} \leftarrow \hat{\boldsymbol{\theta}}(\Lambda^*)$

End

5.5 Monte Carlo study

We use Monte Carlo to empirically the performance of the adaptive positive LASSO estimator. The empirical minimum, maximum, mean, medium, mode (for ARCH lag order only), standard error, bias, MSE, MAD, and selection proportion were summarized. The definitions of empirical bias, MSE, and MAD are listed below for reference:

$$\widehat{Bias}(\hat{q}^{dapL}) = \hat{E}[\hat{q}^{dapL}] - q = \frac{1}{M}\sum_{m=1}^{M}(\hat{q}^{dapL})^{(m)} - q$$

$$\widehat{MSE}(\hat{q}^{dapL}) = \hat{E}[\hat{q}^{dapL} - q]^2 = \frac{1}{M}\sum_{m=1}^{M}((\hat{q}^{dapL})^{(m)} - q)^2$$

$$\widehat{MAD}(\hat{q}^{dapL}) = \hat{E}|\hat{q}^{dapL} - q| = \frac{1}{M}\sum_{m=1}^{M}|(\hat{q}^{dapL})^{(m)} - q|$$

$$\widehat{Bias}(\hat{\theta}_j^{dapL}) = \hat{E}[\hat{\theta}_j^{dapL}] - \theta_j^* = \frac{1}{M}\sum_{m=1}^{M}(\hat{\theta}_j^{dapL})^{(m)} - \theta_j^*$$

$$\widehat{MSE}(\hat{\theta}_j^{dapL}) = \hat{E}[\hat{\theta}_j^{dapL} - \theta_j^*]^2 = \frac{1}{M}\sum_{m=1}^{M}\left((\hat{\theta}_j^{dapL})^{(m)} - \theta_j^*\right)^2$$

$$\widehat{MAD}(\hat{\theta}_j^{dapL}) = \hat{E}|\hat{\theta}_j^{dapL} - \theta_j^*| = \frac{1}{M}\sum_{m=1}^{M}|(\hat{\theta}_j^{dapL})^{(m)} - \theta_j^*|$$

where M denotes the total number of MC runs.

We use the function *mvBEKK.sim* in R package *mgarch* developed by Schmidbauer and Tunalioglu to generate 44 data sets of sample size $T = 1000$ from the following sparse trivariate BEKK VARCH(2) model.

$$\boldsymbol{H}_t = \boldsymbol{C}\boldsymbol{C}' + \boldsymbol{A}_1\boldsymbol{y}_{t-1}\boldsymbol{y}_{t-1}'\boldsymbol{A}_1' + \boldsymbol{A}_2\boldsymbol{y}_{t-2}\boldsymbol{y}_{t-2}'\boldsymbol{A}_2', \tag{5.24}$$

where

$$\boldsymbol{C} = \begin{pmatrix} 0.75 & 0 & 0 \\ 0.16 & 0.68 & 0 \\ 0.34 & 0 & 0.47 \end{pmatrix}, \boldsymbol{A}_1 = \begin{pmatrix} 0.32 & 0 & 0.35 \\ 0 & 0.27 & 0 \\ 0.18 & 0 & 0.45 \end{pmatrix}, \boldsymbol{A}_2 = \begin{pmatrix} 0.23 & 0.25 & 0.46 \\ 0.14 & 0.31 & 0 \\ 0 & 0 & 0.35 \end{pmatrix}. \tag{5.25}$$

Pretending that we did not know the true lag order q, which is 2 in this case, of the underlying bivariate BEKK VARCH process, we set the maximum order $h = 4$. For the sake of simplicity we used $h = 4$ for all 44 models. To find an approximately optimal combination of λ_T, γ_0, γ_1, and γ_2, we used grid-search method and the BIC criteria. Specifically, let $\mathcal{G} = \lambda_T \times \gamma_0 \times \gamma_1 \times \gamma_2 = [0.25, 1.7]_{\Delta=0.25} \times 2 \times [1.0, 2.0]_{\Delta=0.25} \times [1.0, 2.0]_{\Delta=0.25}$. [4] For the sake of simplicity, the same 4-dimensional grid $\mathcal{G}$ was used for all 1000 models.

[4] Δ in the subscript represents the increment of the sequence.

We used R package *Rmpi* [5] (Yu, 2002) for parallel computing system to expedite the optimization process. For each data set generated from the model (5.24) of the size 1,000, we parallelized the optimization tasks on 150 grid nodes of tuning and weighting parameters and distributed 150 optimizations to 48 CPUs.

The optimization was very slow. For one data set, it would take about 24 hours for the clustering computing system with 48 CPU to fit 150 models (150 grid nodes) via the doubly adaptive LASSO procedure. The major reason for slow computation may be caused by the fact that the conditional variance matrix involved in the likelihood function depends on time index t, and often has to be inverted for all t in every iteration. Another reason might be that our coding was in R language. It would have been better if we had used, say, C language. In addition, the convergence is slow. We set maximum number of iteration steps to be 300. Quite a few of 150 optimizations had not converged yet when the number of iterations reached 300. And the BIC might choose non-convergent results.

Among 44 replications, 43% of times (19 runs) the BIC chose non-convergent results, 57% of times the BIC chose convergent results. Table 5.1 and Table 5.2 summarize for lag order estimates from these 25 convergent results. Table 5.3 summarizes the results for coefficients estimates from these 25 convergent results. Because we have only 25 replications that were convergent, we cannot reach a confirmatory results. But the tables do show some promising prospect.

[5] R package *Rmpi* is an interface, or wrapper, to MPI. It provides an interface to low-level MPI functions from R so that users do not have to know details of the MPI implementations (C or Fortran)

Table 5.1: Empirical distribution of the doubly adaptive LASSO estimates for the trivariate BEKK ARCH(2) order based on 25 convergent replications each of size T=1,000, generated from trivariate BEKK ARCH(2) model with coefficients defined in (5.25). Set h=4. Use the BIC to choose λ, γ_0, γ_1, and γ_2.

Lag Order	2	3	4
Percentage	72%	2%	24%

Table 5.2: Empirical statistics of the doubly adaptive LASSO estimates for the trivariate BEKK ARCH(2) order based on 25 convergent replications each of size T=1,000, generated from trivariate BEKK ARCH(2) model with coefficients defined in (5.25). Set h=4. Use the BIC to choose λ, γ_0, γ_1, and γ_2.

True	Minimum	Maximum	Mean	Median	Mode	SE	Bias	MSE	MAD
2	2	4	2.52	2	2	0.872	0.52	1	0.52

Table 5.3: Empirical statistics of the doubly adaptive LASSO estimates for the trivariate ARCH coefficients $\boldsymbol{C}$, $\boldsymbol{A_1}$ and $\boldsymbol{A_2}$ based on 25 convergent replications each of size T=1,000, generated from trivariate ARCH(2) model with coefficients defined in (5.25). Set h=4. Use the BIC to choose λ, γ_0, γ_1, and γ_2.

Coeff	True	Min	Max	Mean	Median	SE	Bias	MSE	MAD	Prop
C_{11}	0.75	0.6535	0.8052	0.7320	0.7296	0.0331	-0.0180	0.0014	0.0300	1
C_{21}	0.16	0.1077	0.2342	0.1630	0.1615	0.0320	0.0030	0.0010	0.0267	1
C_{31}	0.34	0.0891	0.7391	0.3094	0.3219	0.1150	-0.0306	0.0136	0.0738	1
C_{22}	0.68	0.6568	0.7592	0.6928	0.6805	0.0273	0.0128	0.0009	0.0199	1
C_{32}	0	0	0.0569	0.0082	0	0.0174	0.0082	0.0004	0.0082	0.24
C_{33}	0.47	0	0.5355	0.3022	0.4579	0.2332	-0.1678	0.0804	0.1875	0.64
$A_{11,1}$	0.32	0.2302	0.7653	0.3624	0.3561	0.0994	0.0424	0.0113	0.0651	1
$A_{21,1}$	0	0	0.1654	0.0149	0	0.0379	0.0149	0.0016	0.0149	0.24
$A_{31,1}$	0.18	0	0.8728	0.3782	0.3644	0.1419	0.1982	0.0586	0.2126	0.96
$A_{12,1}$	0	0	0.4601	0.0327	0	0.0978	0.0327	0.0103	0.0327	0.28
$A_{22,1}$	0.27	0.1341	0.3437	0.2588	0.2648	0.0561	-0.0112	0.0031	0.0443	1
$A_{32,1}$	0	0	0.1282	0.0196	0	0.0368	0.0196	0.0017	0.0196	0.28
$A_{13,1}$	0.35	0	0.2752	0.1753	0.1786	0.0554	-0.1747	0.0335	0.1747	0.96
$A_{23,1}$	0	0	0.0836	0.0069	0	0.0190	0.0069	0.0004	0.0069	0.16
$A_{33,1}$	0.45	0.0044	0.6044	0.4511	0.4736	0.1186	0.0011	0.0135	0.0826	1
$A_{11,2}$	0.23	0.0571	0.3837	0.2311	0.2225	0.0701	0.0011	0.0047	0.0490	1
$A_{21,2}$	0.14	0.1467	0.4123	0.2341	0.2265	0.0612	0.0941	0.0125	0.0941	1
$A_{31,2}$	0	0.3435	1.1475	0.5281	0.4861	0.1662	0.5281	0.3054	0.5281	1
$A_{12,2}$	0.25	0	0.2369	0.1215	0.1334	0.0642	-0.1285	0.0205	0.1285	0.88
$A_{22,2}$	0.31	0.1809	0.3842	0.3117	0.3175	0.0507	0.0017	0.0025	0.0392	1
$A_{32,2}$	0	0	0.0846	0.0068	0	0.0215	0.0068	0.0005	0.0068	0.16
$A_{13,2}$	0.46	0	0.0887	0.0061	0	0.0213	-0.4539	0.2065	0.4539	0.08
$A_{23,2}$	0	0	0.0343	0.0016	0	0.0069	0.0016	0.0000	0.0016	0.08
$A_{33,2}$	0.35	0.0972	0.7577	0.3469	0.3404	0.1109	-0.0031	0.0118	0.0616	1
$A_{11,3}$	0	0	0.2540	0.0153	0	0.0560	0.0153	0.0032	0.0153	0.08
$A_{21,3}$	0	0	0.1281	0.0051	0	0.0256	0.0051	0.0007	0.0051	0.04
$A_{31,3}$	0	0	0.4709	0.0414	0	0.1257	0.0414	0.0169	0.0414	0.12
$A_{12,3}$	0	0	0.2586	0.0178	0	0.0625	0.0178	0.0041	0.0178	0.08
$A_{22,3}$	0	0	0	0	0	0	0	0	0	0
$A_{32,3}$	0	0	0.5368	0.0822	0	0.1770	0.0822	0.0368	0.0822	0.2
$A_{13,3}$	0	0	0.0397	0.0016	0	0.0079	0.0016	0.0001	0.0016	0.04
$A_{23,3}$	0	0	0.0716	0.0041	0	0.0153	0.0041	0.0002	0.0041	0.08
$A_{33,3}$	0	0	0.2434	0.0097	0	0.0487	0.0097	0.0024	0.0097	0.04
$A_{11,4}$	0	0	0.1280	0.0101	0	0.0350	0.0101	0.0013	0.0101	0.08
$A_{21,4}$	0	0	0.0128	0.0005	0	0.0026	0.0005	0.0000	0.0005	0.04
$A_{31,4}$	0	0	0.4414	0.0570	0	0.1384	0.0570	0.0216	0.0570	0.16
$A_{12,4}$	0	0	0.1168	0.0047	0	0.0234	0.0047	0.0005	0.0047	0.04
$A_{22,4}$	0	0	0	0	0	0	0	0	0	0
$A_{32,4}$	0	0	0.4747	0.0441	0	0.1247	0.0441	0.0169	0.0441	0.12
$A_{13,4}$	0	0	0	0	0	0	0	0	0	0
$A_{23,4}$	0	0	0	0	0	0	0	0	0	0
$A_{33,4}$	0	0	0	0	0	0	0	0	0	0

Chapter 6

Discussion and Future Work

In the previous chapters we proposed the doubly adaptive LASSO methodology tailored to time series analysis, and we conducted asymptotic analysis and the simulation studies. The methodology seems to have some nice properties such as consistency and normality. Now, we are at the stage of discussion.

The adaptive LASSO and the doubly adaptive LASSO approaches are both computationally intensive due to choosing hyper-parameters over a grid. The latter is even more computationally costly than the former because two additional weighting parameters are involved in the latter. Although it shows promising results in modeling time series data, the doubly adaptive LASSO methodology has much higher computational costs compared to the cost incurred by the adaptive LASSO.

It is also worth mentioning the fact that this thesis deals with those processes with fixed parameters and fixed lag order only. Readers are also notified that asymptotic properties of the doubly adaptive LASSO estimators in this thesis are not uniform but pointwise. This thesis did not answer the criticism reviewed in Section 1.2.6.

We introduced the notion of the surrogate of approximated likelihood in order to implement the algorithm for maximizing likelihood function. Although the algorithm may be relatively robust, we do not have the mathematical justification for using the surrogate of second-order likelihood approximation. We have not assessed the computational performance of the algo-

rithm. In addition, the surrogate may be one reason for slow convergence of BEKK VARCH(q) optimization. We need to develop robust, stable and efficient computational algorithms for optimization of likelihood function.

The doubly adaptive LASSO seems to excel in identifying the correct lag order of a time series model. By construction, the doubly adaptive LASSO gives more favor to recent values, which is natural and reasonable because more recent values are more relevant in prediction. But it may give unduly favor to those autoregressors that are recent but irrelevant. As we have seen in the simulation studies, the doubly adaptive LASSO tend to include more recent but insignificant autoregressors in a model with a high probability. One solution for this problem is to adopt two-step adaptive LASSO approach, namely, in the first step, one may identify lag order, as in classical methodology, and in the second step, one may apply the adaptive LASSO methods to get a sparse AR model.

The cross-validation seems to be not only computationally costly, but also difficult to implement for time series anlysis. The BIC criteria has been reported to perform variable selection much better than other approaches and the BIC seem more appropriate and more feasible for time series models, but we are not clear what the mathematical reasoning stands behind our favour for it. How to select optimum tuning and weighting parameters is an open question.

The results on doubly adaptive LASSO estimator for BEKK VARCH(q) models are only preliminary. We have not shown the oracle properties for the double adaptive LASSO estimator for BEKK VARCH(q) models. The results from simulation study is very limited albeit they do show promising prospect. In the future, we will investigate oracle properties.

In thesis, due to time constraint, we did not conduct empirical studies and comparative studies. For example, how to compare the forecast abilities between the doubly adaptive LASSO estimators and other estimators such as the QML estimator, the SCAD estimator, the adaptive LASSO estimator. In the future we will conduct empirical and comparative studies.

In this thesis, we did not touch upon the inferential issues. To attach forecast intervals to point estimators are common practice, which requires the computation of standard errors. We did not investigate the issue of standard errors. We did not touch on important issues on statistical tests, p-value, and so on.

R package is needed to facilitate applications of the doubly adaptive LASSO to practical data analysis. We need to develop R package for this purpose in the future.

There are maby models for time series analysis. It is possible to extend the doubly adaptive LASSO to other models such as ARMA(p,q), GARCH(p,q), VARMA(p,q), BEKK VGARCH(p,q,k), and so on.

Appendix A

Some Definitions and Theorems in Probability

A.1 Stationarity

Definition (**Univariate strict stationarity**). The time series $y_t, t \in \mathbb{Z} = \{0, \pm 1, \pm 2, \cdots\}$ is said to be strictly stationary if the joint distribution of $(y_{t_1}, \cdots, y_{t_k})'$ and $(y_{t_1+h}, \cdots, y_{t_k+h})'$ are the same for all $k \in \mathbb{Z}^+$, and $t_1, \cdots, t_k, h \in \mathbb{Z}$.

Definition (**Multivariate strict stationarity**). The K-variate time series $\boldsymbol{y}_t, t \in \mathbb{Z} = \{0, \pm 1, \pm 2, \cdots\}$ is said to be strictly stationary if the joint distribution of $(\boldsymbol{y}_{t_1}, \cdots, \boldsymbol{y}_{t_k})'$ and $(\boldsymbol{y}_{t_1+h}, \cdots, \boldsymbol{y}_{t_k+h})'$ are the same for all $k \in \mathbb{Z}^+$, and $t_1, \cdots, t_k, h \in \mathbb{Z}$.

Definition (**Univariate second-order stationarity**). The time series $y_t, t \in \mathbb{Z} = \{0, \pm 1, \pm 2, \cdots\}$ is said to be covariance stationary if its first and second moments are time invariant, namely, (i) $E[y_t] = \mu$ for all $t \in \mathbb{Z}$, with μ is a constant, (ii) $Var[X_t] < \infty$, and (iii) the autocovariace function $\text{Cov}(y_u, y_v) = \gamma(v-u)$ where $\gamma(v-u)$ is a function only of $v-u$.

Definition (**Multivariate second-order stationarity**). The K-variate time series $\boldsymbol{y}_t = (y_{1t}, \cdots, y_{Kt})'$, $t \in \mathbb{Z} = \{0, \pm 1, \pm 2, \cdots\}$ said to be covariance stationary if its first and second moments are time invariant, namely, (i) $E[\boldsymbol{y}_t] = (E[y_{1t}], E[y_{2t}], \cdots, E[y_{Kt}])' = \boldsymbol{\mu} = (\mu_1, \mu_2, \cdots, \mu_K)'$ is a constant vector, and (ii) the cross-covariance between y_{iu} and y_{jv} for all $i, j = 1, \cdots, K$ are functions only of $(v-u)$, or $Cov(\boldsymbol{y}_u, \boldsymbol{y}_v) = \Gamma(v-u)$, where $\Gamma(s)$ is the lag-s corss-covariance matrix function for

$\boldsymbol{y}_t$ defined as

$$\Gamma(s) = \text{Cov}(\boldsymbol{y}_t, \boldsymbol{y}_{t+s}) = \text{Cov}(\boldsymbol{y}_{t-s}, \boldsymbol{y}_t) = \begin{pmatrix} \gamma_{11}(s) & \gamma_{12}(s) & \cdots & \gamma_{1K}(s) \\ \gamma_{21}(s) & \gamma_{22}(s) & \cdots & \gamma_{2K}(s) \\ \vdots & \vdots & & \vdots \\ \gamma_{K1}(s) & \gamma_{K2}(s) & \cdots & \gamma_{KK}(s) \end{pmatrix}, \tag{A.1}$$

where

$$\gamma_{ij}(s) = E[(y_{it} - \mu_i)(y_{j,t+s} - \mu_j)] = E[(y_{i,t-s} - \mu_i)(y_{jt} - \mu_j)]$$

for $s = 0, \pm 1, \pm 2, \cdots$, $i = 1, \cdots, K$ and $j = 1, \cdots, K$.

Note that $\gamma_{ii}(s)$ is the autocovariance function for the ith component process y_{it}, and $\gamma_{ij}(s), i \neq j$ is the cross-covariance function between the ith and jth component processes. Also note that $\Gamma(0)$ is the contemporaneous variance and covariance matrix of the vector process. The lag-s autocorrelation matrix function $\boldsymbol{\rho}(s)$ for the vector process $\boldsymbol{y}_t$ is accordingly defined as $\boldsymbol{\rho}(s) = D^{-1/2}\Gamma(s)D^{-1/2}$, where $D = \text{diag}(\gamma_{11}(0), \gamma_{22}(0), \cdots, \gamma_{KK}(0))$. The autocovariance matrix function and the autocorrelation matrix function are positive semidefinite. Note that $\Gamma(s) = \Gamma'(-s)$, and $\boldsymbol{\rho}(s) = \boldsymbol{\rho}'(-s)$.

Note that no moment conditions are required for the definition of strict stationarity. Therefore, strict stationarity does not necessarily imply second-order stationarity. Note also that second-order stationarity does not imply strict stationarity.

A.2 White Noise

Definition (**Univariate white noise**). The time series $\epsilon_t, t \in \mathbb{Z} = \{0, \pm 1, \pm 2, \cdots\}$ is said to be a white noise process, written as

$$\epsilon_t \sim \text{WN}\left(0,\ \sigma_\epsilon^2\right), \tag{A.2}$$

if $E[\epsilon_t] = 0$, $E\left[\epsilon_t^2\right] = \sigma_\epsilon^2$, and $E\left[\epsilon_t \epsilon_{t-j}\right] = 0, \forall j \neq 0$.

Definition (**Multivariate white noise**). The K-variate time series $\boldsymbol{\epsilon}_t, t \in \mathbb{Z} = \{0, \pm 1, \pm 2, \cdots\}$ is said to be a vector white noise process, written as

$$\boldsymbol{\epsilon}_t \sim \text{WN}_K\left(\mathbf{0},\ \Sigma_\epsilon\right), \tag{A.3}$$

if it satisfies $E[\boldsymbol{\epsilon}_t] = \mathbf{0}$, $E\left[\boldsymbol{\epsilon}_t \boldsymbol{\epsilon}_t'\right] = \Sigma_\epsilon$, which is positive definite, and $E\left[\boldsymbol{\epsilon}_t \boldsymbol{\epsilon}_{t-j}'\right] = \mathbf{0}, \forall j \neq 0$.

A.3 Ergodicity

Definition (**Ergodicity**)[1] A strictly stationary process $\{y_t\}$ is said to be ergodic if, for any two bounded functions $f : \mathbb{R}^{k+1} \mapsto \mathbb{R}$ and $f : \mathbb{R}^{l+1} \mapsto \mathbb{R}$,

$$\lim_{s\to\infty} |E[g(y_t,\cdots,y_{t+k})g(y_{t+s},\cdots,y_{t+s+l})]| = |E[f(y_t,\cdots,y_{t+k})]| \times |E[g(y_{t+s},\cdots,y_{t+s+l})]|.$$

A strictly stationary process that is ergodic is said to be ergodic stationary.

Note that the definition of ergodicity does not require the existence of moments of $\{y_t\}$.

Theorem A.3.1 ***(Ergodicity of functions)***[2]. *Let $\boldsymbol{f}$ be a $\mathcal{F}$-measurable function into $\mathbb{R}^k$ and define $\boldsymbol{z}_t = f(\cdots, \boldsymbol{y}_t, \boldsymbol{y}_{t-1}, \cdots)$, where $\boldsymbol{y}_t$ is $q \times 1$ vector. (i) If $\{\boldsymbol{y}_t\}$ is stationary, then $\{\boldsymbol{z}_t\}$ is stationary. (ii) If $\{y_t\}$ be ergodic stationary, then $\{z_t\}$ is ergodic stationary.*

See Stout (1974) p.182 for proof.

Theorem A.3.2 ***(Ergodic theorem)***. *Let $\{y_t\}$ be ergodic stationary with $E[y_t] = \mu < \infty$. Then*

$$\bar{y}_T = \frac{1}{T}\sum_{t=1}^{T} y_t \longrightarrow \mu \ \ a.s. \ as \ T \to \infty.$$

Let the K-variate vector process $\{\boldsymbol{y}_t\}$ be ergodic stationary with $E[\boldsymbol{y}_t] = \boldsymbol{\mu}$ where $E[y_{i,t}] = \mu_i < \infty$ for all $i = 1, \cdots, K$. Then

$$\bar{\boldsymbol{y}}_T = \frac{1}{T}\sum_{t=1}^{T} \boldsymbol{y}_t \longrightarrow \boldsymbol{\mu} \ \ a.s.$$

See Stout (1974) p.181 for proof. The ergodic theorem says that the time average of an ergodic stationary process converges to the ensemble mean almost surely.

A.4 Martingale Difference

Definition (**Martingale difference**). $\{\boldsymbol{\nu}_t\}$ is said to be a sequence of vector martingale differences (MDS) if and only if

$$E[\boldsymbol{\nu}_{t+1}|\mathcal{F}_t] = \mathbf{0}.$$

[1] See Hayashi (2000) p.101.
[2] See White (1999) p.39-46.

Theorem A.4.1 *(**The CLT for ergodic stationary MDS** (Billingsley, 1961)). Let* $\{\boldsymbol{\nu}_t\}$ *be an ergodic stationary sequence of square integrable martingale difference vectors such that* $Var[\boldsymbol{\nu}_t] \equiv \Sigma_\nu^2$ *whose all entries exist and finite , Then*

$$\frac{1}{\sqrt{T}} \sum_{t=1}^{T} \boldsymbol{\nu}_t \xrightarrow{D} N(\mathbf{0}, \ \Sigma_\nu^2).$$

See Billingsley (1961) for proof.

A.5 Stochastic Boundedness

Definition (**Stochastic Boundedness**). A sequence of random variables $\{X_t\}$ is said to be stochastically bounded if $\forall \epsilon \in (0,1)$ $\exists M \in (0,\infty)$ such that $\inf_{t\geq 1} P(|X_t| \leq M) > 1-\epsilon$, denoted by $X_t = O_p(1)$.

Note that $X_t = O_p(a_t)$ with a_t being a sequence of variables means that X_t/a_t is stochastically bounded.

The necessity of stochastic boundedness for convergence in law follows from the following theorem.

Theorem A.5.1 . *Convergence in distribution implies stochastic boundedness.*

See, for example, Bierens p.158 for proof.

Appendix B

Some Definitions and Formulae in Matrix Calculus

(1) The Kronecker product

Let $A = (a_{ij})$ and $B = (b_{ij})$ be $m \times n$ and $p \times q$ matrices, respectively. The $mp \times nq$ matrix

$$A \otimes B = \begin{pmatrix} a_{11}B & \cdots & a_{1n}B \\ \vdots & & \vdots \\ a_{m1}B & \cdots & a_{mn}B \end{pmatrix}$$

is the Kronecker product of A and B.

(2) The *vec* and *vech* operators

The *vec* operator transforms an $m \times n$ matrix into an $mn \times 1$ vector by stacking the columns. The *vech* operator transforms an $m \times m$ square matrix into an $m(m+1)/2 \times 1$ vector by stacking the entries on and below the main diagonal. For example,

$$vec\begin{pmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \end{pmatrix} = [a_{11}\ a_{21}\ a_{12}\ a_{22}\ a_{13}\ a_{23}]',$$

$$vech\begin{pmatrix} b_{11} & b_{12} & b_{13} \\ b_{21} & b_{22} & b_{23} \\ b_{31} & b_{32} & b_{33} \end{pmatrix} = [b_{11}\ b_{21}\ b_{31}\ b_{22}\ b_{32}\ b_{33}]'.$$

(3) Elimination matrix

For an $m \times m$ *square* matrix A, the *elimination matrix* L_m is an $m(m+1)/2 \times m^2$ matrix defined such that

$$vech(A) = L_m vec(A).$$

(3) Duplication matrix

For an $m \times m$ *symmetric* matrix B, the *duplication matrix* D_m is an $m^2 \times m(m+1)/2$ matrix defined such that

$$vec(B) = D_m vech(B).$$

The rank of D_m is $m(m+1)/2$. The matrix $D_m' D_m$ is invertible. Let D_m^+ be the Moore-Penrose inverse of D_m, namely,

$$D_m^+ = (D_m' D_m)^{-1} D_m'.$$

The *vec* and *vech* of the symmetric matrix B is also related by D_m^+ as follows

$$D_m^+ vec(B) = vech(B).$$

(3) Communication matrix

For an $m \times n$ matrix C, the *communication matrix* K_{mn} is an $mn \times mn$ matrix defined such that

$$vec(C') = K_{mn} vec(C),$$

or, equivalently,

$$vec(C) = K_{nm} vec(C').$$

(4)

$$K_{mm} = 2D_m D_m^+ - \boldsymbol{I}_{m^2}$$

(5)

$$\frac{\partial log|\boldsymbol{X}|}{\partial vec\boldsymbol{X}} = vec((\boldsymbol{X}^{-1})')$$

(6) Let $\boldsymbol{X}(m \times m)$ be lower triangular.

$$\begin{aligned} \frac{\partial vech(\boldsymbol{X}'\boldsymbol{X})}{\partial vech(\boldsymbol{X})'} &= 2D_m^+(\boldsymbol{I}_m \otimes \boldsymbol{X}')L_m' \\ \frac{\partial vech(\boldsymbol{X}\boldsymbol{X}')}{\partial vech(\boldsymbol{X})'} &= 2D_m^+(\boldsymbol{X} \otimes \boldsymbol{I}_m)L_m' \end{aligned}$$

(7) $\boldsymbol{x}_{m\times 1}$, $\boldsymbol{Y}_{n\times p} = \boldsymbol{Y}(\boldsymbol{x})$, $\boldsymbol{Z}_{p\times q} = \boldsymbol{Z}(\boldsymbol{x})$.

$$\frac{\partial vec(\boldsymbol{Y}\boldsymbol{Z})}{\partial \boldsymbol{x}'} = (\boldsymbol{I}_q \otimes \boldsymbol{Y})\frac{\partial vec\boldsymbol{Z}}{\partial \boldsymbol{x}'} + (\boldsymbol{Z}' \otimes \boldsymbol{I}_n)\frac{\partial vec\boldsymbol{Y}}{\partial \boldsymbol{x}'}$$

(8) $\boldsymbol{x}_{m\times 1}$, $\boldsymbol{A}_{s\times n}$, $\boldsymbol{Y}_{n\times p} = \boldsymbol{Y}(\boldsymbol{x})$, $\boldsymbol{B}_{p\times q}$, $\boldsymbol{Z}_{q\times r} = \boldsymbol{Z}(\boldsymbol{x})$, and $\boldsymbol{C}_{r\times k}$.

$$\frac{\partial vec(\boldsymbol{AYBZC})}{\partial \boldsymbol{x}'} = (\boldsymbol{C}\otimes\boldsymbol{AYB})\frac{\partial vec\boldsymbol{Z}}{\partial \boldsymbol{x}'} + (\boldsymbol{C}'\boldsymbol{Z}'\boldsymbol{B}'\otimes\boldsymbol{A})\frac{\partial vec\boldsymbol{Y}}{\partial \boldsymbol{x}'}$$

(9)

$$\begin{aligned}\frac{\partial vec(\boldsymbol{YBZ})}{\partial \boldsymbol{x}'} &= \frac{\partial vec(\boldsymbol{I}_n\boldsymbol{YBZI}_r)}{\partial \boldsymbol{x}'}\\ &= (\boldsymbol{I}_r\otimes\boldsymbol{I}_n\boldsymbol{YB})\frac{\partial vec\boldsymbol{Z}}{\partial \boldsymbol{x}'} + (\boldsymbol{I}_r\boldsymbol{Z}'\boldsymbol{B}'\otimes\boldsymbol{I}_n)\frac{\partial vec\boldsymbol{Y}}{\partial \boldsymbol{x}'}\\ &= (\boldsymbol{I}_r\otimes\boldsymbol{YB})\frac{\partial vec\boldsymbol{Z}}{\partial \boldsymbol{x}'} + (\boldsymbol{Z}'\boldsymbol{B}'\otimes\boldsymbol{I}_n)\frac{\partial vec\boldsymbol{Y}}{\partial \boldsymbol{x}'}\end{aligned}$$

(10) $\boldsymbol{x}_{m\times 1}$, $\boldsymbol{Y}_{n\times p} = \boldsymbol{Y}(\boldsymbol{x})$, $\boldsymbol{Z}_{p\times r} = \boldsymbol{Z}(\boldsymbol{x})$.

$$\frac{\partial(vec\boldsymbol{Y}\otimes vec\boldsymbol{Z})}{\partial \boldsymbol{x}'} = \frac{\partial vec\boldsymbol{Y}}{\partial \boldsymbol{x}'}\otimes vec\boldsymbol{Z} + vec\boldsymbol{Y}\otimes\frac{\partial vec\boldsymbol{Z}}{\partial \boldsymbol{x}'}$$

(11) $\boldsymbol{A}_{n\times m}$, $\boldsymbol{X}_{m\times m}$ nonsingular, $\boldsymbol{B}_{m\times p}$.

$$\frac{\partial vec(\boldsymbol{AX}^{-1}\boldsymbol{B})}{\partial vec\boldsymbol{X}'} = -\boldsymbol{B}'\boldsymbol{X}'^{-1}\otimes\boldsymbol{AX}^{-1}$$

(12) $\boldsymbol{B}_{r\times m}$, $\boldsymbol{X}_{m\times n}$, $\boldsymbol{C}_{n\times s}$, and $\boldsymbol{A}_{p\times q}$.

$$\begin{aligned}\frac{\partial vec(\boldsymbol{BXC}\otimes\boldsymbol{A})}{\partial vec\boldsymbol{X}'} &= (\boldsymbol{I}_s\otimes\boldsymbol{K}_{qr}\otimes\boldsymbol{I}_p)(\boldsymbol{C}'\otimes\boldsymbol{B}\otimes vec(\boldsymbol{A}))\frac{\partial vec(\boldsymbol{A})\otimes\boldsymbol{BXC}}{\partial vec\boldsymbol{X}'}\\ &= (\boldsymbol{I}_q\otimes\boldsymbol{K}_{sp}\otimes\boldsymbol{I}_r)(vec(\boldsymbol{A})\otimes\boldsymbol{C}'\otimes\boldsymbol{B})\end{aligned}$$

(13) $\boldsymbol{X}_{m\times m}$ lower triangular.

$$\frac{\partial vec(\boldsymbol{X})}{\partial vech(\boldsymbol{X})'} = \boldsymbol{L}_m'$$

Appendix C

The Partial Lag Autocorrelation Matrix Function

In his PhD dissertation, *Partial Lag Autocorrelation and Partial Process Autocorrelation for Vector Time Series, with applications*, Heyse (1985) defined the notion of partial lag autocorrelation (PLAC) matrix function, which serves as a diagnostic aid for determining the order of a vector autoregressive model. Heyse (1985) also proposed an recursive algorithm for computing the sample partial lag autocorrelation matrix. The PLAC function play an important role in the doubly adaptive LASSO for vector AR(p) and BEKK VARCH(q) processes, so we document the definition, derivation, estimation, programming of the partial lag autocorrelation matrix function. For more details, please see Heyse (1985) or Wei (2006, p.408 - 414).

C.1 Autocorrelation Matrix Function

Let $\boldsymbol{y}_t = (y_{1t}, \cdots, y_{Kt})^{'}, t = 0, \pm 1, \pm 2, \cdots$ be jointly stationary vector process such that $\mathrm{E}[y_{it}] = \mu_i$, and cross-covariance between y_{it} and y_{jt} for all $i, j = 1, \cdots, K$ are functions only of $(s-t)$. The mean of the vector process $\boldsymbol{y}_t$ is defined as

$$\mathrm{E}[\boldsymbol{y}_t] = \boldsymbol{\mu} = (\mu_1, \mu_2, \cdots, \mu_K)^{'}.$$

The lag-s autocovariance matrix function $\Gamma(s)$ for the vector process $\boldsymbol{y}_t$ is defined as

$$\Gamma(s) = \mathrm{Cov}(\boldsymbol{y}_t, \boldsymbol{y}_{t+s}) = \mathrm{Cov}(\boldsymbol{y}_{t-s}, \boldsymbol{y}_t) = \begin{pmatrix} \gamma_{11}(s) & \gamma_{12}(s) & \cdots & \gamma_{1K}(s) \\ \gamma_{21}(s) & \gamma_{22}(s) & \cdots & \gamma_{2K}(s) \\ \vdots & \vdots & & \vdots \\ \gamma_{K1}(s) & \gamma_{K2}(s) & \cdots & \gamma_{KK}(s) \end{pmatrix}, \tag{C.1}$$

where

$$\gamma_{ij}(s) = \mathrm{E}[(y_{it} - \mu_i)(y_{j,t+s} - \mu_j)] = \mathrm{E}[(y_{i,t-s} - \mu_i)(y_{jt} - \mu_j)]$$

for $s = 0, \pm 1, \pm 2, \cdots$, $i = 1, \cdots, K$ and $j = 1, \cdots, K$. Note that $\gamma_{ii}(s)$ is the autocovariance function for the ith component process y_{it}, and $\gamma_{ij}(s), i \neq j$ is the cross-covariance function between the ith and jth component processes. Also note that $\Gamma(0)$ is the contemporaneous variance and covariance matrix of the vector process.

The lag-s autocorrelation matrix function $\boldsymbol{\rho}(s)$ for the vector process $\boldsymbol{y}_t$ is defined as

$$\boldsymbol{\rho}(s) = D^{-1/2}\Gamma(s)D^{-1/2} = \begin{pmatrix} \rho_{11}(s) & \rho_{12}(s) & \cdots & \rho_{1K}(s) \\ \rho_{21}(s) & \rho_{22}(s) & \cdots & \rho_{2K}(s) \\ \vdots & \vdots & & \vdots \\ \rho_{K1}(s) & \rho_{K2}(s) & \cdots & \rho_{KK}(s) \end{pmatrix}, \tag{C.2}$$

where

$$D = \mathrm{diag}(\gamma_{11}(0), \gamma_{22}(0), \cdots, \gamma_{KK}(0))$$

$$\rho_{ij}(s) = \frac{\gamma_{ij}(s)}{\sqrt{\gamma_{ii}(0)\gamma_{jj}(0)}}$$

for $s = 0, \pm 1, \pm 2, \cdots$, $i = 1, \cdots, K$ and $j = 1, \cdots, K$. Note that $\rho_{ii}(s)$ is the autocorrelation function for the ith component process y_{it} whereas $\rho_{ij}(s), i \neq j$ is the cross-correlation function between the ith and jth component processes.

The autocovariance matrix function and the autocorrelation matrix function are positive semidefinite in the sense that

$$\sum_{i=1}^{n}\sum_{j=1}^{n} \boldsymbol{\alpha}_i^{'}\Gamma(t_i - t_j)\boldsymbol{\alpha}_j \geq 0,$$

$$\sum_{i=1}^{n}\sum_{j=1}^{n} \boldsymbol{\alpha}_i^{'}\boldsymbol{\rho}(t_i - t_j)\boldsymbol{\alpha}_j \geq 0,$$

for any set of time points $t_1, \cdots, t_n$ and any set of real vectors $\boldsymbol{\alpha}_1, \cdots, \boldsymbol{\alpha}_n$. The results follows immediately by evaluating the variance of $\sum_{i=1}^{n} \boldsymbol{\alpha}_i^{'}\boldsymbol{y}_{t_i}$ and its standardization.

Note that $\gamma_{ij}(s) \neq \gamma_{ij}(-s)$ for $i \neq j$, and hence $\Gamma(s) \neq \Gamma(-s)$. Instead, because $\gamma_{ij}(s) = \mathrm{E}[(y_{it} - \mu_i)(y_{j,t+s} - \mu_j)] = \mathrm{E}[(y_{j,t+s} - \mu_j)(y_{it} - \mu_i)] = \gamma_{ji}(-k)$, we have

$$\Gamma(s) = \Gamma^{'}(-s), \tag{C.3}$$

$$\boldsymbol{\rho}(s) = \boldsymbol{\rho}^{'}(-s). \tag{C.4}$$

C.2 Partial Lag Autocorrelation Matrix

In extending the partial autocorrelation concept to vector time series Heyse (1985) introduced the notion of partial lag autocorrelation matrix function, which is the autocorrelation matrix between the elements of $\boldsymbol{y}_t$ and $\boldsymbol{y}_{t+s}$, after removing the linear dependence of each on the intervening vectors $\boldsymbol{y}_{t+1}, \cdots, \boldsymbol{y}_{t+s-1}$. This is defined as the ordinary correlation between the elements of residuals,

$$\boldsymbol{u}_{s-1,t+s} = \boldsymbol{y}_{t+s} - (\Psi_{s-1,1}\boldsymbol{y}_{t+s-1} + \cdots + \Psi_{s-1,s-1}\boldsymbol{y}_{t+1}), \tag{C.5}$$

and

$$\boldsymbol{v}_{s-1,t} = \boldsymbol{y}_t - (\Theta_{s-1,1}\boldsymbol{y}_{t+1} + \cdots + \Theta_{s-1,s-1}\boldsymbol{y}_{t+s-1}). \tag{C.6}$$

The partial lag partial lag autocorrelation matrix function is defined as

$$\boldsymbol{P}(s) = D_{\boldsymbol{v}}(s)^{-1/2}\boldsymbol{V}_{\boldsymbol{vu}}(s)D_{\boldsymbol{u}}(s)^{-1/2}, \tag{C.7}$$

where

$$\begin{aligned} \boldsymbol{V}_{\boldsymbol{u}}(s) &= \mathrm{Var}[\boldsymbol{u}_{s-1,t+s}], \\ \boldsymbol{V}_{\boldsymbol{v}}(s) &= \mathrm{Var}[\boldsymbol{v}_{s-1,t}], \\ \boldsymbol{V}_{\boldsymbol{vu}}(s) &= \mathrm{Cov}(\boldsymbol{v}_{s-1,t}, \boldsymbol{u}_{s-1,t+s}), \end{aligned}$$

and $D_{\boldsymbol{v}}(s)$ and $D_{\boldsymbol{u}}(s)$ are the diagonal matrices of $\boldsymbol{V}_{\boldsymbol{v}}(s)$ and $\boldsymbol{V}_{\boldsymbol{u}}(s)$, respectively.

In the rest of this subsection that follows, we derive the expressions for $\boldsymbol{V}_{\boldsymbol{u}}(s)$, $\boldsymbol{V}_{\boldsymbol{v}}(s)$, and $\boldsymbol{V}_{\boldsymbol{vu}}(s)$. First we re-express $\boldsymbol{u}(s)$ and $\boldsymbol{v}(s)$ as

$$\boldsymbol{u}_{s-1,t+s} = \begin{cases} \boldsymbol{y}_{t+s} - \sum_{k=1}^{s-1} \Psi_{s-1,k}\boldsymbol{y}_{t+s-k} \\ \boldsymbol{y}_{t+1} \end{cases} = \begin{cases} \boldsymbol{y}_{t+s} - \boldsymbol{\Psi}(s)\boldsymbol{y}_t(s), & s \geq 2 \\ \boldsymbol{y}_{t+1}, & s = 1 \end{cases} \tag{C.8}$$

$$\boldsymbol{v}_{s-1,t} = \begin{cases} \boldsymbol{y}_t - \sum_{k=1}^{s-1} \Theta_{s-1,k}\boldsymbol{y}_{t+k} \\ \boldsymbol{y}_t \end{cases} = \begin{cases} \boldsymbol{y}_t - \boldsymbol{\Theta}(s)\boldsymbol{y}_t(s), & s \geq 2 \\ \boldsymbol{y}_t, & s = 1 \end{cases} \tag{C.9}$$

where the matrices $\boldsymbol{\Psi}(s)$ and $\boldsymbol{\Theta}(s)$, and the vector $\boldsymbol{y}_t(s)$ for $s \geq 2$ are defined as

$$\boldsymbol{\Psi}'(s) = \begin{pmatrix} \Psi'_{s-1,1} \\ \Psi'_{s-1,2} \\ \vdots \\ \Psi'_{s-1,s-1} \end{pmatrix}, \quad \boldsymbol{\Theta}'(s) = \begin{pmatrix} \Theta'_{s-1,s-1} \\ \Theta'_{s-1,s-2} \\ \vdots \\ \Theta'_{s-1,1} \end{pmatrix}, \quad \boldsymbol{y}_t(s) = \begin{pmatrix} \boldsymbol{y}_{t+s-1} \\ \boldsymbol{y}_{t+s-2} \\ \vdots \\ \boldsymbol{y}_{t+1} \end{pmatrix}.$$

Define the following matrices $\boldsymbol{A}(s)$, $\boldsymbol{B}(s)$, and $\boldsymbol{C}(s)$ for $s \geq 2$ using lag-k covariance matrices $\Gamma(k), k = 0, \cdots, s-1$:

$$\boldsymbol{A}(s) = \begin{pmatrix} \Gamma(0) & \Gamma'(1) & \cdots & \Gamma'(s-2) \\ \Gamma(1) & \Gamma(0) & \cdots & \Gamma'(s-3) \\ \vdots & \vdots & & \vdots \\ \Gamma(s-2) & \Gamma(s-3) & \cdots & \Gamma(0) \end{pmatrix}, \quad \boldsymbol{B}(s) = \begin{pmatrix} \Gamma'(s-1) \\ \Gamma'(s-2) \\ \vdots \\ \Gamma'(1) \end{pmatrix}, \quad \boldsymbol{C}(s) = \begin{pmatrix} \Gamma(1) \\ \Gamma(2) \\ \vdots \\ \Gamma(s-1) \end{pmatrix}.$$

We see that

$$\begin{aligned}
\mathrm{Var}[\boldsymbol{y}_t(s)] &= \mathrm{E}[\boldsymbol{y}_t(s)\boldsymbol{y}'_t(s)] \\
&= \mathrm{E}\begin{pmatrix} \boldsymbol{y}_{t+s-1}\boldsymbol{y}'_{t+s-1} & \boldsymbol{y}_{t+s-1}\boldsymbol{y}'_{t+s-2} & \cdots & \boldsymbol{y}_{t+s-1}\boldsymbol{y}'_{t+1} \\ \boldsymbol{y}_{t+s-2}\boldsymbol{y}'_{t+s-1} & \boldsymbol{y}_{t+s-2}\boldsymbol{y}'_{t+s-2} & \cdots & \boldsymbol{y}_{t+s-2}\boldsymbol{y}'_{t+1} \\ \vdots & \vdots & & \vdots \\ \boldsymbol{y}_{t+1}\boldsymbol{y}'_{t+s-1} & \boldsymbol{y}_{t+1}\boldsymbol{y}'_{t+s-2} & \cdots & \boldsymbol{y}_{t+1}\boldsymbol{y}'_{t+1} \end{pmatrix} \\
&= \begin{pmatrix} \Gamma(0) & \Gamma(-1) & \cdots & \Gamma(-(s-2)) \\ \Gamma(1) & \Gamma(0) & \cdots & \Gamma(-(s-3)) \\ \vdots & \vdots & & \vdots \\ \Gamma(s-2) & \Gamma(s-3) & \cdots & \Gamma(0) \end{pmatrix} = \boldsymbol{A}(s), \\
\mathrm{E}[\boldsymbol{y}_t(s)\boldsymbol{y}'_t] &= \mathrm{E}\begin{pmatrix} \boldsymbol{y}_{t+s-1}\boldsymbol{y}'_t \\ \boldsymbol{y}_{t+s-2}\boldsymbol{y}'_t \\ \vdots \\ \boldsymbol{y}_{t+1}\boldsymbol{y}'_t \end{pmatrix} = \begin{pmatrix} \Gamma(-(s-1)) \\ \Gamma(-(s-2)) \\ \vdots \\ \Gamma(-1) \end{pmatrix} = \begin{pmatrix} \Gamma'(s-1) \\ \Gamma'(s-2) \\ \vdots \\ \Gamma'(1) \end{pmatrix} = \boldsymbol{B}(s), \\
\mathrm{E}[\boldsymbol{y}_t(s)\boldsymbol{y}'_{t+s}] &= \mathrm{E}\begin{pmatrix} \boldsymbol{y}_{t+s-1}\boldsymbol{y}'_{t+s} \\ \boldsymbol{y}_{t+s-2}\boldsymbol{y}'_{t+s} \\ \vdots \\ \boldsymbol{y}_{t+1}\boldsymbol{y}'_{t+s} \end{pmatrix} = \begin{pmatrix} \Gamma(1) \\ \Gamma(2) \\ \vdots \\ \Gamma(s-1) \end{pmatrix} = \boldsymbol{C}(s).
\end{aligned}$$

The coefficients matrices $\boldsymbol{\Psi}_{s-1,k}$ and $\boldsymbol{\Theta}_{s-1,k}$ are those that minimize $\mathrm{E}\left[|\boldsymbol{u}_{s-1,t+s}|^2\right]$ and $\mathrm{E}\left[|\boldsymbol{v}_{s-1,t}|^2\right]$, respectively. Consider the minimization of

$$\begin{aligned}
\mathrm{E}\left[|\boldsymbol{u}_{s-1,t+s}|^2\right] &= \mathrm{E}\left[\left(\boldsymbol{y}_{t+s} - \boldsymbol{\Psi}(s)\boldsymbol{y}_t(s)\right)\left(\boldsymbol{y}_{t+s} - \boldsymbol{\Psi}(s)\boldsymbol{y}_t(s)\right)'\right] \\
&= \Gamma(0) - \boldsymbol{\Psi}(s)\boldsymbol{C}(s) - \boldsymbol{C}'(s)\boldsymbol{\Psi}'(s) + \boldsymbol{\Psi}(s)\boldsymbol{A}(s)\boldsymbol{\Psi}'(s),
\end{aligned}$$

$$\begin{aligned}
\mathrm{E}\left[|\boldsymbol{v}_{s-1,t+s}|^2\right] &= \mathrm{E}\left[(\boldsymbol{y}_t - \boldsymbol{\Theta}(s)\boldsymbol{y}_t(s))(\boldsymbol{y}_t - \boldsymbol{\Theta}(s)\boldsymbol{y}_t(s))'\right] \\
&= \Gamma(0) - \boldsymbol{\Theta}(s)\boldsymbol{B}(s) - \boldsymbol{B}'(s)\boldsymbol{\Theta}'(s) + \boldsymbol{\Theta}(s)\boldsymbol{A}(s)\boldsymbol{\Theta}'(s).
\end{aligned}$$

Taking the derivative with respect to the elements of $\boldsymbol{\Psi}(s)$ and $\boldsymbol{\Theta}(s)$ and setting the resulting equations equal to 0 gives, [Graham (1981, p.54)],

$$\boldsymbol{A}(s)\boldsymbol{\Psi}'(s) = \boldsymbol{C}(s), \tag{C.10}$$

$$\boldsymbol{A}(s)\boldsymbol{\Theta}'(s) = \boldsymbol{B}(s), \tag{C.11}$$

which are the multivariate normal equations for the autoregression of $\boldsymbol{y}_{t+s}$ and $\boldsymbol{y}_t$ on $\boldsymbol{y}_{t+s-1}, \cdots, \boldsymbol{y}_{t+1}$, respectively. Solving the system yields the multivariate linear regression coefficients

$$\boldsymbol{\Psi}'(s) = \boldsymbol{A}(s)^{-1}\boldsymbol{C}(s), \tag{C.12}$$

$$\boldsymbol{\Theta}'(s) = \boldsymbol{A}(s)^{-1}\boldsymbol{B}(s). \tag{C.13}$$

The linear combinations of $\boldsymbol{\Psi}(s)\boldsymbol{y}_t(s)$ and $\boldsymbol{\Theta}(s)\boldsymbol{y}_t(s)$ define the linear projections of $\boldsymbol{y}_{t+s}$ and $\boldsymbol{y}_t$ onto the space spanned by $\boldsymbol{y}_{t+1}, \cdots, \boldsymbol{y}_{t+s-1}$, respectively. Since

$$\begin{aligned}
\mathrm{E}[\boldsymbol{y}_t(s)\boldsymbol{u}_{s-1,t+s}] &= \mathrm{E}\left[\boldsymbol{y}_t(s)(\boldsymbol{y}_{t+s} - \boldsymbol{\Psi}(s)\boldsymbol{y}_t(s))'\right] = \boldsymbol{C}(s) - \boldsymbol{A}(s)\boldsymbol{\Psi}'(s) = \boldsymbol{0}, \\
\mathrm{E}[\boldsymbol{y}_t(s)\boldsymbol{v}_{s-1,t}] &= \mathrm{E}\left[\boldsymbol{y}_t(s)(\boldsymbol{y}_t - \boldsymbol{\Theta}(s)\boldsymbol{y}_t(s))'\right] = \boldsymbol{B}(s) - \boldsymbol{A}(s)\boldsymbol{\Theta}'(s) = \boldsymbol{0},
\end{aligned}$$

we have that $\boldsymbol{y}_t(s)$ and $\boldsymbol{u}_{s-1,t+s}$, and $\boldsymbol{y}_t(s)$ and $\boldsymbol{v}_{s-1,t}$ are both uncorrelated and

$$\begin{aligned}
\mathrm{Var}(\boldsymbol{y}_{t+s}) &= \Gamma(0) \\
&= \mathrm{Var}[\boldsymbol{u}_{s-1,t+s}] + \mathrm{Var}[\boldsymbol{\Psi}(s)\boldsymbol{y}_t(s)] \\
&= \boldsymbol{V}_{\boldsymbol{u}}(s) + \boldsymbol{\Psi}(s)\boldsymbol{A}(s)\boldsymbol{\Psi}'(s) \\
&= \boldsymbol{V}_{\boldsymbol{u}}(s) + \boldsymbol{\Psi}(s)\boldsymbol{C}(s), \\
\mathrm{Var}(\boldsymbol{y}_t) &= \Gamma(0) \\
&= \mathrm{Var}[\boldsymbol{v}_{s-1,t}] + \mathrm{Var}[\boldsymbol{\Theta}(s)\boldsymbol{y}_t(s)] \\
&= \boldsymbol{V}_{\boldsymbol{v}}(s) + \boldsymbol{\Theta}(s)\boldsymbol{A}(s)\boldsymbol{\Theta}'(s) \\
&= \boldsymbol{V}_{\boldsymbol{v}}(s) + \boldsymbol{\Theta}(s)\boldsymbol{B}(s), \\
\mathrm{Cov}(\boldsymbol{v}_{s-1,t}, \boldsymbol{u}_{s-1,t+s}) &= \mathrm{E}\left[(\boldsymbol{y}_t - \boldsymbol{\Theta}(s)\boldsymbol{y}_t(s))(\boldsymbol{y}_{t+s} - \boldsymbol{\Psi}(s)\boldsymbol{y}_t(s))'\right] \\
&= \Gamma(s) - \boldsymbol{\Theta}(s)\boldsymbol{C}(s) - \boldsymbol{B}'(s)\boldsymbol{\Psi}'(s) + \boldsymbol{\Theta}(s)\boldsymbol{A}(s)\boldsymbol{\Psi}'(s) \\
&= \Gamma(s) - \boldsymbol{B}'(s)\boldsymbol{\Psi}'(s),
\end{aligned}$$

so that the formulae for $\boldsymbol{V_u}(s)$, $\boldsymbol{V_v}(s)$, and $\boldsymbol{V_{vu}}(s)$ for $s \geq 2$ are

$$\boldsymbol{V_u}(s) = \Gamma(0) - \boldsymbol{\Psi}(s)\boldsymbol{C}(s) = \Gamma(0) - \sum_{k=1}^{s-1} \Psi_{s-1,k}\Gamma(k), \tag{C.14}$$

$$\boldsymbol{V_v}(s) = \Gamma(0) - \boldsymbol{\Theta}(s)\boldsymbol{B}(s) = \Gamma(0) - \sum_{k=1}^{s-1} \Theta_{s-1,k}\Gamma'(k), \tag{C.15}$$

$$\boldsymbol{V_{vu}}(s) = \Gamma(s) - \boldsymbol{B}'(s)\boldsymbol{\Psi}'(s) = \Gamma(s) - \sum_{k=1}^{s-1} \Gamma(s-k)\Psi'_{s-1,k}. \tag{C.16}$$

For the case $s = 1$ since there are no intervening vectors between $\boldsymbol{y}_t$ and $\boldsymbol{y}_{t+s}$ we have that

$$\begin{aligned} \boldsymbol{V_u}(1) &= \mathrm{Var}(\boldsymbol{y}_{t+1}) = \Gamma(0), \\ \boldsymbol{V_v}(1) &= \mathrm{Var}(\boldsymbol{y}_t) = \Gamma(0), \\ \boldsymbol{V_{vu}}(1) &= \mathrm{Cov}(\boldsymbol{y}_t, \boldsymbol{y}_{t+1}) = \Gamma(1), \end{aligned}$$

and

$$\boldsymbol{P}(1) = \boldsymbol{D}^{-1/2}\Gamma(1)\boldsymbol{D}^{-1/2} = \boldsymbol{\rho}(1),$$

where $\boldsymbol{D}$ is the diagonal matrix of $\Gamma(0)$, and $\boldsymbol{\rho}(1)$ the regular autocorrelation matrix at lag 1.

We call the $K \times K$ matrix $\boldsymbol{P}(s)$ the partial lag autocorrelation matrix at lag s, which is the autocorrelation matrix between the elements of $\boldsymbol{y}_t$ and $\boldsymbol{y}_{t+s}$ after the their linear dependence on the vectors at the intervening lags have been removed.

$\boldsymbol{P}(s)$, as a function of the lag s, is a vector extension of the partial autocorrelation function in the same manner as the autocorrelation matrix function is a vector extension of the autocorrelation function. In the case $K = 1$, the partial lag autocorrelation matrix function $\boldsymbol{P}(s)$ reduces to the partial autocorrelation function $P(s)$. To see this, notice that ,

$$\boldsymbol{A}(s) = \begin{pmatrix} \gamma(0) & \gamma(1) & \cdots & \gamma(s-2) \\ \gamma(1) & \gamma(0) & \cdots & \gamma(s-3) \\ \vdots & \vdots & & \vdots \\ \gamma(s-2) & \gamma(s-3) & \cdots & \gamma(0) \end{pmatrix}, \quad \boldsymbol{B}(s) = \begin{pmatrix} \gamma(s-1) \\ \gamma(s-2) \\ \vdots \\ \gamma(1) \end{pmatrix}, \quad \boldsymbol{C}(s) = \begin{pmatrix} \gamma(1) \\ \gamma(2) \\ \vdots \\ \gamma(s-1) \end{pmatrix},$$

$$\boldsymbol{\Psi}(s) = \begin{pmatrix} \psi_1 \\ \psi_2 \\ \vdots \\ \psi_{s-1} \end{pmatrix}, \quad \boldsymbol{\Theta}(s) = \begin{pmatrix} \Theta_{s-1} \\ \Theta_{s-2} \\ \vdots \\ \Theta_1 \end{pmatrix}.$$

So we have

$$\begin{aligned} V_u(s) &= \gamma(0) - \sum_{k=1}^{s-1} \psi_k \gamma(k), \\ V_v(s) &= \gamma(0) - \sum_{k=1}^{s-1} \theta_k \gamma(k), \\ V_{vu}(s) &= \gamma(s) - \sum_{k=1}^{s-1} \psi_k \gamma(s-k), \end{aligned}$$

and therefore

$$P(s) = \frac{V_{vu}(s)}{\sqrt{V_u(s)}\sqrt{V_v(s)}} = \frac{\gamma(s) - \sum_{k=1}^{s-1} \psi_k \gamma(s-k)}{\gamma(0) - \sum_{k=1}^{s-1} \psi_k \gamma(k)} = \frac{\rho(s) - \sum_{k=1}^{s-1} \psi_k \rho(s-k)}{1 - \sum_{k=1}^{s-1} \psi_k \rho(k)},$$

which is exactly the formula for the partial autocorrelation function at lag s.

Analogous to the partial autocorrelation function for the univariate case the partial lag autocorrelation matrix, $\boldsymbol{P}(s)$ has the cut-off property for autoregressive processes. So if $\{\boldsymbol{y}_t\}$ is a vector autoregressive process of order p then $\boldsymbol{P}(s)$ will be nonzero for $s = p$ and will equal 0 for $s > p$. This property makes $\boldsymbol{P}(s)$ a useful tool for identifying VAR processes.

Before we start discuss the computing algorithm, we take an excursion to partial autoregression matrix defined by Tiao and Box (1981).

C.3 Partial Autoregression Matrix Function

Tiao and Box (1981) define the partial autoregression matrix at lag s for a vector time series $\{\boldsymbol{y}_t\}$ to be the last matrix coefficient when the data is fitted to a VAR process of order s. This is a direct extension of the Box and Jenkins (1976, p. 64) definition of the partial autocorrelation function for univariate time series. It is equal to $\Psi_{s,s}$ in the multivariate linear regression

$$\boldsymbol{y}_{t+s} = \Psi_{s,1}\boldsymbol{y}_{t+s-1} + \cdots + \Psi_{s,s}\boldsymbol{y}_t + \boldsymbol{e}_{s,t+s},$$

where the $K \times K$ matrix coefficients $\Psi_{s,k}, k = 1, 2, \cdots, s$ are those that minimize

$$\mathrm{E}\left[\left|\boldsymbol{y}_{t+s} - \Psi_{s,1}\boldsymbol{y}_{t+s-1} - \cdots - \Psi_{s,s}\boldsymbol{y}_t\right|^2\right].$$

Differentiating wrt $\Psi_{s,1}$ and then setting to $\mathbf{0}$ yields

$$\begin{aligned}\mathbf{0} &= \mathrm{E}\left[-\boldsymbol{y}_{t+s-1}\left(\boldsymbol{y}_{t+s}-\Psi_{s,1}\boldsymbol{y}_{t+s-1}-\cdots-\Psi_{s,s}\boldsymbol{y}_t\right)'\right] \\ &= -\Gamma(1)+\Gamma(0)\Psi'_{s,1}+\cdots+\Gamma(-(s-1))\Psi'_{s,s},\end{aligned}$$

or

$$\Gamma(0)\Psi'_{s,1}+\cdots+\Gamma'(s-1)\Psi'_{s,s}=\Gamma(1).$$

So by differentiating wrt all $\Psi_{s,k}$ matrix we get the Yule-Walker equations in unnormalized form,

$$\begin{pmatrix}\Gamma(0) & \Gamma'(1) & \cdots & \Gamma'(s-1)\\ \Gamma(1) & \Gamma(0) & \cdots & \Gamma'(s-2)\\ \vdots & \vdots & & \vdots\\ \Gamma(s-1) & \Gamma(s-2) & \cdots & \Gamma(0)\end{pmatrix}\begin{pmatrix}\Psi'_{s,1}\\ \Psi'_{s,2}\\ \vdots\\ \Psi'_{s,s}\end{pmatrix}=\begin{pmatrix}\Gamma(1)\\ \Gamma(2)\\ \vdots\\ \Gamma(s)\end{pmatrix},$$

or

$$\begin{pmatrix}\boldsymbol{A}(s) & \boldsymbol{B}(s)\\ \boldsymbol{B}'(s) & \Gamma(0)\end{pmatrix}\begin{pmatrix}\boldsymbol{\Psi}'_{s-1}\\ \Psi'_{s,s}\end{pmatrix}=\begin{pmatrix}\boldsymbol{C}(s)\\ \Gamma(s)\end{pmatrix},$$

where

$$\boldsymbol{\Psi}'_{s-1}=\begin{pmatrix}\Psi'_{s,1}\\ \Psi'_{s,2}\\ \vdots\\ \Psi'_{s,s-1}\end{pmatrix}.$$

Solving for $\Psi_{s,s}$ gives

$$\begin{aligned}\Psi'_{s,s} &= \left(\Gamma(0)-\boldsymbol{B}'(s)\boldsymbol{A}(s)^{-1}\boldsymbol{B}(s)\right)^{-1}\left(\Gamma(s)-\boldsymbol{B}'(s)\boldsymbol{A}(s)^{-1}\boldsymbol{B}(s)\right)\\ &= \left(\Gamma(0)-\boldsymbol{B}'(s)\boldsymbol{\Theta}'(s)\right)^{-1}\left(\Gamma(s)-\boldsymbol{B}'(s)\boldsymbol{\Psi}'(s)\right),\end{aligned}$$

or

$$\Psi_{s,s}=\left(\Gamma(s)-\boldsymbol{B}'(s)\boldsymbol{\Psi}'(s)\right)'\left(\Gamma(0)-\boldsymbol{\Theta}(s)\boldsymbol{B}(s)\right)^{-1}=\boldsymbol{V}'_{\boldsymbol{vu}}(s)\boldsymbol{V}_{\boldsymbol{v}}(s)^{-1}. \tag{C.17}$$

For $s=1$, $\Psi_{s,s}=\Gamma'(1)\Gamma(0)^{-1}$.

Similarly, we can also compute $\Theta_{s,s}$ in the multivariate linear regression

$$\boldsymbol{y}_t=\Theta_{s,1}\boldsymbol{y}_{t+1}+\cdots+\Theta_{s,s}\boldsymbol{y}_{t+s}+\boldsymbol{e}_{s,t},$$

where the $K \times K$ matrix coefficients $\Theta_{s,k}, k = 1, 2, \cdots, s$ are those that minimize

$$\mathrm{E}\left[\left|\boldsymbol{y}_t - \Theta_{s,1}\boldsymbol{y}_{t+1} - \cdots - \Theta_{s,s}\boldsymbol{y}_{t+s}\right|^2\right].$$

Differentiating wrt $\Theta_{s,1}$ and then setting to $\mathbf{0}$ yields

$$\begin{aligned}\mathbf{0} &= \mathrm{E}\left[-\boldsymbol{y}_{t+1}\left(\boldsymbol{y}_t - \Theta_{s,1}\boldsymbol{y}_{t+1} - \cdots - \Theta_{s,s}\boldsymbol{y}_{t+s}\right)'\right] \\ &= -\Gamma(-1) + \Gamma(0)\Theta'_{s,1} + \cdots + \Gamma(s-1)\Theta'_{s,s},\end{aligned}$$

or

$$\Gamma(0)\Theta'_{s,1} + \cdots + \Gamma(s-1)\Theta'_{s,s} = \Gamma'(1).$$

So by differentiating wrt all $\Theta_{s,k}$ matrix we get the Yule-Walker equations in unnormalized form,

$$\begin{pmatrix} \Gamma(0) & \Gamma'(1) & \cdots & \Gamma'(s-1) \\ \Gamma(1) & \Gamma(0) & \cdots & \Gamma'(s-2) \\ \vdots & \vdots & & \vdots \\ \Gamma(s-1) & \Gamma(s-2) & \cdots & \Gamma(0) \end{pmatrix} \begin{pmatrix} \Theta'_{s,s} \\ \Theta'_{s,s-1} \\ \vdots \\ \Theta'_{s,1} \end{pmatrix} = \begin{pmatrix} \Gamma'(s) \\ \Gamma'(s-1) \\ \vdots \\ \Gamma'(1) \end{pmatrix},$$

or

$$\begin{pmatrix} \Gamma(0) & \boldsymbol{C}'(s) \\ \boldsymbol{C}(s) & \boldsymbol{A}(s) \end{pmatrix} \begin{pmatrix} \Theta'_{s,s} \\ \boldsymbol{\Theta}'_{s-1} \end{pmatrix} = \begin{pmatrix} \Gamma'(s) \\ \boldsymbol{B}(s) \end{pmatrix},$$

where

$$\boldsymbol{\Theta}'_{s-1} = \begin{pmatrix} \Theta'_{s,s-1} \\ \Theta'_{s,s-2} \\ \vdots \\ \Theta'_{s,1} \end{pmatrix}.$$

Solving for $\Theta_{s,s}$ gives

$$\begin{aligned}\Theta'_{s,s} &= \left(\Gamma(0) - \boldsymbol{C}'(s)\boldsymbol{A}(s)^{-1}\boldsymbol{C}(s)\right)^{-1}\left(\Gamma(s) - \boldsymbol{C}'(s)\boldsymbol{A}(s)^{-1}\boldsymbol{B}(s)\right) \\ &= \left(\Gamma(0) - \boldsymbol{C}'(s)\boldsymbol{\Psi}'(s)\right)^{-1}\left(\Gamma'(s) - \boldsymbol{\Psi}(s)\boldsymbol{B}(s)\right),\end{aligned}$$

or

$$\Theta_{s,s} = \left(\Gamma(s) - \boldsymbol{B}'(s)\boldsymbol{\Psi}'(s)\right)(\Gamma(0) - \boldsymbol{\Psi}(s)\boldsymbol{C}(s))^{-1} = \boldsymbol{V}_{\boldsymbol{vu}}(s)\boldsymbol{V}_{\boldsymbol{u}}(s)^{-1} \tag{C.18}$$

For $s = 1$, $\Theta_{s,s} = \Gamma(1)\Gamma(0)^{-1}$.

C.4 Recursive Algorithm

The recursive procedure for computing partial lag autocorrelation matrices introduced by Heyse (1985) is a vector generalization of Durbin's (1960) recursive computational procedure for univariate partial autocorrelations.

From the previous subsection we have that for $s \geq 2$,

$$\begin{aligned} \boldsymbol{y}_{t+s} &= \sum_{k=1}^{s-1} \Psi_{s-1,k} \boldsymbol{y}_{t+s-k} + \boldsymbol{u}_{s-1,t+s}, \\ \boldsymbol{y}_t &= \sum_{k=1}^{s-1} \Theta_{s-1,k} \boldsymbol{y}_{t+k} + \boldsymbol{v}_{s-1,t}. \end{aligned}$$

Consider the regressions

$$\begin{aligned} \boldsymbol{y}_{t+s+1} &= \sum_{k=1}^{s} \Psi_{s,k} \boldsymbol{y}_{t+s+1-k} + \boldsymbol{u}_{s,t+s+1}, \\ \boldsymbol{y}_t &= \sum_{k=1}^{s} \Theta_{s,k} \boldsymbol{y}_{t+k} + \boldsymbol{v}_{s,t}. \end{aligned}$$

Corresponding to the definition of the partial lag autocorrelation matrix, our interest is in the autocorrelation between $\boldsymbol{v}_{s,t}$ and $\boldsymbol{u}_{s,t+s+1}$ and for this we need to compute the multivariate linear regression coefficients $\Psi_{s,k}$ and $\Omega_{s,k}$. Let

$$\boldsymbol{u}_{s-1,t+s} = \Psi_s^* \boldsymbol{v}_{s-1,t} + \boldsymbol{u}_{t+s}^* \tag{C.19}$$

$$\boldsymbol{v}_{s-1,t} = \Theta_s^* \boldsymbol{u}_{s-1,t+s} + \boldsymbol{v}_t^* \tag{C.20}$$

where

$$\Psi_s^* = \mathrm{Cov}(\boldsymbol{u}_{s-1,t+s}, \boldsymbol{v}_{s-1,t}) \mathrm{Var}(\boldsymbol{v}_{s-1,t})^{-1} = \boldsymbol{V}_{\boldsymbol{vu}}'(s) \boldsymbol{V}_{\boldsymbol{v}}(s)^{-1} \tag{C.21}$$

$$\Theta_s^* = \mathrm{Cov}(\boldsymbol{v}_{s-1,t}, \boldsymbol{u}_{s-1,t+s}) \mathrm{Var}(\boldsymbol{u}_{s-1,t})^{-1} = \boldsymbol{V}_{\boldsymbol{vu}}(s) \boldsymbol{V}_{\boldsymbol{u}}(s)^{-1} \tag{C.22}$$

Note that

$$\Psi_s^* = \Psi_{s,s}, \qquad \Theta_s^* = \Theta_{s,s} \tag{C.23}$$

$$\boldsymbol{u}_{t+s}^* = \boldsymbol{u}_{s,t+s+1}, \quad \boldsymbol{v}_t^* = \boldsymbol{v}_{s,t+1}. \tag{C.24}$$

So we have

$$\begin{aligned}\boldsymbol{u}_{s-1,t+s} &= \Psi_{s,s}\boldsymbol{v}_{s-1,t} + \boldsymbol{u}_{s,t+s+1} \\ \boldsymbol{v}_{s-1,t} &= \Theta_{s,s}\boldsymbol{u}_{s-1,t+s} + \boldsymbol{v}_{s,t+1}\end{aligned}$$

Substituting the expressions for $\boldsymbol{u}_{s-1,t+s}$ and $\boldsymbol{v}_{s-1,t}$ yields

$$\begin{aligned}\boldsymbol{y}_{t+s} - \sum_{k=1}^{s-1}\Psi_{s-1,k}\boldsymbol{y}_{t+s-k} &= \Psi_{s,s}(\boldsymbol{y}_t - \sum_{k=1}^{s-1}\Theta_{s-1,k}\boldsymbol{y}_{t+k}) + \boldsymbol{u}_{s,t+s+1} \\ \boldsymbol{y}_t - \sum_{k=1}^{s-1}\Theta_{s-1,k}\boldsymbol{y}_{t+k} &= \Theta_{s,s}(\boldsymbol{y}_{t+s} - \sum_{k=1}^{s-1}\Psi_{s-1,k}\boldsymbol{y}_{t+s-k}) + \boldsymbol{v}_{s,t+1}\end{aligned}$$

Rearranging the equations yields

$$\begin{aligned}\boldsymbol{y}_{t+s} &= \sum_{k=1}^{s-1}(\Psi_{s-1,k} - \Psi_{s,s}\Theta_{s-1,s-k})\boldsymbol{y}_{t+s-k} + \Psi_{s,s}\boldsymbol{y}_t + \boldsymbol{u}_{s,t+s+1} \\ \boldsymbol{y}_t &= \sum_{k=1}^{s-1}(\Theta_{s-1,k} - \Theta_{s,s}\Psi_{s-1,s-k})\boldsymbol{y}_{t+k} + \Theta_{s,s}\boldsymbol{y}_{t+s} + \boldsymbol{v}_{s,t+1}\end{aligned}$$

We thus have recursive formulae for $\Psi_{s,k}$ and $\Theta_{s,k}$:

$$\Psi_{s,k} = \Psi_{s-1,k} - \Psi_{s,s}\Theta_{s-1,s-k} \tag{C.25}$$

$$\Theta_{s,k} = \Theta_{s-1,k} - \Theta_{s,s}\Psi_{s-1,s-k} \tag{C.26}$$

Algorithm 12: Recursive algorithm for the partial lag autocorrelation matrix function

Input: Sample autocorrelation matrix function $\widehat{\Gamma}(s), s = 1, \cdots, h$

Output: Sample partial lag autocorrelation matrices $\widehat{\mathbf{P}}(s), s = 1, \cdots, h$

Start

$$
\begin{aligned}
\mathbf{V_u}(1) &= \mathbf{V_v}(1) = \Gamma(0) \\
\mathbf{V_{vu}}(1) &= \Gamma(1) \\
D_{\mathbf{u}}(1) &= D_{\mathbf{v}}(1) = \text{Diag}(\gamma_{11}(0), \cdots, \gamma_{KK}(0)] \\
\mathbf{P}(1) &= D_{\mathbf{v}}(1)^{-1/2} \mathbf{V_{vu}}(1) D_{\mathbf{u}}(1)^{-1/2} \\
\Psi_{1,1} &= \Gamma'(1)\Gamma(0)^{-1} \\
\Theta_{1,1} &= \Gamma(1)\Gamma(0)^{-1}
\end{aligned}
$$

for $s \leftarrow 2$ ***to*** h **do**

$$
\begin{aligned}
\mathbf{V_u}(s) &= \Gamma(0) - \sum_{k=1}^{s-1} \Psi_{s-1,k}\Gamma(k) \\
\mathbf{V_v}(s) &= \Gamma(0) - \sum_{k=1}^{s-1} \Theta_{s-1,k}\Gamma'(k) \\
\mathbf{V_{vu}}(s) &= \Gamma(s) - \sum_{k=1}^{s-1} \Gamma(s-k)\Psi'_{s-1,k} \\
D_{\mathbf{u}}(s) &= \text{Diag}([\mathbf{V_u}(s)]_{ii}, i = 1, \cdots, K) \\
D_{\mathbf{v}}(s) &= \text{Diag}([\mathbf{V_v}(s)]_{ii}, i = 1, \cdots, K) \\
\mathbf{P}(s) &= D_{\mathbf{v}}(s)^{-1/2} \mathbf{V_{vu}}(s) D_{\mathbf{u}}(s)^{-1/2} \\
\Psi_{s,s} &= \mathbf{V'_{vu}}(s)\mathbf{V_v}(s)^{-1} \\
\Psi_{s,k} &= \Psi_{s-1,k} - \Psi_{s,s}\Theta_{s-1,s-k}, \quad k = 1, \cdots, s-1 \\
\Theta_{s,s} &= \mathbf{V_{vu}}(s)\mathbf{V_u}(s)^{-1} \\
\Theta_{s,k} &= \Theta_{s-1,k} - \Theta_{s,s}\Psi_{s-1,s-k}, \quad k = 1, \cdots, s-1.
\end{aligned}
$$

Output $\widehat{\mathbf{P}}(s), s = 1, \cdots, h$

End

For example,

$$
\begin{aligned}
s=1: \quad \boldsymbol{V_u}(1) &= \Gamma(0) \\
\boldsymbol{V_v}(1) &= \Gamma(0) \\
\boldsymbol{V_{vu}}(1) &= \Gamma(1) \\
D_{\boldsymbol{u}}(1) &= \text{Diag}([\boldsymbol{V_u}(1)]_{ii}) = \text{Diag}(\gamma_{11}(0), \cdots, \gamma_{KK}(0)) \\
D_{\boldsymbol{v}}(1) &= \text{Diag}([\boldsymbol{V_v}(1)]_{ii}) = \text{Diag}(\gamma_{11}(0), \cdots, \gamma_{KK}(0))
\end{aligned}
$$

$$
\boldsymbol{P}(1) = D_{\boldsymbol{v}}(1)^{-1/2} \boldsymbol{V_{vu}}(1) D_{\boldsymbol{u}}(1)^{-1/2}
$$

$$
\begin{aligned}
\Psi_{1,1} &= \Gamma^{'}(1)\Gamma(0)^{-1} \\
\Theta_{1,1} &= \Gamma(1)\Gamma(0)^{-1}
\end{aligned}
$$

$$
\begin{aligned}
s=2: \quad \boldsymbol{V_u}(2) &= \Gamma(0) - \Psi_{1,1}\Gamma(1) \\
\boldsymbol{V_v}(2) &= \Gamma(0) - \Theta_{1,1}\Gamma^{'}(1) \\
\boldsymbol{V_{vu}}(2) &= \Gamma(2) - \Gamma(1)\Psi^{'}_{1,1} \\
D_{\boldsymbol{u}}(2) &= \text{Diag}([\boldsymbol{V_u}(2)]_{11}, \cdots, [\boldsymbol{V_u}(2)]_{KK}) \\
D_{\boldsymbol{v}}(2) &= \text{Diag}([\boldsymbol{V_v}(2)]_{11}, \cdots, [\boldsymbol{V_v}(2)]_{KK})
\end{aligned}
$$

$$
\boldsymbol{P}(2) = D_{\boldsymbol{v}}(2)^{-1/2} \boldsymbol{V_{vu}}(2) D_{\boldsymbol{u}}(2)^{-1/2}
$$

$$
\begin{aligned}
\Psi_{2,2} &= \boldsymbol{V^{'}_{vu}}(2)\boldsymbol{V_v}(2)^{-1} \\
\Psi_{2,1} &= \Psi_{1,1} - \Psi_{2,2}\Theta_{1,1} \\
\Theta_{2,2} &= \boldsymbol{V_{vu}}(2)\boldsymbol{V_u}(2)^{-1} \\
\Theta_{2,1} &= \Theta_{1,1} - \Theta_{2,2}\Psi_{1,1}
\end{aligned}
$$

$$
\begin{aligned}
s=3: \quad \boldsymbol{V_u}(3) &= \Gamma(0)-\Psi_{2,1}\Gamma(1)-\Psi_{2,2}\Gamma(2)\\
\boldsymbol{V_v}(3) &= \Gamma(0)-\Theta_{2,1}\Gamma'(1)-\Theta_{2,2}\Gamma'(2)\\
\boldsymbol{V_{vu}}(3) &= \Gamma(3)-\Gamma(2)\Psi'_{2,1}-\Gamma(1)\Psi'_{2,2}\\
D_{\boldsymbol{u}}(3) &= \text{Diag}([\boldsymbol{V_u}(3)]_{11},\cdots,[\boldsymbol{V_u}(3)]_{KK})\\
D_{\boldsymbol{v}}(3) &= \text{Diag}([\boldsymbol{V_v}(3)]_{11},\cdots,[\boldsymbol{V_v}(3)]_{KK})
\end{aligned}
$$

$$
\boldsymbol{P}(3) = D_{\boldsymbol{v}}(3)^{-1/2}\boldsymbol{V_{vu}}(3)D_{\boldsymbol{u}}(3)^{-1/2}
$$

$$
\begin{aligned}
\Psi_{3,3} &= \boldsymbol{V}'_{\boldsymbol{vu}}(3)\boldsymbol{V_v}(3)^{-1}\\
\Psi_{3,1} &= \Psi_{2,1}-\Psi_{3,3}\Theta_{2,2}\\
\Psi_{3,2} &= \Psi_{2,2}-\Psi_{3,3}\Theta_{2,1}\\
\Theta_{3,3} &= \boldsymbol{V_{vu}}(3)\boldsymbol{V_u}(3)^{-1}\\
\Theta_{3,1} &= \Theta_{2,1}-\Theta_{3,3}\Psi_{2,2}\\
\Theta_{3,2} &= \Theta_{2,2}-\Theta_{3,3}\Psi_{2,1}
\end{aligned}
$$

C.5 Estimation and Inference

Sample Autocorrelation Matrix

Given a sample realization $\boldsymbol{y}_1,\boldsymbol{y}_2,\cdots,\boldsymbol{y}_T$ of an K-dimensional vector time series the sample autocovariance matrix at lag s is computed by

$$
\widehat{\Gamma}(s)=\frac{1}{T}\sum_{t=1}^{T-s}(\boldsymbol{y}_t-\bar{\boldsymbol{y}})(\boldsymbol{y}_t-\bar{\boldsymbol{y}})',
$$

where $\bar{\boldsymbol{y}}$ is the vector of sample mean.

The sample autocorrelation matrix at lag s is computed by

$$
\widehat{\boldsymbol{\rho}}(s)=\widehat{D}^{-1/2}\widehat{\Gamma}(s)\widehat{D}^{-1/2},
$$

where $\widehat{D}$ is the diagonal matrix whose ith diagonal element is the ith diagonal element of $\widehat{\Gamma}(0)$.

Hannan (1970, p.228) showed that (i) $\widehat{\boldsymbol{\rho}}(s)$ is a consistent estimator for $\boldsymbol{\rho}(s)$, and (ii) $\widehat{\boldsymbol{\rho}}(s)$ is asymptotically normally distributed. Bartlett (1966) gives the asymptotic covariance between the estimates $\widehat{\rho}_{ij}(s)$ and $\widehat{\rho}_{ij}(s+1)$. For the case in which $\{\boldsymbol{z}_t\}$ consists of K independent white noise series Bartlett's approximation simplifies to

$$\mathrm{Cov}\left(\widehat{\rho}_{ij}(s),\widehat{\rho}_{ij}(s+1)\right) \approx 1/(T-s),$$

which is of practical importance because at the identification stage of the model building process one is often interested in comparing values of $\widehat{\rho}_{ij}(s)$ to benchmarks appropriate to the null hypothesis of the ith and jth series being independent white noise. Tiao and Box (1981) recommend using "**+**" to indicate that $\widehat{\rho}_{ij}(s) > 2/\sqrt{T}$, "**−**" to indicate that $\widehat{\rho}_{ij}(s) < -2/\sqrt{T}$, and "**·**" to indicate that $-2/\sqrt{T} \le \widehat{\rho}_{ij}(s) \le 2/\sqrt{T}$.

Sample Partial Lag Autocorrelation Matrix

The sample partial lag autocorrelation matrix, $\widehat{\boldsymbol{P}}(s)$, can be obtained by using $\widehat{\Gamma}(r)$ of $\Gamma(r)$ for $r = 0, \cdots, s-1$ in the recursive algorithm.

Under the null hypothesis that $\{\boldsymbol{y}_t\}$ is a vector AR$(s-1)$ process, the two series of residuals $\{\boldsymbol{u}_{s-1,t+s}\}$ and $\{\boldsymbol{v}_{s-1,t}\}$ are uncorrelated, and each consists of K independent white noise series. Using Quenouille (1957, p.41) and Hannan(1970, p.400), the elements of $\widehat{\boldsymbol{P}}(s)$, denoted by $\widehat{P}_{ij}(s)$, are asymptotically $N(0, 1/T)$ distributed. Use Tiao and Box's notations "**+**" to indicate that $\widehat{P}_{ij}(s) > 2/\sqrt{T}$, "**−**" to indicate that $\widehat{P}_{ij}(s) < -2/\sqrt{T}$, and "**·**" to indicate that $-2/\sqrt{T} \le \widehat{P}_{ij}(s) \le 2/\sqrt{T}$.

In addition, $T\left(\widehat{P}_{ij}(s)\right)^2 \sim \chi^2(1)$ asymptotically, which implies that asymptotically

$$X(s) = T\sum_{i=1}^{K}\sum_{i=1}^{K}\left(\widehat{P}_{ij}(s)\right)^2 \sim \chi^2(K^2). \tag{C.27}$$

$X(s)$ provides a diagnostic aid for determining the order of a vector autoregressive model.

Appendix D

Analytical Score and Hessian for BEKK VARCH(q) Model

In this appendix, we derive the analytical negative score gradient and analytical Hessian matrix for the negative log quasi-likelihood function of the BEKK VARCH(q) model.

D.1 The Negative Log Quasi-likelihood of BEKK VARCH(q) Models

Suppose we have on a sample of size T d-variate time series $\boldsymbol{y}_t$, $t = 1, \cdots, T$. The negative log quasi-likelihood function $L_T(\boldsymbol{\theta})$ is defined as

$$\begin{aligned} L_T(\boldsymbol{\theta}) &= \sum_{t=1}^{T}(-\ell_t(\boldsymbol{\theta})) \\ &= \frac{1}{2}dT\log(2\pi) + \frac{1}{2}\sum_{t=1}^{T}\log|\boldsymbol{H}_t| + \frac{1}{2}\sum_{t=1}^{T}\boldsymbol{y}_t{}'\boldsymbol{H}_t^{-1}\boldsymbol{y}_t, \end{aligned} \tag{D.1}$$

where parameter vector $\boldsymbol{\theta} = \left(vech(\boldsymbol{C})', vec(\boldsymbol{A}_1)', \cdots, vec(\boldsymbol{A}_q)'\right)' = (\boldsymbol{c}', \boldsymbol{a}_1', \cdots, \boldsymbol{a}_q')' = (\boldsymbol{c}', \boldsymbol{a}')'$, and $\boldsymbol{c} = vech(\boldsymbol{C})$, $\boldsymbol{a}_j = vec(\boldsymbol{A}_j)$, and $\boldsymbol{a} = (\boldsymbol{a}_1', \cdots, \boldsymbol{a}_q')'$.

D.2 The Negative Score Gradient

Lucchetti (2001) derived the analytical score for BEKK(1,1,1) model. In this section, we derive analytical negative score gradient $\boldsymbol{S}_T(\boldsymbol{\theta})$ for BEKK(0, q, 1) model.

$$\boldsymbol{S}_T(\boldsymbol{\theta}) = \sum_{t=1}^{T}\boldsymbol{s}_t(\boldsymbol{\theta}) = \sum_{t=1}^{T}\left(-\frac{\partial\ell_t(\boldsymbol{\theta})}{\partial\boldsymbol{\theta}'}\right) \tag{D.2}$$

$$\begin{aligned} \boldsymbol{s}_t(\boldsymbol{\theta}) &= \frac{1}{2}\frac{\partial log|\boldsymbol{H}_t|}{\partial \boldsymbol{\theta}'} + \frac{1}{2}\frac{\partial(\boldsymbol{y}_t{}'\boldsymbol{H}_t^{-1}\boldsymbol{y}_t)}{\partial \boldsymbol{\theta}'} \\ &= \frac{1}{2}\frac{\partial log|\boldsymbol{H}_t|}{\partial \boldsymbol{h}_t'}\frac{\partial \boldsymbol{h}_t}{\partial \boldsymbol{\theta}'} + \frac{1}{2}\frac{\partial(\boldsymbol{y}_t{}'\boldsymbol{H}_t^{-1}\boldsymbol{y}_t)}{\partial \boldsymbol{h}_t'}\frac{\partial \boldsymbol{h}_t}{\partial \boldsymbol{\theta}'} \\ &= \frac{1}{2}\left(\frac{\partial log|\boldsymbol{H}_t|}{\partial \boldsymbol{h}_t'} + \frac{\partial(\boldsymbol{y}_t{}'\boldsymbol{H}_t^{-1}\boldsymbol{y}_t)}{\partial \boldsymbol{h}_t'}\right)\frac{\partial \boldsymbol{h}_t}{\partial \boldsymbol{\theta}'} \end{aligned} \tag{D.3}$$

D.2.1 Derivation of $\partial log|\boldsymbol{H}_t|/\partial \boldsymbol{h}_t'$ and $\partial(\boldsymbol{y}_t{}'\boldsymbol{H}_t^{-1}\boldsymbol{y}_t)/\partial \boldsymbol{h}_t'$

Firstly,

$$\frac{\partial log|\boldsymbol{H}_t|}{\partial \boldsymbol{h}_t'} = vec(\boldsymbol{H}_t^{-1\prime})' = vec(\boldsymbol{H}_t^{-1})'. \tag{D.4}$$

Secondly, note that

$$\begin{aligned} \boldsymbol{y}_t{}'\boldsymbol{H}_t^{-1}\boldsymbol{y}_t &= tr(\boldsymbol{y}_t{}'\boldsymbol{H}_t^{-1}\boldsymbol{y}_t) \\ &= tr(\boldsymbol{y}_t\boldsymbol{y}_t{}'\boldsymbol{H}_t^{-1}) \\ &= vec(\boldsymbol{y}_t\boldsymbol{y}_t{}')'vec(\boldsymbol{H}_t^{-1}) \\ &= (\boldsymbol{y}_t\otimes\boldsymbol{y}_t)'vec(\boldsymbol{H}_t^{-1}), \end{aligned}$$

and

$$\frac{\partial vec(\boldsymbol{H}_t^{-1})}{\partial \boldsymbol{h}_t'} = -\boldsymbol{H}_t^{-1\prime}\otimes\boldsymbol{H}_t^{-1} = -\boldsymbol{H}_t^{-1}\otimes\boldsymbol{H}_t^{-1} \tag{D.5}$$

so that

$$\begin{aligned} \frac{\partial(\boldsymbol{y}_t{}'\boldsymbol{H}_t^{-1}\boldsymbol{y}_t)}{\partial \boldsymbol{h}_t'} &= -(\boldsymbol{y}_t\otimes\boldsymbol{y}_t)'\left(\boldsymbol{H}_t^{-1}\otimes\boldsymbol{H}_t^{-1}\right) \\ &= -\left(\boldsymbol{y}_t{}'\boldsymbol{H}_t^{-1}\right)\otimes\left(\boldsymbol{y}_t{}'\boldsymbol{H}_t^{-1}\right). \end{aligned} \tag{D.6}$$

D.2.2 Derivation of $\partial \boldsymbol{h}_t/\partial \boldsymbol{\theta}'$

$$\frac{\partial \boldsymbol{h}_t}{\partial \boldsymbol{\theta}'} = \frac{\partial}{\partial \boldsymbol{\theta}'}\left(vec(\boldsymbol{C}\boldsymbol{C}') + \sum_{j=1}^{q} vec(\boldsymbol{A}_j\boldsymbol{y}_{t-j}\boldsymbol{y}_{t-j}'\boldsymbol{A}_j')\right). \tag{D.7}$$

Note that

$$\begin{aligned} \frac{\partial vec(\boldsymbol{C}\boldsymbol{C}')}{\partial \boldsymbol{c}'} &= \frac{\partial(\boldsymbol{D}_d vech(\boldsymbol{C}\boldsymbol{C}'))}{\partial vech(\boldsymbol{C})'} = \boldsymbol{D}_d\frac{\partial vech(\boldsymbol{C}\boldsymbol{C}')}{\partial vech(\boldsymbol{C})'} \\ &= 2\boldsymbol{D}_d\boldsymbol{D}_d^{+}(\boldsymbol{C}\otimes\boldsymbol{I}_d)\boldsymbol{L}_d', \end{aligned}$$

where $\boldsymbol{D}_d$ is the $d^2 \times d(d+1)/2$ duplication matrix, $\boldsymbol{D}_d^+$ is the Moore-Penrose inverse of the duplication matrix $\boldsymbol{D}_d$, and $\boldsymbol{L}_d$ is the $d(d+1)/2 \times d^2$ elimination matrix. Recall that $\boldsymbol{K}_{dd} = 2\boldsymbol{D}_d\boldsymbol{D}_d^+ - \boldsymbol{I}_{d^2}$, where $\boldsymbol{K}_{dd}$ is the commutation matrix. Hence

$$\frac{\partial vec(\boldsymbol{C}\boldsymbol{C}')}{\partial \boldsymbol{c}'} = (\boldsymbol{I}_{d^2} + \boldsymbol{K}_{dd})(\boldsymbol{C} \otimes \boldsymbol{I}_d)\boldsymbol{L}_d'. \tag{D.8}$$

Note also that

$$\begin{aligned}
\frac{\partial vec(\boldsymbol{A}_j \boldsymbol{y}_{t-j}\boldsymbol{y}_{t-j}'\boldsymbol{A}_j')}{\partial \boldsymbol{a}_j'} &= (\boldsymbol{A}_j \boldsymbol{y}_{t-j}\boldsymbol{y}_{t-j}' \otimes \boldsymbol{I}_d) + (\boldsymbol{I}_d \otimes \boldsymbol{A}_j \boldsymbol{y}_{t-j}\boldsymbol{y}_{t-j}')\boldsymbol{K}_{dd} \\
&= (\boldsymbol{A}_j \boldsymbol{y}_{t-j}\boldsymbol{y}_{t-j}' \otimes \boldsymbol{I}_d) + \boldsymbol{K}_{dd}(\boldsymbol{A}_j \boldsymbol{y}_{t-j}\boldsymbol{y}_{t-j}' \otimes \boldsymbol{I}_d) \\
&= (\boldsymbol{I}_{d^2} + \boldsymbol{K}_{dd})(\boldsymbol{A}_j \boldsymbol{y}_{t-j}\boldsymbol{y}_{t-j}' \otimes \boldsymbol{I}_d)
\end{aligned} \tag{D.9}$$

Substituting (D.8) and (D.9) into (D.7) yields

$$\begin{aligned}
\frac{\partial \boldsymbol{h}_t}{\partial \boldsymbol{\theta}'} &= \left[(\boldsymbol{I}_{d^2} + \boldsymbol{K}_{dd})(\boldsymbol{C} \otimes \boldsymbol{I}_d)\boldsymbol{L}_d' \vdots (\boldsymbol{I}_{d^2} + \boldsymbol{K}_{dd})(\boldsymbol{A}_1 \boldsymbol{y}_{t-1}\boldsymbol{y}_{t-1}' \otimes \boldsymbol{I}_d) \vdots \cdots \vdots (\boldsymbol{I}_{d^2} + \boldsymbol{K}_{dd})(\boldsymbol{A}_q \boldsymbol{y}_{t-q}\boldsymbol{y}_{t-q}' \otimes \boldsymbol{I}_d)\right] \\
&= (\boldsymbol{I}_{d^2} + \boldsymbol{K}_{dd})\left[(\boldsymbol{C} \otimes \boldsymbol{I}_d)\boldsymbol{L}_d' \vdots (\boldsymbol{A}_1 \boldsymbol{y}_{t-1}\boldsymbol{y}_{t-1}' \otimes \boldsymbol{I}_d) \vdots \cdots \vdots (\boldsymbol{A}_q \boldsymbol{y}_{t-q}\boldsymbol{y}_{t-q}' \otimes \boldsymbol{I}_d)\right] \\
&= (\boldsymbol{I}_{d^2} + \boldsymbol{K}_{dd})\left[(\boldsymbol{C} \otimes \boldsymbol{I}_d)\boldsymbol{L}_d' \vdots \left[\boldsymbol{A}_1 \boldsymbol{y}_{t-1}\boldsymbol{y}_{t-1}' \vdots \cdots \vdots \boldsymbol{A}_q \boldsymbol{y}_{t-q}\boldsymbol{y}_{t-q}'\right] \otimes \boldsymbol{I}_d\right] \\
&= (\boldsymbol{I}_{d^2} + \boldsymbol{K}_{dd})\left[(\boldsymbol{C} \otimes \boldsymbol{I}_d)\boldsymbol{L}_d' \vdots (\boldsymbol{A}\boldsymbol{Y}_{t-1}) \otimes \boldsymbol{I}_d\right].
\end{aligned} \tag{D.10}$$

D.2.3 Derivation of $\boldsymbol{s}_t(\boldsymbol{\theta})$

Substituting (D.4), (D.6) and (D.10) into (D.3) yields the following negative score gradient for one observation:

$$\boldsymbol{s}_t(\boldsymbol{\theta}) = \left[vec(\boldsymbol{H}_t^{-1})' - \left(\boldsymbol{y}_t'\boldsymbol{H}_t^{-1}\right) \otimes \left(\boldsymbol{y}_t'\boldsymbol{H}_t^{-1}\right)\right] \frac{(\boldsymbol{I}_{d^2} + \boldsymbol{K}_{dd})}{2} \left[(\boldsymbol{C} \otimes \boldsymbol{I}_d)\boldsymbol{L}_d' \vdots (\boldsymbol{A}\boldsymbol{Y}_{t-1}) \otimes \boldsymbol{I}_d\right], \tag{D.11}$$

or

$$\boldsymbol{s}_t(\boldsymbol{\theta}) = \left[vec(\boldsymbol{H}_t^{-1})' - \left(\boldsymbol{y}_t'\boldsymbol{H}_t^{-1}\right) \otimes \left(\boldsymbol{y}_t'\boldsymbol{H}_t^{-1}\right)\right] \boldsymbol{D}_d\boldsymbol{D}_d^+ \left[(\boldsymbol{C} \otimes \boldsymbol{I}_d)\boldsymbol{L}_d' \vdots (\boldsymbol{A}\boldsymbol{Y}_{t-1}) \otimes \boldsymbol{I}_d\right]. \tag{D.12}$$

Denoting

$$\begin{aligned}
\boldsymbol{Q}_t(\boldsymbol{\theta}) &= vec(\boldsymbol{H}_t^{-1})' - \left(\boldsymbol{y}_t'\boldsymbol{H}_t^{-1}\right) \otimes \left(\boldsymbol{y}_t'\boldsymbol{H}_t^{-1}\right), && \text{(D.13)} \\
\boldsymbol{N}_d &= \boldsymbol{D}_d\boldsymbol{D}_d^+ = \frac{(\boldsymbol{I}_{d^2} + \boldsymbol{K}_{dd})}{2}, && \text{(D.14)} \\
\boldsymbol{R}_{t-1}(\boldsymbol{\theta}) &= \left[(\boldsymbol{C} \otimes \boldsymbol{I}_d)\boldsymbol{L}_d' \vdots (\boldsymbol{A}\boldsymbol{Y}_{t-1}) \otimes \boldsymbol{I}_d\right], && \text{(D.15)}
\end{aligned}$$

we also express (D.12) as

$$\boldsymbol{s}_t(\boldsymbol{\theta}) = \boldsymbol{Q}_t(\boldsymbol{\theta})\boldsymbol{N}_d\boldsymbol{R}_{t-1}(\boldsymbol{\theta}) \tag{D.16}$$

for $t = 1, \cdots, T$, where $\boldsymbol{Q}_t$ is a $1 \times d^2$ matrix, $\boldsymbol{N}_d$ is a $d^2 \times d^2$ matrix, and $\boldsymbol{R}_{t-1}$ is a $d^2 \times q'$ matrix where $q' = (d(d+1)/2 + qd^2)$ represents the total number of parameters in the BEKK multivariate ARCH(q) model.

D.3 The Analytical Hessian Matrix

Hafner and Herwartz (2008) studied analytical quasi maximum likelihood inference in some multivariate volatility models such VEC(1, 1), BEKK(1, 1, 1) and CCC models. In this section, we derive analytical Hessian Matrix $\boldsymbol{J}_T(\boldsymbol{\theta})$ for BEKK(0, q, 1) model.

$$\boldsymbol{J}_T(\boldsymbol{\theta}) = \frac{\partial \boldsymbol{S}_t(\boldsymbol{\theta})}{\partial \boldsymbol{\theta}} = \sum_{t=1}^{T} \frac{\partial \boldsymbol{s}_t(\boldsymbol{\theta})}{\partial \boldsymbol{\theta}} \tag{D.17}$$

$$\begin{aligned}
\frac{\partial \boldsymbol{s}_t(\boldsymbol{\theta})}{\partial \boldsymbol{\theta}} &= \frac{\partial(\boldsymbol{Q}_t\boldsymbol{N}_d\boldsymbol{R}_{t-1})}{\partial \boldsymbol{\theta}} \\
&= \frac{\partial vec(\boldsymbol{Q}_t\boldsymbol{N}_d\boldsymbol{R}_{t-1})'}{\partial \boldsymbol{\theta}} \\
&= \frac{\partial vec(\boldsymbol{R}_{t-1})'}{\partial \boldsymbol{\theta}}(\boldsymbol{I}_{q'} \otimes \boldsymbol{N}_d\boldsymbol{Q}_t') + \frac{\partial vec(\boldsymbol{Q}_t)'}{\partial \boldsymbol{\theta}}\boldsymbol{N}_d\boldsymbol{R}_{t-1} \\
&= \frac{\partial vec(\boldsymbol{R}_{t-1})'}{\partial \boldsymbol{\theta}}(\boldsymbol{I}_{q'} \otimes \boldsymbol{N}_d\boldsymbol{Q}_t') + \frac{\partial \boldsymbol{Q}_t}{\partial \boldsymbol{\theta}}\boldsymbol{N}_d\boldsymbol{R}_{t-1}
\end{aligned} \tag{D.18}$$

where the subscript of the identity matrix $q' = d(d+1)/2 + qd^2$, which represents the total number of parameters in the BEKK(0, q, 1)model.

D.3.1 Derivation of $\partial \boldsymbol{Q}_t'/\partial \boldsymbol{\theta}'$

$$\begin{aligned}
\frac{\partial \boldsymbol{Q}_t'}{\partial \boldsymbol{\theta}'} &= \frac{\partial}{\partial \boldsymbol{h}_t'}\left(vec(\boldsymbol{H}_t^{-1}) - (\boldsymbol{H}_t^{-1}\boldsymbol{y}_t) \otimes (\boldsymbol{H}_t^{-1}\boldsymbol{y}_t)\right)\frac{\partial \boldsymbol{h}_t}{\partial \boldsymbol{\theta}'} \\
&= \left(\frac{\partial vec(\boldsymbol{H}_t^{-1})}{\partial \boldsymbol{h}_t'} - \frac{\partial\left((\boldsymbol{H}_t^{-1}\boldsymbol{y}_t) \otimes (\boldsymbol{H}_t^{-1}\boldsymbol{y}_t)\right)}{\partial \boldsymbol{h}_t'}\right)\frac{\partial \boldsymbol{h}_t}{\partial \boldsymbol{\theta}'}
\end{aligned} \tag{D.19}$$

Note that $\partial vec(\boldsymbol{H}_t^{-1})/\partial \boldsymbol{h}_t'$ and $\partial \boldsymbol{h}_t/\partial \boldsymbol{\theta}'$ were derived in the sections D.2.1 and D.2.2. In this section we derive $\partial(\boldsymbol{H}_t^{-1}\boldsymbol{y}_t) \otimes (\boldsymbol{H}_t^{-1}\boldsymbol{y}_t)/\partial \boldsymbol{h}_t'$.

Because

$$\frac{\partial\left((\boldsymbol{H}_t^{-1}\boldsymbol{y}_t)\otimes(\boldsymbol{H}_t^{-1}\boldsymbol{y}_t)\right)}{\partial\boldsymbol{h}_t'} = \frac{\partial(\boldsymbol{H}_t^{-1}\boldsymbol{y}_t)}{\partial\boldsymbol{h}_t'}\otimes(\boldsymbol{H}_t^{-1}\boldsymbol{y}_t)+(\boldsymbol{H}_t^{-1}\boldsymbol{y}_t)\otimes\frac{\partial(\boldsymbol{H}_t^{-1}\boldsymbol{y}_t)}{\partial\boldsymbol{h}_t'},$$

and

$$\frac{\partial(\boldsymbol{H}_t^{-1}\boldsymbol{y}_t)}{\partial\boldsymbol{h}_t'} = \frac{\partial(\boldsymbol{I}_d\boldsymbol{H}_t^{-1}\boldsymbol{y}_t)}{\partial\boldsymbol{h}_t'} = -\boldsymbol{y}_t'\boldsymbol{H}_t'^{-1}\otimes\boldsymbol{I}_d\boldsymbol{H}_t^{-1} = -\boldsymbol{y}_t'\boldsymbol{H}_t^{-1}\otimes\boldsymbol{H}_t^{-1},$$

we have

$$\frac{\partial\left((\boldsymbol{H}_t^{-1}\boldsymbol{y}_t)\otimes(\boldsymbol{H}_t^{-1}\boldsymbol{y}_t)\right)}{\partial\boldsymbol{h}_t'} = -\boldsymbol{y}_t'\boldsymbol{H}_t^{-1}\otimes\boldsymbol{H}_t^{-1}\otimes\boldsymbol{H}_t^{-1}\boldsymbol{y}_t-\boldsymbol{H}_t^{-1}\boldsymbol{y}_t\otimes\boldsymbol{y}_t'\boldsymbol{H}_t^{-1}\otimes\boldsymbol{H}_t^{-1}. \tag{D.20}$$

D.3.2 Derivation of $\partial vec\boldsymbol{R}_{t-1}/\partial\boldsymbol{\theta}'$

$$\frac{\partial vec\boldsymbol{R}_{t-1}}{\partial\boldsymbol{\theta}'} = \frac{\partial}{\partial\boldsymbol{\theta}'}vec\left(\left[(\boldsymbol{C}\otimes\boldsymbol{I}_d)\boldsymbol{L}_d'\vdots(\boldsymbol{A}\boldsymbol{Y}_{t-1})\otimes\boldsymbol{I}_d\right]\right) \tag{D.21}$$

Noticing that

$$vec\left(\left[(\boldsymbol{C}\otimes\boldsymbol{I}_d)\boldsymbol{L}_d'\vdots(\boldsymbol{A}\boldsymbol{Y}_{t-1})\otimes\boldsymbol{I}_d\right]\right) = \begin{bmatrix} vec((\boldsymbol{C}\otimes\boldsymbol{I}_d)\boldsymbol{L}_d') \\ \hdashline vec([\boldsymbol{A}\boldsymbol{Y}_{t-1}\otimes\boldsymbol{I}_d]) \end{bmatrix},$$

we express the $(q'd^2\times q')$ matrix $\partial vec\boldsymbol{R}_{t-1}/\partial\boldsymbol{\theta}'$ in the following block format,

$$= \left[\begin{array}{c:c} \frac{\partial vec((\boldsymbol{C}\otimes\boldsymbol{I}_d)\boldsymbol{L}_d')}{\partial\boldsymbol{c}'} & \boldsymbol{0} \\ \hdashline \boldsymbol{0} & \frac{\partial vec([\boldsymbol{A}\boldsymbol{Y}_{t-1}\otimes\boldsymbol{I}_d])}{\partial\boldsymbol{a}'} \end{array}\right],$$

or more compactly, in direct sum format,

$$\frac{\partial vec\boldsymbol{R}_{t-1}}{\partial\boldsymbol{\theta}'} = \frac{\partial vec((\boldsymbol{C}\otimes\boldsymbol{I}_d)\boldsymbol{L}_d')}{\partial\boldsymbol{c}'}\oplus\frac{\partial vec([\boldsymbol{A}\boldsymbol{Y}_{t-1}\otimes\boldsymbol{I}_d])}{\partial\boldsymbol{a}'} \tag{D.22}$$

where $\partial vec((\boldsymbol{C}\otimes\boldsymbol{I}_d)\boldsymbol{L}_d')/\partial\boldsymbol{c}'$ is a $d^3(d+1)/2\times d(d+1)/2$ matrix that is formulated as

$$\begin{aligned}
\frac{\partial vec((\boldsymbol{C}\otimes\boldsymbol{I}_d)\boldsymbol{L}_d')}{\partial\boldsymbol{c}'} &= (\boldsymbol{L}_d\otimes\boldsymbol{I}_{d^2})\frac{\partial vec(\boldsymbol{C}\otimes\boldsymbol{I}_d)}{\partial\boldsymbol{c}'} \\
&= (\boldsymbol{L}_d\otimes\boldsymbol{I}_{d^2})\frac{\partial vec(\boldsymbol{C}\otimes\boldsymbol{I}_d)}{\partial vec(\boldsymbol{C})'}\frac{\partial vec(\boldsymbol{C})}{\partial\boldsymbol{c}'} \\
&= (\boldsymbol{L}_d\otimes\boldsymbol{I}_{d^2})\frac{\partial vec(\boldsymbol{C}\otimes\boldsymbol{I}_d)}{\partial vec(\boldsymbol{C})'}\boldsymbol{L}_d' \\
&= (\boldsymbol{L}_d\otimes\boldsymbol{I}_{d^2})(\boldsymbol{I}_d\otimes\boldsymbol{K}_{dd}\otimes\boldsymbol{I}_d)(\boldsymbol{I}_{d^2}\otimes vec(\boldsymbol{I}_d))\boldsymbol{L}_d' \ ,
\end{aligned} \tag{D.23}$$

and $\partial vec([\boldsymbol{A}\boldsymbol{Y}_{t-1} \otimes \boldsymbol{I}_d])/\partial \boldsymbol{a}'$ is a $qd^4 \times qd^2$ matrix that can be expressed as

$$\begin{aligned} \frac{\partial vec([\boldsymbol{A}\boldsymbol{Y}_{t-1} \otimes \boldsymbol{I}_d])}{\partial \boldsymbol{a}'} &= \frac{\partial vec([\boldsymbol{I}_d\boldsymbol{A}\boldsymbol{Y}_{t-1} \otimes \boldsymbol{I}_d])}{\partial \boldsymbol{a}'} \\ &= (\boldsymbol{I}_{qd} \otimes \boldsymbol{K}_{dd} \otimes \boldsymbol{I}_d)(\boldsymbol{Y}'_{t-1} \otimes \boldsymbol{I}_d \otimes vec(\boldsymbol{I}_d)). \end{aligned} \tag{D.24}$$

Therefore,

$$\begin{aligned} \frac{\partial vec(\boldsymbol{R}_{t-1})'}{\partial \boldsymbol{\theta}} = &\Big[\boldsymbol{L}_d(\boldsymbol{I}_{d^2} \otimes vec(\boldsymbol{I}_d)')(\boldsymbol{I}_d \otimes \boldsymbol{K}_{dd} \otimes \boldsymbol{I}_d)(\boldsymbol{L}'_d \otimes \boldsymbol{I}_{d^2})\Big] \\ &\oplus \Big[(\boldsymbol{Y}_{t-1} \otimes \boldsymbol{I}_d \otimes vec(\boldsymbol{I}_d)')(\boldsymbol{I}_{qd} \otimes \boldsymbol{K}_{dd} \otimes \boldsymbol{I}_d)\Big] \end{aligned} \tag{D.25}$$

The nicety of the matrix $\partial vec(\boldsymbol{R}_{t-1})'/\partial \boldsymbol{\theta}$ is that it is not predicated on the parameters of the model, and only the order of the ARCH model (q) and the dimension of the time series vector (d) matter.

Bibliography

[1] Akaike, H. (1969). *Fitting autoregressive models for prediction*. Annals of the Institute of Statistical Mathematics 21: 243-247.

[2] Akaike, H. (1971). *Autoregressive model fitting for control*. Annals of the Institute of Statistical Mathematics 23: 163-180.

[3] Akaike, H. (1973). *Information theory and an extension of the maximum likelihood principle*. Second International Symposium on Information Theory, pp. 267-281.

[4] Akaike, H. (1974). *A new look at the statistical model identification*. IEEE Transactions on Automatic Control AC-19: 716-723.

[5] Akaike, H. (1978). *A Bayesian analysis of the minimum AIC procedure*. Annals of the Institute of Statistical Mathematics 30, Part A, 9-14.

[6] Allen, D.M. (1974). *The relationship between variable selection and data augmentation and a method for prediction*. Technometrics 16, 125-127.

[7] Andrews, D. (1999). *Estimation when a parameter is on a boundary*. Econometrica, 67, 1341-1384.

[8] Audrino, F. and Camponovo L. (2013) *Oracle Properties and finite sample inference of the adaptive LASSO for time series regression models*. arXiv:1312.1473 [stat.ME].

[9] Barbosa, S.M. (2014). *Multivariate AutoRegressive analysis*. R package version 1.1-2, URL: http://cran.at.r-project.org/web/packages/mAr

[10] Bartlett, M.S. (1938). *Further aspects of the theory of multiple regression*. Proceedings of the Cambridge Philosophicak Society, 34:33-40

[11] Bartlett, M.S. (1966). *An introduction to stochastic processes*, 2nd edition. London, U.K.: Cambridge University Press.

[12] Bierens, H.J. (2004). *Introduction to the Mathematical and Statistical Foundations of Econometrics*. London, U.K.: Cambridge University Press.

[13] Bollerslev, T. (1986). *Generalized autoregressive conditional heteroskedasticity*. J. Econometrics 31: 307-327.

[14] Bougerol, P. and Picard, N. (1992). *Strict stationarity of generalized autoregressive processes*. Annals of Probability 20: 1714-1729.

[15] Box, G. E. P., Jenkins, G. M. and Reinsel, G. C. (1994). *Time series analysis: forecasting and control*(3rd edn). San Francisco: Holden-Day.

[16] Brannstrom, T. 1995). *Bias approximation and reduction in vector autoregressive models*. PhD thesis, Stockholm School of Economics.

[17] Breiman, L. (1996). *Heuristics of instability and stabilization in model selection*. Annals of Statistics 24(6), 2350-2383.

[18] Bühlmann, P. and van de Geer, S. (2011). *Statistics for High-dimensional Data*. Berlin-Heidelberg: Springer.

[19] Caner, M. and Knight K. (2013) *An alternative to unit root tests: Bridge estimators differentiate between nonstationary versus stationary models and select optimal lag*. J. Statistical Planning and Inference 143:691-715.

[20] Chand, S. (2011). *Goodness of Fit and Lasso Variable Selection in Time Series Analysis*. PhD thesis, University of Nottingham.

[21] Chernoff, H. (1954). *On the distribution of the likelihood ratio*. Annals of Mathematical Statistics, 54:573-578.

[22] Choi, B. (1992). *ARMA Model Identification*. New York: Springer-Verlag.

[23] Chong, E. K. P. and Zak, S. H. (2008). *An introduction to Optimization*. John Wiley & Sons Inc., Hoboken, New Jersey.

[24] Cleveland, W. S. (1971). *The inverse autocorrelations of a time series and their applications*. Technometrics 14, 277-98.

[25] Craven P. and Wahba G. (1979). *Smoothing noisy data with spline functions*. Numer. Math. 31:377-403.

[26] de Gooijer, J. G., Abraham, B., Gould, A., Robinson , L. (1985). *Methods for determining the order of an autoregressive-Moving average process: a survey*. International Statistical Review, Vol. 53, No. 3, pp. 301-329.

[27] De Jong, P. (1976). *The recursive fitting of autoregressions.* Biometrika, Vol. 63, No. 3, pp. 525-530.

[28] Donoho, D.L., and Johnstone, I. (1994). *Ideal spatial adaptation by wavelet shrinkage*. Biometrika 81:425-455.

[29] Donoho, D.L., Michael Elad, M., and Temlyakov, V.N. (2006) *Stable recovery of sparse overcomplete representations in the presence of noise*. IEEE Transactions on Iinformation Ttheory, Vol. 52, No. 1.

[30] Durbin, J.(1960). *The Fitting of Time Series Models*. Review of the Institute of International Statistics, 28: 233-244.

[31] Engle, R.F. (1982). *Autoregressive conditional heteroskedasticity with estimates of the variance of U.K. inflation*. Econometrica 50, 987-1008.

[32] Engle, R. and Kroner, F. (1995). *Multivariate simultaneous generalized ARCH*. Economic Theory, 11:122-150.

[33] Efron, B., Hastie, T., Johnstone, I., and Tibshirani, R. (2004). *Least Angle regression*. Annals of Statistics, 32(2) 407-499.

[34] Fan, J. and Li, R. (2001). *Variable selection via nonconcave penalized likelihood and its oracle properties*. J. American Statistical Association, 96: 1348-1360.

[35] Francq, C. and J. M. Zakoïan (2007) *Quasi-maximum likelihood estimation in GARCH processes when some coefficients are equal to zero*. Stochastic Processes and their Applications 117, 1265-1284.

[36] Francq, C. and J. M. Zakoïan (2009) *Testing the nullity of GARCH coefficients: correction of the standard tests and relative efficiency comparisons*. J. American Statistical Association 104, 313-324.

[37] Francq, C. and J.-M. Zakoïan (2010). *GARCH models: structure, statistical inference and financial applications*. West Sussex, U.K.: John Wiley & Sons Ltd.

[38] Frank, I. and Friedman, J. (1993). *A statistical view of some chemometrics regression tools (with discussion)*. Technometrics 35(2):109-148.

[39] Friedman, J., Hastie, T., Hoefling, H. and Tibshirani, R. (2007). Pathwise coordinate optimization. Annals of Applied Statistics 2(1): 302-332.

[40] Fu, W. (1998). *Penalized regressions: the bridge vs. the lasso*. J. Computational and Graphical Statistics 7(3): 397-416.

[41] Fujita, A., Sato, J.R., Garay-Malpartida, H. M., Yamaguchi, R., Miyano, S., Sogayar, M.C., and Ferreira, C. E. (2007). *Modeling gene expression regulatory networks with the sparse vector autoregressive model*. BMC Systems Biology, 1:39.

[42] Geyer, C.(1994). *On the asymptotics of constrained M-Estimation*. Annals of Statistics, 22, 1993-2010

[43] Hafner, C. M., and Herwartz, H. (2008). Analytical quasi maximum likelihood inference in multivariate volatility models Metrika, 67:219-239.

[44] Hamilton, J.D. (1994. *Time Series Analysis*. Princeton, NJ: Princeton University Press.

[45] Hannan, E.J.(1970). *Multiple Time Series*. New York: John Wiley.

[46] Hannan, E. J. and Quinn, B. G. (1979).*The determination of the order of an autoregression*. J. the Royal Statistical Society B41:190-195.

[47] Hastie, T., Tibshirani, R. and Friedman, J. (2009)*The Elements of Statistical Learning: Data Mining, Inference, and Prediction* (2nd edn). Springer-Verlag.

[48] Haufe, S., Müller, K. R., Nolte, G. and Krämer, N. (2008). *Sparse Causal Discovery in Multivariate Time Series*. Neural information processing systems. 97-106.

[49] Hayashi, F. (2000). *Econometrics*. Princeton, N. J.: Princeton University Press.

[50] Heyse, J.F.(1985). *Partial lag autocorrelation and partial process autocorrelation for vector time series, with applications*. PhD Dissertation, Temple University.

[51] Hipel, K.W., McLeod, A.I. and Lennox, W.C. (1977). *Advances in Box-Jenkins modelling Part 1, Model construction*. Water Resources Res. 13, 567-575.

[52] Hoerl, A. E. and Kennard, R. (1970). *Ridge regression: biased estimation for nonorthogonal problems*. Technometrics 12: 55-67.

[53] Hurwicz L. (1950). *Least-Squares Bias in Time Series*. In *Statistical inference in dynamic economic models* (T. Koopmans, eds). John Wiley & Sons, New York, Chapman & Hall, Limited, London.

[54] Hsu, N., Hung, H., and Chang, Y. (2008). *Subset selection for vector autoregressive processes using LASSO*. Computational Statistics and Data Analysis 52: 3645-3657.

[55] Knight, K., and Fu, W. (2000). *Asymptotics for LASSO-type estimators*, Annals of Statistics, 28: 1356-1378.

[56] Kock, A.B. (2012). *On the oracle property of the adaptive lasso in stationary and nonstationary autoregressions*. CREATES Research Papers 2012-05, Aarhus University.

[57] Kock, A.B. and Callot, L.A.F. (2012).*Oracle inequalities for high dimensional vector autoregressions*. CREATES Research Paper 2012-12, Aarhus University.

[58] Leeb, H. and Pötscher B. M. (2003). *The finite-sample distribution of post-model selection estimators and uniform versus nonuniform approximations*. Econometric Theory, 19:100-142.

[59] Leeb, H. and Pötscher B. M. (2005). *Model selection and inference: facts and fiction*, Econometric Theory, 21:21-59 .

[60] Leeb, H. and Pötscher B. M. (2008). *Sparse estimators and the oracle property, or the return of Hodges' estimator*, J. Econometrics, 142:201-211.

[61] Ling, S. and McAleer, M. (2002). *Necessary and sufficient moment conditions for the GARCH(r, s) and asymmetric power GARCH(r, s) models*. Econometric Theory 18: 722-729.

[62] Lockhart R., Taylor, J., Tibshirani R. J., and Tibshirani, R. (2014). *A significance test for the LASSO*. Annals of Statistics, Vol. 42, No. 2, 413-468.

[63] Lucchetti, R. (2001). *Analytical score for multivariate GARCH models*. Computational Economics, 19:133-143.

[64] Lütkepohl, H. (1996). *Handbook of Matrices*, New York: Wiley.

[65] Lütkepohl, H. (2006). *New Introduction to Multiple Time Series Analysis*. Berlin:Springer-Verlag.

[66] Madigan, D. and Ridgeway, G. (2004). *Discussion of "Least Angle Regression" by Efron et al*.Annals of Statistics, Vol 32, No. 2, 465-469.

[67] McLeod, A. I. and Hipel, K. W. (1995) *Exploratory spectral analysis of hydrological time series*. J. Stochastic Hydrology and Hydraulics 9, 171-205.

[68] McLeod, A.I., Hipel, K.W., and Lennox, W.C. (1977). *Advances in Box-Jenkins modelling, Part 2, Applications*. Water Resources Research, 13:576-586.

[69] McLeod, A. I. and Zhang, Y.(2005). *Partial autocorrelation parameterization for subset autoregression*. J. Time Series Analysis, 27:599-612

[70] McQuarrie, A. D. R. and Tsai, C-L (1998). *Regression and Time Series Model Selection*. World Scientific:Singapore.

[71] Medeiros, M.C and Mendes, E.F. (2012). *Estimating High-Dimensional Time Series Models*. CREATES Research Paper 2012-37.

[72] Meinshausen, N. (2007). *Lasso with relaxation*. Computational Statistics and Data Analysis, 52(1):374-293.

[73] Meinshausen, N. and Bühlmann, P. (2006). *High-dimensional graphs and variable selection with the Lasso*. Annals of Statistics, 34 1436-1462.

[74] Mallows, C. L. (1973). *Some Comments on Cp*. Technometrics 15(4): 661-675.

[75] Monti, A. C. (1994). *A proposal for a residual autocorrelation test in linear models*, Biometrika, Vol. 81, No. 4, 776-80.

[76] Nardi, Y. and Rinaldo, A. (2011). *Autoregressive Process modeling via the LASSO Procedure*, J. Multivariate Analysis, Vol 102, 3:528-549.

[77] Nelson, D.B. and Cao, C.Q. (1992). *Inequality constraints in the univariate GARCH model*. Journal of Business and Economic Statistics 10, 229-235.

[78] Nicholls, D. F. and Pope, A.L. (1988). *Bias in the Estimation of Multivariate Autoregressions*,Australian Journal of Statistics, 30A, 296-309.

[79] Osborne, M., Presnell, B. and Turlach, B. (2000a). *A new approach to variable selection in least squares problems*. IMA Journal of Numerical Analysis 20: 389-404.

[80] Osborne, M., Presnell, B. and Turlach, B. (2000b). *On the lasso and its dual*. J. Computational and Graphical Statistics 9: 319-337.

[81] Park H. and Sakaori F. (2013) *Lag weighted lasso for time series model*. Computational Statistics 28:493-504.

[82] Pötscher, B. M. and Leeb, H. (2009). *On the distribution of penalized maximum likelihood estimators: the LASSO, SCAD, and thresholding*, J. Multivariate Analysis 100:2065-2082.

[83] Pötscher, B. and Schneider (2009). *On the distribution of the adaptive LASSO estimator*. J. Statistical Planning and Inference 139:2775-2790.

[84] Pötscher, B. and Schneider (2010). *Confidence sets based on penalized maximum likelihood estimators*. Electronic Journal of Statistics, Vol 4, 334-360.

[85] Quenouille, M. H.(1949). *Approximate tests of correlation in time series*. J. Royal Statistical Society, Series B 11: 68-84.

[86] Quenouille, M. H.(1957). *The analysis of multiple time Series*. London: Griffin.

[87] Ren, Y. and Zhang, X. (2010), *Subset Selection for Vector Autoregressive Processes via the Adaptive LASSO*. Statistics and Probability Letters 80: 1705-1712.

[88] Shin, K-I. and Kang, H-J. (2001). *A study on the effect of power transformation in the ARMA(p,q) model*. J. Applied Statistics, 28, 8:1019-1028.

[89] Schwarz, G. (1978). *Estimating the dimension of a model*. Annals of Statistics 6(2): 461-464.

[90] Song, S. and Bickel, P.J. (2011). it Large Vector Auto Regressions. arXiv:1106.3915v1 [stat.ML].

[91] Tang L., Zhou, Z., and Wu C. (2012) *Efficient estimation and variable selection for infinite variance autoregressive models*. J. Applied Mathematica Computing. 40:399-413

[92] Tiao, G.C. and Box, G.E.P.(1981). *Modeling multiple time series with applications*. J. American Statistical Association, 76: 802-816.

[93] Tibshirani, R. (1996). *Regression shrinkage and selection via the lasso*. J. Royal Statistical Society, Series B 58 (1): 267-288.

[94] Tjostheim, D. and J. Paulsen (1983). *Bias of some commonly-used time series estimates*, Biometrika, 70, 389-399

[95] Tsai, H. and Chan, K.-S. (2008). *A note on inequality constraints in the GARCH model*. Econometric Theory 24, 823-828.

[96] Valdés-Sosa, P. A., Sánchez-Bornot, J. M., Lage-Castellanos, A., Vega-Hernández, M., Bosch-Bayard, J., Melie-García, L., and Canales-Rodríguez, E. (2005). it Estimating Brain Functional Connectivity with Sparse Multivariate Autoregression. Philosophical Transactions R. Soc. B, 360(1457):969-981.

[97] Wagener, J., and Dette, H. (2012). *Bridge estimators and the adaptive LASSO under heteroscedasticity*. Mathematical Methods of Statistics, Vol. 21, No. 2, pp. 109-126.

[98] Woodward, W.A. and Gray, H.L. (1978). *New ARMA models for Wolfer's sunspot data*. Commun. Statist. - Simula., Computa,. B7(1), 97-115

[99] Wang, H., Leng, C., 2007. *Unified LASSO estimation via least squares approximation*. J. American Statistical Association, 101, 1418-1429.

[100] Wang, H., Li, G., and Tsai, C.(2007). *Regression coefficients and autoregressive order shrinkage and selection via the LASSO*. J. Royal Statistical Society, Series B 69 (1):63-78.

[101] Wei, W.S. (2005). *Time Series Analysis: Univariate and Multivariate Methods* (2nd Edn). Reading, MA: Addison-Wesley.

[102] Xu G., Xiang Y., Wang S. and Lin Z. (2012) *Regularization and variable selection for infinite variance autoregressive models*. J. Statistical Planning and Inference 142:2545-2553.

[103] Yoon, Y.j., Park, and C., Lee, T. (2013)*Penalized regression models with autoregressive error terms*. J. Statistical Computation and Simulation, Vol. 83, No. 9, 1756-1772

[104] Yu, H. (2002) *Rmpi: Parallel Statistical Computing in R*. R News, Vol. 2, No. 2., pp. 10-14.

[105] Zhao, P. and Yu, B. (2006). *On model selection consistency of Lasso*. J. Machine Learning Research, 7:2541-2563.

[106] Zou, H. (2006). *The adaptive LASSO and its oracle properties*. J. American Statistical Association, 101: 1418-1429.

Curriculum Vitae

Name:	Zi Zhen Liu Born in Tongwei County, Gansu Province, China Canadian Citizen
Education and Degrees:	University of Western Ontario 2009 – 2014 Doctor of Philosophy in Statistics
	University of Toronto 2007 – 2009 Master of Science in Statistics
	University of Toronto 2004 – 2008 Honours Bachelor of Science Specialist in Actuarial Science Major in Statistics Minor in Economics