2015

The Bootstrap Estimation In Time Series

Yun Liu
Michigan Technological University, yliu26@mtu.edu

THE BOOTSTRAP ESTIMATION IN TIME SERIES

By

Yun Liu

A REPORT

Submitted in partial fulfillment of the requirements for the degree of

MASTER OF SCIENCE

In Mathematical Sciences

MICHIGAN TECHNOLOGICAL UNIVERSITY

2015

This report has been approved in partial fulfillment of the requirements for the Degree of MASTER OF SCIENCE in Mathematical Sciences.

Department of Mathematical Sciences

Report Advisor: *Dr. Yeonwoo Rho*

Committee Member: *Dr. Latika Lagalo*

Committee Member: *Dr. Seokwoo Choi*

Committee Member: *Dr. Min Wang*

Department Chair: *Dr. Mark S. Gockenbach*

Contents

List of Figures

List of Tables

Abstract

Time series, a special case in dependent data sequence, is widely used in many fields. In time series, linear process models are quite popularly used. General form of linear process indicates the time dependence property of time series, AR(p), MA(q) and ARMA(p, q) models are all linear process models. In this report, simulations are based on the simplest models of these linear process models, such as AR(1), MA(1) and ARMA(1,1) models. AR(1)-SEASON, which is developed based on AR(1) model by changing the weight of residuals, is also considered in this report.

To deal with dependent data sequence, common methods which aim to deal with independent data are no longer accurate to do inference. For dependent data, a conventional method involves consistent estimation of the long run variance, for example, Andrews [2]. However, in Andrews method, it might be hard to determine the bandwidth. As an alternative, bootstrap methods can be used to approximate the limiting distribution. Block based bootstrap methods, such as moving block bootstrap, non-overlapping block bootstrap and circular block bootstrap, can be used for dependent data. Stationary bootstrap, which is with flexible block length following

a geometric distribution with parameter p_S, has also been proved to be consistent. AR-Sieve bootstrap aims to construct a fitted model of AR($\hat{p}$) and resampling the data with the fitted model. In our simulations, we compare finite sample confidence interval coverage rates. We also consider these bootstrap methods with Andrews estimation of variance [2] and simulations results show that with the help of Andrews estimation, the estimations are more accurate.

A further discussion of determining an optimal block length for AR(1) model is also mentioned in our report.

Chapter 1

Introduction

The development of statistics is pretty quick along with social and technological development. At the beginning of statistical research, many of theorems are introduced based on i.i.d. data series. However, with the wide use of statistical methods in many other fields, e.g. economics, finance and other fields, i.i.d. data series can no longer satisfy researchers requirements. Then, more theorems came up to deal with dependent data series. A special kind of dependent data series is time series, in which data are dependent with respect to time. One of the popular methods of time series is linear process, some of these linear proces are listed in Chapter 2, which are AR(p), MA(q) abd ARMA(p, q). In the simulations in

Chapter 5, we consider several simplest linear process models with different methods.

Unlike the methods we use to estimate parameters of i.i.d. data series, the variance of dependent data cannot be estimated in a simple form. Andrews (1991) [2] introduced a method to estimate the variance of limiting distribution in regression with time-dependent variables. However, in finite samples, the greater time-dependence in each model, the less accurate of Andrews variance estimation is. For example, Andrews method has to determine the bandwidth during estimating, however, even though Andrews had listed some useful guide for the choice of bandwidth, they are suitable ony for a few kinds of data generating processes. Researchers might want to find other methods to do more accurate inference of dependent data infinite samples.

Bootstrap methods aimed to deal with i.i.d. random variables when they were first introduced. Efron (1979) [11] showed that bootstrap was developed from jackknife method to make up the shortage of jackknife method. In most bootstrap methods, two levels of estimations are calculated. Take the estimation of population mean μ as an example. The lower level is to use the sample mean $\overline{X}_n$ to estimate μ; the upper level is to use the bootstrap version of sample mean $\overline{X}_n^*$, which is the sample

mean of new data constructed by original data, to estimate the sample mean $\overline{X}_n$ of the original data. Such kind of bootstrap methods for time series were firstly introduced by Carlstein (1986) [9] for univariate time series based on bootstrap methods with i.i.d. data series, also known as non-overlapping block bootstrap method. In non-overlapping block bootstrap, original data are divided into several blocks, and correlation of data is maintained within blocks, and independence is assumed between blocks. The introduction of non-overlapping block bootstrap methods helped a lot for dependent series, especially time series. However, with different determination of block length in non-overlapping block bootstrap method, the last few observations may not be used into calculation, and this shortage would influence the result somehow. In 1989 [16], moving block bootstrap was introduced to make up the shortage of non-overlapping block bootstrap method. With moving block bootstrap, the starting observation in each block is the second observation in the previous block. And generating of blocks will stop once the last observation is contained into the last block. Therefore, all of the observations are considered into calculation. At the same time, some parts of original data will be calculated more than once, and several beginning and ending observations will be only counted once. In 1994, Politis and Romano came up with circular bootstrap method [20], in which the series was wrapped into a circle. All observations will be counted more

than once, and if the total number of original data is n, there will be n blocks. These three block bootstrap methods are based on a fixed block length to divide the original data into blocks. Another kind of block bootstrap, also known as stationary bootstrap, divides the original data into blocks with block length following a geometric distribution with parameter p. With non-fixed block length, results shown in Chapter 4 indicates the accuracy of stationary bootstrap, even though with large correlation, stationary bootstrap may not be as accurate as expected. With the development of bootstrap based on dependent data series, another kind of bootstrap was introduced. AR-Sieve bootstrap method is based on the reconstruction of fitted model based on original data. Instead of dividing the original data into blocks and then calculating estimations by using block bootstrap methods, AR-Sieve bootstrap constructs new a series based on the fitted model of original data, and with the new series, AR-Sieve bootstrap calculates the estimations. In practice, we usually cannot get the model and then do calculations. In this paper, simulations of bootstraps are based on the given models, and data series can be generated as the original data from the given model. We use aforementioned bootstrap methods and the Andrews method to construct confidence intervals of μ, and compare average converage rrates. If average converage rate is closer to the nominal level 95%, it means that this method delivers more accurate cofidence interval in finite samples. We also

consider the average length of confidence interval. The smaller the confidence interval, the more precise the estimation is.

As we mentioned above, original data series is divided into blocks in block bootstrap methods. We consider a simpler but not necessarily the optimal way to calculate the block length and to construct blocks. However, more optional block length is given in Chapter 5 with circular block bootstrap and stationary bootstrap methods, as introduced by Politis and White (2004) [14]. Further, we calculate the theoretical optimal block length of circular block bootstrap and theoretical optimal expected block length of stationary bootstrap for our simulation models in Chapter 4. Even though Politis and White (2004) [14] gave the estimation of block length for these two bootstrap methods, since we only consider most basic models in this paper, then the estimation of block length did not significantly improve the results. With new block length in circular block bootstrap and new parameter p in stationary bootstrap, we choose to do simulations for AR(1) and ARMA(1,1). Results and detailed comparisons are also shown in Chapter 5.

Simulations and models contained in this paper are quite basic. The goal is to

compare the bootstrap methods with different basic models of time series. Further work can be done with more compicated models, or with other bootstrap methods. Bootstrap methods based on time series are still developing. More methods can be dedicated into dealing with dependent data, and we will continue to work on this.

Chapter 2

Time Series

In the development of Statistics, many statistical methods related to independent or uncorrelated data, were introduced by scientists to deal with analyzing the natural facts from collecting data. However, in many practical situations, collected data are correlated. In some cases, data are related among themselves; and in other cases, data are collected over time. We name the data collected sequentially in time as *time series*. Time series are widely used in economics, quantitative finance, meteorology and so on. Researchers might find out the trend of some observations with respect to time. Often, these observations are not independent in most cases, therefore, it is natural to think about methods that are appropriate for time series analysis.

Denote the real valued observations in times as $\cdots, X_{-2}, X_{-1}, X_0, X_1, X_2, \cdots$, the time series is denoted as $\{X_t\}$. The random variables are indexed by all integers $\mathbb{Z}$. The most simple problem in time series is to estimate the population mean μ. Due to the time dependence, we generally need to use different methods. In the rest of this chapter, we introduce some basic concepts of time series, including stationarity, models based on stationarity and Andrews(1991) [2] methods to estimate the longrun variance of dependent data.

2.1 Stationary Processes

To deal with the time dependence, we often make assumptions on the form of the time dependence to make the analysis easier. The typical assumption in time series analysis is stationarity. For example, the mean or variance, does not change over time for stationary time series. Two definitions of stationarity are usually applied, strict stationarity and second order stationarity/weak stationarity.[25]

Definition 2.1.1 *For any integer finite sequence $t_1, t_2, \cdots, t_l$ and any integer p, if*

$(X_{t_1}, X_{t_2}, \cdots, X_{t_l})$ and $(X_{t_1+p}, X_{t_2+p}, \cdots, X_{t_l+p})$ have the same distribution, then the time series X_t is lth- order stationary and it is said to be strictly stationary.

Most of the time, the strict stationarity is too strong to be satisfied. Thus, an alternative definition comes up and it might be easier to work under this assumption. In the weak stationarity, the first moment and autocovariance do not change over time in stead of having the same distribution for $(X_{t_1}, X_{t_2}, \cdots, X_{t_l})$ and $(X_{t_1+p}, X_{t_2+p}, \cdots, X_{t_l+p})$.

Definition 2.1.2 *If, for all integer t and l, the mean of the time series $\{X_t\}$ is constant, and if the covariance of X_t and X_{t+l} only depends on the magnitude of l, the time series $\{X_t\}$ is said to be second order stationary, or weak stationary.In other words, the time series $\{X_t\}$ is second order stationary, or weak stationary if $E(X_t) = \mu$ and $cov(X_t, X_{t+l}) = \gamma_l$ where μ is a constant and γ_l is independent of t. $\{\gamma_l\}$ with integers l is called the autocovariance function at lag l. And we also define autocorrelation function of X_t at lag l as*

$$\rho_l = \frac{\gamma_l}{\gamma_0} = corr(X_t, X_{t+l}).$$

Remark 2.1.1 *[25]*

1. *Autocoraviance has properties:*

$$\gamma_0 = Var(X_t), \quad \gamma_l = \gamma_{-l}, \quad |\gamma_l| \leq \gamma_0.$$

2. *The strict stationarity does not necessarily mean weak stationarity. However, if we further assume finite second moment, i.e.,* $E|X_t^2| < \infty$*, then all strictly stationary series are also second order stationary.*

3. *If the strict stationarity is satisfied while the second moment is infinite, then the second order stationarity can not be implied.*

4. *The only case where the weak stationarity implies the strict stationary is a weakly stationary Gaussian time series.*

Stationarity is widely used in time series models to deal with practical cases.

2.2 Models Based on Stationarity

2.2.1 Linear Processes

One of the most popular methods to model the time dependence is to express the variable as a linear combination of an i.i.d. sequence. Let $\{\epsilon_t\}$ be a sequence of identically distributed independent random variables with mean zero. A linear process $\{X_t\}, t \in \mathbb{Z}$ is defined as

$$X_t = \sum_{i=-\infty}^{\infty} \psi_i \epsilon_{t-i}. \tag{2.1}$$

If $\sum_{i=1}^{\infty} \psi_i^2$ is infinite, then the variance of X_t will be infinite. Therefore, it is reasonable to assume that $\sum_{i=1}^{\infty} \psi_i^2 < \infty$. Models mentioned below are special cases of linear processes that are commonly used in practice.

2.2.1.1 Moving Average Processes

Based on the linear processes expression in 2.1, if only a finite number of ψ_i are nonzero, then we can get a *moving average process*,

$$X_t = \sum_{i=0}^{q} \theta_i \epsilon_{t-i}, \tag{2.2}$$

where θ_i are fixed constants, $\theta_0 = 1$, constant q indicates the finite number of nonzero θ_i, and white noise $\{\epsilon_t\}$ is a sequence of independent random variables with mean zero and variance σ^2. (2.2) is called the *moving average process* of order q, denoted as MA(q). MA(1) and MA(q) are stationary for every finite θ or every finite sequence $\{\theta_i\}, i = 1, 2, \cdots, q$. Since according to 2.2,

$$\begin{aligned}
E(X_t) &= 0, \\
Var(X_t) &= \sigma^2 \sum_{i=0}^{q} \theta_i^2, \\
\gamma_{t-i} &= E(X_t X_{t-i}) \\
&= E(\sum_{i=0}^{q} \theta_i \epsilon_{t-i} X_{t-i}) \\
&= \begin{cases} \sigma^2 \left(\sum_{i=0}^{q} \theta_i \theta_{i+|l|} \right), & \text{for } |l| \leq q \\ 0, & \text{otherwise.} \end{cases}.
\end{aligned}$$

Then the sum is finite when every θ is finite. And γ_{t-i} does not depend on t, i.e., MA(1) and MA(q) are stationary.

In a similar way, we can show that MA(∞) is stationary if the coefficients are absolutely summable, i.e., $\sum_{i=0}^{\infty} |\theta_i| < \infty$.

2.2.1.2 Autoregressive Processes

The autoregressive process of order p, AR(p), can be defined by

$$X_t - \epsilon_t = \sum_{i=1}^{p} \phi_i X_{t-i}, \tag{2.3}$$

where ϵ_t is a independent random variables sequence with mean zero and variance σ^2, and ϕ_i are fixed constants. AR(1) process can be defined as

$$X_t - \epsilon_t = \phi_1 X_{t-1}.$$

In fact, AR(1) can be proved as the second order stationarity. To prove it, we firstly rewrite the definition function of X_t as

$$\begin{aligned} X_t - \epsilon_t &= \phi_1(\epsilon_{t-1} + \phi(\epsilon_{t-2} + \phi_1(\epsilon_{t-3} + \cdots))) \\ &= \phi_1\epsilon_{t-1} + \phi_1^2\epsilon_{t-2} + \phi_1^3\epsilon_{t-3} + \cdots, \\ X_t &= \sum_{i=0}^{t-1} \phi_1^i \epsilon_{t-i}. \end{aligned}$$

Therefore, since ϵ_t is an independent sequence, from the "new" definition function we can figure out that AR(1) is a linear process with the mean of X_t being 0, and the autocovariance function being as follow:

$$\begin{aligned} \gamma_0 &= E(X_t^2 - E(X_t)^2) = E(X_t^2) \\ &= E\left(\sum_{i=0}^{\infty} \phi_1^i \epsilon_{t-i}\right)^2 = E\left(\sum_{i=0}^{\infty} \phi_1^{2i} \epsilon_{t-i}^2\right) \\ &= \sigma^2\left(\sum_{i=0}^{\infty} \phi_1^{2i}\right) = \sigma^2 \frac{1}{1-\phi_1^2}, \\ \gamma_l &= E(X_t X_{t+l}) = E\left(\sum_{i=0}^{\infty} \phi_1^i \epsilon_{t-i} \sum_{j=0}^{\infty} \phi_1^j \epsilon_{t-j+l}\right) \\ &= \sigma^2 \frac{\phi_1^l}{1-\phi_1^2}. \end{aligned}$$

Note that not all AR(p) process is stationary. For example, if $\phi_1 = 1$ for AR(1) model, $Var(X_i)$ diverge to infinity. To find the conditions on the AR coefficients so

that in AR(p) process is stationary, we do the followings.

Since in AR(p) model, X_t and X_{t-1} are related. Back-shift operator B is defined as

$$BX_t = X_{t-1} \text{ and } B^k X_t = X_{t-k} \text{ for all } k,$$

then AR(p) can be rewritten with back-shift operator as

$$\begin{aligned} X_t &= \sum_{i=1}^{p} \phi_i X_{t-i} + \epsilon_t \\ &= \sum_{i=1}^{p} \phi_i B^i X_t + \epsilon_t. \end{aligned} \tag{2.4}$$

Some operations can be applied on 2.4 to calculate the conditions of ϕ_i's.

$$\begin{aligned} X_t - \sum_{i=1}^{p} \phi_i X_{t-i} &= \epsilon_t \\ (1 - \sum_{i=1}^{p} \phi_i B^i) X_t &= \epsilon_t. \end{aligned}$$

Denote $1 - \sum_{i=1}^{p} \phi_i B^i$ as $\Phi(B)$ which is named as autoregressive polynomial of X_t, then

$$\Phi(B) X_t = \epsilon_t$$

$$X_t = \Phi(B)^{-1}(\epsilon_t).$$

Calculate moments of X_t,

$$E(X_t) = \Phi(B)^{-1}E(\epsilon_t) = 0, \tag{2.5}$$

$$Var(X_t) = \Phi(B)^{-2}Var(\epsilon_t) > 0. \tag{2.6}$$

From 2.5, $\Phi(B) \neq 0$. From 2.6, $\Phi(B)^{-2} > 0$, where $\Phi(B)^{-2} = 1/(\Phi(B)^2)$. So, $\Phi(B)^2 > 0$

By using the fundamental theorem of algebra, characteristic function $\Phi(z)$ can be factored as

$$\Phi(z) = \prod_{i=1}^{p}(1 - \frac{z}{r_i}),$$

where $r_1, \cdots, r_p \in \mathbb{C}$ is the roots of $\Phi(z)$ and $\mathbb{C}$ is complex number set. Then AR(p) is stationary and ergodic if and only if $|r_i| > 1$ for all i where $|r_i|$ is the modulus of r_i.

2.2.1.3 ARMA Processes

Another common process belong to the linear processes is the *autoregressive moving average process* of orders p and q, denoted as ARMA(p, q), where it's time dependence is modeled using an autoregressive representation as well as moving average

representation. To define ARMA(p, q), we define the function as

$$X_t = \sum_{i=1}^{p} \phi_i X_{t-i} + \sum_{j=0}^{q} \theta_j \epsilon_{t-j}.$$

Treat the sum of residuals as a whole part, then ARMA(p, q) can be written as the form of AR(p) model. Then it is a linear process with calculation.

Consider the special case, ARMA$(1, 1)$, the defining equation can be written as

$$X_t = \phi_1 X_{t-1} + \epsilon_t - \theta_1 \epsilon_{t-1}. \tag{2.7}$$

The condition of stationarity for ARMA(p, q) is the same as the condition of AR(p) model, i.e., all roots r_i's of $\Phi(B) = 0$ lie outside the unit circle, $|r_i| > 1$.

2.2.2 α-Mixing Processes

Another well known process is α-mixing process. If the process is stochastic, "mixing" represents the dependence between X_{t_1} and X_{t_2} approaches to zero when $|t_1 - t_2|$ is increasing, or means "asymptotically independent".

Consider the probability space, $(\Omega, \mathcal{F}, P)$, and let $\mathcal{F}_i^j$, $-\infty \leq i \leq j \leq \infty$ be the σ-field generated from the random variables X_t, where $i \leq t \leq j$ and $t \in \mathbf{Z}$. The

dependence coefficient for any positive integer n and given random sequence X is defined as

$$\alpha(n) = \sup_{A \in \mathcal{F}_{-\infty}^{m},\ B \in \mathcal{F}_{m+n}^{\infty},\ m \in \mathbf{Z}} |P(A)P(B) - P(A \cap B)|.$$

Then, a strictly stationary stochastic process $\{X_t\}$ is said to be α-*mixing* if

$$\alpha(n) \to 0 \text{ as } n \to \infty.$$

Remark 2.2.1 *$ARMA(p, q)$ is a α-mixing process.*

2.3 Conventional Method to Conduct Testing of Confidence Interval for μ

After the introduction of the stationary process above, we can consider the estimation of the mean.

Suppose that we are observing process which satisfy

$$X_t = \mu + \epsilon_t,$$

where $\{\epsilon_t\}$ is a stationary time series with mean zero and summable covariances, i.e. $\sum |\gamma_l|$ is infinite. Our purpose is to looking at the estimation of the mean μ. The unbiased estimation of the mean, μ, is the sample mean, given as

$$\overline{X}_n = \frac{\sum_{i=1}^{n} X_i}{n}$$

Consider the variance of $\overline{X}_n$, we could easily obtain that

$$\begin{aligned} Var(\overline{X}_n) &= \frac{1}{n} \sum_{l=-n}^{n} \left(1 - \frac{|l|}{n}\gamma_l\right) \\ &= \frac{1}{n} \left\{ \gamma_0 + 2 \sum_{1 \le l \le n} \left(1 - \frac{l}{n}\gamma_l\right) \right\}. \end{aligned}$$

Let $\sigma_n^2 = Var(\sqrt{n}\overline{X}_n)$, then Theorem 7.1.1 in [4] shows that as $n \to \infty$,

$$Var(\overline{X}_n) \to 0 \text{ if } \gamma_n \to 0,$$

and

$$\sigma_n^2 \to \sum_{l=-\infty}^{\infty} \gamma_l \text{ if } \sum_{l=-\infty}^{\infty} |\gamma_l| < \infty.$$

Refer to Theorem 27.4 in [3], suppose that X_t is stationary and α-mixing process with $\alpha(n) = O(n^{-5})$ and $E(X_t) = 0$, $E(X_t^{12}) < \infty$, then

$$Var(\sqrt{n}\overline{X}_n) \to \sum_{l=-\infty}^{\infty} \gamma_l = V,$$

$$\sqrt{n}(\overline{X}_n - \mu) \xrightarrow{d} N(0, V). \tag{2.8}$$

When estimating V, Andrews (1991) [2] states that the result of V equals the spectral density function at $\lambda = 0$ multiplied by 2π motivates the use of spectral density function. Therefore, we consider to use the spectral density function to estimate V. Consider the estimator of V as

$$\hat{V} = \frac{n}{n-1} \sum_{l=-n+1}^{n-1} k\left(\frac{l}{S_n}\right) \hat{\gamma}_l = \frac{n}{n-1} \left(2 \sum_{l=1}^{n-1} k\left(\frac{l}{S_n}\right) \hat{\gamma}_l + k(0)\hat{\gamma}_0 \right), \tag{2.9}$$

$\hat{\gamma}_l$ is one of the estimator of γ_l and $\hat{\gamma}_l = \dfrac{1}{n} \sum_{t=|l|+1}^{n} \hat{\epsilon}_t \hat{\epsilon}_{t-l}$ where $\hat{\epsilon}_t = X_t - \overline{X}_n$ for $|l| < n$.

In 2.9, S_n is a band-width parameter. The class of kernels $\mathscr{K}$, is a set of the real-valued kernels, $k(\cdot)$, which is given by

$$\mathscr{K} = \left\{ k(x) \in [-1,1],\ x \in R \;\middle|\; \begin{array}{c} k(x) = k(-x),\ k(0) = 1,\ \forall x \in R, \\ \int_{-\infty}^{\infty} k^2(x)dx < \infty, \\ k(x) \text{ is continuous at 0 at all but finite other points} \end{array} \right\}.$$

According to Parzen (1957) [17] and Andrews (1991) [2] mentioned, we consider the corresponding class of kernel estimators of the spectral density function. Parzen (1957) [17] recommends several kinds of kernels and Andrews (1991) [2] selected five of them. We consider two kernels that are commonly used, Bartlett and Quadratic Spectral kernel.

$$\begin{aligned} &\text{Bartlett:} && k_{BT}(x) = \begin{cases} 1 - |x| & \text{for } |x| \le 1, \\ 0 & \text{otherwise,} \end{cases} \\ &\text{Quadratic Spectral:} && k_{QS}(x) = \frac{25}{12\pi^2 x^2}\left(\frac{\sin(6\pi x/5)}{6\pi x/5} - \cos(6\pi x/5)\right). \end{aligned} \tag{2.10}$$

Andrews (1991) [2] presented how to get an original bandwidth parameter. The optimal bandwidth estimators, $\hat{S}_n$, for Bartlett and Quadratic Spectral kernels, are quite different. The estimators for two kernels can be calculated as follows.

$$\begin{aligned} &\text{Bartlett:} && \hat{S}_n = 1.1447((\hat{\alpha}(1)n)^{1/3}\,, \\ &\text{Quadratic Spectral:} && \hat{S}_n = 1.3221((\hat{\alpha}(2)n)^{1/5}\,. \end{aligned} \tag{2.11}$$

where $\alpha(q)$ is a function of the unknown spectral density function $f(\cdot)$, and $\hat{\alpha}(q)$ is the estimator.

As for $\alpha(q)$, the function of the spectral density function, for AR(1) model, let ψ be the autoregressive variance parameter, and ϕ_1 be the AR coefficient. Let $\hat{\psi}$ and $\hat{\phi}_1$ be their estimators respectively. Then, as mentioned by Andrews (1991) [2],

$$\begin{aligned} \hat{\alpha}(1) &= \left(\frac{2\hat{\psi}}{(1-\hat{\phi}_1)(1+\hat{\phi}_1)}\right)^2, \\ \hat{\alpha}(2) &= \left(\frac{2\hat{\psi}}{(1-\hat{\phi}_1)^2}\right)^2. \end{aligned} \tag{2.12}$$

For ARMA(1,1) and MA(1) models, the estimated functions $\hat{\alpha}(q)$ are not the same as AR(1). Andrews (1991) [2] derived the function for $\hat{\alpha}(1)$ and $\hat{\alpha}(2)$ for ARMA(1,1) and MA(1) separately.

In ARMA(1,1), the estimation of ARMA parameter (ϕ_1, θ_1) is denoted as $(\hat{\phi}_1, \hat{\theta}_1)$, then

$$\begin{aligned} \hat{\alpha}(1) &= \left(\frac{2(1+\hat{\phi}_1\hat{\theta}_1)(\hat{\phi}_1+\hat{\theta}_1)}{(1-\hat{\phi}_1)(1+\hat{\phi}_1)(1+\hat{\theta}_1)^2}\right)^2, \\ \hat{\alpha}(2) &= \left(\frac{2(1+\hat{\phi}_1\hat{\theta}_1)(\hat{\phi}_1+\hat{\theta}_1)}{(1-\hat{\phi}_1)^2(1+\hat{\theta}_1)^2}\right)^2. \end{aligned} \tag{2.13}$$

The general form of the function for MA(1) model is also given by Andrews (1991) [2].

Chapter 3

Bootstrap Methods

3.1 Jackknife and Bootstrap Methods for i.i.d. data

3.1.1 Jackknife Method

Before we start to introduce the bootstrap, jackknife method should be introduced first and we can obtain a general idea of how jackknife works. The "delete-one"

jackknife was first introduced by Quenouille (1949) [24]. Suppose that $X_1, \cdots, X_n$ is i.i.d. random sample, let $\hat{\theta}_n$ be the estimator of θ, for some function f_n, $\hat{\theta}_n = f_n(X_1, \cdots, X_n)$. Then the definition of the jackknife average is

$$\hat{\theta}_{n,i} = f_{n-1}(X_1, \cdots, X_{i-1}, X_{i+1}, \cdots, X_n).$$

The function denotes a statistic calculated with all sample except one observation. Leaving out each observation at one time from the dataset, the jackknife estimator calculates the estimate and find the average of the replicates.

The jackknife estimator of the bias $E(\hat{\theta}_n) - \theta$ can be obtained as

$$Bias = \frac{n-1}{n} \sum_{i=1}^{n} (\hat{\theta}_{n,i} - \hat{\theta}_n).$$

Set $\hat{\theta}^* = n\hat{\theta}_n - (n-1)\hat{\theta}_{n,i}$, $i = 1, \cdots, n$, which is called the ith pseudo value. Therefore, the jackknife estimator of $\hat{\theta}_n$, is given as $\bar{\theta}^* = \hat{\theta}_n - Bias$, which can be calculated

$$\bar{\theta}^* = \frac{1}{n} \sum_{i=1}^{n} \hat{\theta}^*.$$

Remark 3.1.1 *Expand $E(\hat{\theta}_n)$ in powers of n^{-1} according to the property of Quias,*

$$E_n \equiv = E(\hat{\theta}_n) = \theta + \sum_{i=1}^{\infty} \frac{a_k}{n^k},$$
$$E_{n-1} \equiv = E(\hat{\theta}_n) = \theta + \sum_{i=1}^{\infty} \frac{a_k}{(n-1)^k},$$
$$E(\bar{\theta}^*) = nE_n - (n-1)E_{n-1} = \theta - \frac{b}{n(n-1)}.$$

Remark 3.1.2 *The jackknife estimator of $Var(\hat{\theta}_n)$ is given as*

$$\widehat{Var}(\hat{\theta}_n) = \frac{1}{n(n-1)} \sum_{i=1}^{n} (\hat{\theta}^* - \bar{\theta}_n^*) = \frac{n-1}{n} \widetilde{Var}(\hat{\theta}_{n-1}).$$

The jackknife estimator $\widetilde{Var}(\hat{\theta}_{n-1})$ of $Var(\hat{\theta}_{n-1})$ is always upwards.

Unfortunately, the jackknife estimator of variance fails for many non-sufficiently smooth functions. Take the median as an example. The median of a dataset may be influenced a lot when one observation is left out. Then the calculation of the jackknife estimator of variance will not be accurate enough to describe the original dataset.

3.1.2 Bootstrap Methods

As we mentioned before, Efron (1981) [12] verified that the bootstrap is more dependable than the jackknife. Here we give a brief introduction of bootstrap methods and some particular bootstrap methods.

Suppose that $x = (x_1, \cdots, x_n)$ is an observed random sample from a distribution with cdf $F(x)$. If X^* is selected at random from x, then

$$P(X^* = x_i) = \frac{1}{n}, \quad i = 1, \cdots, n.$$

Resampling generates a random sample $X_1^*, \cdots, X_n^*$ by sampling with replacement from x. The random variables X_i^* are i.i.d., uniformly distributed on the set $x_1, \cdots x_n$.

The empirical distribution function (ecdf) $F_n(x)$ is an estimator of F_X,

$$F_n(x) = \frac{1}{n} \sum_{i=1}^{n} I\{X_i \leq x\},$$

where

$$I\{X_i \leq x\} = \begin{cases} 1 & \text{if } X_i \leq x, \\ 0 & \text{else,} \end{cases}$$

It can be shown that $F_n(x)$ is a sufficient statistic for $F(x)$ and

$$sup_x|F_n(x) - F_X| \xrightarrow{a.s.} 0, \tag{3.1}$$

according to Gilvenko-Cantelli Theorem. Moreover, $F_n(x)$ is the distribution function of a random variable that is uniformly distributed on the set $x_1, \cdots, x_n$. Hence the empirical cdf F_n is the cdf of X^*, F_n^*.

Thus in bootstrap, there are two approximations: The ecdf F_n is an approximation to the cdf F_X, which we can treat this estimation as the first or the lower layer; the ecdf F_n^* of the bootstrap replicates is an approximation to the ecdf F_n, which can be treated as the second layer or the higher level. Resampling from the sample x is equivalent to generating random samples from the distribution F_n.

To generate a bootstrap random sample by resampling x, generate n random integers $i_1, \cdots, i_n$ uniformly distributed on $1, \cdots, n$ and select the bootstrap sample $x = (x_{i_1}, \cdots, x_{i_n})$.

Suppose θ is the parameter of interest (θ could be a vector), and $\hat{\theta}$ is an estimator of θ. Then the bootstrap estimate of the distribution of $\hat{\theta}$ is obtained as follows.

1. For each bootstrap replicate, indexed $b = 1, \cdots, B$:

   (a) Generate sample $x^{*(b)} = x_1^*, \cdots, x_n^*$ by sampling with replacement from the observed sample $x_1, \cdots, x_n$.

   (b) Compute the b^{th} replicate $\hat{\theta}^{(b)}$ from the b^{th} bootstrap sample.

2. The bootstrap estimate of $F_{\hat{\theta}}(\cdot)$ is the empirical distribution of the replicates $\hat{\theta}^{(1)}, \cdots, \hat{\theta}^{(B)}$.

The bootstrap is applied to estimate the standard error and the bias of an estimator in the following sections.

Bootstrap Estimation of Standard Error

The bootstrap estimate of standard error of an estimator $\hat{\theta}$ is the sample standard deviation of the bootstrap replicates $\hat{\theta}^{(1)}, \cdots, \hat{\theta}^{(B)}$.

$$\hat{se}(\hat{\theta}^*) = \sqrt{\frac{1}{B}\sum_{b=1}^{B}(\hat{\theta}^{(b)} - \bar{\hat{\theta}}^*)^2}$$

where $\bar{\hat{\theta}}^* = \frac{1}{B}\sum_{b=1}^{B}\hat{\theta}^{(b)}$.

Bootstrap Estimation of Bias

If $\hat{\theta}$ is an unbiased estimator of θ, $E[\hat{\theta}] = \theta$. The bias of an estimator $\hat{\theta}$ for θ is

$$bias(\hat{\theta}) = E[\hat{\theta} - \theta] = E[\hat{\theta}] - \theta$$

The bootstrap estimation of bias uses the bootstrap replicates of $\hat{\theta}$ to estimate the sampling distribution of $\hat{\theta}$. For the finite population $x = (x_1, \cdots, x_n)$, the parameter is $\hat{\theta}(x)$ and there are B independent and identically distributed estimators $\hat{\theta}^{(b)}$. The sample mean of the replicates $\hat{\theta}^{(b)}$ is unbiased for its expected value $E[\hat{\theta}]$, so the

bootstrap estimate of bias is

$$\hat{bias}(\hat{\theta}) = \bar{\hat{\theta}}^* - \hat{\theta},$$

where $\bar{\hat{\theta}}^* = \frac{1}{B}\sum_{b=1}^{B}\hat{\theta}^{(b)}$, and $\hat{\theta} = \hat{\theta}(x)$ is the estimate computed from the original observed sample.

The bootstrap method for i.i.d. data is two estimations. To apply bootstrap methods into dependent data, especially time series, covariance of the dependent data is necessary to consider into different bootstraps since covariance is quite important to describe the dependent data. Therefore, some different bootstrap methods are introduced.

3.2 Bootstrap Methods for Dependent Data

As we mentioned before, the conventional method to estimate the population mean is based on the limiting distribution being standard normal distribution. However, in time series, Andrews method to estimate the variance can not be applied every time. First, Andrews method is based on the selection of bandwidth in the kernel calculation. Even though Andrews listed several selection of bandwidth, it is not enough to cover all kind of models. Selection of bandwidth is a hard thing to be

determined in Andrews methods. Second, it is known that Andrews method may not deliver accurate size or coverage rates in finite samples. Hereby, we introduce the bootstrap methods to approximate the limiting distribution. The main idea of most bootstraps methods is re-sampling. The most popular bootstrap methods for dependent data, especially time series, are block bootstrap, sieve bootstrap and stationary bootstrap.

3.2.1 Block Bootstraps

Bootstrap methods are aiming to re-sample the data to get the estimator not related on the limiting distribution. However, in normal cases, the bootstrap which was introduced by Efron (1979) [11] is dealing with i.i.d dataset. Bootstrap has been developed by several researchers to explore the fields that it can deal with to make it more useful and developed. The principal ideas of different bootstraps are quite similar, but with different models and procedures. The block bootstraps are introduced to deal with dependent data series, especially time series.

Instead of resampling the whole dataset which was introduced by Efron (1979) [11], subsamples are randomly captured from the whole series which is divided into several blocks, with block length $l < n$ and b blocks, to keep the dependent structure of neighbored observations. Hereby, we introduce two block bootstrap methods.

3.2.1.1 Moving Block Bootstrap

Moving block bootstrap, in short MBB, is one of the popular bootstrap methods in inference of time series. Overlapping block bootstrap was introduced [13] in the same setting. Suppose that the series $\{X_t\}$ is a sequence of weak stationary variables, divide the series into several blocks under the condition

$$l \to \infty \text{ as } n \to \infty \text{ but with } \frac{l}{n} \to 0,$$

where l is the block length and n is the number of total observations in time series. Under the condition of weak stationary, dependence are kept within each block, and these blocks can be assumed as i.i.d.

In MBB, after determing the length of blocks, l, we can get $N = n - l + 1$ i.i.d blocks $Y_i = (X_i, \cdots, X_{i+l-1}),\ 1 \leq i \leq N$.

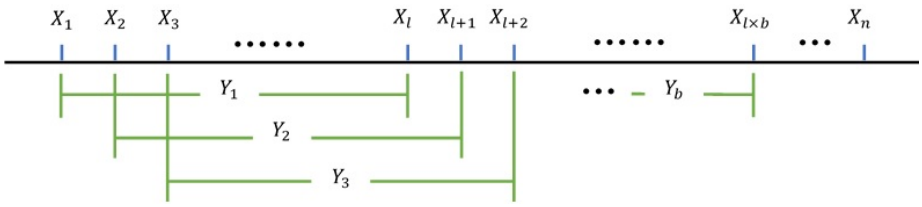

Figure 3.1: Moving Block Bootstrap

In N i.i.d blocks, we randomly select $b = \lfloor n/l \rfloor$ blocks, $Y_1^*, \cdots, Y_b^*$, to construct a new series with these selected blocks as $\{X_1^*, \cdots, X_m^*\}$ where $m = lb$. The total number of the samples in the new series is $l \times b$. With the new series, we can calculate the new sample mean $\overline{X}_n^*$ in each replicates, denoted by $\overline{X}_n^{*(b)}$, and it is easy to get the confidence interval with B replicates.

Theorem 6 in [16] shows the consistency of MBB under some conditions.

Theorem 3.2.1 *Suppose that random variables* $\{X_t\},\ t = 1, \cdots, n$ *construct a stationary m-dependent sequence with* $EX_1 = \mu$ *and* $E|X_1|^{4+\delta} < \infty$ *for some positive* δ. *If* $l/n \to 0$ *when* $n \to \infty$, *then*

$$\sup_x |P^*\{\sqrt{m}(\overline{X^*}_m - E^*(\overline{X^*}_m)) \leq x\} - P\{\sqrt{n}(\overline{X}_n - \mu) \leq x\}| \xrightarrow{P} 0, \tag{3.2}$$

where P^* is the bootstrap probability and E^* is the mean under the bootstrap probability and $\overline{X}^*_m = \frac{\sum_{t=1}^{m} X_t^*}{m}$.

If the condition $l/n \to 0$ can be replaced by $l/\sqrt{n} \to 0$ as $n \to \infty$, then $E^*\overline{X^*}_m$ can be replaced by $\overline{X}$ in 3.2.

In Theorem 3.2.1, m-dependence is defined as follows. Let $\{X_1, X_2, \cdots\}$ be a sequence with random variables, let A be an event based on the sequence $\{X_1, \cdots, X_h\}$ and B be an event based on $\{X_{h+1+m}, \cdots, \}$. If any pair of events A and B are independent, then the sequence $\{X_t\}$ is m-dependent. As Liu and Singh mentioned in [16], "The notion of m-dependence is probably the most basic model which takes into account such dependence." The proof of this theorem can be found in [16].

In Theorem 3.2.1, two different cumulative density functions are involved, and two estimations are considered, which can be treated as two different layers. The first estimation or the first layer, is the estimation of population with the given sample, which is expressed as $P\{\sqrt{n}(\overline{X}_n - \mu) \leq x\}$. This is the usual layer we are looking at without bootstrap. If the limiting distribution is involves nuisance parameters, bootstrap can be used to approximate the limiting distribution of a function of

given data, for example, the mean. Therefore, the second estimation, or the second layer, which is expressed by $P^*\{\sqrt{m}(\overline{X}_m^* - E^*(\overline{X}_m^*)) \leq x\}$, can be generated. Then Theorem 3.2.1 shows that the difference between the two cumulative density functions (one of them is the cdf of bootstrap method, and another one is the cdf of the sample series) convergent to 0 in probability.

3.2.1.2 Non-overlapping Block Bootstrap

The nonoverlapping block bootstrap, as known as NBB, is firstly introduced by Carlstein in 1986 for univariate time series. [9] NBB and MBB only have several differences on defining the blocks. In MBB, the total number of blocks is $n - l + 1$, and the blocks are defined overlapping. However, in NBB, we divide the whole series into approximately independent blocks $\{Y_i\}$, where $Y_i = \{X_{(i-1)l+1}, \cdots, X_{il}\}$, $i = 1, \cdots, b$, and then we sample b blocks $Y_1^*, \cdots, Y_b^*$, putting these blocks together to get $X_1^*, \cdots, X_m^*$.

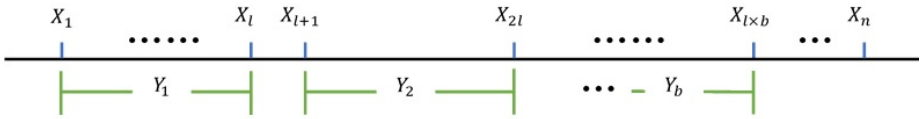

Figure 3.2: Non-overlapping Block Bootstrap

Since we already showed the consistency of MBB, comparing MBB with NBB, if the difference between MBB and NBB can be insignificant and negligible, then NBB is consistent.

The estimated parameter in bootstrap version in general, can be denoted as $\theta^*_{bl,n} = T(F^*_{bl,n})$, and in special, we are discussing the mean for all of these methods. To separate MBB and NBB, the estimator of MBB in the bootstrap version is denoted as $\theta^{*(M)}_{bl,n}$, while the estimator of NBB is $\theta^{*(N)}_{bl,n}$.

$$\theta^{*(M)}_{bl,n} = \frac{1}{bl}\sum_{j=1}^{bl} X_j^{*(M)}, \text{ and } \theta^{*(N)}_{bl,n} = \frac{1}{bl}\sum_{j=1}^{bl} X_j^{*(N)}.$$

The probability of selecting any block is N^{-1},

$$\begin{aligned} & P\{(X_1^*, \cdots, X_l^*) = (X_i, \cdots, X_{i+l-1})|X_n\} \\ =& P\{Y_j = i|X_n\} \\ =& N^{-1} \text{ ,for } 1 \le i \le N. \end{aligned} \tag{3.3}$$

While the probability of selecting any block is b^{-1},

$$\begin{aligned} & P\{(X_1^*, \cdots, X_l^*) = (X_{(i-1)l+1}, \cdots, X_{il}) | X_n\} \\ =& P\{Y_j = i | X_n\} \\ =& b^{-1} \text{ ,for } 1 \leq i \leq b. \end{aligned} \tag{3.4}$$

Hence, from 3.3, we get

$$\begin{aligned} E(\theta_{bl,n}^{*(M)}) &= \frac{1}{N} \sum_{j=1}^{N} \left(\frac{1}{l} \sum_{i=1}^{l} X_{j+i-1} \right) \\ &= \frac{1}{Nb} \left(\sum_{r=1}^{l-1} r(X_r + X_{N_r+1}) + l \sum_{s=l}^{N} X_s \right) \\ &= \frac{1}{N} \left(\sum_{r=1}^{n} X_r - \frac{1}{l} \sum_{s=1}^{l-1} (l-s)(X_s + X_{n-s+1}) \right). \end{aligned} \tag{3.5}$$

And from 3.4, we get

$$\begin{aligned} E(\theta_{bl,n}^{*(N)}) &= \frac{1}{b} \sum_{j=1}^{b} \left(\frac{1}{l} \sum_{i=1}^{l} X_{(j-1)l+i} \right) \\ &= \frac{1}{bl} \left(\sum_{r=1}^{n} X_r - \sum_{i=bl+1}^{n} X_i \right). \end{aligned} \tag{3.6}$$

Notice that when the process $\{X_t\}$ is under some conditions and some standard moment, the bias of these two estimators is $E(\theta_{bl,n}^{*(M)}) - E(\theta_{bl,n}^{*(N)})$, and the expectation

of the squared bias is $E(E(\theta^{*(M)}_{bl,n}) - E(\theta^{*(N)}_{bl,n}))^2 = O(l/n^2)$, that is, for large sample size, the difference between these two methods is insignificant and negligible. [15]

3.2.1.3 Circular Block Bootstrap

Another popular bootstrap method with fixed block length, is circular block bootstrap, also known as CBB. After the introduction NBB and MBB, CBB was firstly introduced by Politis and Romano in 1994 [20], which 'wrap' the series into a circle, to make fully use of every observation in the series, comparing with NBB. As we can see, in NBB, the last several elements cannot be included in any block when the length of the block is not a divisor of the total number of observations. Then, when we construct bootstrap procedure, these not-included observations are deleted by resampling. However, in CBB, when the series is wrapped like a circle, and there are n blocks, while $n - l + 1$ blocks are constructed in MBB. The "wrapped circle" is defined as $X_i \equiv X_j$ for $i > n$, where $j = i \mod n$, and $X_0 = X_n$.

As mentioned above, let the series be $\{X_1, \cdots, X_n\}$, then the whole arc of the "wrapped circle" is $\{X_1, \cdots, X_n, X_1, \cdots, X_{n+l-1}\}$, thus, by setting the block length as l, there are totally n blocks, and each block is defined as $Y_i = \{X_i, \cdots, X_{i+l-1}\}$.The

block set is $\{Y_1, \cdots, Y_n\}$. Sampling with replacement with probability $1/n$, randomly select b i.i.d blocks from the set where $lb = m \approx n$. Then we have a new series as $X_1^*, \cdots, X_m^*$. With the new series, the confidence interval can be calculated by following the similar procedure mentioned above, and we can also get the true percentage.

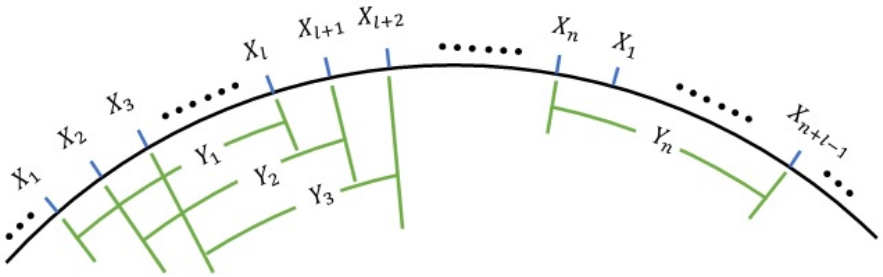

Figure 3.3: Circular Block Bootstrap

The following theorem were stated by Politis, D.N., Romano, J.P. [20] shows the consistency and asymptotic accuracy of CBB. The theorem is aiming to show the consistency of circular block bootstrap with α-mixing series.

Theorem 3.2.2 *Let $\{X_t\}$ be a sequence assumed to be with weak stationarity. Specially, the α-mixing condition ($\alpha(h) \to 0$ as $h \to \infty$, where $\alpha(h)$ is defined in Chapter 2) is assumed. Assume that $E|X_t|^{6+\delta} < \infty$ for some positive δ, and*

$\sum_{h=1}^{\infty} n^2(\alpha(h))^{\frac{\delta}{6+\delta}} < \infty$. *As* $n \to \infty$, *let* $m/n \to 1$ *where* $m = lb$, *and let* $l \to \infty$ *but* $b/n \to 0$. *Then* $\sigma_n^2 \equiv Var(\sqrt{n}\overline{X}_N)$ *is limiting to* $\sigma_\infty^2 < \infty$, *and* $Var^*(\sqrt{m}\overline{X}^*_m) \xrightarrow{P} \sigma_\infty^2$, *where* Var^* *is the variance of* $\sqrt{m}\overline{X}^*_m$ *under the bootstrap probability* P^*, *and*

$$sup_x|P^*\{\sqrt{m}(\overline{X}^*_m - \overline{X}_n) \leq x\} - P\{\sqrt{n}(\overline{X}_n - \mu) \leq x\}| \xrightarrow{p} 0, \tag{3.7}$$

for almost all sample series $\{X_t\}$ *for* $t = 1, \cdots, n$.

The meaning of Theorem 3.2.2 is similar with the Theorem 3.2.1. The theorem shows the consistency of CBB. Since blocks are defined similar in MBB, NBB and CBB, the proofs of consistency of three block bootstraps are quite similar.

3.2.1.4 Stationary Bootstrap

In the previous three methods, the block length is fixed. A new resampling method, which is also generally available and useful for stationary weakly time series, called stationary bootstrap, also known as SB, was introduced by Politis and Romano [21] in 1994. In contrast to the former mentioned block bootstraps, resampling procedure is repeated in stationary bootstrap to get an approximation to the

distribution of the statistic with generating the pseudo-time series as a stationary time series, i.e. stationary bootstrap aims to retain the stationarity of the original series in resampling. The block is of random length, and the block length can be assigned to have some kind of distribution. Usually, the selected distribution of the block length is a geometric distribution.

Suppose that $\{X_t\}$ is a given stationary weakly time series as mentioned before. Define each block as $B_{i,l} = \{X_i, X_{i+1}, \cdots, X_{i+l-1}\}$, where l is length of the block. Since when we randomly choose the starting point of each block and the block length has a geometric distribution, then observations in some block may exceed the given sample sequence. To avoid "cutting off" some observation in some blocks, "wrapping" the series may help to keep all observations which should be included in the blocks. Define the "wrap" as $X_i \equiv X_j$ for $i > n$, where $j = i(\mod n)$, and $X_0 = X_n$. Since the block lengths may be different, then define $l_1, l_2, \cdots$ be the block length independent to X_i, and $\{l_i\}$ is a sequence of i.i.d random variables which have a geometric distribution $P\{l_i = u\} = (1 - p_S)^{u-1} p_S$ for $u = 1, 2, \cdots$. Then for any given p_S, we can generate the sequence $\{l_i\}$. As for the index of the starting point of each block i_i independent to X_i and l_i, randomly select from $\{1, \cdots, n\}$, i.e. $\{i_i\}$ is a sequence of i.i.d random variables having the discrete uniform distribution

on $\{1, \cdots, n\}$. Therefore, the sampled sequence of blocks with random length is $\{B_{i_1,l_1}, B_{i_2,l_2}, \cdots\} = \{X_{i_1}, X_{i_1+1}, \cdots, X_{i_1+l_1-1}, X_{i_2}, X_{i_2+1}, \cdots, X_{i_2+l_2-1}, \cdots\}$. If the sequence length is greater then the sample size n, then stop the process once n observations are generated. By simulating a large number of pseudo time series, the distribution of T^* can be approximated, the critical value and confidence interval can be calculated then.

Like the previous theorems mentioned above, the consistency of stationary bootstrap aims to show the difference of two estimations convergent to 0. Theorem 1 in [21] shows the consistency of stationary bootstrap.

Theorem 3.2.3 *Assume that* $\{X_t\}$ *for* $t = 1, 2, \cdots$ *is a strictly stationary process which the autocovariance* $\gamma(\cdot)$ *satisfies* $\gamma(0) + \sum_r |r\gamma(r)| < \infty$. *Assume the condition 3.8 holds where* $\kappa_4(u, v, w)$ *is the fourth joint cumulant of the distribution of* $(X_i, X_{i+u}, X_{i+v}, X_{i+u+v+w})$.

$$\sum_{u,v,w} |\kappa_4(u, v, w)| = K < \infty. \tag{3.8}$$

Assume $E|X|^{2+\delta} < \infty$ *and* $\sum_h [\alpha(h)]^{\frac{\delta}{2+\delta}}$ *for some positive* δ *where* $\alpha(h)$ *is defined in Chapter 2.*

Then, $\sigma_\infty^2 = Var(X_1) + 2\sum_{i=1}^{\infty} cov(X_1, X_{1+i})$ *is finite.*

If $\sigma_\infty^2 > 0$*, then*

$$sup_x |P\{\sqrt{n}(\overline{X}_n - \mu) \leq x\} - \Phi(x/\sigma_\infty)| \to 0, \tag{3.9}$$

where $\Phi(\cdot)$ *is the standard normal distribution function.*

Assume that the lag $p_S \to 0$ *in the geometric distribution and* $np_S \to \infty$ *, then the bootstrap distribution is close to the true sampling distribution given as*

$$sup_x |P^*\{\sqrt{n}(\overline{X^*} - \overline{X}_n) \leq x\} - P\{\sqrt{n}(\overline{X}_n - \mu) \leq x\}| \xrightarrow{P} 0,$$

where P^* *is the bootstrap probability mentioned above.*

It has to be mentioned that, the form of the convergence in Theorem 3.2.3 is quite similar with the previous theorems only with $\sqrt{n}$ in the first cumulative function instead of m. Since in stationary bootstrap, we stop generating sequence once n variables are generated, then the total length of the new sequence is the same with the original data. It is possible to get n variables in NBB, MBB or CBB if the block

length is chosen in some way.

3.2.2 AR-Sieve Bootstrap

AR-sieve bootstrap develops another way. Instead of constructing blocks from the original data, AR-sieve bootstrap aims to establish an AR($\hat{p}$) time series model and construct series based on the model to estimate the parameter. The essentials in AR-sieve bootstrap is to estimate the parameters in AR($\hat{p}$) model and to determine the residuals from the original series in [5], [7], [6].

Let $\{X_1, \cdots, X_n\}$ be the original time series. The purpose of AR-sieve approximation is to construct an AR($\hat{p}$) model

$$X_t - \mu = \sum_{i=1}^{\hat{p}} \phi_i (X_{t-i} - \mu) + \epsilon_t,$$

where μ is $E(X_t)$ and ϵ_t is an innovation sequence independent of $\{X_s; s < t\}$, which is of i.i.d random variables with expectation 0. Since we need to estimate the series with an AR($\hat{p}$) model, then the first thing is to determine the order $\hat{p}$.

Usually Akaike information criterion can help to find the autoregressive order $\hat{p}$. Let $\hat{\phi}_{\hat{p}} = (\hat{\phi}_{1,n}, \hat{\phi}_{2,n}, \cdots, \hat{\phi}_{\hat{p},n})$ be the Yule-Walker autoregressive parameter estimators, then

$$\hat{\phi}_{\hat{p}} = \hat{\Gamma}_{\hat{p}}^{-1} \hat{\gamma}_{\hat{p}},$$

$$\hat{\gamma}_k = \frac{1}{n} \sum_{t=1}^{n-|k|} (X_t - \overline{X}_n)(X_{t-|k|} - \overline{X}_n), \text{ for } 0 \leq k \leq \hat{p},$$

where $\overline{X}_n = \dfrac{\sum_{t=1}^{n} X_t}{n}$, $\hat{\Gamma}_{\hat{p}} = \hat{\gamma}(|s-r|)$ for $s, r = 1, \cdots, \hat{p}$, and $\hat{\gamma}_{\hat{p}} = (\hat{\gamma}(1), \cdots, \hat{\gamma}(\hat{p}))'$.

With Yule-Walker estimators, some part of the estimated AR($\hat{p}$) can be determined. Then another important part is the residuals. To keep the information of the original data, let $\hat{\epsilon}_t = X_t - \sum_{i=1}^{\hat{p}} \hat{\phi}_i X_{t-i},\ i = \hat{p}+1, \cdots, n$ be the residuals of the model fit, then we can define the empirical distribution function of i.i.d. residuals.

$$F_{\hat{\epsilon}_t}(x) = \frac{1}{n - \hat{p}} \sum_{t=\hat{p}+1}^{n} I[\hat{\epsilon}_t - \bar{\hat{\epsilon}}_t \leq x],$$

where $\bar{\hat{\epsilon}}_t = \dfrac{1}{n-\hat{p}} \sum_{t=\hat{p}+1}^{n} \hat{\epsilon}_t$.

The last thing of AR-sieve bootstrap, is to construct a new series based on the fitted

model. The fitted AR($\hat{p}$) model is given as

$$X_t^* - \overline{X}_n = \sum_{i=1}^{\hat{p}} \hat{\phi}_i (X_{t-i}^* - \overline{X}_n) + \hat{\epsilon}_t^*, \tag{3.10}$$

where $\hat{\epsilon}_t^*$ is i.i.d. residuals with marginal distribution $F_{\hat{\epsilon}_t}(x)$. Since if we set $\hat{p}$, then the first $\hat{p}$'s estimated values cannot be calculated according to 3.10, then in order to get a series, it is necessary to set these estimated values. Several ways can help to set these values, for example, set these as 0, or as $\overline{X}_n$. Assuming that $Y = f(X_1, \cdots, X_n)$ is a function of the original data, then the AR-sieve bootstrap function can be defined as $Y^* = f(X_1^*, \cdots, X_n^*)$. To show the consistency of sieve bootstrap, Theorem 3.1 in [5] gives a proof for general cases with AR(∞) model. Assume that $\{X_t\}$ for $t \in Z$ is a stationary process with $EX_t = \mu$. As what has been mentioned in [5], according to Wold's Theorem, if the sequence $\{X_t\}$ is purely stochastic, it can be written as one-sided MA(∞) process according to Section 2.10 in [15]

$$X_t - \mu = \sum_{i=0}^{\infty} \psi_i \epsilon_{t-i}, \tag{3.11}$$

where $\psi_0 = 1$ and $\{\epsilon_t\}$ is a sequence constructed with uncorrelated variables, with $E\epsilon_t = 0$ and $\sum_{i=0}^{\infty} \psi_i^2 < \infty$.

Under some additional assumptions of invertibility that we require of the process in

3.11(see [1], Theorem 7.6.9), $\{X_t\}$ can be represented as a one-sided AR(∞) model

$$\sum_{i=0}^{\infty} \phi_i (X_{t-i} - \mu) = \epsilon_t, \tag{3.12}$$

where $\phi_0 = 1$ and $\sum_{i=0}^{\infty} \phi_j^2 < \infty$.

Write

$$\Phi(z) = \sum_{i=0}^{\infty} \phi_i z^i, \ \phi_0 = 1, \ z \in C,$$
$$\Psi(z) = \sum_{i=0}^{\infty} \psi_i z^i, \ \psi_0 = 1, \ z \in C.$$

Then model 3.11 and 3.12 can be written as

$$\Phi(B)(X - \mu) = \epsilon, \ X - \mu = \Psi(B)\epsilon,$$

where B is the back shift operate $(BX)_t = X_{t-1}$, $x \in R^Z$. So $\Phi(z) = 1/\Psi(z)$.

Let $\mathcal{F}_t$ be the σ-field generated by $\{\epsilon_t\}_{s=-\infty}^{t}$, then $\mathcal{F}_t = \sigma(\{\epsilon_s; s \leq t\})$. Before we start to show the consistency of sieve bootstrap, some assumptions are necessary to mention. The following assumptions are shown as Assumtion A1, A1', A2 and B in [6]

Assumption 3.2.1 *$X_t - \mu = \sum_{i=0}^{\infty} \psi_i \epsilon_{t-i}$ where $\psi_0 = 1$ with $\{\epsilon_t\}$ stationary, ergodic. And $E[\epsilon_t|\mathcal{F}_{t-1}] \equiv 0$, $E[\epsilon_t^2|\mathcal{F}_{t-1}] \equiv \sigma^2 < \infty$, $E|\epsilon_t|^s < \infty$ for some $s \geq 4$.*

Since sieve bootstrap process get independent residual from the innovations, usually it is unable to satisfy the assumption with non-independent variables $\{\epsilon_t\}$. Therefore, we strength the assumption above.

Assumption 3.2.2 *$X_t - \mu = \sum_{i=0}^{\infty} \psi_i \epsilon_{t-i}$ where $\psi_0 = 1$ with $\{\epsilon_t\}$ i.i.d and $E[\epsilon_t] = 0$, $E|\epsilon_t|^s < \infty$ for some $s \geq 4$.*

Assumption 3.2.3 *$\Psi(z)$ is bounded away from zero for $|z| \leq 1$, $\sum_{i=0}^{\infty} i^r |\psi_i| < \infty$ for some $r \in N$.*

Assumption 3.2.4 *let $\hat{p} = \hat{p}(n)$, $\hat{p}(n) \to \infty$ and $\hat{p}(n) = o(n)$ as $n \to \infty$. And $\hat{\phi}_{\hat{p}} = (\hat{\phi}_{1,n}, \cdots, \hat{\phi}_{\hat{p},n})'$ satisfy the empirical Yule-Walker equations*

$$\hat{\phi}_{\hat{p}} = -\hat{\Gamma}_{\hat{p}}^{-1} \hat{\gamma}_{\hat{p}}.$$

Theorem 3.2 in [6] will show that the sieve bootstrap is consistent, even with non-independent innovations in Assumption 3.2.1.

Theorem 3.2.4 *Let $s = 4$ in Assumption 3.2.1, $r = 1$ in Assumption 3.2.3, $p(n) \to o((n/\log(n))^{1/4})$ in Assumption 3.2.4, then*

1. $Var^*\left(\frac{\sum_{t=1}^n X_t^*}{\sqrt{n}}\right) - Var\left(\frac{\sum_{t=1}^n X_t}{\sqrt{n}}\right) = o_{\hat{p}}(1)$ *as* $n \to \infty$;

2. *If, in addition,* $\frac{\sum_{t=1}^n (X_t - \mu)}{\sqrt{n}} \xrightarrow{d} N(0, \sum_{h=-\infty}^{\infty} \gamma(h))$, *then as* $n \to \infty$

$$sup_x \left| P^* \left\{ n^{-1/2} \sum_{t=1}^n (X_t^* - \overline{X^*}) \leq x \right\} - P\left\{ n^{-1/2} \sum_{t=1}^n (X_t - \mu) \leq x \right\} \right| = o_{\hat{p}}(1).$$

Theorem 3.2.4 shows that the consistency of sieve bootstrap in general case when the lag is infinity. Our model is AR(1) model, which is a special case of AR(∞). Therefore, sieve bootstrap is also consistent when the lag $p = 1$.

We will do simulation of each bootstrap method in the next chapter, and also compare the simulation results.

Chapter 4

Simulations of Methods

In Chapter 3, several bootstrap methods for time series are shown to be consistent. However, different bootstrap methods have different results for basic models. Here, we use four basic linear process models: three are AR(1), MA(1) and ARMA(1,1), one is based on AR(1), AR(1)-SEASON, which considers the weights, like for 12 months, based on AR(1) model.

Andrews estimation of variance in [2] is quite simple and straightforward. The process is shown in 4.1.1. As for the bootstrap methods, we would like to see

which bootstrap methods is better to deal with the certain kind of time series models by calculating the the average convidence interval coverage rate of $\mu = 0$ lying in the confidence interval by Monte Carlo method. Different bootstraps have different ways to approximate quantiles of limiting distribution for calculating confidence interval from the empirical distribution, however, the main ideas are the same.

Suppose we wish to approximate the distribution of

$$T^2 = \left(\frac{\sqrt{n}(\overline{X}_n - \mu)}{\hat{\sigma}_n} \right)^2 . \tag{4.1}$$

By using Monte Carlo simulations, we can draw many bootstrap samples and find out the estimators to create the bootstrap version of T^2:

$$T^{*2} = \left(\frac{\sqrt{n}(\overline{X}^*_n - \overline{X}_n)}{\hat{\sigma}^*_n} \right)^2 . \tag{4.2}$$

Once we get the histogram of the distribution, we can find out the quantiles of this distribution, and even create the confidence interval.

$$\overline{X}_n \pm \sqrt{T^{*2}_\alpha} \frac{\hat{\sigma}_n}{\sqrt{n}},$$

where T_α^{*2} is the 95% quantile of bootstrapped T^{*2}.

By replicating computing the confidence interval, we can get the average convidence interval coverage rate of 0 ling in the confidence interval in finite samples. When $\phi_1 = 0$, i.e. the series contains random i.i.d. variables, the percentage we calculate, should be close to 95%, as we set the confidence level being 95%.

Since we have to estimate the variance to get the confidence interval, one way to avoid the estimation, is to change the form of the equations. We all know that the confidence interval calculated with the critical value is

$$\overline{X}_n \pm \sqrt{T_\alpha^{*2}} \frac{\hat{\sigma}_n}{\sqrt{n}}.$$

4.1 and 4.2 can be rewritten as

$$(T\hat{\sigma}_n)^2 = (\sqrt{n}(\overline{X}_n - \mu))^2,$$

$$(T^*\hat{\sigma}_n^*)^2 = (\sqrt{n}(\overline{X}_n^* - \overline{X}_n))^2,$$

Since the sample are generated from the same population, the standard deviation

should be the same. Therefore, let the second equation as M^*, then

$$\pm M^* = \sqrt{n}(\overline{X}_n - \mu),$$
$$\frac{\pm M^*}{\sqrt{n}} = \overline{X}_n - \mu.$$

Therefore, the confidence interval can also be calculated as

$$\overline{X}_n \pm \frac{M^*}{\sqrt{n}}.$$

Here we calculate two percentages with two different ways to construct the confidence interval as mentioned above in each bootstrap method.

We also consider about the average length of confidence interval. If the coverage rates are similar, then comparing the interval length, the less the average interval length, the more accurate the appxomation of the method based on a certain model.

4.1 Simulation Process

4.1.1 Andrews Estimation Process

The Andrews estimation of variance method in [2] to estimate the variance for dependent data is mentioned in Chapter 1. According to the formula, calculation of estimated variance is quite straightforward.

1. Generate $R = 10000$ replicates for Bartlett kernel or Quadratic Spectral kernel separately:

   (a) Generate $\{X_t\}, t = 1, 2, \cdots, n$ from AR(1) model with certain ϕ_1;

   (b) Calculate the $T^2 = \dfrac{n * (\overline{X}_n - \mu)^2}{\hat{V}}$, where $\overline{X}_n = \dfrac{1}{n}\sum_{i=1}^{n} X_i$, $\hat{V}$ is the Andrews estimated variance with Bartlett kernel or Quadratic Spectral kernel according to the formulas in Chapter 1, 2.9.

(c) Compare T^2 with 95% quantile T_0^2 of χ^2 distribution. If $T^2 < T_0^2$, then count it, i.e. $q + 1$.

2. Calculate the the average convidence interval coverage rate by q/R.

4.1.2 Bootstrap Processes

When simulating the bootstrap methods, we calculate the result with or without Andrews estimation of variance in [2] using Quadratic Spectral kernel to compare the influence of using the estimation of variance for each model. In the following subsections, we only list the process of AR(1) model. The processes of other models are the same. We also consider AR(1)-SEASON model mentioned by Andrews (1991) [2] even though this model is not stationary.

4.1.2.1 Moving Block Bootstrap Process

The process of MBB bootstrap is given as follows:

1. Generate the R replicates to get the the average convidence interval coverage rate:

   (a) Generate the AR(1) time series;

   (b) Generate the B replicates for bootstrap of the series:

      i. Divide the series into $n - l + 1$ blocks $Y_1, Y_2, \cdots, Y_{n-l+1}$ with length l;

      ii. Randomly select b blocks with replacement from these blocks, to get a new series, $Y_1^*, Y_2^*, \cdots, Y_b^*$, which can also be represented as $X_1^*, X_2^*, \cdots, X_m^*$;

iii. Calculate the value T^{2*} or M^{2*};

(c) Get the distribution of T^{2*} or M^{2*} and their 95% quantile;

(d) Calculate the 95% confidence interval;

(e) Compare 0 with the confidence interval; if 0 lies in the confidence interval, then count this replicate; otherwise, do not count it.

2. Get the percentage of the counted number respect to the R replicates.

The process of non-overlapping block method and circular bootstrap method are quite similar. We also list the process in the following subsections. We consider these three methods with the same block length.

4.1.2.2 Non-overlapping Block Bootstrap

The procedure of calculating the the average convidence interval coverage rate with NBB is quite similar with MBB procedure, the only difference is the way to determine blocks and the total number of blocks. Similar to MBB, the process is given as:

1. Generate the R replicates to get the the average convidence interval coverage rate:

   (a) Generate the AR(1) time series;

   (b) Generate the B replicates for bootstrap of the series:

      i. Divide the series into b blocks $Y_1, Y_2, \cdots, Y_b$ with length l, where $b \times l \approx n$;

      ii. Randomly select b blocks with replacement from these blocks, to get a new series, $Y_1^*, Y_2^*, \cdots, Y_b^*$, which can also be represented as $X_1^*, X_2^*, \cdots, X_m^*$;

      iii. Calculate the value T^{*2} or M^{*2};

   (c) Get the distribution of T^{*2} or M^{*2} and their 95% quantile;

   (d) Calculate the 95% confidence interval;

(e) Compare 0 with the confidence interval; if 0 lies in the confidence interval, then count this replicate; otherwise, do not count it.

2. Get the percentage of the counted number respect to the R replicates.

4.1.2.3 Circular Block Bootstrap

The process of CBB bootstrap is as follows:

1. Generate the R replicates to get the the average convidence interval coverage rate:

   (a) Generate the AR(1) time series;

   (b) Generate the B replicates for bootstrap of the series:

      i. "Wrap" the series as a circle;

ii. Divide "wrapped circle" into n blocks $\{Y_1, Y_2, \cdots, Y_n\}$ with length l;

iii. Randomly select b blocks with replacement from these blocks, to get a new series, $Y_1^*, Y_2^*, \cdots, Y_b^*$, which can also be represented as $X_1^*, X_2^*, \cdots, X_m^*$;

iv. Calculate the value T^{*2} or M^{*2};

(c) Get the distribution of T^{*2} or M^{*2} and their 95% quantile;

(d) Calculate the 95% confidence interval;

(e) Compare 0 with the confidence interval; if 0 lies in the confidence interval, then count this replicate; otherwise, do not count it.

2. Get the percentage of the counted number respect to the R replicates.

These three block bootstrap methods mentioned above are with the fixed block length and the total number of randomly selected blocks. Another kind of block bootstrap with random block length having some distribution was introduced to compare with the fix-block-length block bootstraps.

4.1.2.4 Stationary Bootstrap

In stationary bootstrap, the block length is not fixed, it follows a geometric distribution. We choose several values for parameter p_S to compare the result. Lahiri (1999) [14] mentioned the way to calculate the expected block length of moving bootstrap method and stationary bootstrap. With the detailed calculation, we can get the expected block length of stationary bootstrap, furthermore, we can get the parameter p_S. We will get the result in Chapter 5. The process of stationary bootstrap is given as follows.

1. Generate the R replicates to get the the average convidence interval coverage rate:

   (a) Generate the AR(1) time series;

   (b) Generate the B replicates for bootstrap:

      i. Set the probability p_S in the geometric distribution. Generate a

sequence $\{l_i\}$ which is formed as i.i.d random variables having the same distribution;

ii. Generate sequence $\{i_i\}$ having discrete uniform distribution on $\{1, cdots, n\}$;

iii. Construct block sequence $B_{i_1,l_1}, B_{i_2,l_2} \cdots$ with random length l_i, stop once n observations are generated. A new series can be represented as $X_1^*, X_2^*, \cdots, X_m^*$;

iv. Calculate the value T^{*2} or M^{*2};

(c) Get the distribution of T^{*2} or M^{*2} and their 95% quantile;

(d) Calculate the 95% confidence interval;

(e) Compare 0 with the confidence interval; if 0 lies in the confidence interval, then count this replicate; otherwise, do not count it.

2. Get the percentage of the counted number respect to the R replicates.

Block bootstrap methods mentioned above are the most common block bootstraps to deal with time series. Another kind of method, which is based on "rebuilding" the series with AR($\hat{p}$) model to estimate the series, is AR-Sieve bootstrap.

4.1.2.5 AR-Sieve Bootstrap

Similar to the previous bootstraps, the process of AR-sieve bootstrap is as follows.

1. Generate the R replicates to get the the average convidence interval coverage rate:

   (a) Generate the AR(1) time series, a sequence will be generated and assumed as the given data sequence;

   (b) Generate the B replicates for bootstrap:

      i. Calculate the estimated parameters in fitted model AR(p) with

Yule-Walker estimators;

ii. Calculate the residuals $\{\hat{\epsilon}_t\}$ from the model and the original series;

iii. Construct the empirical distribution of the centered residual $F_{\hat{\epsilon}_t}$;

iv. Generate a new sequence based on the fitted model and set the beginning values as $\overline{X}_n$;

v. Calculate the value T^{*2} or M^{*2};

(c) Get the distribution of T^{*2} or M^{*2} and their 95% quantile;

(d) Calculate the 95% confidence interval;

(e) Compare 0 with the confidence interval; if 0 lies in the confidence interval, then count this replicate; otherwise, do not count it.

2. Get the percentage of the counted number respect to the R replicates.

The processes of bootstrap methods we mentioned above are quite similar except AR-Sieve bootstrap. The results of all simulations are in tables in Section 4.3. We

can compare the results of all models and all methods correspondingly.

4.2 Simulation Modes

We mentioned processes of five common bootstrap methods for time series above. In the simulations, we consider AR(1), MA(1), ARMA(1,1) models with different values of parameters to compare these five bootstrap methods. AR(1) model:

$$X_t = \phi_1 X_{t-1} + \epsilon_t, \tag{4.3}$$

MA(1) model:

$$X_t = \epsilon_t + \theta_1 \epsilon_{t-1}, \tag{4.4}$$

ARMA(1,1) model:

$$X_t = \phi_1 X_{t-1} + \epsilon_t + \theta_1 \epsilon_{t-1}, \tag{4.5}$$

where ϕ_1 is the parameter in AR(1) model, θ_1 is the parameter in MA(1) model, ϵ_t is i.i.d. normally distributed, i.e., ϵ_t $N(0, \sigma^2)$. In Andrews (1991) [2], another kind of time series model was mentioned, which is AR-SEASON model. However,

AR-SEASON is not stationary model because of the unconditionally heteroskedastic errors, which can also be considered as a seasonal weight. The model of AR(1)-SEASON is given as

$$X_t = \phi_1 X_{t-1} + a_t \epsilon_t, \tag{4.6}$$

where $a_t = \{1, 1, 1, 2, 3, 1, 1, 1, 1, 2, 4, 6\}$ mentioned in [22].

With each $\phi_1 = 0, 0.3, 0.5, 0.7, 0.9, 0.95$ and $\theta_1 = 0.3, 0.5, 0.7, 0.99$, we can generate the replicates and calculate the average conficence interval coverage rates. With different values of parameters and the total number of sample is $n = 128$, we can find out the results of each methods, including Andrews estimation of variance, moving block bootstrap (MBB), non-overlapping block bootstrap (NBB), circular block bootstrap (CBB), stationary bootstrap (SB) and AR-Sieve bootstrap method. In Andrews method, we only have one estimation, then by using Monte Carlo method, we generate 10,000 times. As for bootstrap methods, set $B = 500$ replicates in bootstrap method and $R = 2000$ replicates in Monte Carlo method.

It is necessary to mention that block length is important for moving block bootstrap, non-overlapping Bootstrapping and circular bootstrap methods. To determine the

block length of these three block bootstrap methods, we can let $l = n^{1/3}$ mentioned in [8]. More details about block length determination will be discussed in Chapter 5. By calculating $l = n^{1/3}$, the block length of models with the total number of Observation is $n = 128$ is $l = 5$. The number of blocks in NBB is approximately 25. Then we create 25 blocks, as shown in Figure 3.2, and randomly re-sample 25 blocks $\{Y_1^*, \cdots, Y_{25}^*\}$ to create the bootstrap data $\{X_1^*, \cdots, X_{125}^*\}$. In MBB, according to how we determine the blocks shown in Figure 3.1, there should be 124 blocks, and 128 blocks for CBB. However, in stationary bootstrap in which the block length is not fixed, the values of parameter p_S that we select is $0.5, 0.3, 0.1, 0.05$. In AR-Sieve bootstrap, we use AR(1) model to fit the original data since we use AR(1) model to get one of the original series.

In the next section, we will show the simulation results and compare the methods.

4.3 Simulation Results

According to all descriptions of methods mentioned above, the results are shown in the following tables. Each table is about one certain model with five different

methods, even with different parameters in the methods. In the following tables, we use "per" to denote the average confidence interval coverage rate, and "l" to denote the average length of confidence interval.

Table 4.1
Andrews Simulation Results

			ϕ_1					
			0	0.3	0.5	0.7	0.9	0.95
θ_1	0	BT	0.9471	0.9179	0.8965	0.8344	0.5551	0.3547
		QS	0.9447	0.9184	0.9012	0.8585	0.7172	0.5984
	0.3	BT	0.9258	0.8809	0.8586	0.8064	0.6043	0.4418
		QS	0.9309	0.9077	0.8902	0.8398	0.6344	0.4682
	0.5	BT	0.9223	0.8834	0.8628	0.8245	0.6384	0.4885
		QS	0.9289	0.9078	0.8908	0.8537	0.6622	0.5045
	0.7	BT	0.9263	0.8839	0.8638	0.8263	0.6534	0.5110
		QS	0.9329	0.9082	0.8920	0.8551	0.6703	0.5165
	0.99	BT	0.9302	0.8843	0.8639	0.8248	0.6709	0.5228
		QS	0.9233	0.9049	0.8935	0.8540	0.6847	0.5308

In Table 4.1, when $\phi_1 = 0$, θ_1 changes, then it is a MA(1) model, and if $\theta_1 = 0$, then it is a AR(1) model. Other pairs of ϕ_1 and θ_1 values will determine the exact expression in ARMA$(1, 1)$ model.

Table 4.2
AR(1) Simulation Results

		MBB		NBB		CBB		SB								Sieve	
								$p_S = 0.5$		0.3		0.1		0.05			
ϕ_1		No V	V	No V	V	No V	V	No V	V	No V	V	No V	V	No V	V	No V	V
0	per	0.943	0.950	0.942	0.946	0.937	0.946	0.934	0.938	0.9385	0.944	0.907	0.913	0.872	0.882	0.936	0.942
	l(10^{-4})	1.585	1.704	1.986	2.058	1.659	1.721	1.718	1.809	1.598	1.477	1.470	1.520	0.944	0.950	1.886	1.714
0.3	per	0.914	0.940	0.914	0.938	0.908	0.938	0.895	0.941	0.900	0.929	0.897	0.915	0.856	0.875	0.935	0.942
	l(10^{-4})	2.171	2.356	2.265	2.560	2.251	2.425	1.924	2.365	2.122	2.223	2.195	2.364	1.949	2.190	2.501	2.636
0.5	per	0.895	0.940	0.892	0.943	0.893	0.936	0.842	0.925	0.862	0.929	0.881	0.913	0.858	0.889	0.925	0.943
	l(10^{-4})	2.855	3.197	3.181	3.553	2.908	3.409	2.830	3.992	2.564	3092	2.341	2.501	3.688	4.232	3.246	3.479
0.7	per	0.824	0.922	0.826	0.923	0.820	0.920	0.752	0.896	0.816	0.909	0.840	0.901	0.837	0.878	0.912	0.929
	l(10^{-4})	3.895	4.909	4.163	5.269	4.002	5.026	3.278	4.737	3.733	5.013	2.940	3.216	3.022	3.639	4.677	6.286
0.9	per	0.600	0.795	0.595	0.795	0.599	0.787	0.498	0.787	0.610	0.824	0.716	0.849	0.720	0.831	0.823	0.892
	l(10^{-4})	7.972	11.42	7.915	12.07	8.015	11.53	4.789	8.591	7.633	13.01	9.240	12.89	9.193	13.23	12.19	12.15
0.95	per	0.451	0.708	0.453	0.705	0.454	0.698	0.347	0.687	0.433	0.707	0.596	0.792	0.620	0.780	0.723	0.818
	l(10^{-4})	7.279	11.43	7.358	11.84	7.260	11.69	6.587	14.77	8.601	16.26	12.06	21.10	19.25	35.47	15.72	26.36

Table 4.3
MA(1) Simulation Results

		MBB		NBB		CBB		SB								Sieve	
								$p_S = 0.5$		0.3		0.1		0.05			
θ_1		No V	V	No V	V	No V	V	No V	V	No V	V	No V	V	No V	V	No V	V
0.3	per	0.932	0.951	0.931	0.947	0.930	0.949	0.902	0.933	0.918	0.939	0.899	0.924	0.867	0.881	0.948	0.949
	l(10^{-4})	2.017	2.305	2.391	2.535	2.021	2.373	2.107	2.358	1.917	1.919	1.795	1.426	1.223	1.863	2.627	2.656
0.5	per	0.922	0.944	0.920	0.947	0.918	0.946	0.907	0.938	0.908	0.939	0.903	0.925	0.858	0.883	0.965	0.961
	l(10^{-4})	2.314	2.524	2.416	2.897	2.403	2.350	2.140	2.655	2.235	2.115	2.355	1.639	2.053	2.178	2.996	3.066
0.7	per	0.927	0.949	0.934	0.948	0.924	0.950	0.902	0.938	0.912	0.940	0.900	0.926	0.873	0.884	0.966	0.959
	l(10^{-4})	2.615	2.861	2.930	3.231	2.736	2.850	2.654	2.967	2.418	2.370	2.215	1.878	3.247	2.479	3.403	3.356
0.99	per	0.931	0.955	0.931	0.949	0.928	0.951	0.895	0.935	0.911	0.941	0.891	0.927	0.878	0.885	0.973	0.968
	l(10^{-4})	3.131	3.383	3.244	3.841	3.101	3.282	2.852	3.530	2.957	2.681	2.220	2.226	2.077	2.925	3.725	3.661

Table 4.4
AR(1)-SEASON Simulation Results

		MBB		NBB		CBB		SB								Sieve	
								$p_S = 0.5$		0.3		0.1		0.05			
ϕ_1		No V	V	No V	V	No V	V	No V	V	No V	V	No V	V	No V	V	No V	V
		No V	V	No V	V	No V	V	No V	V	No V	V	No V	V	No V	V	No V	V
0	per	0.939	0.948	0.934	0.950	0.938	0.947	0.933	0.943	0.933	0.942	0.910	0.922	0.871	0.884	0.939	0.948
	l(10^{-4})	5.216	5.246	5.678	6.281	5.118	5.420	4.186	4.413	4.813	4.674	2.794	2.867	2.376	2.390	4.658	4.534
0.3	per	0.913	0.942	0.913	0.944	0.910	0.940	0.902	0.944	0.892	0.924	0.897	0.919	0.856	0.883	0.934	0.942
	l(10^{-4})	5.405	5.932	5.839	6.773	5.817	6.080	4.994	6.704	5.519	6.367	6.796	7.136	3.553	3.799	6.590	7.275
0.5	per	0.892	0.943	0.890	0.944	0.888	0.937	0.841	0.925	0.866	0.930	0.875	0.913	0.857	0.888	0.928	0.947
	l(10^{-4})	8.060	10.43	8.381	10.41	8.057	10.67	6.240	8.429	8.057	10.33	10.72	12.15	7.488	8.478	8.881	12.85
0.7	per	0.840	0.916	0.840	0.920	0.831	0.918	0.736	0.896	0.791	0.905	0.840	0.910	0.826	0.877	0.909	0.936
	l(10^{-4})	8.075	9.831	8.638	10.56	8.136	10.10	9.513	14.23	7.864	10.13	9.304	11.38	8.226	9.781	13.55	20.20
0.9	per	0.591	0.786	0.593	0.790	0.590	0.782	0.479	0.791	0.598	0.822	0.714	0.859	0.711	0.836	0.822	0.902
	l(10^{-4})	24.24	35.97	24.16	37.58	23.41	37.23	13.68	26.21	20.76	35.84	30.66	49.18	27.67	50.73	32.36	42.27
0.95	per	0.434	0.670	0.435	0.673	0.435	0.664	0.359	0.687	0.433	0.704	0.569	0.770	0.617	0.804	0.723	0.857
	l(10^{-4})	15.42	21.98	14.90	21.22	15.55	23.80	18.17	42.15	16.44	30.69	26.75	36.96	41.24	73.53	56.17	60.36

Table 4.5
ARMA(1, 1) Simulation Results

			MBB		NBB		CBB		SB								Sieve	
									$p_S = 0.5$		0.3		0.1		0.05			
ϕ_1	θ_1		No V	V	No V	V	No V	V	No V	V	No V	V	No V	V	No V	V	No V	V
0.3	0.3	per	0.905	0.939	0.914	0.949	0.901	0.936	0.857	0.934	0.890	0.935	0.887	0.913	0.861	0.884	0.957	0.960
		l(10^{-4})	2.728	3.153	3.129	3.690	2.785	3.246	2.678	3.356	2.372	2.528	2.444	2.766	1.649	1.760	3.708	3.798
	0.5	per	0.906	0.939	0.909	0.938	0.902	0.938	0.865	0.933	0.881	0.936	0.892	0.912	0.853	0.885	0.970	0.963
		l(10^{-4})	3.101	3.682	3.292	4.187	2.952	3.926	2.785	3.841	2.955	2.849	3.247	3.182	2.903	2.010	4.688	4.458
	0.7	per	0.906	0.950	0.908	0.942	0.902	0.939	0.862	0.933	0.886	0.936	0.888	0.912	0.866	0.885	0.975	0.965
		l(10^{-4})	4.251	4.886	3.250	3.984	3.930	4.395	3.457	4.457	3.311	3.174	2.926	3.590	4.502	2.302	5.658	5.105
	0.99	per	0.916	0.950	0.905	0.952	0.906	0.939	0.862	0.935	0.886	0.936	0.881	0.911	0.868	0.885	0.976	0.966
		l(10^{-4})	3.831	4.369	4.338	4.720	3.392	5.204	3.631	5.232	3.895	3.675	2.962	4.209	2.928	2.717	6.896	6.207
0.5	0.3	per	0.889	0.944	0.883	0.931	0.888	0.934	0.812	0.926	0.860	0.933	0.877	0.911	0.843	0.890	0.951	0.955
		l(10^{-4})	3.574	4.274	3.607	4.646	3.674	5.087	3.052	4.766	3.771	3.433	3.486	3.884	2.733	2.420	5.079	4.647
	0.5	per	0.885	0.941	0.886	0.936	0.881	0.934	0.822	0.926	0.850	0.931	0.868	0.909	0.848	0.887	0.960	0.961
		l(10^{-4})	4.362	5.813	4.498	5.370	4.630	6.202	3.310	5.356	3.537	4.022	4.938	4.546	4.767	2.795	6.094	5.485
	0.7	per	0.880	0.937	0.895	0.945	0.877	0.938	0.824	0.925	0.859	0.929	0.869	0.910	0.847	0.888	0.966	0.965
		l(10^{-4})	4.984	4.965	4.407	6.314	4.414	7.298	5.144	5.995	4.527	4.446	5.021	5.124	2.759	3.147	7.151	6.310
	0.99	per	0.880	0.931	0.872	0.933	0.875	0.936	0.826	0.924	0.834	0.932	0.865	0.910	0.849	0.887	0.969	0.963
		l(10^{-4})	4.933	5.886	5.129	8.740	4.569	9.024	4.525	6.956	4.457	5.209	8.352	5.881	6.967	3.686	8.460	7.349
0.7	0.3	per	0.819	0.927	0.825	0.922	0.813	0.915	0.729	0.900	0.808	0.916	0.853	0.911	0.830	0.888	0.946	0.967
		l(10^{-4})	4.888	6.454	5.667	9.728	5.083	7.203	4.101	6.857	4.366	5.474	3.993	6.493	4.424	3.593	6.818	8.291
	0.5	per	0.821	0.913	0.815	0.918	0.816	0.919	0.726	0.897	0.804	0.915	0.837	0.910	0.810	0.890	0.958	0.974
		l(10^{-4})	6.340	8.435	5.952	7.870	5.733	8.031	4.206	7.898	5.756	6.393	4.316	7.522	8.508	4.109	8.150	10.67
	0.7	per	0.813	0.928	0.809	0.921	0.810	0.927	0.727	0.896	0.791	0.916	0.846	0.909	0.828	0.889	0.964	0.977
		l(10^{-4})	7.454	10.27	6.620	9.097	7.498	8.929	5.405	8.970	6.490	7.102	7.136	8.522	7.154	4.699	9.565	13.85
	0.99	per	0.811	0.925	0.812	0.918	0.809	0.925	0.726	0.901	0.793	0.917	0.831	0.907	0.822	0.888	0.967	0.978
		l(10^{-4})	6.737	7.659	9.333	1.373	6.797	10.53	4.879	10.90	9.305	8.442	8.969	9.932	4.165	5.588	11.43	15.96

Table table – *Continued from previous page*

0.9	0.3	per	0.596	0.810	0.615	0.763	0.594	0.741	0.497	0.804	0.599	0.831	0.715	0.859	0.731	0.854	0.873	0.974
		l(10^{-4})	10.35	15.29	8.943	11.55	10.17	15.61	5.841	13.86	9.483	14.38	11.56	19.21	12.40	10.69	17.75	32.10
	0.5	per	0.598	0.825	0.595	0.774	0.588	0.753	0.492	0.803	0.598	0.831	0.708	0.858	0.717	0.854	0.888	0.980
		l(10^{-4})	10.42	20.59	9.991	16.30	10.16	17.73	7.717	18.09	9.847	16.41	16.61	21.96	8.863	12.16	21.84	43.25
	0.7	per	0.594	0.819	0.596	0.799	0.589	0.769	0.491	0.801	0.591	0.832	0.728	0.859	0.722	0.854	0.892	0.983
		l(10^{-4})	14.24	26.74	17.53	27.44	14.04	20.35	11.11	20.66	10.26	18.35	15.40	24.69	21.17	13.39	25.34	49.39
	0.99	per	0.586	0.831	0.595	0.779	0.579	0.775	0.484	0.801	0.600	0.838	0.720	0.857	0.729	0.855	0.897	0.975
		l(10^{-4})	10.90	18.96	16.82	29.86	10.92	23.56	9.065	24.83	16.05	21.01	21.04	27.48	26.48	15.58	29.98	61.67
0.95	0.3	per	0.448	0.728	0.436	0.583	0.449	0.585	0.329	0.706	0.430	0.740	0.567	0.789	0.606	0.804	0.776	0.967
		l(10^{-4})	9.171	14.97	12.60	21.65	9.444	12.53	16.15	25.65	13.90	27.74	9.394	26.25	30.05	19.63	23.37	38.34
	0.5	per	0.456	0.734	0.451	0.615	0.454	0.612	0.333	0.708	0.453	0.741	0.572	0.789	0.603	0.804	0.788	0.975
		l(10^{-4})	18.84	41.83	14.90	28.36	18.16	15.23	7.213	29.81	15.11	31.59	10.89	29.79	34.98	21.35	27.70	50.33
	0.7	per	0.434	0.722	0.441	0.616	0.429	0.626	0.348	0.704	0.421	0.740	0.560	0.788	0.586	0.804	0.794	0.980
		l(10^{-4})	23.78	52.83	11.82	19.98	24.43	18.43	14.40	33.70	21.24	35.30	13.59	33.05	13.21	23.21	31.64	63.58
	0.99	per	0.450	0.723	0.454	0.661	0.444	0.631	0.344	0.707	0.423	0.738	0.569	0.788	0.607	0.804	0.797	0.977
		l(10^{-4})	12.18	19.72	20.37	18.32	11.39	21.56	13.63	43.15	16.87	40.35	30.69	37.30	45.62	27.51	36.96	77.31

Table 4.1 - 4.5 present the coverage rates and average length of confidence interval. In Table 4.1, we use AR(1), MA(1) and ARMA(1,1) models separately to compare the coverage rate and average length of confidence interval of two different kernels mentioned by Andrews [2]. In Table 4.2, Table 4.3 and Table 4.5, we use five bootstrap methods separately and consider about using Andrews estimation of variance during the simulation of each bootstrap method to get the coverage rates and average length.

Except for the only non-stationary model considered in this paper, AR(1)-SEASON model, simulations results show that the bootstrap methods with Andrews estimation of variance is more accurate than other methods in general.

Take a look at each table separately. In Table 4.1, coverage rates are more accurate if Andrews estimation of variance is used with quadratic spectral kernel than with Bartlett kernel, closer to 0.95 in general, especially with large ϕ_1's. The Andrews estimation does not change the result a lot if the model is MA(1). As for Andrews estimation for ARMA$(1,1)$, with ϕ_1 increasing, quadratic spectral kernel shows more accurate coverage rates than Bartlett kernel, no matter what ϕ_1 is. Under the same ϕ_1, the results are similar with different θ_1 in ARMA$(1,1)$ model.

The results from AR(1) model with MBB, NBB and CBB are quite similar because of the similar determinations of blocks based on the original data. In each fix-length

block bootstrap method, we also generate the results with Andrews estimation of variance. With the help of Andrews estimation, coverage rates are closer to 0.95 than simulation results without the estimation of variance. It seems that MBB method is quite more accurate than other two methods. With stationary bootstrap, different value of parameter p_S give different coverage rate. According to Table 4.2, when $p_S = 0.1$, the simulation results are more accurate when ϕ_1 becomes larger. However, the proper value of parameter p_S has to be determined by ϕ_1 in AR(1) model. As for AR-Sieve method, since we determine the fitted model by setting an AR(1) model, then AR-Sieve bootstrap method is suitable for original data based on AR(1) model, and the coverage rates are much closer to 0.95.

The simulation results do not seem to have much differences for MA(1) model. The reason why simulation results are quite close and similar is that MA(1) model does not rely on time too much. According to the definition of MA(q) model, the relation between residuals are essential, however, the relation between two variables are not that strong. Therefore, no matter which method we use, except for AR-Sieve method, the simulations results are quite acceptable, especially for MBB, NBB and CBB. Stationary bootstrap also shows similar results, but with different parameter p_S to determine the block length for each block in stationary bootstrap, results are not exactly the same. Parameter p_S influences the results and it may be different

with different lag q in MA(q) model. AR-Sieve bootstrap should work for MA model, or ARMA models asymptotically as well as AR models, the reason why the simulation results do not meet our expectation may be that coverage rates are not as accurate in finite samples.

ARMA$(1,1)$ model is a little bit complex. As ϕ_1 being different, the changes of stationary bootstrap with parameters p_S being larger are different. Further discussion with optimal block length under different ϕ_1's will be shown in Chapter 5. MBB method also shows more accurate coverage rates for ARMA(1,1) model. As ϕ_1 and θ_1 become large, the methods which can get more accurate coverage rates are MBB, stationary bootstrap with proper p_S and AR-Sieve bootstrap. MBB, NBB and CBB share similar simulation results and the larger ϕ_1 becomes, the less accurate the coverage rate. This is similar with the previous inferences. Also similar with what is mentioned above, different parameter p_S in stationary bootstrap method have different results. It is hard to conclude which parameter p_S is better during calculating the coverage rate and the average length of confidence interval of the original data. However, it may help if we calculate the optimal block length for ARMA(1,1) model. AR-Sieve method for ARMA(1,1) is abnormal since the simulation coverage rate is much more than the expected value. One reason why this circumstance happens may be that ARMA(1,1) model can not be fitted as

AR(p) model. And because of a fault fitted model, the results are not reasonable to be accepted. The simulation results are quite similar to the results of AR(1) model when ϕ_1 becomes 0.95. The possible reason may be that the parameter θ_1 of MA(1) in ARMA(1,1) does not influence a lot on AR(1) in ARMA(1,1), then when we use AR-Sieve method, which means we use an AR(1) model to fit the original data, it is quite accurate.

As for AR(1)-SEASON model, the results are close to the simulation results under AR(1) model. Similar with the results of AR(1) model, MBB and AR-Sieve methods get more accurate coverage rates of original data.

Through all tables shown above, compare Table 4.1 by using quadratic spectral kernel with Table 4.2 and Table 4.5, Andrews method shows similar results with fix-length block bootstrap methods without Andrews estimation of variance. For these two methods, the method with more accurate coverage rage should be AR-Sieve bootstrap. For different ϕ_1's, the results show similar inference. The value of θ_1 does not influence a lot on the results under the same ϕ_1 in ARMA(1,1) model. The second accurate methods according to tables shown above is MBB and stationary bootstrap with certain parameter p_S. NBB and CBB are similar to MBB so that we do not need to consider about these two methods too much.

As for the comparison of average length of confidence interval, the simulations with Andrews method have larger average length in general. Compare the fixed length block bootstrap, which are MBB, NBB and CBB, average length of confidence interval will be smaller in MBB. And with different value of parameter p_S in stationary bootstrap, by choosing p_S properly, average length of confidence interval can be decreased.

In AR(1) and AR(1)-SEASON model, it seems that MBB presents the smallest average length among all three fixed-length bootstrap methods. And consider the coverage rate at the same time, MBB might be the best choice among MBB, NBB and CBB. By using stationary bootstrap, different p_S leads to different results. The smallest average length will be when $p_S = 0.5$. However, if we take the coverage rate into consideration as well, then when $p_S = 0.5$, the results indicate that this is not a better choice. Although when $p_S = 0.01$, the average length is not the smallest, however, with Andrews estimation method, the coverage rate is the largest in stationary bootstrap. Then we can consider that when $p_S = 0.01$, stationary bootstrap might be also accurate for AR(1) model. It is not hard to determine the exact p_S value which should be chosen for stationary bootstrap, however, we would not talk about the method to choose p_S in our report.

In MA(1) model, the inference of average length of confidence interval is similar to

the inference of coverage rate because of weak time-dependence of MA(1) model. However, average lengths are also different for stationary bootstrap when p_S is not the same.

In ARMA(1,1) model, results might be complicated. Comparing MBB, NBB and CBB, when ϕ_1 becomes larger, it seems that MBB with Andrews method gives a better results consider the average length and the coverage rate as well. Consider about stationary bootstrap, when ϕ_1 and θ_1 become large, then when $p_S = 0.05$, the simulation results give more accurate coverage rates and with wider average length. However, whether stationary bootstrap is accurate or not, dependes on the parameter value we choose. With a proper value, stationary bootstrap method might be more accurate than MBB. As for AR-Sieve method, as we mentioned above, the coverage rates of ARMA(1,1) may be closer to AR(1) model, however, the average length of confidence interval is going to be very large, even more than twice of the average length of AR(1) model when ϕ_1 becomes 0.95. It is not what we expected.

In conclusion, bootstrap methods show the advantages of dealing with time series, especially MBB based on different models, for coverage confidence interval rate, average length of confidence interval, model suitability. As what we mentioned above, optimal block length can be taken into consideration when simulating

different bootstrap methods. It might be more accurate if we choose more proper optimal block lengths for block bootstraps. We will discuss the theoretical optimal block length of CBB and stationary bootstrap under AR(1) model in Chapter 5.

Chapter 5

Further Discussion

5.1 Determination of Block Length for Block Bootstraps

Even though ways to define how blocks would be in NBB, MBB, or CBB are different, all of them are based on the selection of the block length. The block length we take to help to define the blocks in Chapter 4 is $l = n^{1/3}$, which is based on the common use of $l = c \cdot n^{1/3}$ where c is a constant. However, it is not convincing that $c = 1$, that we chose to define the block length, is a good choice

to decide the block length under different circumstances, e.g. different sample size n, different ϕ_1's. Theorem 3.1, 3.2 and 3.3 in [23] showed how block length should be determined and estimations of expected block lengths with different bootstraps. Since NBB, MBB and CBB are quite similar, then it is enough to show what the theoretical block length is with CBB in different cases. As for stationary bootstrap, since the block lengths of stationary bootstrap are i.i.d random variables having geometric distribution with parameter p, then we can calculate the theoretical expected block length for stationary bootstrap as a comparison.

According to Lahiri's theorem in [14] and [18], detailed approximations to the first two moments of $\hat{\sigma}^2_{SB}$ and $\hat{\sigma}^2_C B$ are provided in the theorem. Block length can be calculated with further steps.

Theorem 5.1.1 *Assume that both* $E|X_t|^{6+\delta}$ *and* $\sum_{h=1}^{\infty} h^2(\alpha(h))^{\frac{\delta}{6+delta}}$ *are finite for*

some positive δ. If $l \to \infty$ as $n \to \infty$, with $l = o(n^{1/2})$. Define

$$g(w) = \sum_{s=-\infty}^{\infty} \gamma(s)\cos(ws),\ G = \sum_{h=-\infty}^{\infty} |h|\gamma(h),$$
$$D_{SB} = 2g^2(0) + \frac{2}{\pi}\int_{-\pi}^{\pi} G^2(w)dw,$$
$$D_{CB} = \frac{4}{3}g^2(0).$$

Then

$$\begin{aligned} bias(\hat{\sigma}^2_{SB}) &= -\frac{1}{l}G + o(1/l),\ Var(\hat{\sigma}^2_{SB}) = \frac{l}{n}D_{SB} + o(l/n), \\ bias(\hat{\sigma}^2_{CB}) &= -\frac{1}{l}G + o(1/l),\ Var(\hat{\sigma}^2_{SB}) = \frac{l}{n}D_{CB} + o(l/n). \end{aligned} \tag{5.1}$$

In Theorem 5.1.1, the bias and the variance of the variance of two bootstraps depend on the block length l and the original data. Therefore, if we would like to determine the block length, it is a way to consider minimize MSE to get the theoretical optimal block length, see [23]. In [23] provided by Politis and White, estimations of theoretical block length are shown for the stationary bootstrap and the circular bootstrap in general cases. To show how to get the optimal theoretical block length, we consider the Mean Squared Error (MSE), according to Politis and White [23].

The definition of MSE is given as

$$MSE(\theta) = bias^2(\theta) + Var(\theta).$$

Since we already get bias and variance of σ^2 with different bootstrap methods, it is straightforward to find MSE of the variance.

$$MSE(\hat{\sigma}^2) = \frac{G^2}{l^2} + \frac{Dl}{n} + o(1/l^2) + o(l/n).$$

To minimize the large samle MSE, we could choose

$$l = \left(\frac{2G^2}{D}\right)^{1/3} n^{1/3}.$$

Let $c = \left(\frac{2G^2}{D}\right)^{1/3}$, then

$$l = cn^{1/3},$$

as what we did above.

Therefore, we can calculate the exact value of the constant to get optimal block length.

5.2 Optimal Expected Block Length

With corresponding formulas in different bootstrap methods, we can calculate the optimal expected block length for circular block bootstrap and stationary bootstrap methods separately. Table 5.1 shows the result of optimal expected block length of stationary bootstrap and circular bootstrap as well.

According to Table 5.1, different block length can be generated with different ϕ_1's. From the optimal expected block length of stationary bootstrap, we can also calculate the parameter p in the geometric distribution. The expectation of geometric distribution with parameter p is given as $E(l) = \frac{1}{p}$, then from each expectation of block length for stationary bootstrap, p can be calculated correspondingly.

Table 5.1
Optimal Expected Block Length

ϕ_1	0	0.3	0.5	0.7	0.9	0.95
SB	1.044713	1.390087	2.089164	2.615999	2.700121	2.747743
p	0.9572	0.7194	0.4787	0.3823	0.3704	0.3639
CBB	1.870101	2.469882	3.651564	4.515018	4.650964	4.727709

With the results shown in Table 5.1, we can do simulations again with circular block bootstrap and stationary bootstrap methods. Here we only take two models as examples, AR(1) with the same ϕ_1's and ARMA(1,1) with the same values of parameters (ϕ_1, θ). The simulation results are shown in Table 5.2.

Table 5.2
AR(1) Simulation Results with Optimal Block Length

ϕ_1		CBB		SB	
		No V	V	No V	V
0	per	0.939	0.920	0.949	0.950
	l(10^{-4})	1.668	1.634	1.712	1.735
0.3	per	0.898	0.938	0.861	0.930
	l(10^{-4})	2.242	2.869	1.916	2.312
0.5	per	0.888	0.926	0.834	0.916
	l(10^{-4})	2.554	2.964	2.818	3.883
0.7	per	0.818	0.921	0.781	0.910
	l(10^{-4})	4.453	5.782	3.622	5.301
0.9	per	0.594	0.892	0.577	0.827
	l(10^{-4})	7.668	12.00	6.061	10.09
0.95	per	0.446	0.804	0.423	0.706
	l(10^{-4})	11.96	22.36	6.968	11.95

The simulation results are quite similar with the results under a certain block length given as $l = 5$ without Andrews estimation of variance. However, with Andrews

estimation, the results are much more accurate than without Andrews estimation, and they are much more accurate than with a certain block length $l = 5$. The simulation results may be different for large lag l in AR(l) model, and consider the estimation of the optimal block length given by Politis and White [23]. Since the model is quite simple, then it may not be obvious enough for the improvement of the theoretical optimal block length. Simulations with other models can be generated in other work.

Chapter 6

Summary

Time series is closely related to in our daily life. As a special case of dependent data, many methods have been developed to deal with time series original data from methods which can help to solve problems of independent data.

Basic time series models indicate the basic ideas of how time dependence works. Moving average model, MA(q), indicates the relation between several residuals. It is a stationary model in time series, the time dependence in MA model is relatively weak because of the relation between residuals. Autoregressive model, AR(p), states the relation between variables, according to its definition. As time goes by, parameter ρ shows the dependence of the latter variable X_t on the previous variable X_{t-1}

in AR(1) model. With large p, X_t even depends on several previous variables. This model is quite useful and helpful to describe a series which have time dependence. With the development of time series model, people consider about these two models together, and it is called autoregressive moving average model, ARMA(p, q), lag p indicates the parameter in AR(p) and lag q represents the parameter in MA(q). By combining these two models together, ARMA(p, q) become popular in time series research. Researchers already showed the stationarity coditions of these three time series models.

Andrews (1991) [2] came up with one way to estimate the variance of dependent data and the simulation results also shows that the method is quite helpful with some kernels. After that, bootstrap methods [11] for time series have been developed from bootstraps based on independent data. NBB, MBB, CBB and stationary bootstrap methods are based on block selection. With different type of selecting blocks, different block bootstrap methods have different results, and in general, stationary bootstrap method might be more accurate depending on the parameter p_S selection. MBB is also quite accurate for models mentioned in this report. Another kind of bootstrap method for time series is AR-Sieve bootstrap method. It shows strengths to estimate the parameter for AR models by construct a fitted model and resampling based on the fitted model. These methods are quite accurate for MA(1)

because of the lack time dependence of MA(1) model. Take a general look of all simulation results, we also considered bootstrap methods with Andrews estimation of variance at the same time, and it shows more accurate estimations and the results we get are closer to 0.95.

According to the selection of block length in block bootstrap methods, results may be better if we choose an optimal block length for each block bootstrap. Since the selection of blocks for MBB, NBB and CBB are quite similar, then they can share the same way to determine the block length. By using MSE, optimal block length can be calculated for CBB and stationary bootstrap.

In general, bootstrap methods with Andrews estimation of variance would be more accurate for models we consider in this report. Other models or other methods will be considered in the future work.

References

[1] Theodore W. Anderson. *The Statistical Analysis of Time Series*. Wiley Series in Probability and Statistics. Wiley, 2011.

[2] Donald Andrews. Heteroskedasticity and autocorrelation consistent covariance matrix estimation. *Econometrica*, 59(3):817–58, 1991.

[3] Patrick Billingsley. *Probability and Measure*. Wiley-Interscience, 3 edition, April 1995.

[4] Peter J. Brockwell and Richard A. Davis. *Time Series: Theory and Methods*. Springer-Verlag New York, Inc., New York, NY, USA, 1986.

[5] Peter Bühlmann. Sieve bootstrap for time series. *Bernoulli*, 3(2):123–148, 06 1997.

[6] Peter Bühlmann. Sieve bootstrap for time series. *Bernoulli*, 3(2):123–148, 06 1997.

[7] Peter Bühlmann. Bootstraps for time series. *Statist. Sci.*, 17(1):52–72, 05 2002.

[8] Peter Bühlmann. Bootstraps for time series. *Statist. Sci.*, 17(1):52–72, 05 2002.

[9] Edward Carlstein. The use of subseries methods for estimating the variance of a general statistic from a stationary time series. *Ann. Statist*, 14:1171–9, 1986.

[10] Jonathan D. Cryer and Kung-Sik Chan. *Time Series Analysis With Applications in R*. Springer, 2008.

[11] Brad Efron. Bootstrap methods: Another look at the jackknife. *Ann. Statist.*, 7(1):1–26, 01 1979.

[12] Brad Efron and Charles M. Stein. The jackknife estimate of variance. *Ann. Statist.*, 9(3):586–596, 05 1981.

[13] Hans R. Kunsch. The jackknife and the bootstrap for general stationary observations. *Ann. Statist.*, 17(3):1217–1241, 09 1989.

[14] Soumendra N. Lahir. Theoretical comparisons of block bootstrap methods. *Ann. Statist.*, 27(1):386–404, 03 1999.

[15] Soumendra N. Lahir. *Resampling Methods for Dependent Data*. Springer, 2003.

[16] Regina Y. Liu and Kesar Singh. Moving blocks jackknife and bootstrap capture. In Raoul Lepage and Lynne Bilard, editors, *Exploring the limits of bootstrap*, pages 225–248. Wiley-Interscience, 1992.

[17] Emanuel Parzen. On choosing an estimate of the spectral density function of a stationary time series. *Ann. Math. Statist.*, 28(4):921–932, 12 1957.

[18] Andrew Patton, Dimitris Politis, and Halbert White. Correction to Automatic Block-Length Selection for the Dependent Bootstrap by D. Politis and H. White. *Econometric Reviews*, 28(4):372–375, 2009.

[19] Dimitris N. Politis. The impact of bootstrap methods on time series analysis. *Statist. Sci.*, 18(2):219–230, 05 2003.

[20] Dimitris N. Politis and Joseph P. Romano. A circular block resampling procedure for stationary data. In Raoul Lepage and Lynne Bilard, editors, *Exploring the limits of bootstrap*, page 263270. Wiley-Interscience, 1992.

[21] Dimitris N. Politis and Joseph P. Romano. The stationary bootstrap. *Journal of the American Statistical Association*, 89(428):1303–1313, 1994.

[22] Dimitris N. Politis, Joseph P. Romano, and Michael Wolf. *Subsampling*. Springer, 1999.

[23] Dimitris N. Politis and Halbert White. Automatic Block-Length Selection for the Dependent Bootstrap. *Econometric Reviews*, 23:53–70, 1997.

[24] Maurice H. Quenouille. Problems in plane sampling. *Ann. Math. Statist.*, 20(3):355–375, 09 1949.

[25] Suhasini S. Rao. A course in time series analysis. Updated: 2015-08-05.