TECHNISCHE UNIVERSITEIT DELFT
FACULTY OF ELECTRICAL ENGINEERING, MATHEMATICS AND COMPUTER SCIENCE
DELFT INSTITUTE OF APPLIED MATHEMATICS

HET BEPALEN VAN DE FOUT BIJ EEN MARKOV KETEN MONTE CARLO SIMULATIE

BACHELOR EINDPROJECT TECHNISCHE WISKUNDE

Determining the accuracy of a Markov Chain Monte Carlo output

Author
Femke SCHÜRMANN

Supervisor
dr. ir. J. BIERKENS

Other comittee members
Dr. J.-J. CAI
Dr. J.G. SPANDAW

June 26, 2019

TU Delft

Abstract

This is a lively thesis providing different methods for analyzing Markov Chain Monte Carlo output. Written as a series of short chapters and by including a lot of background, students with no prior knowledge in Markov Chain Monte Carlo will be able to read this thesis as well. In the first part, more general statistical definitions and methods such as regression, Bayesian estimation, time series and Monte Carlo simulation are discussed. After a short chapter which includes more information about the use of simulation as a solution for statistical problems, more detailed derivations and proofs are provided for methods which estimate or describe asymptotic variance. The methods for estimating asymptotic variance which are included are: estimation by window estimators, the batch means method and the blocked means method. Furthermore by using historical data, the methods are brought to life in order to get some more feeling about the subjects.

Contents

1 Introduction **4**
- 1.1 Used data . . . 5
- 1.2 Codes from Rstudio . . . 5

2 Frequentist and Bayesian approach to regression analysis **6**
- 2.1 Linear regression model . . . 6
  - 2.1.1 Example of linear regression on the wind speed data . . . 6
- 2.2 Penalised Regression Methods . . . 7
  - 2.2.1 Ridge Regression . . . 7
  - 2.2.2 The Lasso . . . 8
- 2.3 Bayesian methods . . . 9
  - 2.3.1 Bayesian interpretation of Ridge regression and the Lasso . . . 9

3 Integration using simulation techniques **10**
- 3.1 Classical Monte Carlo Integration . . . 10
  - 3.1.1 Example of Monte Carlo Integration: Normal cdf . . . 10
- 3.2 Markov Chain Monte Carlo . . . 12
  - 3.2.1 Gibbs Sampler . . . 12

4 Central limit theorem **15**
- 4.1 CLT for independent and dependent random variables . . . 15
- 4.2 Linear process . . . 18
  - 4.2.1 Example with an AR(1) Process . . . 18
- 4.3 Relevance of the asymptotic variance . . . 19

5 Non parametric estimation **20**
- 5.1 Motivation . . . 20
- 5.2 The batch means method . . . 20
  - 5.2.1 Optimal Batch size for an AR(1) process . . . 22
- 5.3 The blocked means method . . . 25
- 5.4 Estimating the asymptotic variance using window estimators . . . 25
  - 5.4.1 Setting up a confidence interval for the wind speed data using the batch means method . . . 26
- 5.5 Comparison between the different methods . . . 28
  - 5.5.1 General Comparison between the methods . . . 33

6 Conclusion **34**
- 6.1 Further Research . . . 34

7 Appendix **35**
- 7.1 Appendix of section 5.2.1 . . . 35
- 7.2 Appendix of section 5.5 . . . 36

1 Introduction

From a practical perspective, probability distributions can be very complex. In different disciplines (e.g. statistics, physics, optimization and machine learning), such probability distributions are used. Instead of analyzing classic (non-Bayesian) statistical methods, we will focus on Bayesian estimation to deal with this complexity. A method which is very useful in Bayesian statistics is called Markov Chain Monte Carlo (MCMC). This method approximates the probability distribution π which is defined on $\mathbb{R}^d$ using a set of values $x^{(1)}, ..., x^{(n)}$ representing draws from π. With these draws the empirical distribution $\hat{\pi}_n = \frac{1}{n}\sum_{i=1}^{n} \delta_{x^{(i)}}$ can be built to approximate the theoretical distribution π (here = δ_z is the the Dirac delta function in z). The law of large numbers states then that with probability 1 the empirical expectation will converge for a large class of functions $f : \mathbb{R}^d \to \mathbb{R}$ (e.g. $\mathbb{E}_{\hat{\pi}_n}[f(X)] = \frac{1}{n}\sum_{i=1}^{n} f(x^{(i)}) \to \mathbb{E}_{\pi}[f(X)]$ for $n \to \infty$) . We want to determine whether the algorithm is accurate by investigating the quality of the output. Therefore, the problem is stated as:

For a function $f : \mathbb{R}^d \to \mathbb{R}$, *how can we determine the approximation error* $e(x^{(1)}, ..., x^{(n)}) := |\mathbb{E}_{\hat{\pi}_n}[f(X)] - \mathbb{E}_{\pi}[f(X)]|$, *without knowing* $\mathbb{E}_{\pi}[f(X)]$*?*

In the beginning of this thesis, more background of the frequentest approach of regression analysis will be given. Also the Bayesian approach is explained such that the concept of Bayesian statistics becomes clearer. After this, we will dive further into Monte Carlo and Markov Chain Monte Carlo which are very useful tools for Bayesian Statistics. From that point on, the focus is on determining the approximation error of such methods. In order to do this, the central limit theorem is brought in to light. Because this theorem assumes an output of independent variables, it needs to be extended for dependent variables. When doing this, a parameter called *asymptotic variance* has to be calculated. It appears that determining this parameter theoretically is hard and not always possible. Therefore, different methods for estimating asymptotic variance are discussed. A method which can be used for this is *the batch means method* and is for example described by Alexopoulos, Fisherman and Seila (1997) [1]. Such methods are described and compared in chapter 5. Finally, the problem which is stated above will be discussed using the comparison of the different methods for estimating asymptotic variance.

In order to get some more feeling about the subject, historical data from the KNMI (Koninklijk Nederlands Meteorologisch Instituut) is used for some examples. For example, different regression methods are applied on this data. Also a Bayesian approach is introduced, such that the error of this method can be determined using the batch means method.

1.1 Used data

The data [1] which is used, can be downloaded from the KNMI (Koninklijk Nederlands Meteorologisch Instituut) database. The data set contains daily wind speed averages in 0.1 m/s from the KNMI station in Rotterdam. The speeds of days 1-6 are compared with the average wind speed on the seventh day. We use the measurements between the period from 01-05-2016 until 01-05-2019. In figure 1.1.1, a graphical presentation of the data is given. This shows the relation between the wind speeds of every day (1-6) separately and the average wind speed of the 7th. Day 7 is called 'today' and the number of days before are specified in the titles of each plot.

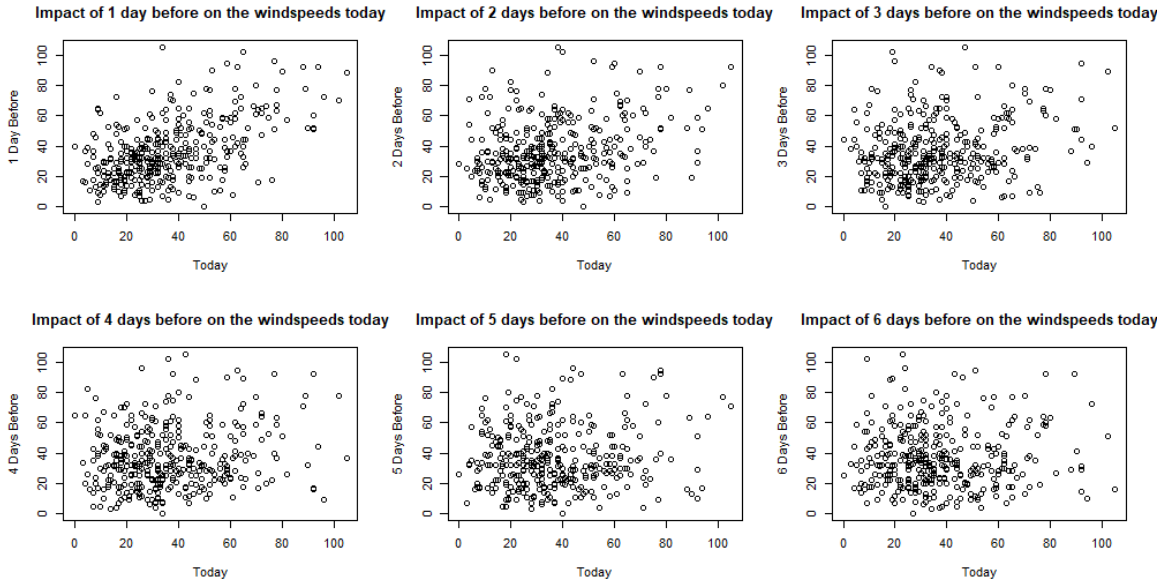

Figure 1.1.1: Plots showing the relation between the wind speeds Rotterdam each day

1.2 Codes from Rstudio

The program 'R' is used in this thesis for implementing the discussed methods. The codes from this can be found on `github.com/FemkeSchurmann/BEP`.

[1] `http://projects.knmi.nl/klimatologie/daggegevens/selectie.cgi`

2 Frequentist and Bayesian approach to regression analysis

In classical (Non-Bayesian) statistics, X is treated as a random variable and has a density or probability mass function $f(x|\theta)$. Here, θ is a fixed (unknown) parameter. A very popular model for determining the relation between variables is called linear regression.

2.1 Linear regression model

The linear regression model describes the relation between the response variable Y and p explanatory variables X.

Definition 2.1 *In matrix-vector notation, a **linear regression model** is considered as:*

$$Y = X\beta + \epsilon. \tag{2.1.1}$$

Here, β is a real valued vector of length p which contains all parameters, X is a real-valued $n \times p$ matrix containing the explanatory values and ϵ is a random vector following an $N_n(0, \sigma^2 I_n)$ distribution modelling the noise.

The Gauss-Markov theorem provides a least squares estimate for β. We denote the least squares estimate by $\hat{\beta}$. This is a linear and unbiased estimator. Indeed, it can easily be derived that the estimator is unbiased:

$$\hat{\beta} = (X'X)^{-1}X'Y,$$

$$E[\hat{B}] = \mathbb{E}[(X'X)^{-1}X'Y] = (X'X)^{-1}X'\mathbb{E}[X\beta + \epsilon] = (X'X)^{-1}X'X\beta = \beta,$$

$$\begin{aligned}\mathrm{Cov}(\hat{\beta}) &= \mathbb{E}[(\hat{\beta} - \beta)(\hat{\beta} - \beta)'] = \mathbb{E}[(X'X)^{-1}X'\epsilon\epsilon'X(X'X)^{-1}] \\ &= (X'X)^{-1}X'\sigma^2 I_n X(X'X)^{-1} = \sigma^2(X'X)^{-1}.\end{aligned}$$

It is proven that this $\hat{\beta}$ is in fact the Best Linear and Unbiased Estimator (BLUE). A linear and unbiased estimator is considered 'best' when among all other linear and unbiased estimators, it has the smallest variance. A version of the proof for the Gauss-Markov theorem is given in J. Fox (2015) [3].

2.1.1 Example of linear regression on the wind speed data

The data which is introduced in subsection 1.1, can be specified by two kinds of variables:
- *1 response variable Y which describes the average wind speed on day 7*
- *6 response variables $X_1, ..., X_6$ which describe data from the past 6 days*

The model for this regression is described as: $Y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + ... + \beta_6 x_6 + \epsilon$ With the 'lm'-function in R, the least squares estimates for the coefficients can be calculated. This output is partly given below in the table:

This example is not a very accurate in describing the relation between the variables because it does not take factors like for example seasonality into account. However, the intention is to give more intuition when practicing linear regression.

Nowadays, there is a lot of data available and it is not uncommon that the number of observations n is (much) smaller than the number of predictors p. Because of this, various problems [7] arise and therefore different models need to be implemented.

Coefficient	Estimate
Intercept (β_0)	16.07
1 day before (β_1)	0.4962
2 days before (β_2)	-0.008210
3 days before (β_3)	0.08427
4 days before (β_4)	0.03557
5 days before (β_5)	-0.04888
6 days before (β_6)	0.0002435

Table 1: Least squares estimates for $\beta_0, ..., \beta_6$

2.2 Penalised Regression Methods

When the amount of predictors p is very large, it can be useful to prefer a penalised regression model. Without loss of generality, assume that $\beta_0 = 0$ such that the model can be written as:

$$Y = X\beta + \epsilon. \tag{2.2.1}$$

Here, ϵ is a random vector with a $N_n(0, \sigma I_n)$ distribution. X is a $n \times p$ matrix where n is the number of samples and p is the number of parameters. The vector $\beta = (\beta_1, ..., \beta_p)$ contains the unknown parameters we want to estimate. In order to calculate the parameters, we want to maximize the likelihood.

Definition 2.2 *Suppose that the sequence of random variables $X_1, .., X_n$ has joint density $f(x_1, x_2, ..., x_n|\theta)$. Given observed values $X_i = x_i$, $i = 1, ..., n$. The **likelihood** of θ as a function of $x_1, ..., x_n$ is defined as:*

$$\mathcal{L}(\theta|X) = \prod_{i=1}^{n} f(x_i|\theta).$$

Firstly, it is necessary to setup the likelihood of model (2.2.1) in order to find the maximized likelihood estimate.

$$\mathcal{L}(\beta|Y) = (2\pi\sigma^2)^{-n/2} \exp(-\frac{||Y - X\beta||^2}{2\sigma^2}).$$

Here, $||\cdot||$ denotes the Euclidean norm. Maximizing this likelihood function is equivalent to minimizing the Least Squares Estimator

$$\hat{\beta} = \text{argmin}_\beta ||Y - X\beta||^2. \tag{2.2.2}$$

This is a general form of the shrinkage and regularization methods for linear models. In the next subsections, two different penalised regression models are described: ridge regression and the lasso. More background can be found in (F. van der Meulen, *Lecture notes mathematical data science* [7]).

2.2.1 Ridge Regression

This type of regression allows some bias while reducing the variance. By setting a penalty, the problem becomes an optimization problem. The ridge regression estimator can be found by minimizing:

$$||Y - X\beta||^2 \text{ subject to } ||\beta||_2^2 \leq t.$$

The restriction is that β lies in a circle with radius t (here the Euclidean norm is used). Solving this optimization problem gives ([7]):

$$\hat{\beta}_r = (X'X + \lambda I_n)^{-1} X'Y. \tag{2.2.3}$$

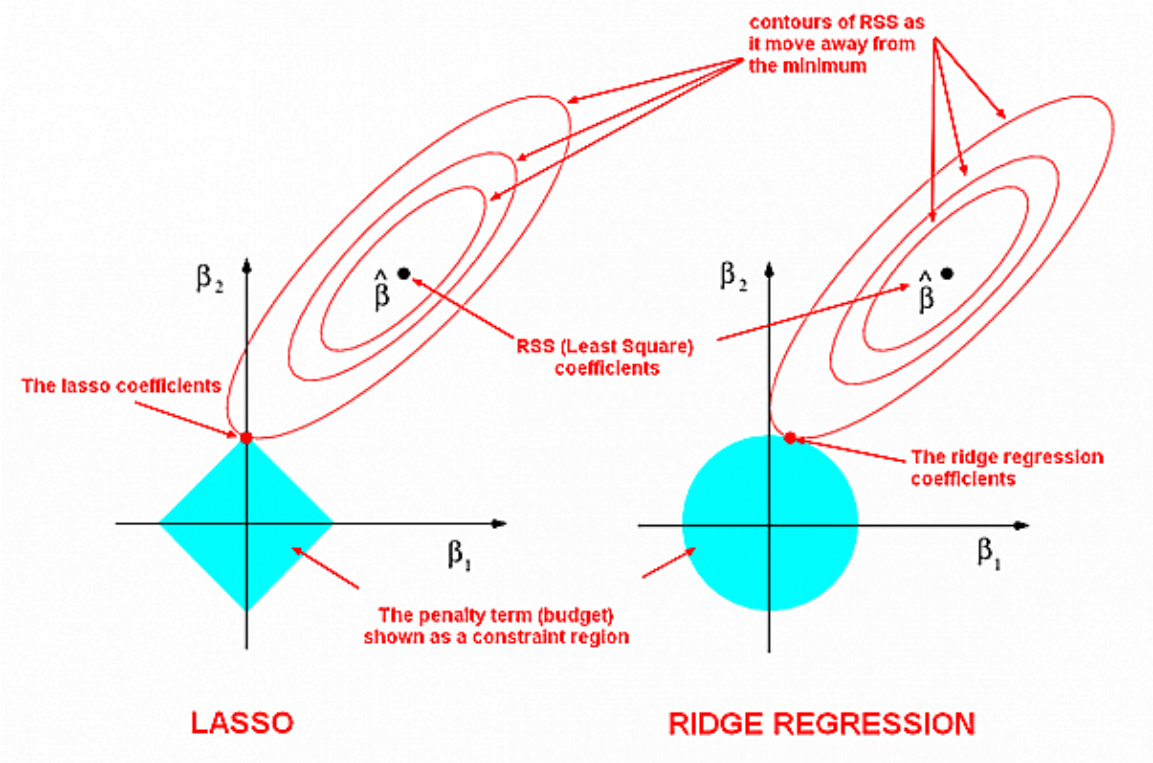

Figure 2.2.1: Graphical view of: right: Ridge regression, left: the Lasso

The parameter $\lambda > 0$ is known as a regularization parameter.

2.2.2 The Lasso

Lasso stands for Least Absolute Shrinkage and Selection Operator. The definition is very similar to ridge regression however the l_2 penalty is substituted for an l_1 penalty. The lasso estimator is given by minimizing:

$$||Y - X\beta||^2 \text{ subject to } ||\beta||_1 \leq t.$$

This is equivalent to minimizing:

$$||Y - X\beta||^2 + \lambda||\beta||_1.$$

Just like in the ridge regression model, the parameter λ is a regularization parameter.

The difference between ridge regression and the lasso becomes clearer when it is explained in a more graphical way (see figure 2.2.1). The penalty term which is used in this figure for the lasso is: $|\beta_1| + |\beta_2| \leq t$ and for the ridge regression: $\beta_1^2 - \beta_2^2 \leq t^2$. In the figure, these are represented as the solid blue areas. The contours are the residual sum of squared as it moves away from the minimum. The coefficients of the regression are the intersections between the red contours and the blue box. From the figure it becomes clear that, for the Lasso, it is more likely that coefficients tend to zero. With ridge regression, this is on the other hand not very likely.

2.3 Bayesian methods

In non-Bayesian statistics (or classical statistics), the focus is on the random variable X, with density function $f(x|\theta)$. Here, θ is a fixed, unknown parameter value. In Bayesian statistics on the other hand, both X and θ are treated as random variables with a joint probability density given by $\pi(\theta)f(x|\theta)$. Here $\pi(\cdot)$ is the prior density of θ and $f(x|\theta)$ is the conditional density of X, with θ known. The goal is to generate a posterior distribution given an updated prior distribution on the parameter θ by using observations of real data. In order to do this, a rule named *Bayes' law* is used.

Definition 2.3 *Let x be an observed value of X. **Bayes' law** states that the posterior distribution is a function which is based on a chosen prior:*

$$\pi(\theta|x) = \frac{\pi(\theta)f(x|\theta)}{\int_\Theta \pi(\theta')f(x|\theta')d\theta'} \propto \pi(\theta)f(x|\theta). \tag{2.3.1}$$

Because $f(x|\theta)$ is treated as a function of θ for fixed x, definition 2.2 can be used and Bayes' law can be written in words as:

$$\text{posterior} \propto \text{prior} \times \text{likelihood}.$$

2.3.1 Bayesian interpretation of Ridge regression and the Lasso

In this example, σ^2 is known, λ is fixed and $\beta \sim N_n(0, \sigma^2\lambda^{-1}I_n)$ is chosen as prior. From definition 2.3, the posterior density of β is:

$$\pi(\beta|D) \propto \exp\left(-\frac{1}{2\sigma^2}(||Y - X\beta||^2 + \lambda||\beta||^2)\right).$$

D is the set which contains all data. By maximising the posterior, the so-called Maximum A Posterior (MAP) estimator can be found and is equal to the ridge regression estimator.

Ridge regression can be interpreted as linear regression for which the coefficients have a normal distribution as a prior. For the lasso, these coefficients have a Laplace distribution prior meaning:

$$\pi(\beta) = \frac{\lambda}{2\sigma}\exp(-\frac{\lambda}{\sigma}||\beta||_1).$$

In this case, the MAP estimator is the lasso estimator.

One of the biggest advantages of Bayesian methods is that, as mentioned in the beginning of this section, they may often be applied in very complex systems where X and θ (or β) are very high dimensional. Especially, computing numerically the normalising constant ($\int_\Theta \pi(\theta')f(x|\theta')d\theta'$) in (2.3.1) such that the posterior density satisfies a proper density function, can be very complex or even impossible.

Instead of integrating numerically, in order to calculate this constant, different simulation techniques can be used. Mainly because in higher (5 or more) dimensions, integrating becomes very impractical and therefore techniques as Monte Carlo integration are a very accessible method for problems as these. More details about (Markov Chain) Monte Carlo are discussed.

3 Integration using simulation techniques

In Bayesian Statistics, simulation techniques are primarily used for numerical approximations of high-dimensional integrals. Especially Markov Chain Monte Carlo makes it possible to compute models which require integration over a very large amount of unknown parameters.

3.1 Classical Monte Carlo Integration

Suppose we want to evaluate the following integral:

$$\mathbb{E}_\pi[h(X)] = \int_\chi h(x)\pi(x)dx. \tag{3.1.1}$$

Let $(X_1, ..., X_n)$ be a sample generated from the probability distribution $\pi(x)$. With this sample the empirical distribution $\hat{\pi}_n(x) = \frac{1}{n}\sum \delta_{X_i}$ can be constructed which approximates the density $\pi(x)$. For n large, the law of large numbers states:

$$\mathbb{E}_{\hat{\pi}_n}[h(X)] = \frac{1}{n}\sum_{i=1}^{n} h(X_i) \to E_\pi[h(X)]. \tag{3.1.2}$$

This method is called *Monte Carlo Simulation*.

3.1.1 Example of Monte Carlo Integration: Normal cdf

Suppose X is a standard normally distributed random variable and we want to know the probability that $0 \leq X \leq 1$. In order to do this, we need to calculate:

$$P(0 \leq X \leq 1) = \int_0^1 \frac{1}{\sqrt{2\pi}} e^{-y^2/2} dy = \int_{-\infty}^{\infty} \frac{1}{\sqrt{2\pi}} e^{-y^2/2} \cdot 1_{[0,1]} dy.$$

When choosing $X \sim U(0,1)$ such that $\pi(X) = 1$ (when $0 \leq X \leq 1$, else $\pi(X) = 0$), the expectation of $h(x) = \frac{1}{\sqrt{2\pi}} e^{-x^2/2}$ is equal to:

$$\mathbb{E}_\pi[h(X)] = \int h(x)\pi(x)dx \to \int_0^1 \frac{1}{\sqrt{2\pi}} e^{-x^2/2} \cdot 1_{[0,1]} dy.$$

Suppose we draw a sample $X_1, X_2, ..., X_n \sim U(0,1)$ such that the empirical distribution $\hat{\pi}_n(X)$ can be built. Then for n large, equation 3.1.2 is used to approximate[2] the integral. Intuitively, this concept can be very well understood. Instead of integrating from 0 to 1, random points between 0 and 1 are drawn and mapped to the standard normal density function. Taking the average gives an approximation for the expected value. From the z-table for standard normal distributions, we find an 'exact' value which is around 0.3413. Building the sample size from 10 to 100000, the values in figure (3.1.2) are the estimates for the expectation (the experiment for every sample size is repeated 1000 times because of the 'randomness' which comes from the pseudo-random number generator). Indeed, when the sample size becomes larger, the boxplot is more accurate for the true value.

[2] `normalmontecarlo.R`

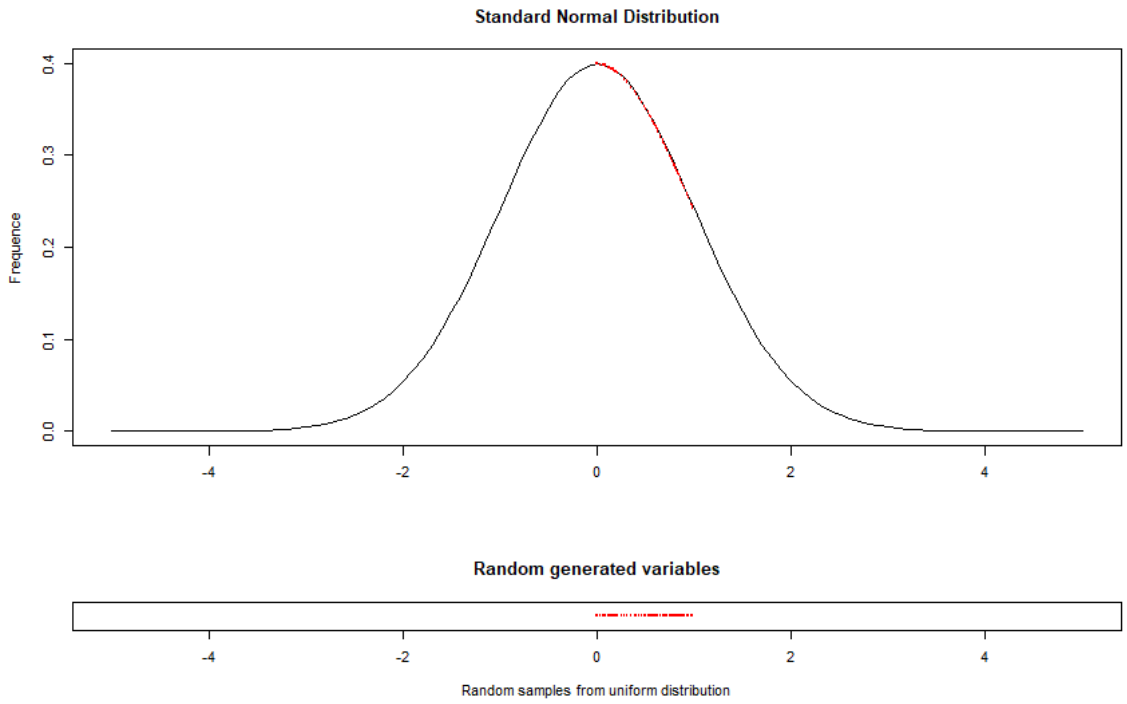

Figure 3.1.1: Visualization of the simulation

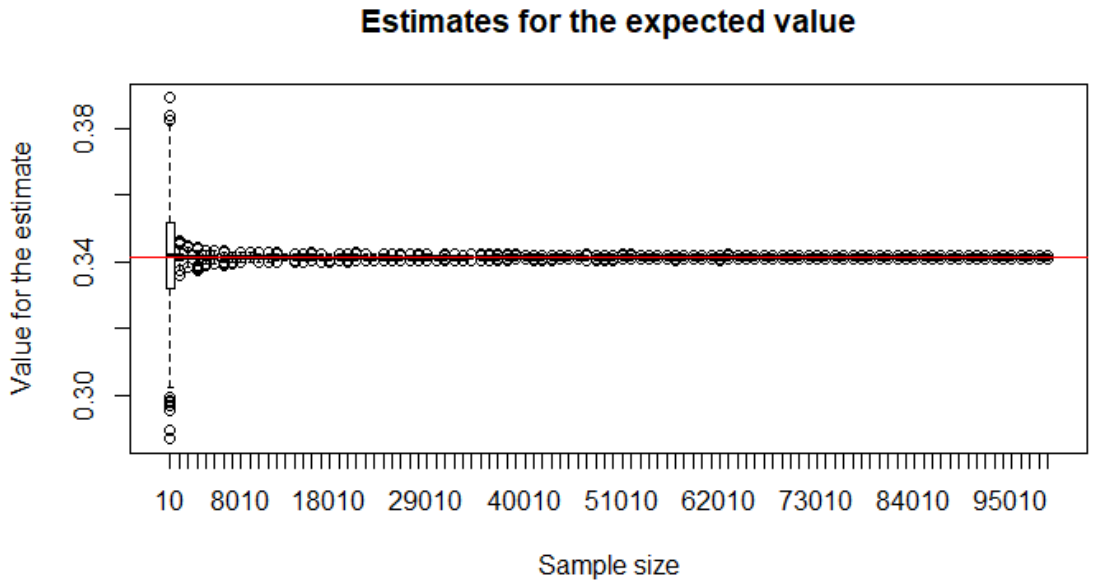

Figure 3.1.2: Estimates for the true value of 0.3415 (red line)

3.2 Markov Chain Monte Carlo

In this section, we will discuss a specific simulation technique called Markov chain Monte Carlo. Because of the fact that entire books can be written about Markov chains, we won't be focusing too much on those definitions. There will be a brief explanation about Markov chains to give some background. However, when it comes to building a Markov chain in order to setup a Markov chain Monte Carlo process, algorithms such as *Gibbs sampler* and *Metropolis-Hastings* can be implemented.

A stochastic process $\{X_t\}_{t\in T}$ is called a *Markov Process* if for all n, $t_1 < ... < t_n$ and $c_1, .., c_n$:

$$P(X_{t_n}|X_{t_1} = c_1, ..., X_{t_{n-1}} = c_{n-1}) = P(X_{t_n}|X_{t_{n-1}} = c_{n-1}),$$

T here is the parameter space and $X(T)$ is the space which contains all states.

The process describes the state X_t of a system on time t. In words this can be interpreted as the conditional probability that, on time t_n, the state of the system equals c_n, while this only depends on state c_{n-1} which describes the state at time t_{n-1}. When a Markov process is discrete in time it is called a *Markov Chain*. More background for Markov Chains is for example described in Norris, J: Markov Chains [9].

One of the main advantages of using Markov Chains in Monte Carlo simulation is that these settings enjoy a very strong stability property. When approximating the integral:

$$\int h(x)\pi(x)dx$$

simulation techniques can be useful. In this section, we will discuss a strategy which shows that with obtaining a sample $X_1, .., X_n$, by using a Markov chain, a proper approximation of $E_\pi[h(X)]$ is done. Here, an estimator such as $\frac{1}{T}\sum_{t=1}^{T} h(X^{(t)})$ is used to estimate the function h of interest.

3.2.1 Gibbs Sampler

A famous MCMC algorithm is called the 'Gibbs sampler' ([17]). The Gibbs sampler is an iterative algorithm that is relatively easy to implement. We want to build a sample from the unknown probability distribution $\pi(\beta)$, in order to ensure that the posterior density function of β is a proper density function. We create a Monte Carlo sample from the distribution of the parameter vector $\beta = (\beta_0, \beta_1, ..., \beta_p)$ such that the empirical distribution $\hat{\pi_n}(\beta)$ can be built. The procedure [13] to create this sample, is given by the following transition from $\beta^{(t)}$ to $\beta^{(t+1)}$:

Given a vector $\beta^{(0)} = (\beta_0^{(0)}, \beta_1^{(0)}, ..., \beta_p^{(0)})$ generate:

Step 1: Let $\beta_1^{(0)}, ..., \beta_p^{(0)}$ fixed and generate a new value β_0 conditional on $\beta_1^{(0)} = \beta_1, ..., \beta_p^{(0)} = \beta_p$ and $X = x$ to obtain a new value $\beta_0^{(1)}$.

Step 2: Similarly, generate a new value $\beta_1 = \beta_1^{(1)}$ from the conditional distribution given $\beta_0^{(1)} = \beta_0, \beta_2^{(0)} = \beta_2, ..., \beta_p^{(0)} = \beta_p$ and $X = x$ to obtain a new value $\beta_1^{(1)}$.

...

Step p: Generate a new value $\beta_p = \beta_p^{(1)}$ from the conditional distribution given $\beta_0^{(1)} = \beta_0, \beta_1^{(1)} = \beta_1, ..., \beta_{p-1}^{(0)} = \beta_{p-1}$ and $X = x$ to obtain a new value $\beta_p^{(1)}$.

This describes one iteration of the Gibbs sampler and generates a new vector $\beta^{(1)}$. Note that at every step, it is only needed to sample from a one-dimensional conditional distribution (these are called full conditionals).Thus, when applying this method even in a high-dimensional problem, all simulations are univariate which is usually an advantage. Because $\beta^{(t)} \sim \pi$ and $\beta^{(t+1)} \sim \pi$, the algorithm only depends on the previous draw and uses only π to draw from. This implies that the Gibbs sampler generates a Markov chain with a stationary distribution (the concept of stationarity is discussed in the next chapter) which converges to the posterior of interest. Therefore, using the law of large numbers, the posterior distribution conditional on the data can be approximated using the Gibbs sampler.

Example of the Gibbs sampler on ridge regression

In this example, we will apply[3] the Gibbs sampler on the data 1.1, by estimating the parameters of the ridge regression (see section 2.3.1). The package 'monomvn' in R, does the sampling for us. This way, we find values for the coefficients of the ridge regression. When using the function 'bridge', it is needed to specify a *burn in*. This is because the Markov chain needs to burn in, which means that it needs to reach its equilibrium distribution. In this example, we will iterate 10000 times and choose a burn in of 1000. The output of this function gives us posteriors of the parameters for the ridge regression. From the 'regress' function, we can attain the maximized a posterior. The first figure contains the estimates for the intercept, while the second figure shows histograms of the estimates for each coefficient.

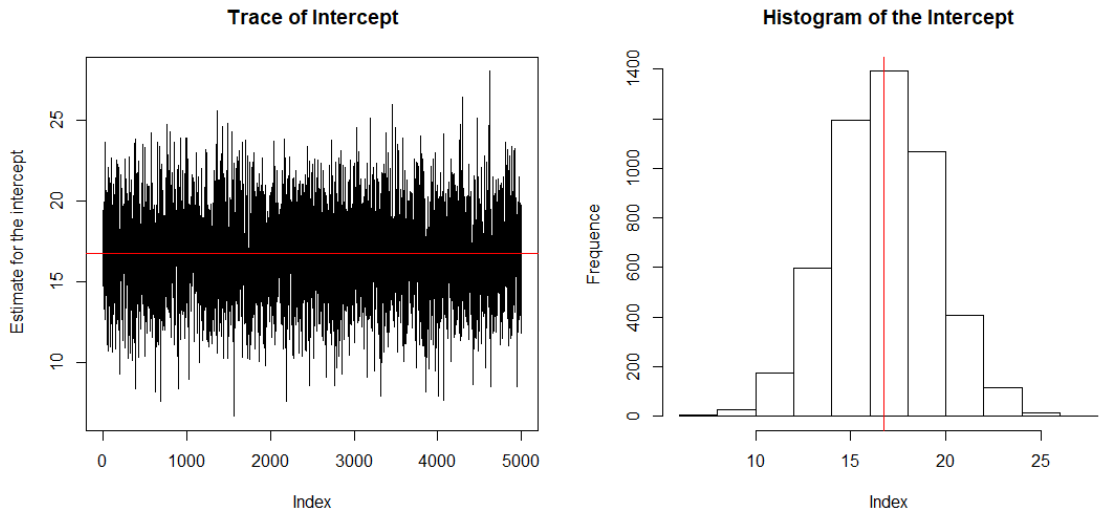

Figure 3.2.1: Trace plot and histogram for the intercept, the red line represents the maximized a posterior

[3] `alledagen.R`

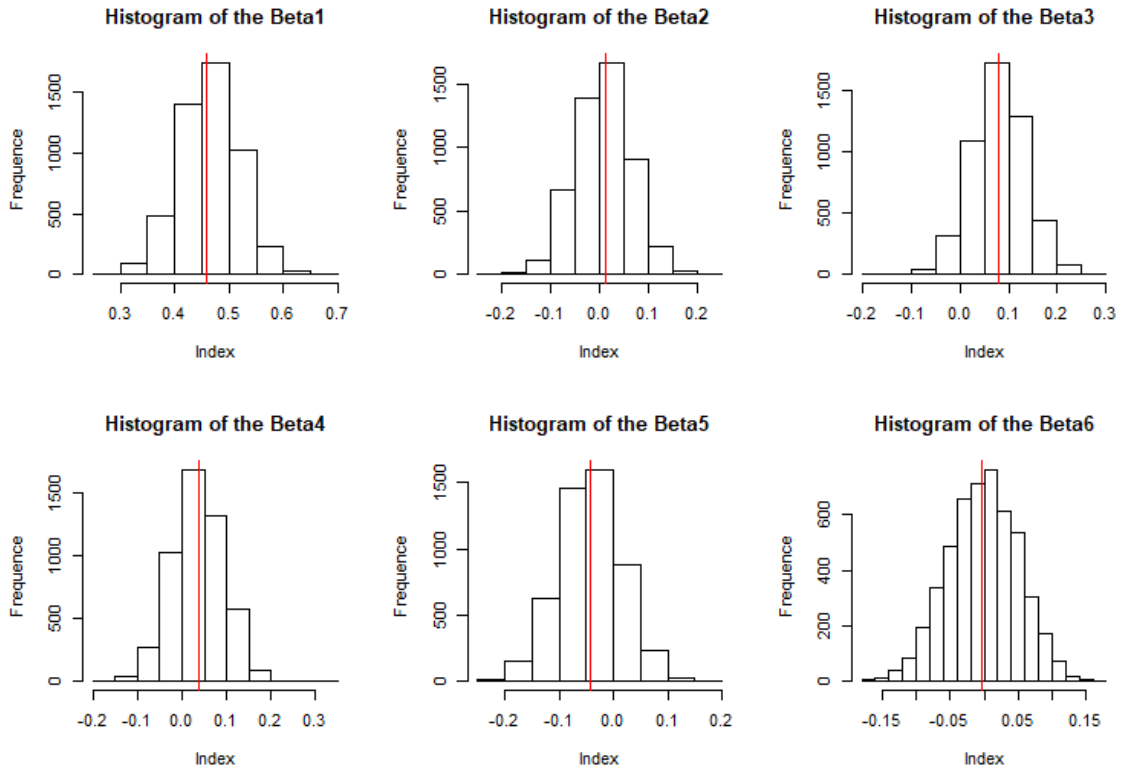

Figure 3.2.2: Histograms for the estimates of the coefficients, the red line represents the maximized a posterior

For each parameter, we estimated a posterior distribution. Because these posteriors are estimated, building a confidence interval works differently from the traditional techniques. This is because the error does not only include a standard error from the model, but also an error which comes from the MCMC simulation. In order to estimate this error, some more statistical definitions are needed. In the next chapter, we will discuss some new definitions in order to find an estimation for the error of an MCMC simulation.

4 Central limit theorem

In the previous chapter, we showed that a posterior distribution π can be approximated using MCMC, which simulates the posterior distribution using random sampling. In this chapter, we will focus on determining the accuracy of a sequence with randomly sampled *dependent* identically distributed variables. In order to do this, we try to estimate the *asymptotic variance*. We start with a more theoretical interpretation on asymptotic variance and later a more practical method for estimating the asymptotic variance is provided.

4.1 CLT for independent and dependent random variables

The classical central limit theorem is stated for independent and identically distributed random variables. It asserts the following:

Theorem 4.1 ***[Central Limit Theorem]*** *Let $\{X_1, ..., X_n\}$ is a sequence of randomly sampled, independent and identically distributed variables, drawn from a distribution with expected value μ and variance σ^2. Then for large n, define the average: $\overline{X}_n = \frac{1}{n}\sum_{i=1}^{n} X_i$. The central limit theorem states that:*

$$\sqrt{n}(\overline{X}_n - \mu) \xrightarrow{\mathcal{D}} N(0, \sigma^2).$$

Because this theorem is stated for independent random variables, it is required to extend this theorem for dependent variables. (More background on this theorem can be found in J. Rice, *Mathematical Statistics and Data Analysis* [12], section 5.3)

Before moving to the extended version of this theorem, it is needed to get some more feeling for a group called *time series*, which is no more than a dataset which is defined on a range of time.

Definition 4.1 *A time series $\{X_t\}$ is called **strictly stationary** [6] if for any times $t_1, ..., t_k$ the distribution of $(X_{t_1+h}, ..., X_{t_k+1})$ is independent of h.*

In other words the likelihood function suffices $\mathcal{L}(X_{t_1}, ..., X_{t_k}) = \mathcal{L}(X_{t_1+h}, ..., X_{t_k+h})$ for all h.

Often, a weaker concept of stationarity is used:

Definition 4.2 *Let $\{X_t\}$ be a time series with $\mathbb{E}[X_t^2] < \infty$, X_t is called **weakly stationary** if:*
(1) $\mathbb{E}[X_t] = \mu$ is independent of t.
(2) $\mathrm{Cov}(X_t, X_{t+h})$ is independent of t for each h.

So from this second property it follows that $\mathrm{Var}(X_t) = \mathrm{Cov}(X_t, X_t)$ is constant. Hence the expectation and variance of weakly stationary time series are constant over time.

Definition 4.3 *Let $\{X_t\}$ be a weakly stationary time series with mean μ.*
*- The **autocovariance function** of $\{X_t\}$ at lag h is*

$$\gamma_X(h) = \mathrm{Cov}(X_{t+h}, X_t) \tag{4.1.1}$$

*- The **autocorrelation function** of $\{X_t\}$ at lag h is*

$$\rho_X(h) = \frac{\gamma_X(h)}{\gamma_X(0)} = \rho(X_{t+h}, X_t) \tag{4.1.2}$$

A time series that is a sequence of dependent random variables, can still have variables which are independent from each other. For example, data from a year ago might not have impact on the behaviour of data today. A formal definition for this sort of dependency is stated in 4.4.

Definition 4.4 *A time series $\{X_t\}$ is called **m-dependent** [16] if random vectors $(..., X_{t-1}, X_t)$ and $(X_{t+1+m}, X_{t+2+m}, ...)$ are independent for every $t \in \mathbb{Z}$.*

With these definitions, the central limit theorem 4.1 can be extended for dependent random variables.

Theorem 4.2 *Let $\{X_t\}$ be a time series which is strictly stationary and m-dependent with mean zero. Let $\overline{X}_n$ be defined as the average and $\gamma_X(h)$ be the autocovariance function. Then $\sqrt{n}\overline{X}_n$ converges to an asymptotic normal distribution with mean 0 and variance $\sigma_\infty^2 := \sum_{h=-m}^{m} \gamma_X(h)$.*

The variance σ_∞^2 which is defined here, is known as the *asymptotic variance*. Note that if $\{X_t\}$ has a non-zero mean, then $\sqrt{n}(\overline{X}_n - \mu) \to N(0, \sigma_\infty)$. A version of the proof for this theorem is given by A. Van der Vaart [16]. A slight variation of this proof is stated below.

Proof. Let m be the integer for which $\{X_t\}$ is m-dependent. Choose a batch size $b > m$ and divide $X_1, ..., X_n$ into $k = \lfloor n/b \rfloor$ batches plus a remainder group of size $n - kb < b$. We define for $1 \leq i \leq k$:

$$A_{i,b} = \sum_{j=1}^{b-m} X_{(i-1)b+j} \quad \text{and} \quad B_{i,b} = \sum_{j=b-m+1}^{b} X_{(i-1)b+j}. \tag{4.1.3}$$

This way, $A_{1,b}, ..., A_{k,b}$ and $B_{1,b}, ..., B_{k,b}$ are sequences of independent and identically distributed random variables with the property that:

$$\sum_{i=1}^{n} X_i = \sum_{j=1}^{k} A_{i,b} + \sum_{j=1}^{k} B_{i,b} + \sum_{i=kb+1}^{n} X_i. \tag{4.1.4}$$

The classical central limit theorem states that if b is fixed and $n \to \infty$ then

$$\frac{1}{\sqrt{n}} \sum_{j=1}^{k} A_{i,b} = \sqrt{\frac{b}{n}} \frac{1}{\sqrt{b}} \sum_{j=1}^{k} A_{i,b} \to \frac{1}{\sqrt{b}} N(0, \text{Var}(A_{i,b})).$$

Because the variables $\frac{1}{\sqrt{n}} \sum_{i=kb+1}^{n} X_i$ have mean zero, the triangle inequality is used:

$$\text{sd}(\frac{1}{\sqrt{(n)}} \sum_{i=kb+1}^{n} X_i) \leq \frac{b}{\sqrt{n}} \text{sd}(X_1) \to 0.$$

Because $\frac{1}{\sqrt{n}} \sum_{j=1}^{k} A_{i,b}$ converges in distribution (see [16], p.36) to $\frac{1}{\sqrt{b}} N(0, \text{Var}(A_{i,b}))$ and, by Chebyshev's inequality, it follows that $\sum_{i=kb+1}^{n} X_i$ tends to zero in probability [4]. Then with Slutsky's lemma [15] it can be derived that, as $n \to \infty$,

$$S_{n,b} := \frac{1}{\sqrt{n}} (\sum_{j=1}^{k} A_{i,b} + \sum_{i=kb+1}^{n} X_i) \to N(0, \frac{1}{b} \text{Var}(A_{i,b})).$$

Now if $b \to \infty$, then

$$\frac{1}{b} \text{Var}(A_{1,b}) = \frac{1}{b} \text{Var}(\sum_{j=1}^{b-m} X_j) = \frac{1}{b} \text{Cov}(\sum_{j=1}^{b-m} X_j, \sum_{k=1}^{b-m} X_k) = \frac{1}{b} \sum_{j=1}^{b-m} \sum_{k=1}^{b-m} \text{Cov}(X_j, X_k).$$

Let $k = j + h$. Then this equation can be formulated as:

$$= \frac{1}{b} \sum_{j=1}^{b-m} \sum_{h=1-j}^{b-m-j} \gamma_X(h).$$

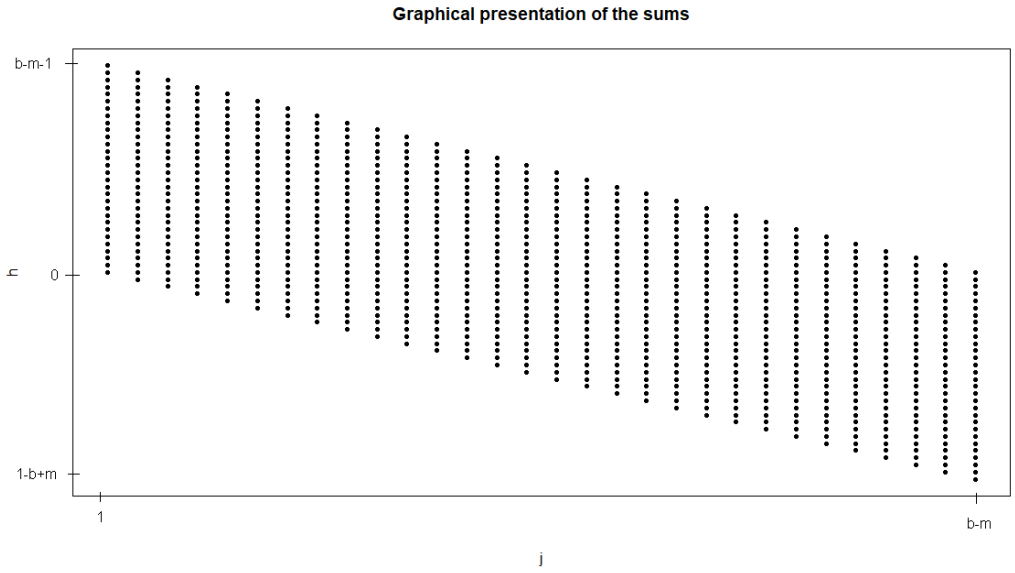

Figure 4.1.1: Graphical presentation of the sums

Flip the sums using the graphical presentation in figure 4.1.1. The sum can be split into 3 parts through the following way:

$$\frac{1}{b}\sum_{h=1-b+m}^{-1}\sum_{j=1-h}^{b-m}\gamma_X(h) + \frac{1}{b}\sum_{1}^{b-m-1}\sum_{j=1}^{b-m-h}\gamma_X(h) + \frac{b-m}{b}\gamma_X(0)$$

$$= \sum_{h=1-b+m}^{-1}\frac{b-m+h}{b}\gamma_X(h) + \sum_{h=1}^{b-m-1}\frac{b-m-h}{b}\gamma_X(h) + \frac{b-m}{b}\gamma_X(0)$$

$$= \frac{b-m}{b}\gamma_X(0) + \sum_{h=1-b+m}^{b-m-1}\frac{b-m-|h|}{b}\gamma_X(h) - \frac{b-m}{b}\gamma_X(0) \to \sum_{h=-m}^{m}\gamma_X(h).$$

This is true because of the symmetry of the autocovariance function. Let $Y_{n,b}$ be a random vector such that $Y_{n,b} \to Y_b$ as $n \to \infty$ for a fixed b, and $Y_b \to Y$ as $b \to \infty$. When this holds, it follows that there exists a sequence $b_n \to \infty$ such that $Y_{n,b_n} \to Y$ as $n \to \infty$. This implies that $b_n \to \infty$ such that $S_{n,b_n} \to N(0, \sigma_\infty)$.

From b_n, a sequence $k_n = \lfloor n/b_n \rfloor$ can be built. Because $\{X_t\}$ is strictly stationary, each $B_{j,b}$ is in distribution equal to $X_1 + ... + X_m$. The variance of $B_{j,b}$ is therefore independent of j, b. By the independence of $B_{1,b}, ..., B_{k,b}$ and by using Chebyshev's inequality:

$$\mathbb{E}\left(\frac{1}{\sqrt{n}}\sum_{j=1}^{k_n} B_{j,b_n}\right)^2 = \frac{b_n}{n}\mathrm{Var}(B_{1,b_n}) \xrightarrow{\mathcal{P}} 0. \qquad \blacksquare$$

Theorem 4.2 is stated for the assumption that a process needs to be strictly stationary and m-dependent. However, it can be very complicated to check these assumptions and in a lot of cases, m might be infinite. For this reason, we will discuss a variant of this theorem which is focused on a specific group of time series called *linear processes*.

[4] $P(d(\sum_{i=kb+1}^{n} X_i, 0) > \epsilon) \to 0, \epsilon > 0$

4.2 Linear process

Define a linear process by a time series which can be written as:

$$X_t = \mu + \sum_{j=-\infty}^{\infty} \psi_j Z_{t-j}$$

here ..., $Z_{-1}, Z_0, Z_1, ...$ are iid variables with mean zero, μ is a constant, and ψ_j are constants with $\sum_j |\psi_j| < \infty$. The autocovariance function of a linear process can be found by:

$$\gamma_X(h) = \text{Cov}(X_t, X_{t+h}) = \text{Cov}(\mu + \sum_{j=-\infty}^{\infty} \psi_j Z_{t-j}, \mu + \sum_{k=-\infty}^{\infty} \psi_k Z_{t-k})$$

$$= \sum_{j=-\infty}^{\infty} \sum_{k=-\infty}^{\infty} \psi_j \psi_k \text{Cov}(Z_{t-j}, Z_{t+h-k}).$$

For j a fixed integer, the covariance term is only nonzero for $k = h + j$. This means that:

$$\gamma_X(h) = \sum_j \psi_j \psi_{j+h} \sigma^2.$$

Here σ^2 is the variance of the $...Z_{-1}, Z_0, Z_1, ..$ variables. Now the asymptotic variance of $\sqrt{n}\overline{X}_n$ is given by:

$$\sigma_\infty^2 := \sigma^2 (\sum_j \psi_j)^2. \tag{4.2.1}$$

Theorem 4.3 *For a linear process with a sequence Z_t of i.i.d variables with mean zero and finite variance, the sequence $\sqrt{n}(\overline{X}_n - \mu) \xrightarrow{\mathcal{D}} \text{Norm}(0, \sigma_\infty^2)$ for large n. [16]*

In the next example, we will dive into a process called *a first order auto regressive process*. This is in particular interesting because the asymptotic variance of this process can easily be derived.

4.2.1 Example with an AR(1) Process

Definition 4.5 *A sequence $\{X_t\}$ of uncorrelated random variables, where $\mathbb{E}[X_t] = 0$ and $\text{Var}(X_t) = \sigma^2$ for each t is called a **white-noise sequence**. [6] We write $X_t \sim WN(0, \sigma^2)$.*

An auto regressive process uses a white noise sequence.

Definition 4.6 *A time series $\{X_t\}$ is follows a **first order auto regressive process** [6], if there exists a white noise process $\{Z_t\}$ and a coefficiënt $\phi \in \mathbb{R}$ such that:*

$$X_t = \phi X_{t-1} + Z_t. \tag{4.2.2}$$

We write $X_t \sim AR(1)$.

Now

$$X_t = \phi \cdot X_{t-1} + Z_t = \phi(\phi X_{t-2} + Z_{t-1}) + Z_t = ... = \phi^k X_{t-k} + \sum_{j=1}^{k-1} \phi^j Z_{t-j}.$$

When $|\phi| < 1$, for k large, it is tempting to state that this limit converges to equation (4.2.3). In fact, we can write a first order autoregressive process as [14]:

$$X_t = \sum_{j=0}^{\infty} \phi^j Z_{t-j}. \tag{4.2.3}$$

Suppose that $\{X_t\}$ is weakly stationary, k, t very large and $|\phi| < 1$.Then:

$$E[X_t] = \phi^k \mathbb{E}[X_{t-k}] + \sum_{j=1}^{k-1} \phi_j \mathbb{E}[Z_{t-j}] = \phi^k \mathbb{E}[X_{t-k}]$$

And because $\{X_t\}$ is stationary, it follows that $\mathbb{E}[X_t] = 0$. For the variance, similarly:

$$\text{Var}(X_t) = \phi^{2k}\text{Var}(X_{t-k}) + \sum_{j=1}^{k-1} \phi^{2j}\text{Var}(Z_t) = \frac{\sigma^2}{1-\phi^2}.$$

The last equality follows from the fact that the first part converges to zero when k becomes large. In the second part, a geometric series with only even powers can be recognized. σ here is the variance of the white noise sequence. Because $\{X_t\}$ is stationary, $\text{Var}(X_t) = \text{Var}(X_{t-k})$ for all k.

We can use the notation of equation 4.2.3 in order to derive the asymptotic variance. Firstly, we will calculate the autocovariance function and secondly, theorem 4.3 can be used to find the asymptotic variance. Because $\{Z_t\}$ is a sequence of i.i.d variables with mean zero and finite variance and because $\sum_j |\phi^j| < \infty$, a first order auto regressive process is linear. Thus, the autocovariance function of an AR(1) process can be found by:

$$\gamma_X(h) = \text{Cov}(X_t, X_t + h) = \text{Cov}(\sum_{j=1}^{\infty} \phi^j Z_{t-j}, \sum_{k=1}^{\infty} \phi^k Z_{t+h-k}) = \sum_{j=1}^{\infty}\sum_{k=1}^{\infty} \phi^{j+k}\text{Cov}(Z_{t-j}, Z_{t+h-k}).$$

This is only nonzero when $k = j + h$ so:

$$= \sum_{j=1}^{\infty} \phi^{2j+h}\text{Var}(Z_{t-j-h}) = \phi^h\sigma^2 \sum_{j=1}^{\infty}(\phi^2)^j = \sigma^2\frac{\phi^h}{1-\phi^2}.$$

This last equality follows from the fact that $|\phi| < 1$ so this is a geometric series. Also note that $\gamma_X(-h) = \gamma_X(h)$ So from theorem 4.3, we can conclude that the asymptotic variance of a first order auto regressive equals:

$$\sigma_\infty^2 = \sigma^2(\sum_{j=-\infty}^{\infty} \frac{\phi^j}{1-\phi^2}) = \frac{\sigma^2}{(1-\phi)^2}. \tag{4.2.4}$$

4.3 Relevance of the asymptotic variance

The goal of this thesis is to determine the accuracy of such a simulation. In this chapter, we have discussed a theoretical approach for finding the asymptotic variance. However, for MCMC it is more useful to estimate the asymptotic variance because it is (in most cases) not even possible to do it theoretically. In the following chapter, different practical methods for approximating the accuracy of an MCMC output using the concept of asymptotic variance will be discussed.

5 Non parametric estimation

5.1 Motivation

When the values $X_1, ..., X_n$ (of an MCMC output) are a stationary time series with mean $\mu_X = \mathbb{E}[X_t]$ and covariance function $h \mapsto \gamma_X(h)$ with an unknown probability distribution (besides that is is stationary), the following *nonparametric* estimators for these parameters can be defined:

$$\hat{\mu_X} = \overline{X}_n = \frac{1}{n}\sum_{t=1}^{n} X_t,$$

$$\hat{\gamma_n}(h) = \frac{1}{n}\sum_{t=1}^{n-h}(X_{t+h} - \overline{X}_n)(X_t - \overline{X}_n).$$

Nonparametric means that the data is not motivated by a statistical model. If the joint distribution of any finite number of the variables X_t is a multivariate normal distribution, the distribution of $\hat{\gamma}_n(h)$ becomes complicated. Therefore, the distribution of these estimators behave more asymptotic. Our goal is to find an asymptotically consistent estimator of the true mean μ_X, with a specified precision. In order to setup an asymptotic confidence interval for μ_X, we need to estimate the asymptotic variance of the sample mean. Typically, such an interval takes the form

$$(\overline{X}_n - \frac{\sigma_\infty}{\sqrt{n}}1.96, \overline{X}_n + \frac{\sigma_\infty}{\sqrt{n}}1.96).$$

If $\sqrt{n}(\overline{X}_n - \mu_X)/\sigma_\infty \rightsquigarrow N(0,1)$, for n large, the confidence level of this interval converges to 95%. For this reason, we want a suitable estimator for σ_∞. Because the variables are not independent and identically distributed, it is much harder to estimate σ_∞. In the sections below, different methods for estimating the asymptotic variance are discussed.

5.2 The batch means method

A method for estimating the asymptotic variance σ_∞ using a model-based estimator, is called the *Batch Means Method* [1]. By dividing an output into batches, the variance of each batch can be estimated. With these estimations, we can find a suitable estimate for the asymptotic variance.

Assumptions For this method, three assumptions are made:
- $\{X_t\}$ is a weakly stationary time series
- $\lim_{n\to\infty} n\text{Var}(\overline{X_t}) < \infty$
- There exists a parameter σ_∞^2 such that for n large $n\text{Var}(\overline{X}_n) \rightsquigarrow \sigma_\infty N(0,1)$. We call this parameter the asymptotic variance

By definition 4.2, this implies that:
- the expected value and variance of these time series are constant
- the autocovariance function $\gamma(h) = \text{Cov}(X_t, X_{t+h})$ depends only on the lag h.

Definitions and notation Let $\{X_t\}$ be a weakly stationary time series. The idea is to split the data into distinct groups called *batches*. Define $n = kb$ where:
- k is the amount of batches
- b is the amount of observations in each batch.

This way, the data is divided in batches as:

$$[X_1, ..., X_b], [X_{b+1}, ..., X_{2b}], ..., [X_{(k-1)b+1}, ..., X_{kb}].$$

Also denote $\sigma_m^2 := \text{Var}(\overline{X}_m)$ for m fixed. By using the variance of each batch, the variance of the total data set can be determined.

Estimating σ_n^2 and σ_b^2 For $i = 1, ..., k$, the ith batch mean is given by:

$$\overline{X_i}(b) = \frac{1}{b}\sum_{j=1}^{b} X_{(i-1)b+j}.$$

The estimator for the mean of the whole sample is:

$$\overline{X}_n = \frac{1}{k}\sum_{i=1}^{k} \overline{X}_i(b).$$

Since the process $\{\overline{X}_i(b), i \geq 1\}$ is also weakly stationary, the covariance is independent for each i and therefore depends only on batch b:

$$\sigma_n^2 = \frac{1}{k^2}(\sum_{i=1}^{k} \text{Var}(\overline{X}_i(b)) + \sum_{i\neq j} \text{Cov}(\overline{X}_i(b), \overline{X}_j(b))) = \frac{\sigma_b^2}{k} + \frac{1}{k^2}\sum_{i\neq j} \text{Cov}(\frac{1}{b}\sum_{l=1}^{b} X_{(i-1)b+l}, \frac{1}{b}\sum_{k=1}^{b} X_{(j-1)b+k})$$

$$= \frac{\sigma_b^2}{k} + \frac{1}{k^2b^2}\sum_{i=1}^{n}\sum_{j=1}^{n} \text{Cov}(X_i, X_j) - \frac{1}{k^2}\sum_{i=1}^{k} \text{Cov}(\overline{X}_i(b), \overline{X}_j(b)) = \frac{\sigma_b^2}{k} + \frac{1}{n^2}\text{Cov}(\sum_{i=1}^{n} X_i, \sum_{j=1}^{n} X_j) - \frac{\sigma_b^2}{k}$$

$$= \frac{\sigma_b^2}{k} + \sigma_n^2 - \frac{\sigma_b^2}{k} = \frac{\sigma_b^2}{k}(1 + \frac{n\sigma_n^2 - b\sigma_b^2}{b\sigma_b^2}). \tag{5.2.1}$$

Here, $n \geq b$ and $\frac{b}{k}(\frac{n\sigma_n^2 - b\sigma_b^2}{b\sigma_b^2}) = \frac{n\sigma_n^2 - b\sigma_b^2}{n\sigma_b^2} \to 0$ when $n \to \infty$. When the sample size n is large enough, $\frac{\sigma_b^2}{k}$ approximates σ_n^2.

For σ_b^2, the following estimator can be used:

$$\hat{V}_k(b) = \frac{1}{k-1}\sum_{i=1}^{k}(\overline{X}_i(b) - \overline{X}_n)^2.$$

One of the questions which arises for implementing the batch means method is on determining the optimal batch size. This appears to vary from process to process. In the next section, an example for determining the optimal batch size for an AR(1) process is described.

5.2.1 Optimal Batch size for an AR(1) process

We can determine the optimal batch size for an AR(1) process, by calculating the minimal mean squared error. This process is an interesting case, because for an AR(1) process the value of the asymptotic variance is known (see example 4.2.1).

Theoretical Approach

Definition 5.1 *The mean squared error of an estimator $\hat{\theta}$ for the parameter θ is defined by:*

$$\mathrm{MSE}(\hat{\theta}) = \mathbb{E}[(\hat{\theta} - \theta)^2] = \mathrm{Var}(\hat{\theta}) - \mathrm{Bias}(\hat{\theta})^2.$$

Suppose we have an AR(1) process as defined in definition 4.6. Let $b_n \to b$ be a sequence for the batch sizes. Under reasonable assumptions, it is generally stated that $n\sigma_n^2 \rightsquigarrow \sigma_\infty^2$ for n large. Therefore we choose as an estimator for the asymptotic variance $b_n \hat{V}_k$ because this approximates $b_n \sigma_b^2 \approx n\sigma_n^2 \rightsquigarrow \sigma_\infty^2$. The theoretical asymptotic variance of an AR(1) process equals $\sigma_\infty^2 = \frac{\sigma^2}{(1-\phi)^2}$ so the bias of this estimator is (Carlstein [2]):

$$\mathrm{Bias}(b_n \hat{V}_k) = \sigma_\infty^2 - \mathbb{E}[b_n \hat{V}_k] = \frac{2\phi}{(1-\phi)(1-\phi^2)b_n} + \mathcal{O}(\frac{1}{b_n}) \tag{5.2.2}$$

and

$$\mathrm{Var}(b_n \hat{V}_k) = \frac{2}{(1-\phi)^4}\frac{b_n}{n} + \mathcal{O}(\frac{b_n}{n}).$$

For the mean squared error, we find:

$$\mathrm{MSE}(b_n \hat{V}_k) = \frac{4\phi^2}{(1-\phi)^2(1-\phi^2)^2 b_n^2} + \frac{2}{(1-\phi)^4}\frac{b_n}{n}. \tag{5.2.3}$$

It can be minimized by taking the derivative and finding its roots.

$$\mathrm{MSE}' = -\frac{8\phi^2}{(1-\phi)^4(1-\phi^2)^2 b_n^3} + \frac{2}{(1-\phi)^4 n} = 0.$$

This becomes:

$$\frac{8\phi^2}{(1-\phi)^4(1-\phi^2)^2 b_n^3} = \frac{2}{(1-\phi)^4 n}.$$

So the optimal batch size is:

$$b_0 = (\frac{2|\phi|}{1-\phi^2})^{2/3} n^{1/3}. \tag{5.2.4}$$

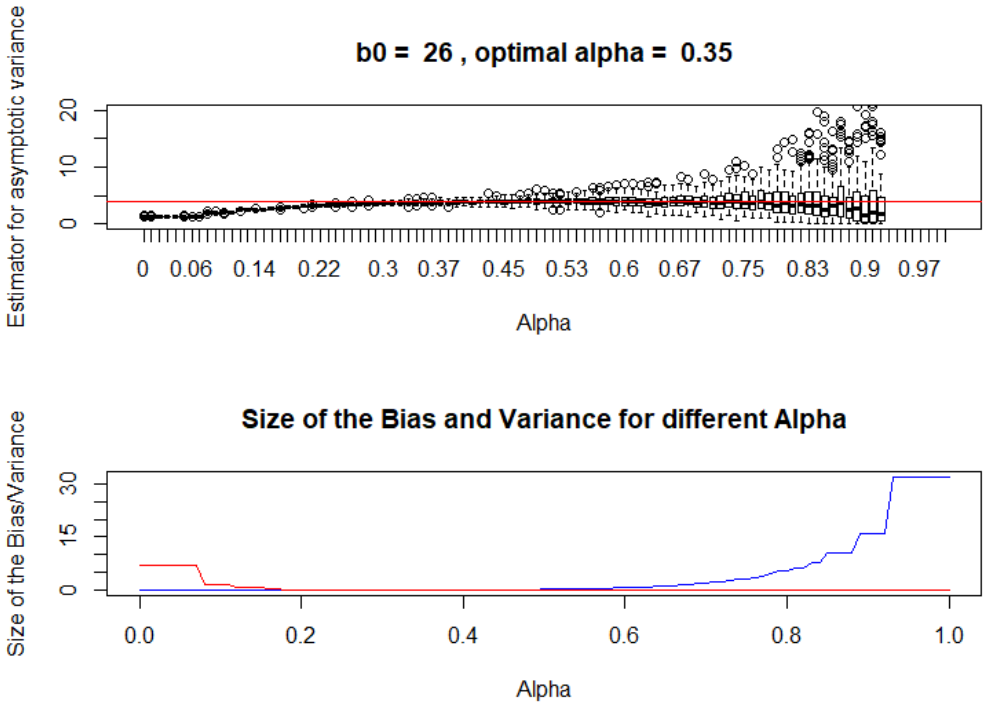

Figure 5.2.1: Boxplots, variance (blue) and bias (red) for different alpha's (1000 times repeated)

Experimental Approach

Because the asymptotic variance of an AR(1) process is known, the theory which is given here can be compared to an experimental approach. This way, the quality of the method becomes clearer which contributes to comparing methods for estimating the asymptotic variance. By the following two experiments[5], we test the batch means method on an AR(1) process (for $\phi = 0.5$ and $Z_t \sim WN(0,1)$). In the first experiment, the optimal batch size will be determined numerically. Secondly, we will look at the impact of varying n while keeping each batch size fixed.

Varying b while n is fixed Fix $n = 10^4$ and define $b_n = n^\alpha$, where $0 \leq \alpha \leq 1$. We expect the optimal batch size to be around $\alpha \approx 1/3$. The experiment is repeated a 1000 times. In figure 5.2.1, the results are given (see appendix for a larger view of this plot). In this figure, also the size of the bias (squared) and variance are shown.

From the figure, it follows that for α between 0.25 and 0.45 the mean squared error (based on the bias and variance), is relatively small. For α smaller than 0.25, the variance is small but the bias is large. On the other hand when α is larger than 0.45, the variance becomes larger.

Varying n while b is fixed Fix $b = b_0$ (5.2.4) and define the sequence $n_\alpha = 100 + 500\alpha$. When estimating the asymptotic variance, we repeat the experiment 50 times. We expect that when n_α becomes larger, the boxplot will be narrower and closer to the theoretical asymptotic variance. In figure 5.2.2, a plot of the results can be found.

[5] `grootexperiment.R`

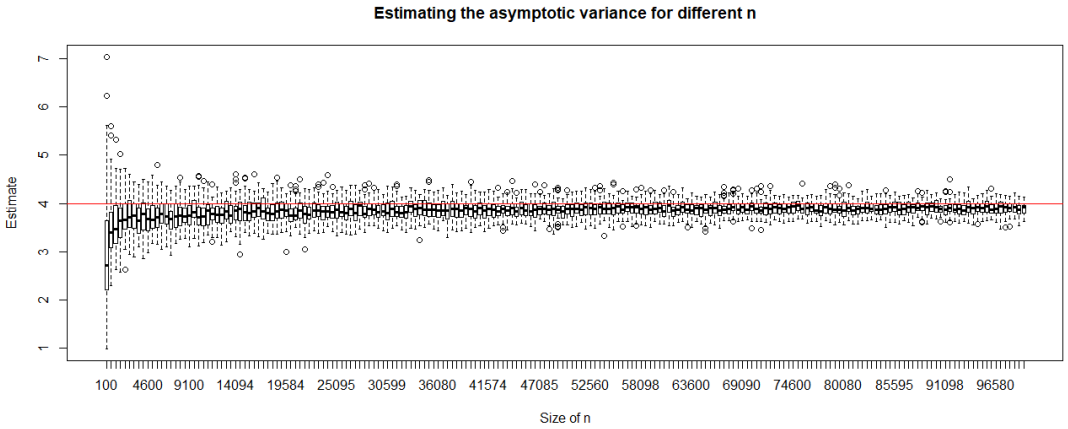

Figure 5.2.2: Boxplots for different n's (50 times repeated)

It seems that even when n becomes larger, the estimate for the asymptotic variance stays biased. Therefore, the question arises whether the experimental optimal value for batch size is different from the theoretical value. The optimal batch size by experiment, can be determined by finding the batch size which has the *Least Squared Error (LSE)* with respect to the true value. In the plot below, we use the output from the experiment we did to setup figure 5.2.1, and plot the MSE of each box-plot (i.e 1000 data points for each batch size). For the results, see figure 5.2.3. The minimum of this plot is the Least Squared Estimate and is shown with a red line. It appears the the optimal batch size of this experiment equals 0.35, which is a good estimate for the exact value $\alpha = 0.33$. The mean of the 1000 samples for $\alpha = 0.35$ equals 3.92 which is indeed a little lower than the theoretical value 4. From this, we may conclude that the experiment gives a biased value. This follows also from the plot of the bias. Theoretically, the bias should go to zero but in practice this seems not to be the case.

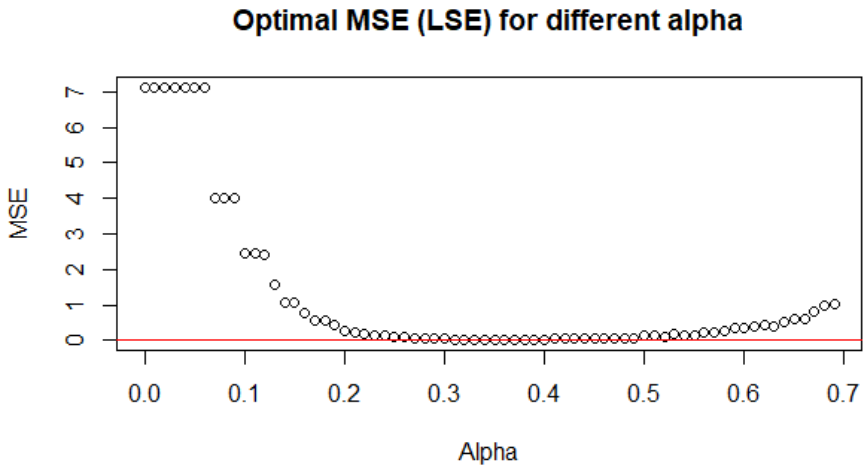

Figure 5.2.3: Mean Squared Errors for different batch sizes (1000 times repeated)

5.3 The blocked means method

In this subsection, we will discuss a method called *the blocked means method*. Because the batch means method uses disjoint batches, it assumes some independence between the batches. However, in most cases these variables are dependent. For this reason, it might be more sufficient to look at this method which is based on independent blocks.

For this method, a block size (or batch size) b is fixed. The sample with n variables can be divided in $n - b + 1$ blocks of size b. This means that the data is split up as follows:

$$[X_1, ..., X_b], [X_2, ..., X_{b+1}], ..., [X_{n-b+1}, ..., X_n].$$

For $i = 1, ..., n - b + 1$, the mean of the i-th block is given by:

$$\overline{X}_i(b) = \frac{1}{b} \sum_{j=1}^{b} X_{(i-1)+j}.$$

For b large, we want to choose b such that $b\text{Var}(\overline{X}_i(b)) = \text{Var}(b\overline{X}_i(b)) \approx \text{Var}(\sqrt{n}\overline{X}_n)$. This suggests that $b\sigma_b^2 \to n\sigma_n^2 \approx \sigma_\infty^2$. Let $b_n \to \infty$ be a sequence that does not converge too fast and is dependent on n. We define the following estimator for σ_b^2:

$$\hat{V}_k(b_n) = \frac{1}{n - b_n + 1} \sum_{i=1}^{n-b_n+1} (\overline{X}_i(b_n) - \overline{X}_n)^2.$$

Theorem (5.3) from (A. van der Vaart: *Time Series* [16]) suggests that under some conditions such as $\frac{b_n}{n} \to 0$ as $b_n \to \infty$ and by the assumption which supposes that $\sqrt{n}(\overline{X}_n - \mu_X) \rightsquigarrow N(0, \sigma_\infty^2)$, for some σ_∞^2, the estimator $b\hat{V}_k(b_n)$ will converge in probability to σ_∞^2 (here $\sigma_\infty^2 = \sum_{h=-m}^{m} \gamma_X(h)$).

5.4 Estimating the asymptotic variance using window estimators

A useful thought for estimating asymptotic variance follows from theorem 4.2. Firstly, we try to estimate the autocovariance function. An estimator for this function is the estimator described in the first part of this chapter:

$$\hat{\gamma}_n(h) = \frac{1}{n} \sum_{t=1}^{n-h} (X_{t+h} - \overline{X}_n)(X_t - \overline{X}_n). \tag{5.4.1}$$

An obvious but naive choice for estimating the asymptotic variance would then be:

$$\hat{\sigma}_\infty^2 \approx \sum_{h=-\infty}^{\infty} \hat{\gamma}_n(h) \approx \sum_{h=-n}^{n} \hat{\gamma}_n(h) = 2 \sum_{h=1}^{n} \hat{\gamma}_n(h) + \hat{\gamma}_n(0).$$

This last equality is true because of the symmetry of the autocovariance function, i.e: $\hat{\gamma}_n(h) = \hat{\gamma}_n(-h)$. This estimate is presumably not very precise because when h is close to n, the sum is taken over very few values which causes high sensibility for unstable values. A solution would be adding a variable to the estimates which down weights the terms when h is large. These variables are called *window estimators* [4] and are implemented in the following way:

$$\hat{\sigma}_\infty^2 \approx 2 \sum_{h=1}^{n} w_n(h)\hat{\gamma}_n(h) + w_n(0)\hat{\gamma}_n(0) \tag{5.4.2}$$

For large values of h, $w_n(h)$ needs to be very small. A window estimator which is based on the Tukey-Hanning lag window [11] is defined as $w_n(h) = \frac{1}{2}(1+\cos(\pi h/K))$ for $|h| \leq K$ and otherwise 0. For K, a value can be chosen so that $\hat{\gamma}_n(h) \approx 0$ for $|h| > K$. In reality, the choice of K seems to be fairly insensitive for the estimated variance. A comparison of more lag window generators is discussed in (Neave (1972) [8]). More information on window estimators can be found in (Priestley (1981) [10]).

5.4.1 Setting up a confidence interval for the wind speed data using the batch means method

In this section, we will try to setup a confidence interval for the estimation of the wind speed data using the batch means method. By applying the Gibbs sampler for ridge regression (which is described in chapter 3) on the data of the wind speeds, using the model:

$$Y = X\beta + \epsilon.$$

Here, $X = \begin{pmatrix} 1 & X_{11} & \cdots & X_{16} \\ \vdots & \vdots & & \vdots \\ 1 & X_{n1} & \cdots & X_{n6} \end{pmatrix}$ and $\epsilon \sim N(0, \sigma^2)$. Furthermore there parameter vector $\beta = \begin{pmatrix} \beta_0 \\ \vdots \\ \beta_6 \end{pmatrix}$ is estimated by the taking the mean of the a posteriors which are shown in figures 3.2.2 and 3.2.1. For observations (X_i, y_i), the posterior π over β can be determined. By the law of total variance, the exact posterior yields:

$$\text{Var}(Y) = \mathbb{E}_\pi[\text{Var}(Y|\beta)] + \text{Var}_\pi(\mathbb{E}[Y|\beta]) = \sigma^2 + \text{Var}_\pi(X\beta) \tag{5.4.3}$$

The goal here is to determine $\text{Var}_\pi(X\beta)$. Generally:

$$\text{Var}_\pi(X\beta) = \mathbb{E}_\pi[(X\beta)^2] - (\mathbb{E}_\pi[X\beta])^2 = \int (X\beta)^2 \pi(\beta) d\beta - (\int X\beta\pi(\beta)d\beta)^2.$$

Here, $\pi(\beta)$ is the posterior distribution of β. So, the estimate for Y would be:

$$\hat{Y} = \int X\beta\pi(\beta)d\beta \pm \sqrt{\sigma^2 + \int (X\beta)^2\pi(\beta)d\beta - (\int X\beta\pi(\beta)d\beta)^2} \tag{5.4.4}$$

$$= f(\pi(X\beta)) \pm g(\pi(X\beta), \pi((X\beta)^2))$$

with $f(x_1) = x_1$ and $g(x_1, x_2) = \sqrt{\sigma^2 - x_1 + x_2}$. Note that $\pi(x) = \int x\pi(\beta)d\beta$. In our case, $\pi(\beta)$ is estimated by MCMC and therefore contains a Monte Carlo error. By using Taylor expansion, this error can be determined. The error for the estimated function f in $\pi(X\beta)$ is:

$$f(\pi(X\beta)) - f(\hat{\pi}(X\beta)) = \frac{df(\hat{\pi}(X\beta))}{d\hat{\pi}(X\beta)}(\pi(X\beta) - \hat{\pi}(X\beta)) + \mathcal{O}(f''(\hat{\pi}(X\beta)))$$

$$= (\pi(X\beta) - \hat{\pi}(X\beta)) + \mathcal{O}(f''(\hat{\pi}(X\beta))) = \sqrt{\frac{\sigma^2_\infty(X\beta)}{n}} + \mathcal{O}(f''(\hat{\pi}(X\beta))).$$

The error of the estimated function g in $(\pi(X\beta), \pi((X\beta)^2))$ is:

$$g(\pi(X\beta), \pi((X\beta)^2)) - g(\hat{\pi}(X\beta), \hat{\pi}((X\beta)^2))$$

$$= \frac{\partial g(\hat{\pi}(X\beta), \hat{\pi}((X\beta)^2))}{\partial\hat{\pi}(X\beta)}(\pi(X\beta) - \hat{\pi}(X\beta)) + \frac{\partial g(\hat{\pi}(X\beta), \hat{\pi}((X\beta)^2))}{\partial\hat{\pi}((X\beta)^2)}(\pi((X\beta)^2) - \hat{\pi}((X\beta)^2)) + \mathcal{O}(g''(\hat{\pi}(X\beta), \hat{\pi}((X\beta)^2))$$

$$= -\frac{\hat{\pi}(X\beta)}{2g(\hat{\pi}(X\beta),\hat{\pi}((X\beta)^2))}(\pi(X\beta) - \hat{\pi}(X\beta)) + \frac{1}{2g(\hat{\pi}(X\beta),\hat{\pi}((X\beta)^2))}(\pi((X\beta)^2) - \hat{\pi}((X\beta)^2)) + \mathcal{O}(g''(\hat{\pi}(X\beta),\hat{\pi}((X\beta)^2))$$

$$= -\frac{\hat{\pi}(X\beta)}{2g(\hat{\pi}(X\beta),\hat{\pi}((X\beta)^2))}\sqrt{\frac{\sigma^2_\infty(X\beta)}{n}} + \frac{1}{2g(\hat{\pi}(X\beta),\hat{\pi}((X\beta)^2))}\sqrt{\frac{\sigma^2_\infty((X\beta)^2)}{n}} + \mathcal{O}(g''(\hat{\pi}(X\beta),\hat{\pi}((X\beta)^2)).$$

So the estimate for Y becomes:

$$\hat{\hat{Y}} = \int X\beta\hat{\pi}(\beta)d\beta$$

$$\pm\Big[\sqrt{\sigma^2 - \hat{\pi}(X\beta)^2 + (\hat{\pi}(X\beta))^2} + \sqrt{\frac{\sigma^2_\infty(X\beta)}{n}}(1 - \frac{\hat{\pi}(X\beta)}{2g(\hat{\pi}(X\beta),\hat{\pi}((X\beta)^2))}) + \frac{1}{2g(\hat{\pi}(X\beta),\hat{\pi}((X\beta)^2))}\sqrt{\frac{\sigma^2_\infty((X\beta)^2)}{n}}\Big].$$

When setting up some sort of confidence interval, we need to know the value for σ^2 and we have to determine the distribution of the standard error. σ^2 could be estimated by MCMC, however, the error would become more complicated and therefore we will use the residual variance from the 'lm' function which is used in section (2.1.1). Note that this is a choice for σ^2 and not necessarily the true value. For the distribution of the standard error, we will make the very weak assumption that it is t-distributed. This is because it is very hard to determine the true distribution for this standard error. By making these weak assumptions, a 'confidence interval' which describes an interval for the estimated values can be built. Comparing this interval to the data gives figure 5.4.1 (here we tried to build a 95% confidence interval). Because the error can be split up in an error for the regression and a Monte Carlo error, we will use different colours to make this visible. The lighter colour describes the standard error without taking the Monte Carlo error into account, while the darker area takes the total standard error into account.

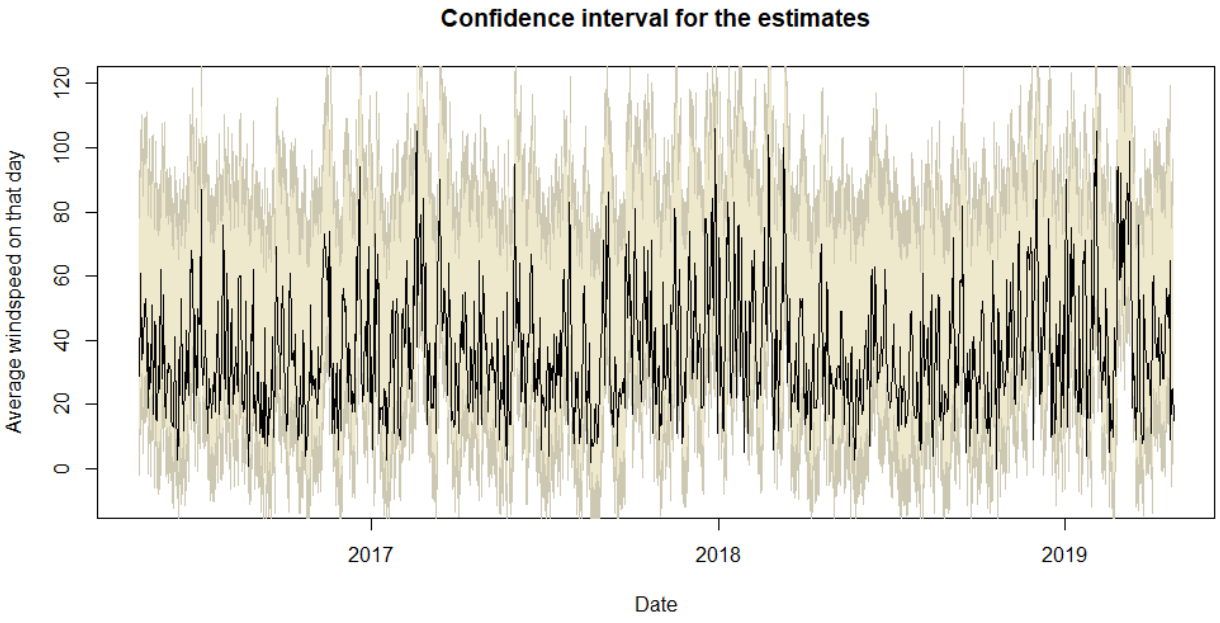

Figure 5.4.1: The black lines are the real data points

It appears that ca. 96.97% of the data points lays inside the estimated confidence interval. When the Monte Carlo error is not taken into account, 85% lays inside the confidence interval based on the regression error. This shows that it is very useful to look at the Monte Carlo error as well.

5.5 Comparison between the different methods

When comparing the three methods which are explained in the previous sections, it is important to focus on performance, as well as the practical aspects of each method (speed, easy to implement, potential). The discussed methods will be applied to four AR(1) processes with different values for ϕ. All figures below are based on a experiment which is repeated a 100 times.

Method In order to compare the discussed methods, we will apply[6] them on an AR(1) process with $\phi = \{-0.5, 0.1, 0.5, 0.9\}$ and $Z_t \sim WN(0,1)$. Here we compare the batch means method, the blocked means method and 2 different methods using window estimators. One method using a window estimator is the naive choice where $w_n = 1$. Because this is a very trivial window estimator, we will call it for the rest of this chapter the method 'without using a window estimator'. The window estimator which is used for the other method with a window estimator, is the one which is based on the Tukey-Hanning lag window (see section 5.4). For the size of the batches and blocks, we use $b_n = n^\alpha$ with $\alpha = 1/3$. On the x-axis, the value of the sample size n is given. Here, n varies between 200 and 10000 (step size 200). The y-axis shows the estimated value for the asymptotic variance and the red line represents the theoretical value for the asymptotic variance which is discussed in example 4.2.1. Also the Mean Squared Error (MSE) of all methods is calculated to give a more quantitative result to compare with.

Results $\phi = -0.5$ In figure 5.5.1, the results for an AR(1) process with $\phi = -0.5$ are given.

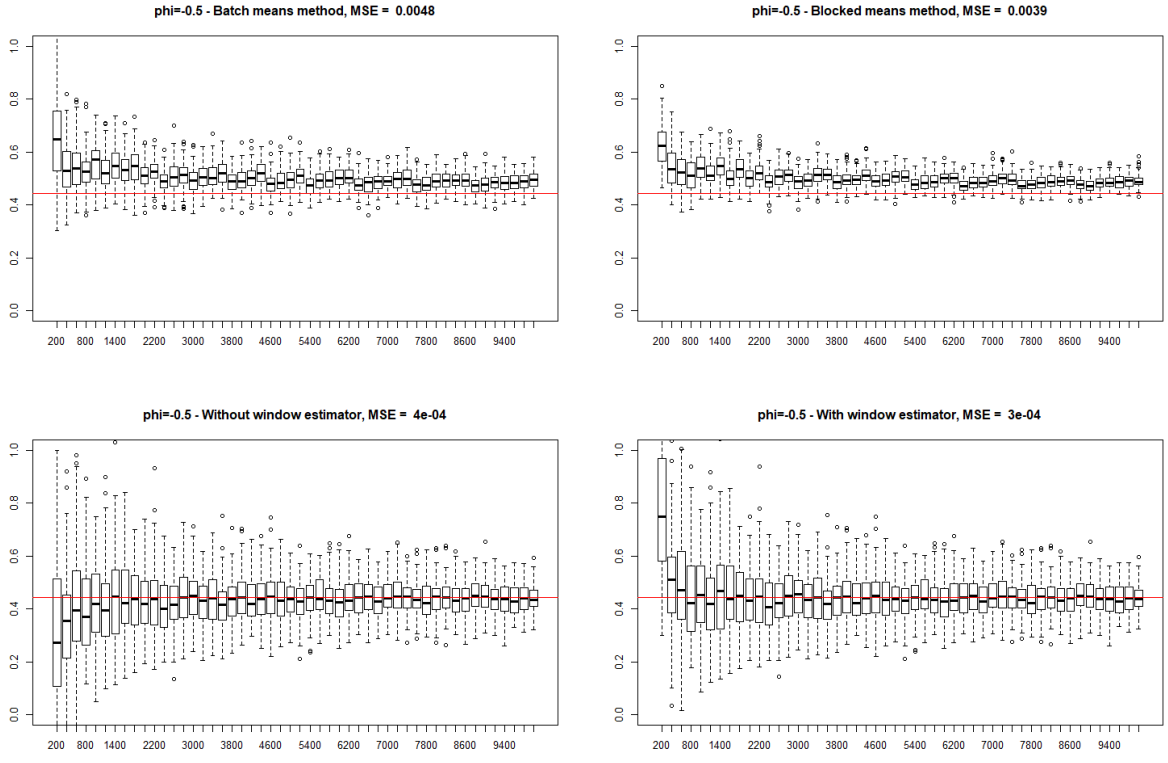

Figure 5.5.1: AR(1) process $X_t = \phi X_{t-1} + Z_t$, $Z_t \sim WN(0,1)$ and $\phi = 0.1$

[6] `methodcomparing.R`

From these plots, it becomes clear that the batch/blocked means method both overestimate the true value for the asymptotic variance. The spread of these estimates is small which causes sensibility for biased estimation. The methods using window estimators are doing much better. Despite the fact that these estimates have a very large spread, the estimates have, on average, better performance than the estimates of the batch/blocked means method. In this experiment, the implementation of the window estimator based on the Tukey-Hanning lag window gives the best results. Note that using this window estimator causes an overcompensation for small values of n.

When looking at these results, the question arises whether using the optimal batch size gives different results. In figure 7.2.1 (see appendix) the optimal batch size for the batch means method is given (using equation 5.2.4). When applying this size for b_n and comparing this to the methods using window estimators, figure 5.5.2 is obtained. Besides the fact that the estimates are little better then the previous results, the presence of the bias is still notable. It shows that for the batch means method (and probably blocked means method as well), the right choice for b_n is important. Though it seems that the method has little potential. Maybe for very large sample sizes the results could be useful because the estimates are more specified than the ones based on window estimators.

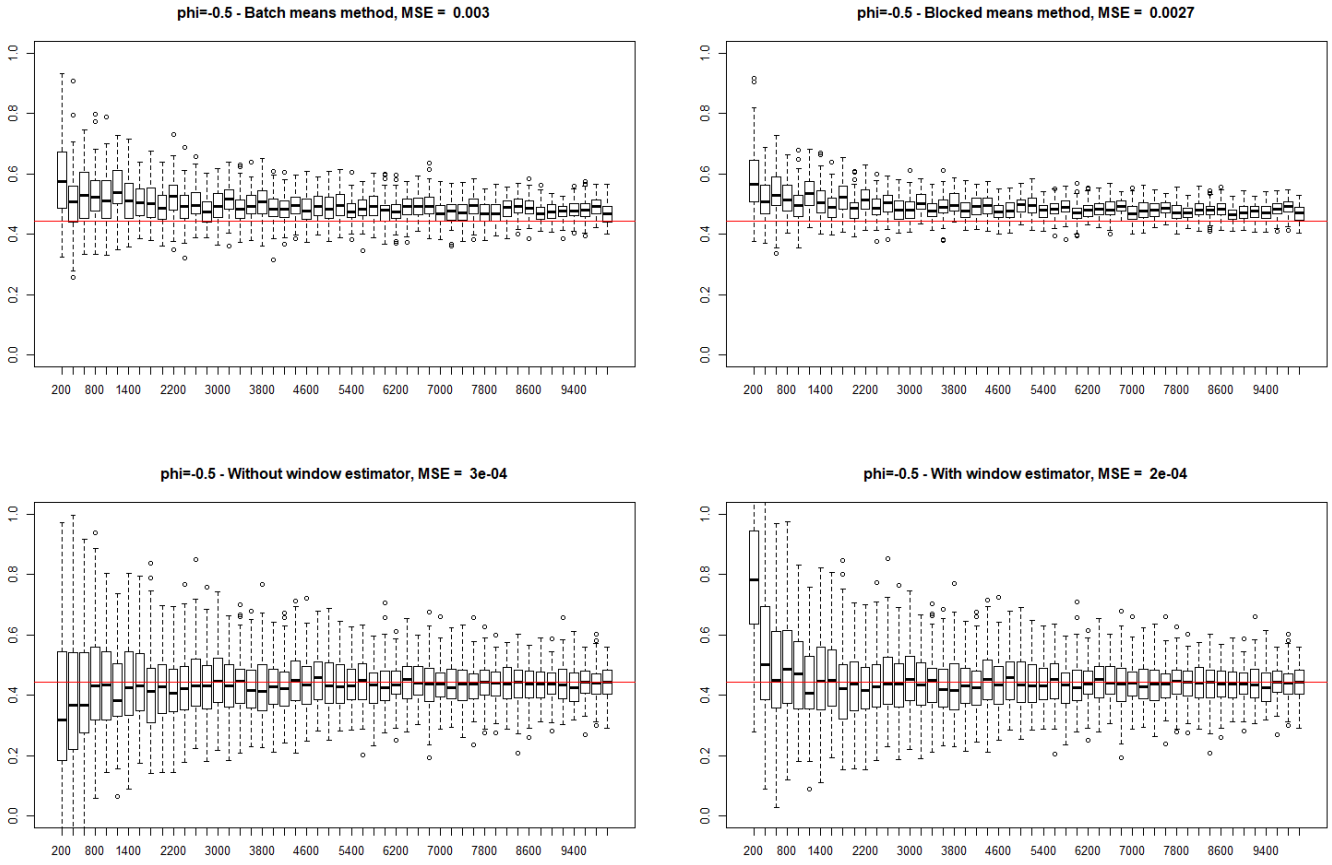

Figure 5.5.2: AR(1) process $X_t = \phi X_{t-1} + Z_t$, $Z_t \sim WN(0,1)$ and $\phi = -0.5$, applied on optimal batch sizes

$\phi = 0.1$: In figure 5.5.3, the results for an AR(1) process with $\phi = 0.1$ are given.

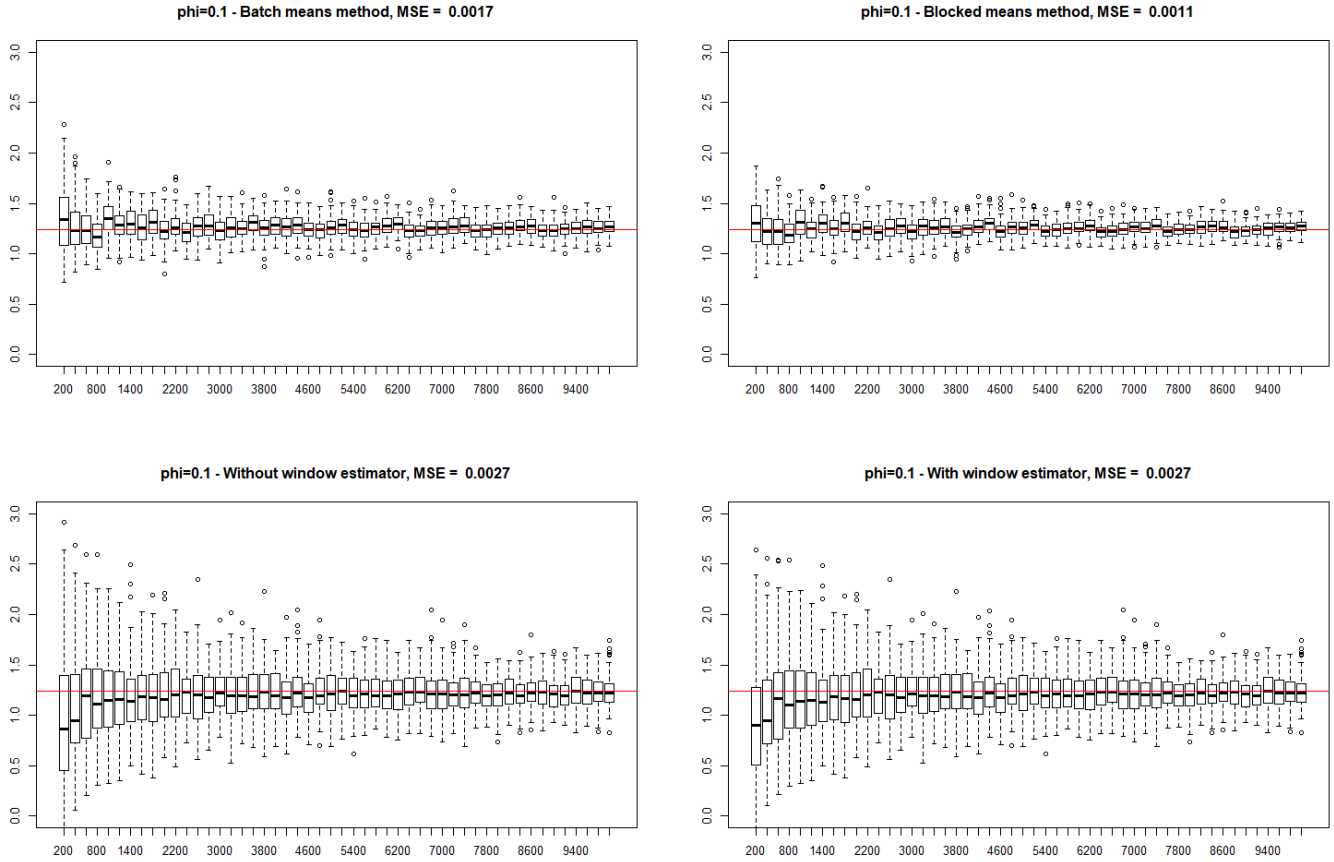

Figure 5.5.3: AR(1) process $X_t = \phi X_{t-1} + Z_t$, $Z_t \sim WN(0, 1)$ and $\phi = 0.1$

Compared to the case where $\phi = -0.5$, the results are different. In this case the batch/blocked means methods give similar results. From the calculated MSE, the blocked means method gives even slightly better results. It is notable that while n is increasing, the estimates are not becoming much more accurate. The blocked/batch means methods give much more specific estimates than the methods using window estimators and are, especially for small values of n, more stable. The other two methods are for this process less accurate (considering the spread). It seems that implementing a window estimator does not give very different results compared to the method without a window estimator.

$\phi = 0.5$: In figure 5.5.4 the results for an AR(1) process with $\phi = 0.5$ are given.

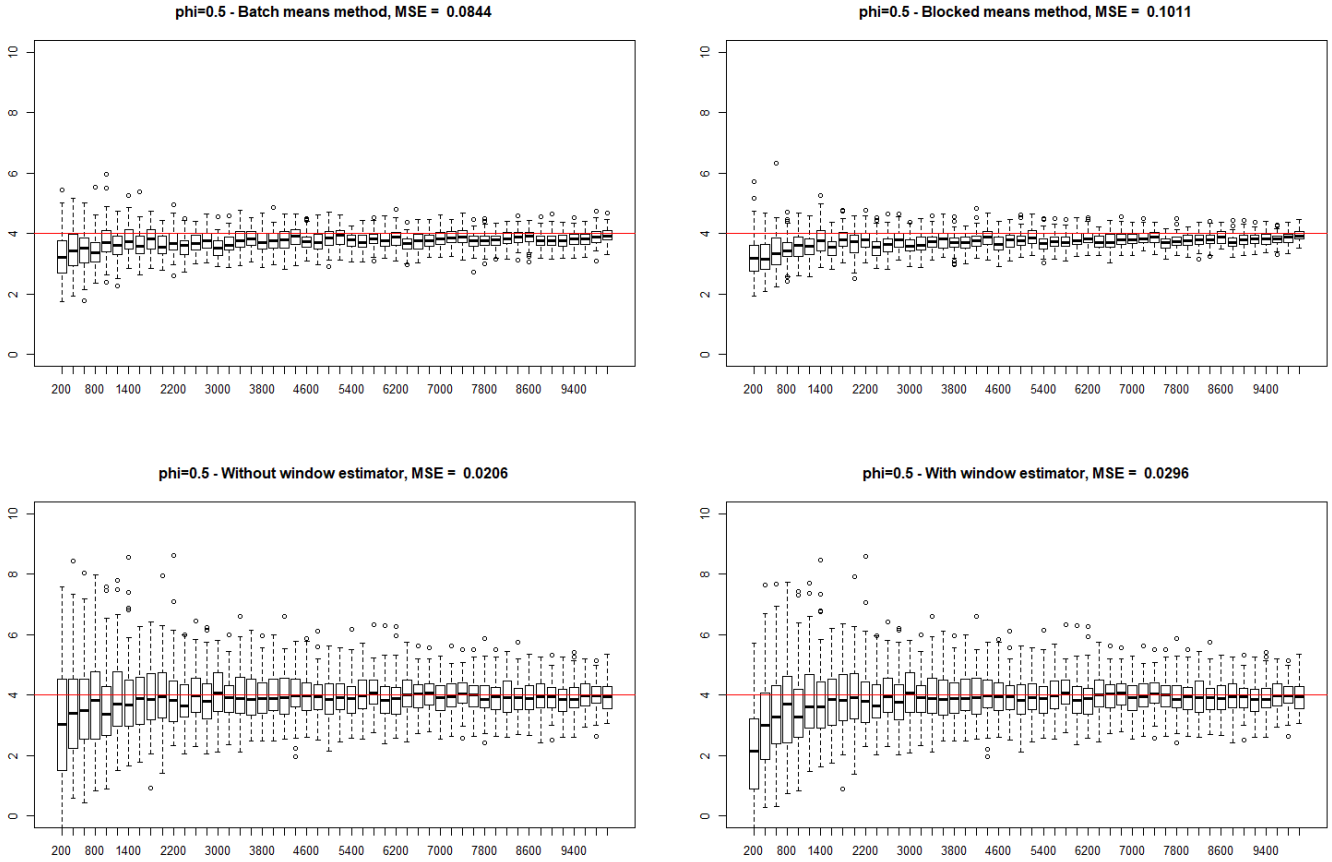

Figure 5.5.4: AR(1) process $X_t = \phi X_{t-1} + Z_t$, $Z_t \sim WN(0,1)$ and $\phi = 0.5$

According to the mean squared error, the batch means method is a little bit more accurate than the blocked means method, in this case. Both the batch means method and the blocked means method have a lower performance compared to the other two methods. Looking at the figures, this result is due to a bias. From an earlier section in this chapter, we have seen that the bias is very sensible for the value of b_n (see figure 5.2.1). However, even for an optimal sequence of batch sizes b_n, the results stay biased (see figure 5.2.2). From the results shown in the figure, we note that the methods using window estimators have much larger spreads than the other two methods. What is again significant is that the estimates which are done without implementing a window estimator have higher performance than the results with a window estimator. This might be because the window estimator is not suitable for this kind of processes and thus a different window estimator would give better results (for example the ones described in the literature which is mentioned in section 5.4).

$\phi = 0.9$**:** In figure 5.5.5 the results for an AR(1) process with $\phi = 0.9$ are given.

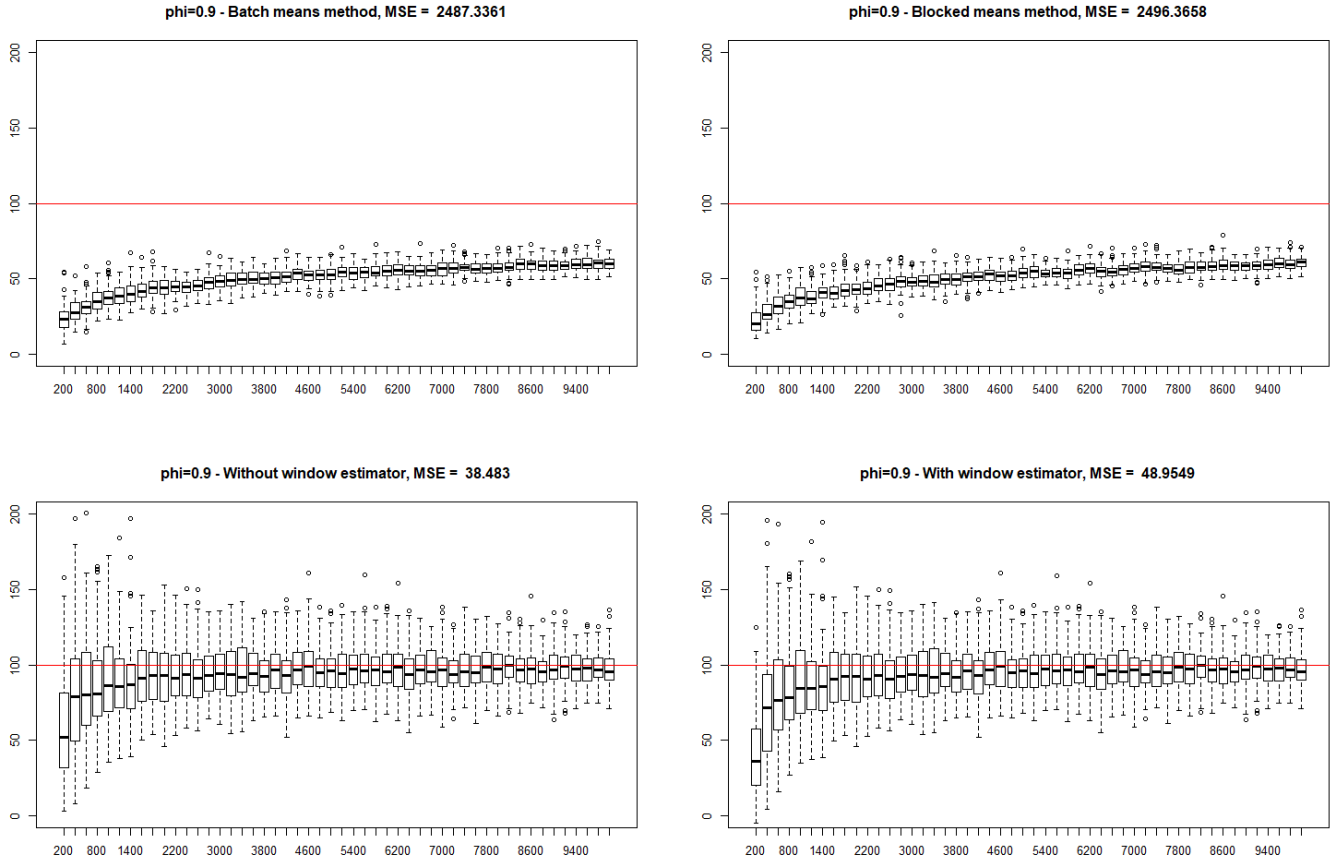

Figure 5.5.5: AR(1) process $X_t = \phi X_{t-1} + Z_t$, $Z_t \sim WN(0,1)$ and $\phi = 0.9$

Clearly, the batch means method and blocked means method consistently underestimate the value for asymptotic variance (for this value of b). The spread of the boxplots of the batch means method and the blocked means method is much smaller compared to the methods using window estimators. This implies that when the batch/blocked means methods have a good estimate, they also have very specific estimates. The methods using window estimators are performing, however, better in this case.

5.5.1 General Comparison between the methods

It is considerable to note that all methods are tested on an AR(1) process. This means that they might not perform well on different processes. For now, we do not take this into account. Note that we have very little focus on using an optimal batch/block size or finding appropriate window estimators for each process.

From the results of the plots which are based on an AR(1) process with $\phi = 0.9$ and $\phi = -0.5$, it became clear that the batch/blocked means method can consistently under/overestimate the value for asymptotic variance. This bias was already pointed out in section 5.2.1. Even when implementing an optimal batch size and comparing it to the methods using window estimators, the estimates are still biased. This is a disadvantage especially because the optimal batch size is for a lot of processes unknown. Therefore, based on these results, the methods which use window estimators give better results for these values of n (even when comparing it to the batch means method with optimal batch size). These methods were more accurate on all processes we tested, which means they might give positive results for a wide range of processes. Despite this, they have more potential as well, given the fact that we only used the window estimator based on the Tukey-Hanning Lag window. It is notable, however, that the difference between implementing or not implementing a window estimator is (based on the results of our experiment) not significant. It seems, in addition, that the method without using a window estimator is slightly better than the one which uses a window estimator.

Apart from the estimates, it is also important to take the implementation of the algorithms into account. All methods are relatively easy to implement. The blocked means method is, however, slower than the batch means method. The methods using window estimators are relatively fast. When using these methods for very large data sets, this is something important to take in to account.

Further Reading In the lecture notes of A. Van der Vaart ([16]), more methods for estimating the asymptotic variance such as: *Blockwise Bootstrap* and *estimation using autocorrelations*, are discussed. Also the potential of the methods using window estimators is significant and therefore interesting for further reading.

6 Conclusion

In this thesis, the goal is focused on finding a solution for the following problem:

For a function $f : \mathbb{R}^d \to \mathbb{R}$, how can we determine the approximation error $e(x^{(1)}, ..., x^{(n)}) := |E_{\hat{\pi}_n}[f(X)] - E_\pi[f(X)]|$, without knowing $E_\pi[f(X)]$?

This is stated in a very general way and it can be interpreted for methods such as Markov Chain Monte Carlo. In the first part of this thesis, the background for understanding the problem was given. In these topics, the complexity of probability distributions is illustrated and methods such as MCMC are elaborated. After extending the central limit theorem for dependent variables the concept of *asymptotic variance* came forward, which provides a solution for approximating the error of such an output. However, after diving into the theory of this concept (using the paper of A. van der Vaart [16]) it became clear that determining the asymptotic variance in an algebraic way can be very difficult or even impossible. Therefore, the solution of estimating the asymptotic variance numerically came forward.

For estimating this parameter, we analyzed three different methods: the batch means method, the blocked means method and estimation using window estimators. By applying these methods on four different AR(1) processes, a conclusion could be stated. The batch means method and the blocked means method had a very similar performance, which was striking because of the fact that the blocked means method takes dependence of the output more into account. Both these algorithms are relatively slow which might be a problem for very large data sets. Both the batch means and blocked means method are very stable but can be very biased as well. Finally, the method using window estimators is discussed. This method performs very well and has potential when using suitable window estimators. It is a fast algorithm and therefore preferable. Also, it performed well on all the processes we tested, which implies that these methods might be implemented on a wide range of processes.

In conclusion, for determining the approximation error $e(x^{(1)}, ..., x^{(n)}) := |E_{\hat{\pi}_n}[f(X)] - E_\pi[f(X)]|$, without knowing $E_\pi[f(X)]$, it is useful to calculate or estimate the asymptotic variance. A good method for this would be estimation using window estimators. When the asymptotic variance is estimated, the approximation error can be determined and used to build an interval which gives an idea of the quality of the output.

6.1 Further Research

Different topics for further research are already mentioned. When interested in rules for batching, more information can be found in Glynn and Iglehart (1990) [5]. A different aspect of this thesis which draws attention, is the performance of estimation for asymptotic variance using window estimators. We have compared only two different window estimators based on a specific experiment although it might be more interesting to look at them in a theoretical way as well. Neave writes about this in his article (Neave (1972) [8]). Also, testing different window estimators on this process might give more surprising results and is therefore a good topic for further research.

7 Appendix

7.1 Appendix of section 5.2.1

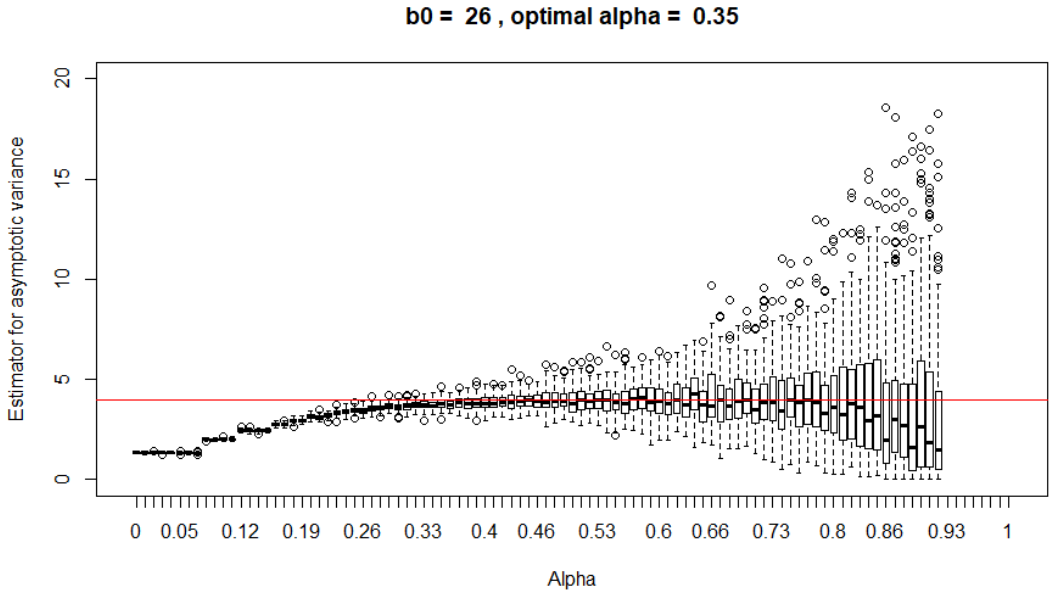

Figure 7.1.1: Larger view of the first part of figure 5.2.1, the red line is the theoretical value for the asymptotic variance

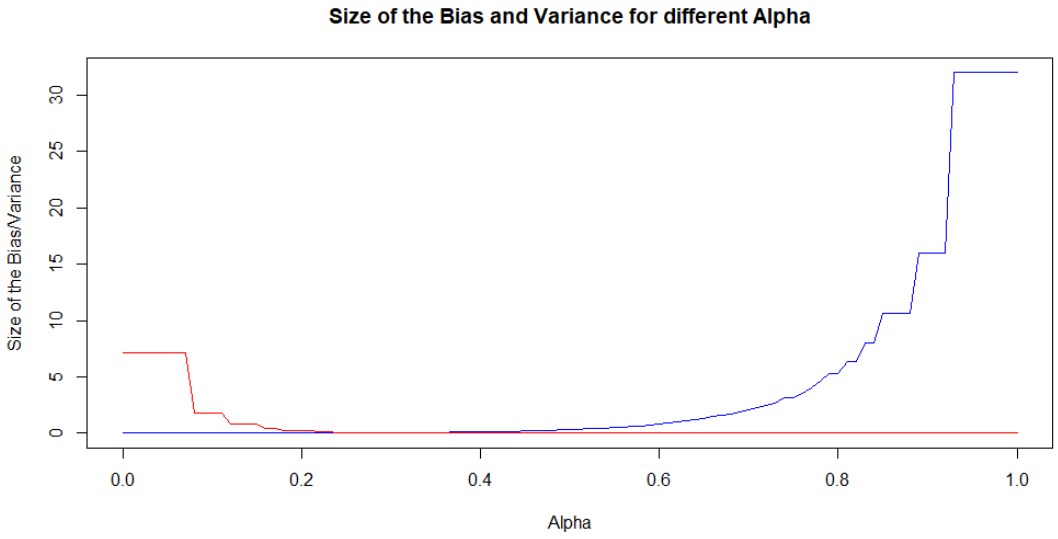

Figure 7.1.2: Larger view of the second part of figure 5.2.1, blue represents the variance, red represents the bias (squared)

7.2 Appendix of section 5.5

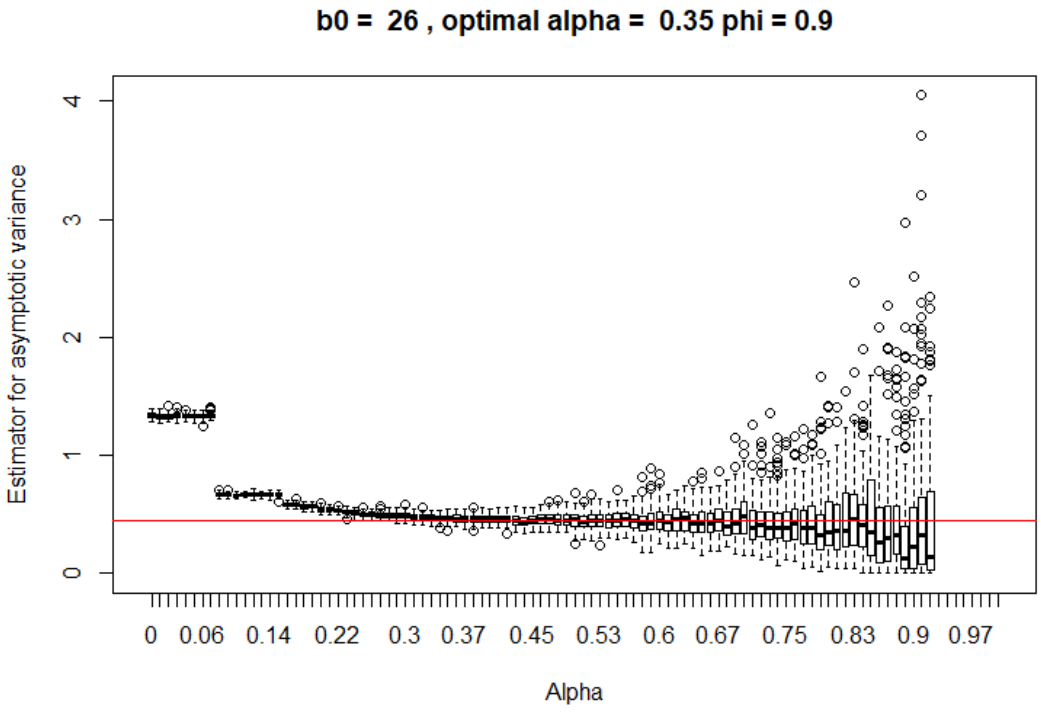

Figure 7.2.1: The batch means method applied on an AR(1) process for different batch sizes ($\phi = -0.5$)

References

[1] Christos Alexopoulos, George S Fishman, and Andrew F Seila. COMPUTATIONAL EXPERIENCE WITH THE BATCH MEANS METHOD. Technical report, 1997.

[2] Edward Carlstein and Others. The use of subseries values for estimating the variance of a general statistic from a stationary sequence. *The annals of statistics*, 14(3):1171–1179, 1986.

[3] John Fox. *Applied regression analysis and generalized linear models*. Sage Publications, 2015.

[4] Charles J Geyer. Practical markov chain monte carlo. *Statistical science*, pages 473–483, 1992.

[5] Peter W. Glynn and Donald L. Iglehart. SIMULATION OUTPUT ANALYSIS USING STANDARDIZED TIME SERIES*. 15, 1990.

[6] Frank Van Der Meulen. Lecture Notes Financial Time Series. Delft, 2018.

[7] Frank Van Der Meulen. Lecture notes mathematical data science. pages 1–24, 2019.

[8] Henry R Neave. A comparison of lag window generators. *Journal of the American Statistical Association*, 67(337):152–158, 1972.

[9] James R Norris. *Markov chains*. Number 2. Cambridge university press, 1998.

[10] Maurice Bertram Priestley. *Spectral analysis and time series*, volume 1. Academic press London, 1981.

[11] Adrian E Raftery and Steven M Lewis. [Practical Markov Chain Monte Carlo]: comment: one long run with diagnostics: implementation strategies for Markov Chain Monte Carlo. *Statistical science*, 7(4):493–497, 1992.

[12] John A Rice. *Mathematical statistics and data analysis*. Cengage Learning, 2006.

[13] Christian Robert and George Casella. *Monte Carlo statistical methods*. Springer Science & Business Media, 2013.

[14] Lynne Seymour, Peter J. Brockwell, and Richard A. Davis. *Introduction to Time Series and Forecasting.*, volume 92. 2006.

[15] Y. Sun. Time Series Lecture Notes. Technical report, University of Michigan, Ann Arbor, 2016.

[16] A.W. van der Vaart. Time Series. *Time Series*, 2010.

[17] G Alastair Young, Richard L Smith, and Others. *Essentials of statistical inference*, volume 16. Cambridge University Press, 2005.