Bayesian Pathway analysis in Epigenetics

by

Alan Wright

A thesis submitted to Plymouth University in partial fulfillment of the requirements for the degree of

DOCTOR OF PHILOSOPHY

School of Computing and Mathematics
Faculty of Science and Technology

Plymouth University

November 2012

Alan Wright

Bayesian Pathway analysis in Epigenetics

Abstract

A typical gene expression data set consists of measurements of a large number of gene expressions, on a relatively small number of subjects, classified according to two or more outcomes, for example cancer or non-cancer. The identification of associations between gene expressions and outcome is a huge multiple testing problem. Early approaches to this problem involved the application of thousands of univariate tests with corrections for multiplicity.

Over the past decade, numerous studies have demonstrated that analyzing gene expression data structured into predefined gene sets can produce benefits in terms of statistical power and robustness when compared to alternative approaches. This thesis presents the results of research on gene set analysis. In particular, it examines the properties of some existing methods for the analysis of gene sets. It introduces novel Bayesian methods for gene set analysis. A distinguishing feature of these methods is that the model is specified conditionally on the expression data, whereas other methods of gene set analysis and IGA generally make inferences conditionally on the outcome.

Computer simulation is used to compare three common established methods for gene set analysis. In this simulation study a new procedure for the simulation of gene expression data is introduced. The simulation studies are used to identify situations in which the established methods perform poorly.

The Bayesian approaches developed in this thesis apply reversible jump Markov chain Monte Carlo (RJMCMC) techniques to model gene expression effects on phenotype. The reversible jump step in the modelling procedure allows for posterior probabilities for activeness of gene set to be produced. These mixture models reverse the generally accepted conditionality and model outcome given gene expression, which is a more intuitive assumption when modelling the pathway to phenotype. It is demonstrated that the two models proposed may be superior to the established methods studied.

There is considerable scope for further development of this line of research, which is appealing in terms of the use of mixture model priors that reflect the belief that a relatively small number of genes, restricted to a small number of gene sets, are associated with the outcome.

Contents

List of Figures

List of Tables

Acknowledgements

Firstly, I would like to thank my Director of studies Dr Rana Moyeed and my second supervisor Professor William Henley for their guidance and support throughout this project. I would also like to express my gratitude to my colleagues both in the School of Computing and Mathematics and in the Centre for Health and Environmental Statistics for their support. Finally, I would like to acknowledge the support that Plymouth University has provided me during my PhD.

Author's declaration

At no time during the registration for the degree of Doctor of Philosophy has the author been registered for any other University award.

Relevant scientific seminars and conferences were regularly attended. One paper is currently in production to be submitted to a refereed journal.

Publications :

Wright, A. & Moyeed, R. & Henley,W. *The analysis of sets of genes: A comparison of existing methods.* (To be submitted)

Presentations given :

2011: *Pathway Analysis in Epigenetics.* Plymouth University, Plymouth, UK.

Conferences and courses attended :

2010: Research Students Conference, University of Warwick, Warwick, UK.

2010: *An introduction to Python.* Plymouth University, Plymouth, UK.

2010/2011: *Academy for PhD Training in Statistics:*

- Statistical Inference (January, University of Cambridge)
- Computational Statistics (January, University of Cambridge)
- Applied Stochastic Processes (June, University of Southampton)
- Computer Intensive Statistics (June, University of Southampton)

Word count for the main body of this thesis: 40,000

Signed: ____________________

Date: ____________________

Abbreviations

A	Adenine
AIC	Akaike Information Criterion
AUC	Area Under the Curve
BIC	Bayesian Information Criterion
BF	Bayes Factor
BGSA	Bayesian Gene Set Analysis
BN	Bayesian Network
BH	Benjamini-Hochberg
C	Cytosine
D	Deviance
DNA	Deoxyribonucleic acid
ES	Enrichment Score
FDR	False Discovery Rate
Fdr	Density based local False Discovery Rate
DIC	Deviance Information Criterion
fdr	Tail area based False Discovery Rate
FWER	Family wise error rate
GSA	Gene Set Analysis
GSEA	Gene Set Enrichment Analysis

G	Guanine
IGA	Individual gene analysis
KEGG	Kyoto Encyclopedia of Genes and Geneomes
MCMC	Markov Chain Monte Carlo
mRNA	Messenger Ribonucleic acid
MSigDB	Molecular Signatures Data Base
NES	Normalized Enrichment Score
PAUC	Partial Area Under the Curve
pFDR	Positive False Discovery Rate
RJMCMC	Reversible Jump Markov Chain Monte Carlo
RNA	Ribonucleic acid
rRNA	Ribosomal Ribonucleic acid
ROC	Receiver Operator Characteristic
SNP	Single-Nucleotide Polymorphism
tRNA	Transfer Ribonucleic acid
T	Thymine
U	Uracil

Chapter 1

Introduction

1.1 Overview

Epigenetics is the study of epigenetic inheritance, which is a set of reversible heritable changes in gene function or phenotype that occur without a change in DNA sequence. These changes can be induced spontaneously, in response to environmental factors, or in response to the presence of a particular allele, even if it is absent from subsequent generations. Epigenetics includes the study of effects that are inherited from one cell generation to the next, whether these occur in embryonic morphogenesis, regeneration, normal turnover of cells, tumors, or the replication of single celled organisms. Although there are many mechanisms and processes that come under the title of epigenetics, this thesis concentrates on the area of gene expression.

Projects in the area of gene expression are often referred to as Gene expression studies and generally attempt to link genetic contributors, along with certain suitable demographic, environmental and lifestyle information, to disease or phenotype.

The term gene expression describes the level of protein production from genes and is quantified by measuring the relative abundance of either protein or ribonucleic acid (RNA) from biological samples. The expression of genes is determined by many factors, both internal (inside the cell) and external (from both other cells and the outside environment) and so expression data is useful when modelling and predicting the occurrence or presence of disease in the here and now, rather than simply saying that a subjects genetic code suggests that at some point they could be susceptible to some disease. In analyzing genetic data it is necessary to have a reasonable understanding of the processes within the cell that result in such data. To this end the following section proceeds to give an introduction to the underlying biological processes behind gene expression and discusses how gene expression is measured and quantified.

1.2 DNA, the gene and gene expression

The human anatomy can be organized into many specific organs and tissues, all of which perform certain roles in the growth, repair and functioning of the body. All organs and tissues are made up of millions of cells, with groups of cells working in concert to a common task. Due to this, the cell is often referred to as the building block of life. Figure 1.1 shows the generalized structure of a cell.

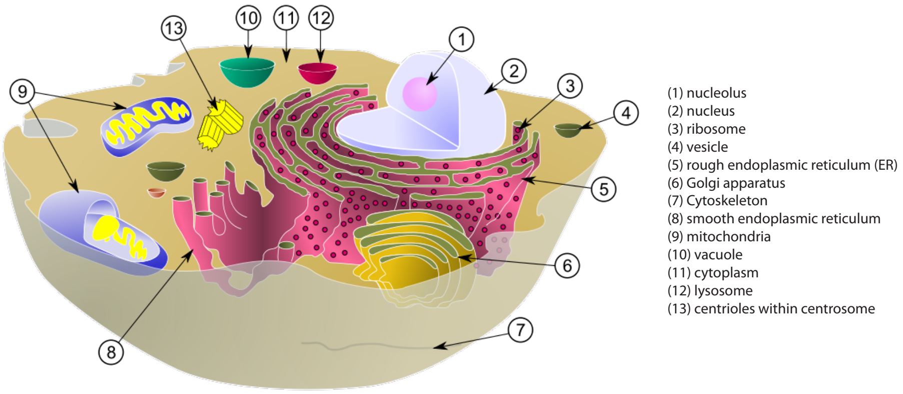


Figure 1.1: Simplified structure of a cell

The nucleus((2)) contains the cells chromosomes, which are made up of deoxyribonucleic acid (DNA), and is where the majority of DNA replication and ribonucleic acid (RNA) synthesis occurs.

DNA is a nucleic acid that contains the information used in the growth and repair of all living organisms. Eukaryotic organisms store most of their DNA inside the cell nucleus and some of their DNA in organelles. The main role of DNA is to store information and is often compared to a set of blueprints, as it carries the information needed to construct ribonucleic acid (RNA) and hence proteins. The DNA segments that carry this genetic information are called genes. Other DNA sequences have structural purposes, or are involved in regulating the use of this genetic information.

DNA is helical in structure and is constructed of two long polymers of units called nucleotides, with backbones made of sugars and phosphate groups joined by ester bonds. These two strands run anti-parallel of one another. Attached to each sugar is one of four molecules, these are Adenine (A), Cytosine (C), Guanine (G) and Thymine (T). These molecules are known as bases. Of these bases only A can bond to T and C to G, thus constraining the two stands to run anti-parallel of one another. It is the sequence of these four bases along the backbone that encodes information for the construction of RNA and hence proteins. Figure 1.2 shows a simplified segment of unravelled DNA.

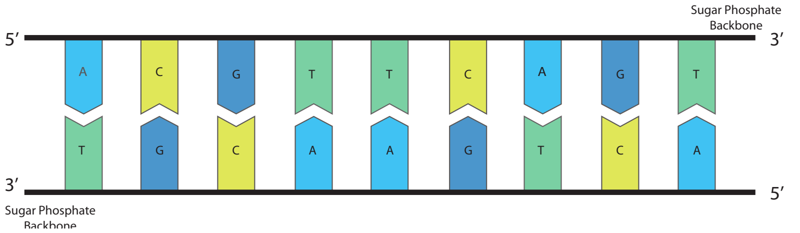


Figure 1.2: Simplified structure of DNA

RNA is a single strand of nucleotides and can be visualized as if it were one strand of the DNA double helix. The main differences between RNA and DNA are

- RNA is generally single stranded, whereas DNA is generally double stranded.
- DNA has a Deoxyribose sugar backbone, whereas RNA has a Ribose sugar backbone. As the name suggests, deoxyribose is the sugar ribose minus an oxygen atom.
- When RNA is transcribed from DNA, the base Thymine is replaced by the base Uracil (U).

There are several distinct types of RNA some of which will be outlined when required in the text to come.

A gene is a section of DNA that codes for RNA chains and hence proteins, which in turn control all processes within the cell. Genes describe and control the heredity in all living organisms and specific genes code for certain phenotypes, for example, there is a gene for eye colour and each variant of that gene is called an allele.

1.2.1 Transcription

Inside the cells nucleus some genes are active almost constantly, but others have to be turned on or off to cater for the needs of the cell or organism. Genes which are turned on or off need some signal to activate/ deactivate them. This is generally facilitated by either a signal from within the cell, for example from a regulatory gene whose job is to turn other genes on or off, or from a signal outside of the cell, for example some hormone. A gene is known as active when the process of transcription is carried out.

Transcription describes the process by which a DNA nucleotide sequence is transcribed into an RNA nucleotide sequence. The DNA sequence is copied by the enzyme RNA polymerase to produce a complimentary nucleotide RNA strand, called messenger RNA (mRNA), which then goes on to carry the genetic message to the protein synthesizing part of the cell. The stretch of DNA transcribed into an RNA molecule is called a transcription unit. A transcription unit that is complimentarily copied into RNA contains sequences which direct and regulate protein synthesis in addition to the coding sequence that is translated into protein itself. There is only one strand of DNA that is transcribed and it is called the template strand. The template strand provides the template for ordering the sequence of complimentary nucleotides in an RNA transcript. The other strand is known as the coding strand and is very similar to the created RNA strand with Uracil being substituted in place of Thymine. Transcription can be split into five stages; *pre-initiation, initiation, promoter clearance, elongation and termination*.

- *Pre-initiation* - RNA polymerase binds to the DNA and along with other enzymes it unravels a section of the DNA to create an initiation bubble. This enables the RNA polymerase access to the template strand of the given transcription unit.

- *Initiation* - The binding of RNA polymerase and hence the initiation of transcription is mediated by a collection of proteins called transcription factors. Only after certain transcription factors are attached to the promoter does the RNA polymerase bind to

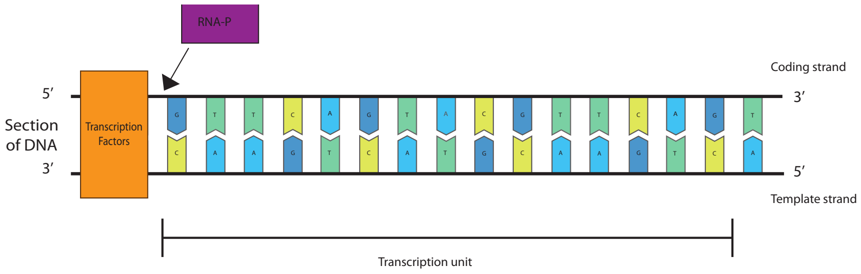


Figure 1.3: Simplified transcription initiation (RNA-P∼RNA polymerase)

it, forming a transcription initiation complex. This can be seen pictorially in Figure 1.3.

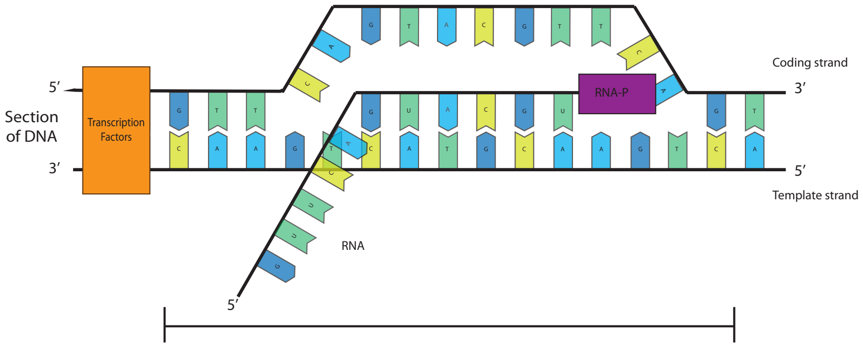


Figure 1.4: Simplified transcription elongation (RNA-P∼RNA polymerase)

- *Promoter clearance* - The RNA polymerase begins to move along the template strand, bonding bases to form the RNA strand. During the time after the first bond is synthesized, when the RNA polymerize must clear the promoter, it is common for the RNA transcript to be released, thus producing a truncated transcript. This occurrence is known as abortive initiation. Once the transcript is around 23 nucleotides long abortive initiation tends not to occur, and so Elongation can begin.

- *Elongation* - As transcription proceeds, RNA polymerase travels along the template strand and produces the complementary base to that on the template strand in order to create the complimentary strand of RNA. This is shown in Figure 1.4. RNA polymerase traverses the template strand from 3' to 5', however the coding strand is generally used as reference (as other than U being substituted for T it is very similar) so transcription is said to go from 5’ to 3’.

- *Termination* - Transcription termination in Eukaryotes is not well understood. It involves the cleavage of the new RNA transcript, followed by addition of A’s at the 3’ end by the process of polyadenylation, for which no template is needed. Figure 1.5 illustrates this.

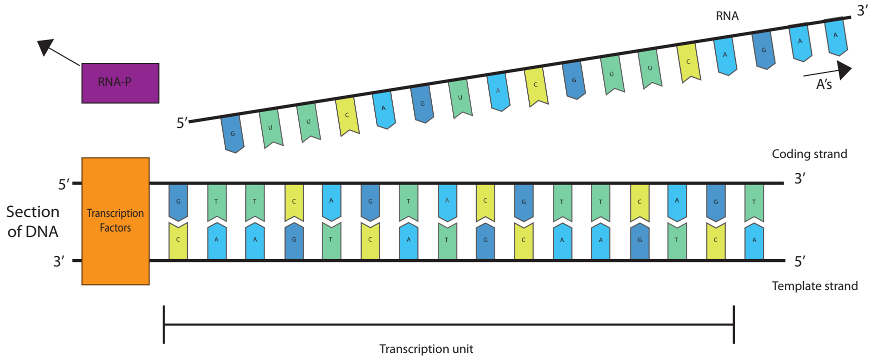


Figure 1.5: Simplified transcription termination (RNA-P∼RNA polymerase)

From this process a section of RNA that will code for a specific protein is produced.

1.2.2 Translation

Translation describes the ‘translation’ of RNA into protein. Before defining this process in detail it is useful to differentiate between the types of RNA involved in Translation:

- Messenger RNA (mRNA)- carries the information for the coding of a protein from the DNA to the ribosome. mRNA is formed by the process of Transcription, which is outlined above;

- Transfer RNA (tRNA)- Is a relatively small RNA molecule. Each tRNA molecule carries one specific amino acid, which it carries to the ribosome. tRNA contains a three base region called the anticodon that can base pair to a complimentary three base codon region on mRNA. Each type of tRNA molecule can be attached to only one type of amino acid and this is how different proteins are specified. tRNA is never translated into protein;

- Ribosomal RNA (rRNA)- There are several types of rRNA, and along with certain specific proteins they form a ribosome. As with tRNA, rRNA is never translated into protein.

Ribosomes are the components of cells that construct proteins. They are found in the cells cytoplasm and they use amino acids, tRNA and mRNA to produce proteins. Amino acids can be thought of here as the building block of proteins; they are molecules all that contain a certain group of elements and what is known as a side chain, they differ by what this side chain is constructed of.

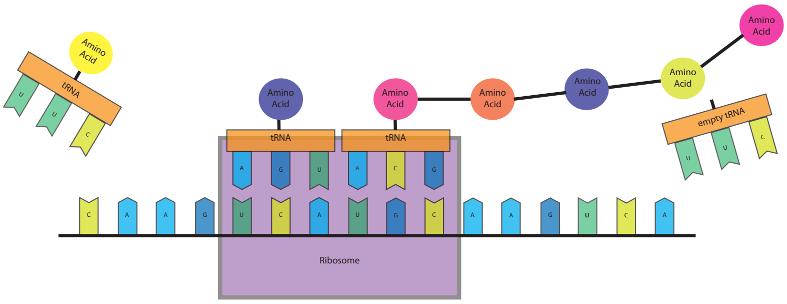


Figure 1.6: Simplified illustration of Translation.

In essence the ribosome translates the genetic code from the mRNA into proteins, and hence the name Translation. They do this by forming a chain of amino acids. This chain of amino acids, also known as a polypeptide, is constructed according to the sequence of bases in the mRNA. The mRNA binds to one active site of the ribosome, while tRNA's which carry one amino acid each, bind to another site. The amino acids are then bound together according to how the complimentary bases on the mRNA and tRNA fit together. This is the process known as translation , i.e. the ribosome translates the genetic information from the mRNA into proteins. Figure 1.6 illustrates this. Translation is part of the overall process of gene expression.

1.2.3 Gene expression and gene expression profiling

The process of gene expression is the primary mechanism by which an organisms genotype results in the phenotype; a gene is expressed and the products from this (such as proteins) give rise to the organisms phenotype, such as eye colour, hair colour, disease status etc.

The term gene expression describes the process by which information from the gene is transcribed into a functional gene product. There are essentially two types of functional gene products; coding RNA (mRNA) and hence proteins and non-coding, functional RNA's such as tRNA and rRNA. Genes are up or down regulated, or turned on or off in response to some stimulus; for example from a regulatory gene whose job is to turn other genes on or off, or from a signal outside of the cell, for example some hormone. Gene expression is a process which encompasses Transcription and Translation, along with some other modulated processes, such as the post translational modification of the protein product, whereby the proteins 'three dimensional' structure is formed. However, in the context of statistical analyses and methodologies it is mainly Transcription and Translation that are of interest, or more strictly the products of these processes, i.e. mRNA and protein.

1.2.4 Quantifying gene expression

The level of expression of a given gene can be measured by determining the number of mRNA molecules produced by that particular gene. A DNA microarray can be used to measure the amount and type of mRNA transcripts present in a sample of cells (blood or tissue). The microarray generally consists of a solid surface made of glass, silicone or nylon to which in specific positions, strands of DNA are attached, these sites are often referred to as probes. Other microarrays, for example Illumina, use microscopic beads instead of a solid platform. Microarrays quantify gene expression by measuring the hybridization of the mRNA from the sample under study to the DNA microarray. Each hybridization can be seen as a separate experiment at each probe on the microarray.

There are many ways in which the microarray is designed and in which hybridization is measured, some of which are discussed in detail by Parmigiani et al. (2003). Measurement of hybridization will not be discussed in detail in this thesis. Once data has been collected from a microarray, it must then be standardized in some way such that data is on a common scale and experimental error is minimal. This standardization is known as normalization. There is often a second level of standardization whereby expressions are transformed to follow a standard $N(0,1)$ distribution. This second level of standardization is generally carried out for ease of analysis.

1.3 Analysis of gene expression data

Modelling gene expression data presents two major problems. Firstly, we have the problem of identifying the group of genes from the thousands of possible candidates that are responsible for a given phenotype. Secondly, we have the problem of determining and/or interpreting how this group of genes affects phenotype.

A typical gene expression data set would consist of the expression of some large number of genes, N, being measured over some relatively small number of subjects, n, across a number of conditions, for example cancer versus non-cancer. Traditional genomics studies often focus on a gene by gene analysis; attempting to relate single genes to phenotype. There are considerations to take into account with such an approach, such as multiple hypothesis testing issues and reliability and interpretation of results Subramanian et al. (2005). It is often unpractical to overcome such problems.

An interesting and useful tool for investigating relationships between gene expression and phenotype is the analysis of sets of genes. There are many published methods for the analysis of gene sets, but prior to going into details of these methods, it is necessary to study the origins of such methods. Individual gene analysis (IGA), as coined by Nam and Kim (2008), is at the root of many if not all methods and models for the analysis of gene sets.

1.3.1 Individual gene analysis (IGA): A typical gene association study

Since the introduction of microarray technology in around 1995, it has been of great interest to identify differentially expressed genes with regard to phenotype or genotype and from this infer on biological processes within the cell. To this end many statistical methods have been developed. The most widely used approach, commonly known as individual gene analysis (IGA) is carried out as follows

1. Calculate some metric of effect of every gene on phenotype. For example t-test test statistic.
2. Calculate corresponding p-values, often using some multiple testing correction, and order from low to high.
3. Choose either some arbitrary cut off for p-values or some arbitrary number of top ranking p-values to determine a list of significant genes.
4. Used pathway annotation files to determine whether there are significantly large numbers of these significant genes in a certain pathway.

The first point of statistical interest is point 2, or more specifically in methods that control or correct for multiple testing. The second point of statistical interest is in determining if the top-ranking list of genes is overrepresented in particular gene sets. To this end, some of the methods that control and account for multiple hypothesis testing on such a scale are described. Following this point 4 is considered.

When performing such testing on large data there will be many false positive results simply by chance, and so considerations of how to control or account for these false positive results must be made. Traditionally, control of the family wise error rate was considered sufficient, however it can be shown that control of the family wise error rate (FWER) can be overly conservative when dealing with such large numbers of tests. More recently false discovery rate (FDR) procedures have been used to account for multiple testing on such large scales and have been used with much success, for example Efron et al. (2001), Efron and Tibshirani (2002) and Mootha et al. (2003). This section continues to introduce and describe some of the methods used to monitor and correct for false positive results when performing analyses such as IGA.

1.3.1.1 Family-wise error rate (FWER)

Testing a single hypothesis can result in two types of error

- Type-1 error: To falsely reject the null hypothesis.
- Type-2 error: To falsely fail to reject the null hypothesis.

This is demonstrated below in Table 1.1.

		Test Declared	
		Non-significant	Significant
Truth	H_0	Correct	Type-1 error
	H_1	Type-2 error	Correct

Table 1.1: Errors in hypothesis testing

Testing relies on control of the type-1 error rate. This is achieved by specifying the level, α, which is the probability of making a type-1 error. In practice the p-value, which is defined as the probability of observing something as or more extreme than the observed test statistic given that H_0 is true, is compared to α to determine whether to reject or fail to reject the null hypothesis. Suppose we took m sample sets, where each sample is comprised of n_1 subjects from a population with condition 1 and n_2 subjects from a population with condition 2. If a test is conducted between the two conditions on each of the m sets then we are testing m null hypothesis, $H_{01}, H_{02}, \ldots, H_{0m}$, and so by chance we will make some errors in our conclusions. Table 1.2 illustrates the sum of the results from the m tests.

		Test Declared		
		Non-significant	Significant	Total
Truth	H_0	U	V	m_0
	H_1	T	S	$m - m_0$
	Total	m-R	R	m

Table 1.2: Numbers of errors in testing m hypotheses.

As can be seen in Table 1.2 V is the number of type-1 errors and T is the number of type-2 errors. The family-wise error rate (FWER) is defined as the probability of making at least one type-1 error.

$$FWER = P[V \geq 1] \tag{1.1}$$

The FWER can be better defined in terms of if the null hypotheses are indeed true.

- The FWER under the complete null (FWEC), given by (1.2), is the probability that at least one type-1 error occurs given that all the null hypotheses are true.

$$FWER = P[V \geq 1 | H_{01}, H_{02}, \ldots, H_{0m} \text{ true}] \tag{1.2}$$

If the FWEC is less than or equal to α then the test procedure is said to control the FWER in the weak sense.

- The FWER under a partial null (FWEP), given by (1.3), is the probability that at least one type-1 error occurs given that a subset of the null hypotheses, call them H_0^s, are true.

  $$FWER = P[V \geq 1 | H_0^s \text{ true}] \tag{1.3}$$

  A test procedure is said to have strong control of FWER when the FWEP is less than or equal to α under all subsets of the null hypotheses. Strong control implies $FWEP \leq \alpha$ for all partial nulls.

Suppose we conduct m hypothesis tests $H_{01}, H_{02}, \ldots, H_{0m}$, and it is specified that each test be controlled at level α. So for any test, the probability of falsely rejecting the null hypothesis (or the type-1 error rate) is less than or equal to α, i.e.

$$P[\text{reject } H_{0i} | H_{0i} \text{ true}] \leq \alpha \qquad \text{for i=1,2,\ldots,m} \tag{1.4}$$

Assume that the tests are independent of one another. Then the probability of rejecting a test is independent of the outcomes of the other m-1 tests. Say we reject n null hypotheses, then the probability of falsely rejecting at least one null hypothesis is given by (1.5)

$$\begin{aligned} FWER &= P[\text{n} \geq 1 | H_{01}, H_{02}, \ldots, H_{0m} \text{ true}] \\ &= 1 - P[\text{n} = 0 | H_{01}, H_{02}, \ldots, H_{0m} \text{ true}] \\ &= 1 - \prod_{i=1}^{m} P[\text{fail to reject } H_{0i} | H_{0i} \text{ true}] \\ &= 1 - \prod_{i=1}^{m} (1 - P[\text{reject } H_{0i} | H_{0i} \text{ true}]) \\ &\leq 1 - \prod_{i=1}^{m} (1 - \alpha) \\ &\leq 1 - (1 - \alpha)^m \qquad (1.5) \end{aligned}$$

So for example if five tests were conducted at the 5% level, then the probability of making at least one type-1 error would be

$$1-(1-0.05)^5 = 0.23 \tag{1.6}$$

A widely used method to control the FWER when testing multiple hypotheses is Bonferroni adjustment. A brief outline of Bonferroni adjustment, which controls the FWER in a weak sense, follows.

The Bonferroni adjustment to control the problem outlined previously is very simple. If we require a significance level α then the Bonferroni adjustment would use (1.7)

$$\alpha' = \frac{\alpha}{m} \tag{1.7}$$

instead of α as the significance level for each individual test. If each test is controlled at the α' level then the overall FWEC will be controlled at the α level. The Bonferroni correction is based on the result that

$$P[\text{reject } H_{01}, \text{reject } H_{02}, \ldots, \text{reject } H_{0m}] \leq P[\text{reject } H_{01}] + P[\text{reject } H_{02}] + \ldots + P[\text{reject } H_{0m}] \tag{1.8}$$

with equality if the events are mutually exclusive. So if we set $P[\text{reject}H_{0i}] = \alpha'$, then $FWER \leq m\alpha'$.

In the context of gene expression studies the Bonferroni correction to p-values can be overly stringent and can often result in many false negatives. For example, say we had 20,000 probes and were carrying out a single gene analysis on cancer status; for a probe/gene to be significant at the 5% level using the Bonferroni correction method, a p-value of 2.5e-06 would have to be achieved. There is also the question of how to choose α' to minimize false positives and false negatives with reasonable justification. The use of the false discovery rate is more suitable in the context of gene expression studies.

1.3.1.2 False discovery rate (FDR)

In recent years, false discovery rate (FDR) methods have become synonymous with many high dimensional model selection procedures and multiple-hypothesis testing problems.

The FDR is defined as the ratio between the number of type-1 errors and the total number of tests that are significant, or alternatively as the proportion of rejected null hypotheses which are erroneously rejected. Suppose that it is required to test m null hypotheses, of which m_0 are true. This situation is summarized in the FWER section in Table 1.2. It is assumed that R is an observable random variable and S, T, U, V are unobservable random variables. Then the FDR is defined as

$$FDR = E\left[\frac{V}{R}\middle| R > 0\right] P[R > 0] \tag{1.9}$$

which is the expectation of the unknown and unobservable random variable

$$V/R \tag{1.10}$$

where $V/R \equiv 0$ if $R = 0$. There are two important properties of this error rate;

1. Suppose that all null hypotheses are true; $V = R$ and so

$$V/R = \begin{cases} 0 & \text{if } V = 0 \\ \\ 1 & \text{if } V \geq 1 \end{cases} \tag{1.11}$$

Then we have (1.12)

$$\begin{aligned} E[V/R] &= 0 \times \mathrm{P}[V = 0] + 1 \times \mathrm{P}[V \geq 1] \\ &= \mathrm{P}[V \geq 1] \\ &= \mathrm{FWER} \end{aligned} \tag{1.12}$$

so if all null hypotheses are true, the FDR and the FWER are equivalent. Therefore control of the FDR implies weak control of the FWER.

2. When it is not the case that all null hypotheses are true ($R > 0$), we have

$$\begin{aligned} E[V/R] &= E[V/R|V \geq 1]\mathrm{P}[V \geq 1] + E[0/R|V = 0]\mathrm{P}[V = 0] \\ &= E[V/R|V \geq 1]\mathrm{P}[V \geq 1] \end{aligned}$$

   and since $V < R$

$$E[V/R] < \mathrm{P}[V \geq 1] \tag{1.13}$$

   Therefore, in general the FDR is less than or equal to the FWER, which implies that if an approach controls the FWER then the FDR will be under control. However, if an approach controls only the FDR it can be more flexible and can lead to a gain in power.

Again lets suppose that we require m null hypotheses to be tested simultaneously, with the null hypotheses given by

$$H_{01}, H_{02}, \ldots, H_{0m} \tag{1.14}$$

With corresponding test statistics

$$z_1, z_2, \ldots, z_m \tag{1.15}$$

and p-values

$$p_1, p_2, \ldots, p_m \tag{1.16}$$

Benjamini and Hochberg (1995) introduced the following procedure to control the number of false discoveries;

- Order the p-values from smallest to largest, so that we have

$$p_{(1)}, p_{(2)}, \ldots, p_{(m)} \tag{1.17}$$

  and denote $H_{(0i)}$as the null hypothesis corresponding to the ordered p-value $p_{(i)}$

- Let

$$k = max\left(i : p_{(i)} \leq q\frac{i}{m} \right) \quad 0 < q < 1$$

then all null hypotheses corresponding to $p_{(1)}, \ldots, p_{(k)}$ are rejected. It can be shown that when the test statistics are independent this procedure controls $E[V/R]$ at level $\leq q$.

From this procedure comes the Benjamini-Hochberg rule (BH rule (1.18)), which is a simple correction of p-values;

$$p_i^{BH} = p_i \frac{m}{\text{order}(p_i)} \qquad i = 1, 2, \ldots, m \tag{1.18}$$

Where

- p_i^{BH} - The BH rule corrected p-value for the i^{th} ordered p value.

- order(p_i) - Equals m for the largest p-value and one for the smallest p-value and so on.

- If $\frac{\text{order}(p_i)}{m} \leq p_i$ then $p_i^{BH} = 1$

The BH rule can be formulated in terms of an empirical Bayes procedure. Suppose that it is small values of z that lead to rejection of H_0. Under the null hypothesis, for a particular test $P[Z \leq z|null] = p$. In general $P[Z \leq z] = order(z)/m$ can be used as an empirical cumulative distribution function for z. If we specify the prior prbability for H_0 being true as p_0 so that, in an obvious notation, $P[null] = p_0$, applying Bayes theorem we have

$$\begin{aligned} P[null|Z \leq z] &= \frac{P[Z \leq z|null]P[null]}{P[Z \leq z]} \\ &= \frac{p_0 p}{order(z)/m} \end{aligned}$$

For the i^{th} order statistic this becomes

$$P[null|Z \leq z_i] = \frac{p_0 p_i}{order(z_i)/m}$$

If we choose to declare significant if this is less than α, then $\frac{p_0 p}{i/m} < \alpha$ and so

$$p < \frac{i}{m}\frac{\alpha}{p_0}$$

this is the BH rule with $q = \alpha/p_0$.

The BH rule is popular due to its simplicity, however, it can often be a conservative estimator of the Fdr. One way to improve the BH rule is to use a more appropriate estimate of p_0. This leads directly to the well-known q-values that are commonly used for large scale multiple testing. There are various methods for the estimation of p_0, some of which are discussed by Storey (2002).

More generally, following Efron (2005), assume that the m tests are divided into the classes null, or non-null, occurring with prior probabilities p_0 and $p_1 = 1 - p_0$ respectively. Also assume that the densities of the test statistics are different depending upon class. This information is summarized below;

$$\begin{aligned} p_0 &= P[\text{null}] \quad \text{and null density} \quad f_0(z) \\ p_1 &= P[\text{non-null}] \quad \text{and non-null density} \quad f_1(z) \end{aligned} \qquad (1.19)$$

Where z is some test statistic. Also let F_0 and F_1 be the cumulative distribution functions (cdf's) for f_0 and f_1 respectively. Consider the two component mixture density.

$$f(z) = p_0 f_0(z) + p_1 f_1(z) \qquad (1.20)$$

The null density f_0 corresponds to the 'uninteresting' test statistics, whereas f_1 is an unspecified alternative density for the 'interesting' test statistics. The corresponding CDF is given by

$$F(z) = p_0 F_0(z) + p_1 F_1(z) \qquad (1.21)$$

The posterior probability of a case being null given that its z-value Z is less than some value z is given by (1.22)

$$\begin{aligned} \mathrm{Fdr}(Z) &= P[null|Z \leq z] \\ &= \frac{p_0 F_0(z)}{F(z)} \end{aligned} \tag{1.22}$$

Note that in the above only the left hand tail areas have been considered, but right hand tail areas and both tail areas could just as well be used.

The aim of most investigations involving large numbers of multiple comparisons is to identify a relatively small set of interesting non-null cases. Because of this, a large value of p_0 is often assumed such as $p_0 \geq 0.9$ Efron (2005) (Note: the BH rule uses $p_0 = 1$ along with the empirical form of F_1 for p-value corrections).

For practical implementation of the methodology described above p_0, f_0 and f_1 are required. In particular, a uniform distribution over $[0,1]$ would be appropriate if p values were used for z. There is a substantial body of literature on methods for obtaining p_0, f_0 and f_1 and this is a research topic in its own right. See, for example, Storey (2002), Efron and Tibshirani (2002), Storey (2003).

False discovery methods are increasingly being applied in the analyses of genetic data to account for such problems, some examples of its use are Efron et al. (2001), Efron and Tibshirani (2002) and Benjamini et al. (2009) gives a great overview of the method of false discovery rates as applied to SNP data. The FDR is also used in Gene Set Enrichment Analysis (GSEA) (See Chapter 2) as described in Mootha et al. (2003) and Subramanian et al. (2005). A particularly good piece of software for FDR analyses is the fdrtool package in R Strimmer (2008). Having outlined the key concepts, this methodology is not pursued further in this thesis, the main focus of which is a Bayesian analysis of gene expression data.

1.3.1.3 Tesing for enrichment

After any p-value corrections have been made and a list of top-ranking genes selected, it needs to be determined whether any particular gene sets are over-represented in that list. One such method to determine the over-representation of a gene set amongst the top-ranking differentially expressed genes is with the use of a 2×2 contingency table. The 2×2 contingency table is constructed as shown in Table 1.3

	Differentially expressed gene	Non-differentially expressed gene	Total
In gene set	n_{GD}	n_{GD^c}	n_G
Not in gene set	n_{G^cD}	$\mathrm{n}_{G^cD^c}$	n_{G^c}
Total	n_D	n_{D^c}	n

Table 1.3: 2×2 contingency table for assessing over representation of differentially expressed genes in a gene set. n_{GD} denoting the number of genes in the set that are differentially expressed and so on.

A number of different tests are proposed to test for independence in Table 1.3, for example the χ^2 test. Khatri and Draghici (2005) provide a detailed account of such IGA methods.

There are several limitations with such a gene-by-gene analysis even after such p-value corrections as described previously are made. These limitations are:

- After correction for multiple hypothesis testing there may not be any significant genes/probes. For example, suppose we had 20,000 probes and were carrying out a single gene analysis on cancer status; for a probe to be significant at the 5% level using the Bonferroni correction method, a p-value of 2.5e-06 would have to be achieved. This means that even with relatively large sample sizes effect sizes have to be large to obtain significance;

- There may be many significant probes/genes even after correction on p-values with no discernible biological theme thus making any interpretation of results very difficult;

- IGA assumes independence of gene effects on phenotype, which is not realistic and will increase false positive results;

- There may be a highly significant pathway effect, which a single gene analysis would not show and in some cases the single probes/genes in the pathway would not be significant on their own;

- The cut-off threshold for the inclusion of genes into the top ranking list is completely arbitrary, and can be shown to severely affect results Pan et al. (2005).

Clearly other methods and models, such as Pathway analysis, also known as gene set analysis, will be more useful for studies aiming at relating pathways to phenotype.

1.3.2 Pathway analysis

A typical frequentist analysis of sets of genes would take some measure of individual gene effect on phenotype, summarize these individual gene effects into some measure of gene set effect, then produce a p-value against some null hypothesis of no gene set effect. It is well known and accepted that genes and hence proteins function in concert to affect phenotype and so it makes sense to construct models to mimic such processes. Such models allow the experimenter to utilize previously accrued biological knowledge to provide a more biologically informed analysis. Analyzing functionally dependent groups of genes rather than single genes not only allows the experimenter an immediate interpretation to any positive results, but results from such analyses are generally more robust, Mootha et al. (2003), Efron and Tibshirani (2006). These functionally dependent groups of genes, known as gene sets, are defined on the basis of prior biological knowledge, such as location on the chromosome. There are many freely available databases defining gene sets, such as Gene Ontology, Ashburner et al. (2000) or the Kyoto encyclopedia of genes and genomes (KEGG), Ogata et al. (1999). The Molecular Signatures data base (MSigDB), which is available at `http://www.broadinstitute.org/gsea/msigdb/index.jsp`, gathers many of these pathway databases into one place, identifying some areas in which the databases of gene sets may be more applicable and more conducive to determining results relative to the study.

It seems as though there are, broadly speaking, three general ways of approaching a gene set analysis. These are defined by the hypothesis which they are testing. These hypotheses are:

- H_0^1: The genes in set s are at most as often differentially expressed as the genes not in set s;
- H_0^2: None of the genes in set s are differentially expressed;
- H_0^3: None of the gene sets are associated with phenotype.

H_0^1 and H_0^2 are defined by Goeman and Buhlmann (2007) as competitive and self contained null hypotheses respectively, Nam and Kim (2008) provide comprehensive details of the methods testing such hypotheses. Gene set enrichment analysis (GSEA), as introduced by Mootha et al. (2003) and developed by Subramanian et al. (2005) defines the third hypothesis.

GSEA is the most widely used method for the analysis of sets of genes and from here on the frequentist methods looked at will focus on GSEA and its derived methods. Several methods originate from GSEA, such as Gene Set Analysis (GSA) Efron and Tibshirani (2006) and those introduced by Jiang and Gentleman (2007) and they follow a general framework:

1. Suppose we have genes $g = 1, 2, \ldots, N$;
2. Begin with a pre-defined collection of gene sets **s**, $s = 1, 2, \ldots, k$;
3. Compute some test statistic z_g for all $g = 1, 2, \ldots, N$ genes;
4. Let $\mathbf{z}_s = (z_{s1}, z_{s2}, \ldots, z_{sm_s})$ be the vector of length m_s of test statistics for the genes $g = 1, 2, \ldots, m_s$ in set s;
5. Compute some gene set score $s(\mathbf{z}_s)$ for each of the k gene sets;
6. Create a null distribution for the gene set scores and test whether the true scores $(s(\mathbf{z}_s))$ are in the extremes of the null distribution.

The way in which this general outline differs from method to method is in the construction of z_g, $s(\mathbf{z}_k)$ and of the null distribution. Clearly, this is a somewhat ad-hoc approach, with no formal modelling involved. To set the problem of analyzing sets of genes into a formal, distribution based model would allow the experimenter to break down the real biological processes behind pathways into sensible, reasoned stages that could be represented by distributions. It would also allow us to discard these different null hypotheses in favour of Bayesian posterior probabilities. The use of Bayesian probabilities also removes the need for the definition of arbitrary cut offs on p-values to determine significant gene sets. There are several Bayesian approaches to the analysis of gene sets such as

Stingo et al. (2011) and Bayesian gene set analysis (BGSA) as presented by Shahbaba et al. (2011). However, there is still much scope for the development and application of Bayesian, model based approaches to the problem of analyzing sets of genes. The rich area of research of Bayesian Pathway analysis provides a great motivation for this thesis. Some established gene set analysis methods will be discussed in detail in Chapter 2.

In order to fully appreciate some of these Bayesian methods and to possibly develop new methods, a grounding in Bayesian statistics is required. The following section outlines such methods, along with other statistical methodology that will be used in this thesis.

1.4 An overview of other relevant statistical methods

The following section presents an outline to some of the more general statistical tools and concepts to be used in the thesis.

1.4.1 The Receiver Operator Characteristic (ROC) Curve

The ROC curve is a graph that allows us to asses the performance of a classification rule. Typical ROC analyses are concerned with a two by two classification table which results from cross classifying the true class of each object by its predicted class. For example a typical two by two situation might be that we construct a probabilistic model which gives the risk that a subject given their data has some disease of interest, then our classification rule c(x) uses some threshold T=t to classify whether a subject is in the disease class, call it c_1, or the unaffected class, call it c_2 due to the computed probability from the model, i.e. a subject with probability $p > t$ is classified as being in c_1 and conversely a subject with probability $p \leq t$ is classified as being in c_2. Then the possible classifications are demonstrated in Table 1.4 More generally, say we get some score s from our model for each subject, which again is classified due to some threshold. The joint probabilities of these classifications are shown below in Table 1.5

There are different ways in which the performance of a classification rule can be assessed due to how the above joint probabilities are summarized. In an ROC analysis

		Classified	
		c_1	c_2
Truth	c_1	True positive (tp)	False negative (fn)
	c_2	False positive (fp)	True negative (tn)

Table 1.4: The cross classification of the true class of an object by its predicted class in a two by two classification situation.

		Classified	
		c_1	c_2
Truth	c_1	$P[s > T, c_1]$	$P[s \leq T, c_1]$
	c_2	$P[s > T, c_2]$	$P[s \leq T, c_2]$

Table 1.5: Probabilities of cross classification of the true class of an object by its predicted class in a two by two classification situation.

these probabilities are summarized as

- $P[s > T|c_1]$ - This is known as the true positive rate or detection rate.
- $P[s > T|c_2]$ - This is known as the false positive rate.
- $P[c_1]$, $P[c_2]$ - marginal probabilities of belonging to either class.

Then the ROC curve is constructed by plotting the true positive rate against the false positive rate with varying threshold T. The idea behind the ROC curve is to provide a summary of the performance of a classifier over the whole range of possible classification thresholds. The ROC curve will generally be a continuous curve lying in the upper triangle of the plot area. The closer to the upper left hand corner the better the performance of the classifier and the closer to a straight line through (0,0), (1,1) the worse the performance of the classifier.

Now, if the two populations are identical then there is equal probability of classifying a subject into either c_1 or c_2 i.e. $P[s|c_1] = P[s|c_2]$, so as t varies the true positive rate will always equal the false positive rate, and therefore the ROC curve will be a straight line through the origin with gradient one. This line is known as the chance diagonal and is often added to ROC plots to see how different classification is from completely random. Alternatively, if there is a complete separation between the two groups, then for all t the true positive rate will be one and the false positive rate will be zero and the ROC curve will follow the x and y axes.

1.4.1.1 Area under the ROC curve (AUC)

The AUC is defined as the area under the ROC curve. Lets consider AUC in the two extreme cases of an ROC curve;

- Perfect separation (where the ROC curve is a line from x=0, y=0 to x=0, y=1 to x=1, y=1)-Then the AUC is the area of a one by one square, i.e. AUC=1.
- Random allocation (ROC curve is a line through (0,0) with gradient one)-Then the AUC is the area of a right angled triangle with height one and width one, i.e. AUC=0.5.

So, the upper bound of the AUC is 1 and the lower bound of the AUC is 0.5; the higher the AUC the better the classifier. An intuitive interpretation of AUC is that the AUC is the average tp rate taken over all possible fp rates. Another useful interpretation is that if s_{c_1} and s_{c_2} are scores given to randomly, independently chosen subjects from c_1 and c_2 then

$$\mathrm{AUC} = P[s_{c_1} > s_{c_2}] \tag{1.23}$$

1.4.2 Bayesian Statistics

The Bayesian approach to statistical inference draws conclusions about model or population parameters from data. Bayesian inference differs from the frequentist approach in that inference is based upon $f(\theta|x)$ rather than $f(x|\theta)$. In essence parameters are considered random variables, rather than fixed quantities.

Say we have parameter θ which we wish to make inference about and a likelihood function $f(x|\theta)$ which measures the probability of observing data x under parameter values θ, also say we have some prior knowledge of θ, $f(\theta)$, then we can obtain a full probability model

$$f(x,\theta) = f(x|\theta)f(\theta) \tag{1.24}$$

then having observed data x then the conditional distribution of θ can be determined using Bayes theorem

$$f(\theta|x) = \frac{f(x|\theta)f(\theta)}{\int f(x|\theta)f(\theta)d\theta} \tag{1.25}$$

The conditional distribution $f(\theta|x)$ is known as the posterior distribution and is at the heart of all Bayesian inference. In many cases $\int f(x|\theta)f(\theta)d\theta$ is intractable and by far the most popular techniques to overcome such problems are Markov chain Monte Carlo (MCMC) methods.

1.4.3 Markov chain Monte Carlo (MCMC)

In many modern applications there is the need to integrate over highly dimensional probability models in order to make inference about model parameters. Markov chain Monte Carlo (MCMC) techniques provide solutions to such problems and therefore give great scope for realistic, complex statistical modelling. Essentially, MCMC uses Markov chains to implement Monte Carlo integration.

1.4.3.1 Monte Carlo integration

Monte Carlo integration evaluates $E[f(X)]$ by drawing samples $X_t, t = 1, 2, \ldots, n$ from $\pi(\cdot)$ and evaluating $E[f(X)]$ by

$$E[f(X)] \approx \frac{1}{n}\sum_{t=1}^{n} f(X_t) \tag{1.26}$$

In essence the population mean is estimated by a sample mean. The sample, X_t, can be generated in any way which draws samples from $\pi(\cdot)$. One specific way of doing this is by a Markov chain with $\pi(\cdot)$ as its stationary distribution.

1.4.3.2 Markov chains

A Markov chain is a sequence of random variables $X_0, X_1, X_2, \ldots$ whereby at any time $t \geq 0$ the next state in the chain X_{t+1} is sampled from a distribution $P(X_{t+1}|X_t)$. $P(\cdot|\cdot)$ is known as the transition kernel of the chain. In essence X_{t+1} depends on X_t only and does not depend on further history of the chain.

Subject to certain regularity conditions, the chain will eventually forget its starting value and $P(X_t|X_0)$ will converge to some unique stationary distribution, independent of t or X_0. As t increases the sampled points will look more and more like dependent samples from the stationary distribution.

1.4.3.3 The Metropolis Hastings algorithm

The aim of the Metropolis-Hastings algorithm is to build a Markov chain on X starting at some initial value X_0. For the Metropolis-Hastings algorithm, at each state X_t, the next state X_{t+1} is chosen by sampling a candidate point Y from some proposal distribution. The candidate point Y is then selected with probability $\alpha(X_t, Y)$

$$\alpha(X_t, Y) = min\left(1, \frac{\pi(Y)q(X_t|Y)}{\pi(X)q(Y|X_t)}\right) \tag{1.27}$$

If Y is accepted, then the next state in the chain becomes $X_{t+1} = Y$, if not then the next state becomes $X_{t+1} = X_t$. As the target distribution $\pi(\cdot)$ only enters the algorithm through the ratio $\pi(Y)/\pi(X)$ then knowledge of the posterior distribution only up to a normalizing constant is required.

Essentially the Metropolis-Hastings algorithm works as follows There are many possible choices for the proposal distribution q. Two of the most common choices are outlined below.

Algorithm 1.1 The Metropolis-Hastings algorithm

```
1. Arbitrarily select some starting value $X_0$
2. Sample a candidate point $Y$ from some proposal distribution $q(Y|X_t)$
3. Compute the acceptance probability $\alpha(X_t, Y)$ using (1.27)
4. Sample u from a uniform distribution $U(0,1)$.
5. If $u \leq \alpha(X_t, Y)$ then $X_{t+1} = Y$, else $X_{t+1} = X_t$
6. Increment t and return to step 2.
```

1.4.3.3.1 1. The Independence sampler For all X, let $q(X_t, Y) = q(Y)$ for some probability density $q(\cdot)$. In essence the proposal density is independent of the current value X_t. This idea is similar to rejection sampling, in that we sample from one probability density

$q(\cdot)$ and then accept or reject the samples as coming from the probability density $\pi(\cdot)$. This method works well when we can find a simple probability density $q(\cdot)$ which closely approximates $\pi(\cdot)$. In this case the acceptance probability can be written

$$\alpha(X_t, Y) = min\left(1, \frac{w(Y)}{w(X_t)}\right) \tag{1.28}$$

where $w(X_t) = \pi(X_t)/q(X_t)$ denotes the relative weights of the two densities at x.

1.4.3.3.2 2. Random-walk Metropolis Suppose that we are currently at X_t. In stationarity it is likely that X_t has high posterior probability. Random-walk Metropolis, therefore, samples proposal values from a proposal density that is centered about the current value X_t. If the proposal density is symmetric about X_t then the acceptance probability can be written as follows.

$$\alpha(X_t, Y) = min\left(1, \frac{\pi(Y)}{\pi(X_t)}\right) \tag{1.29}$$

1.4.3.4 The Gibbs sampler

The Gibbs sampler obtains samples from the multivariate distribution $\theta = (\theta_1, \theta_2, \ldots, \theta_n)$ by successively and repeatedly simulating from the conditional distributions of each component given the other components. In general the Gibbs sampler algorithm is as follows in Algorithm 1.2

Algorithm 1.2 The Gibbs sampler algorithm

1. Initialize with $\theta = (\theta_1^{(0)}, \theta_2^{(0)}, \ldots, \theta_n^{(0)}$.
2. Simulate $\theta_1^{(1)}$ from the full conditional $\theta_1^{(1)} | \theta_2^{(0)}, \ldots, \theta_n^{(0)}$.
3. Simulate $\theta_2^{(1)}$ from the full conditional $\theta_2^{(1)} | \theta_1^{(1)}, \theta_3^{(0)}, \ldots, \theta_n^{(0)}$.
4. Simulate $\theta_n^{(1)}$ from the full conditional $\theta_n^{(1)} | \theta_1^{(1)}, \theta_2^{(1)}, \ldots, \theta_{n-1}^{(1)}$.
5. Iterate.

1.4.3.5 Burn in

Typically, the starting value for a Markov chain will not be drawn from its stationary distribution. Therefore burn in is required such that the chain can converge to its stationary distribution before samples from the chain can be used. The length of burn in depends on the starting value of the chain, the rate of convergence of the chain and how similar the proposal distribution is to the stationary distribution. There is no hard and fast tool for determining burn in and visual inspection of trace plots is still the the most commonly used method, however, this method does have its drawbacks, for example in multi modal distributions.

1.4.3.6 Monitoring Convergence

A general approach to monitoring the convergence of Markov chains, as introduced by Gelman and Rubin (1992), is based upon identifying whether the chain has 'forgotten' its starting point. This can be implemented by comparing several parallel sequences initiated at different starting values and determining whether they are distinguishable.

The most obvious and widely used approach to convergence assessment is to look at trace plots of the given sequences. However, this can be misleading in multi-modal distributions and some prefer a more quantitative approach to convergence monitoring.

Gelman and Rubin (1992) introduce a quantitative approach to convergence monitoring based upon the Analysis of Variance (ANOVA), whereby approximate convergence can be diagnosed when the within chain variance of several parallel sequences, initiated at different starting points, is no greater than the between chain variance. Say we have m parallel sequences, each of length n with realizations X_{ij} $i = 1,2,\ldots,m, \quad j = 1,2,\ldots,n$ then the between chain variance is given by

$$B = \frac{n}{m-1}\sum_{i=1}^{m}(\bar{X}_{i\bullet} - \bar{X}_{\bullet\bullet})^2 \tag{1.30}$$

where

$$\bar{X}_{i\bullet} = \frac{1}{n}\sum_{j=1}^{n} X_{ij} \qquad \text{and} \qquad \bar{X}_{\bullet\bullet} = \frac{1}{m}\sum_{i=1}^{m} X_{i\bullet}$$

and the within chain variance,W, is given by

$$W = \frac{1}{n}\sum_{i=1}^{m} s_i^2 \tag{1.32}$$

where

$$s_i = \frac{1}{n}\sum_{j=1}^{n}(X_{ij} - X_{i\bullet})^2 (1.33)$$

B (1.30) contains a factor of n because it is based on the variance of the within sequence means, $\bar{X}_{i\bullet}$, each of which is an average of n values. From the two variance components (1.30) and (1.32), we construct two estimates of the variance of X in the target distribution.

1. $$\hat{\mathrm{var}}(X) = \frac{n-1}{n}W + \frac{1}{n}B \tag{1.34}$$

   is an estimate of the variance if the starting points of each chain were drawn from the target distribution, i.e. it is an estimate of the variance that is unbiased under stationarity. This will generally be a conservative estimate (over estimate) under the more realistic assumption that the starting points are over-dispersed.

2. The within chain variance, W (1.32), is taken as the second estimate of variance. The within chain variance, for finite n, will be an underestimate of the variance of X as individual chains will not have had time to cover all of the target distribution.

Note that as $n \to \infty$ both $\hat{\mathrm{var}}(X)$ and W will approach $\mathrm{var}(X)$, but from different directions.

The convergence of Markov chains can be monitored by calculating the ratio between the estimated upper and lower bounds for the standard deviation of X, coined the 'estimated potential scale reduction factor'

$$\sqrt{\hat{R}} = \sqrt{\frac{\hat{\mathrm{var}}(X)}{W}} \tag{1.35}$$

As simulation converges, the potential scale reduction factor decreases to 1 and the parallel chains are overlapping. If $\hat{R}$ is large then it is wise to run the chains for longer.

1.4.4 Bayesian Model Comparison

Using the MCMC techniques outlined above, posterior samples for many complex models can be obtained and convergence can be assessed. However, it is often desirable to compare models and assess model fit and adequacy. Below are discussed two methods for Bayesian model comparison; the Deviance information Criterion (DIC) and Bayes Factor (BF).

1.4.4.1 Deviance information Criterion (DIC)

The DIC is a hierarchical modelling generalization of the Akaike information criterion (AIC) and Bayesian information criterion (BIC). It can be particularly useful in the comparison of Bayesian models where posterior distributions of the models are obtained by MCMC simulation.

It is first useful to define deviance. Deviance (D) can be used as a measure of model fit and can be defined for the likelihood $f(x|\theta)$, as

$$D(\theta) = -2logf(x|\theta) \tag{1.36}$$

where x are the data and θ are unknown parameters. The posterior mean deviance is often used as a measure of fit for Bayesian models, where the posterior mean deviance $\bar{D}$ is given by

$$\bar{D} = E[D(\theta)] \tag{1.37}$$

This is a particularly straightforward quantity to calculate within MCMC steps and provides an accurate measure of model fit. However, there will be an infinity of models that fit the data equally well, hence the true model will be indistinguishable from an infinity of correct models. We want the most simple correct model i.e. the model with the least number parameters. Therefore some measure of model complexity is needed to trade off against $\bar{D}$

If we define $\hat{\theta} = E[\theta|x]$, then a model complexity term can be defined as

$$p_D = E_{\theta|x}[D] - D(E_{\theta|x}[\theta]) \tag{1.38}$$

which is essentially the posterior mean deviance minus the deviance evaluated at the posterior mean of the parameters.

Then the DIC is defined as

$$DIC = \bar{D} + p_D \tag{1.39}$$

models with smaller DIC are better supported by the data.

1.4.4.2 Bayes Factors

Suppose we wish to compare how models M_1 and M_2 with parameters θ_1 and θ_2 respectively fit data x. Marginal distributions of x can be found by integration

$$f(x|M_i) = \int f(x|\theta_i, M_i) f(\theta_i) d\theta_i \qquad i = 1,2 \tag{1.40}$$

where $f(\theta)$ is the prior density for θ. Then using Bayes theorem and defining prior densities for the given model, posterior model probabilities can be obtained

$$f(M_i|x) = \frac{f(x|M_i)f(M_i)}{f(x)} \qquad i = 1,2 \tag{1.41}$$

Then the Bayes Factor (BF) is given by

$$\begin{aligned} BF &= \frac{f(M_1|x)/f(M_2|x)}{f(M_1)/f(M_2)} \\ &= \frac{f(x|M_1)}{f(x|M_2)} \end{aligned} \tag{1.42}$$

Which can be thought of as the ratio of the posterior odds of M_1 to the prior odds of M_1. Assuming the models are *a priori* equally probable (i.e. $f(M_1) = f(M_2) = 0.5$) then we have

$$BF = \frac{f(M_1|x)}{f(M_2|x)} \tag{1.43}$$

Considering the case where both models have the same parameterization, i.e. $\theta_1 = \theta_2 = \theta$ then the Bayes Factor is simply the likelihood ratio between the two models. Values of $B > 1$ indicate more support for M_1 under the data and conversely values of $B < 1$ indicate more support for M_2 under the data.

There are other such Bayesian model comparison metrics, such as the Bayesian Information Criterion (BIC). Although these model comparison measures are useful, it is preferable to have an automatic model selection procedure. Reversible jump Markov chain Monte Carlo (RJMCMC) provides not only a model comparison measure, in the form of a posterior model probability, but also performs model selection. RJMCMC is outlined in detail in Chapter 5.

1.5 Outline

The rest of the thesis is as follows. Chapter 2 is devoted to setting the scene and proceeds to discuss in detail and critically appraise three of the more prominent methodologies for the analysis of gene sets, these being GSEA, Gene Set Analysis (GSA) and Bayesian Gene Set Analysis (BGSA). Through discussion it can be seen that all three methods are preferable to IGA. It can also be seen, however, that there are some flaws with the methods, in particular where the methods do not overcome some of the problems with IGA.

Chapter 3 presents a thorough simulation study comparing GSEA, GSA and BGSA. Whereby data simulation models are first discussed in order to decide upon the data simulation model that produces realistic data. Several scenarios concerning ‘activeness’ of gene set are then defined in such a way that aims to highlight the relative strengths and weaknesses of GSEA, GSA and BGSA. The methods are then applied to such data and results discussed. A second, more practically based appraisal can then be made of the three methods.

Chapter 4 proposes a hybrid Bayesian/ frequentist model that builds upon issues raised in Chapters 2 and 3. Chapter 4 also develops a MCMC algorithms to fit this model.

Chapter 5 introduces and defines two Bayesian models for the analysis of gene sets, these being Bayesian Analysis of Gene Sets (BAGS) and Multivariate Bayesian Analysis of Gene Sets (MVBAGS). Chapter 6 compares the introduced methods to GSEA, GSA and BGSA in some case studies based upon data simulated according to the approach outlined in Chapter 3.

Chapter 7 applies the BAGS model to a real data set. This data set, known as the p53 data set, is freely available at `http://www.broadin stitute.org/gsea/datasets.jsp`. Subramanian et al. (2005), Efron and Tibshirani (2006) and Shahbaba et al. (2011) present results of analyses of the p53 data set, and we compare these results to those obtained by applying BAGS to the data.

Finally Chapter 8 discusses the work presented in this thesis and indicates future work in the areas discussed.

Chapter 2

Some existing methodologies for gene set analysis

2.1 Introduction

A typical gene expression data set would consist of the expression of some large number of genes, *N*, being measured over a small number of subjects, *n*, across a number of experimental conditions, for example cancer versus non-cancer. Traditional genomics studies such as individual gene analysis (IGA) often focus on a gene by gene analysis; attempting to relate single genes to phenotype. There are considerations to take into account with such an approach, such as multiple hypothesis testing issues and reliability and interpretation of results Subramanian et al. (2005). It is often unpractical, if not impossible to overcome such problems, hence the need for new approaches to relate gene expression to phenotype.

It is well known and accepted that genes and hence proteins function in concert to affect phenotype. Analyzing functionally dependent sets of genes rather than single genes not only allows the experimenter an immediate interpretation to any positive results, but results from such analyses are generally more robust, Mootha et al. (2003), Efron and Tibshirani (2006). These functionally dependent groups of genes, known as gene sets, are defined on the basis of prior biological study, such as location on the chromosome, and there are several freely available databases defining gene sets, for example `http://www.broadinstitute.org/gsea/msigdb/index.jsp`.

The general theme of methodologies for the analysis of sets of genes is as follows:

- Begin with some pre-defined gene sets;
- Determine some measure of gene effect within gene set;
- Summarize individual within gene set gene effects with some measure encompassing information from these gene effects;
- Determine a p-value for evidence against a null hypothesis of no effect. Alternatively determine a posterior probability of activeness of gene set.

There are many existing methodologies that are used for the analysis of gene sets such as Subramanian et al. (2005); Efron and Tibshirani (2006); Jiang and Gentleman (2007); Bauer et al. (2010); Isci et al. (2011); Shahbaba et al. (2011) and Stingo et al. (2011) amongst others. Nam and Kim (2008) present a useful literature review on many of the existing frequentist approaches to the analysis of gene sets alongside a rather limited simulation study, whilst Hung et al. (2012) provide a thorough description and investigation of GSEA.

This thesis focuses on the methods of GSEA, GSA and BGSA. GSEA brought gene set analysis methods to the forefront and was chosen as it is the most widely cited gene set analysis method to date. One such paper citing GSEA is that by Efron and Tibshirani (2006), who generalize the GSEA framework and introduce a new methodology for the analysis of gene sets that is developed under this general framework, coined GSA. Finally, BGSA was chosen for two reasons, firstly as it represents some of the latest methodology; first published in 2011. Secondly BGSA was chosen as it is one of the few available Bayesian methodologies for the analysis of gene sets.

Publications for the three methods above report the results for the analysis of a gene expression data set on 50 cancer cell lines across 4486 genes, whereby 33 of the cell lines have a p53 mutation and the remaining 27 cell lines are wild type. This freely available p53 data set has become somewhat of a benchmark for comparison when presenting new methods or models for the analysis of sets of genes. The analyses of the p53 data set presented in the three papers will be discussed in Chapter 7.

This chapter proceeds to introduce and describe the methodologies of GSEA, GSA and BGSA in chronological order. Section 2.5 goes on to discuss the relative merits and possible problems with the above methods.

2.2 Gene set enrichment analysis (GSEA)

GSEA aims to determine whether the members of a pre-defined set of genes, s, is randomly distributed across L - a list of genes ordered based upon their correlation with the given outcome call it r_g $g = 1, 2, \ldots, N$ - or predominantly found near the top or bottom of the list, thus indicating a relationship between the gene set s and the outcome.

The process of GSEA is carried out as follows

- An Enrichment score (ES) is calculated, which will indicate if the genes in the set s are found primarily near the top or the bottom of the ordered list L. The ES is a normalized Kolmogorov-Smirnov statistic. Suppose we have the ordered list of genes L, which contains N genes and we have a given gene set s, which contains m_s genes, then we evaluate the fraction of genes in s ('hits') weighted by the magnitude their correlation and the fraction of genes not in s ('misses') present up to a given position i in L

$$\begin{aligned} P_{hit}(s,i) &= \sum_{g \in s, g \leq i} \frac{|r_g|^p}{\sum_{g \in s} |r_g|^p} \\ P_{miss}(s,i) &= \sum_{g \notin s, g \leq i} \frac{1}{N - m_s} \end{aligned} \tag{2.1}$$

  where p is taken to be 1 in Subramanian et al. (2005) and p=0 in Mootha et al. (2003). The ES is defined as the maximum deviation from zero of the cumulative sum of $P_{hit} - P_{miss}$. This is essentially a two sample Kolmogorov-Smirnov test statistic.

- The significance of the ES is then estimated by permuting the outcome and recalculating the ES for the new outcome many times. This provides a null distribution for the ES, from which a p-value for the observed ES can be calculated.

- The final step of GSEA is to adjust for multiple hypothesis testing. This is achieved by first normalizing the ES for each gene set, thus accounting for the size of each set m_s. The normalized ES is known as the normalized enrichment score (NES). The false discovery rate (FDR) (as described in Chapter 1) is then calculated for each NES.

Subramanian et al. (2005) present results for GSEA on six real gene expression data sets, the $p53$ data set being one of them. However, no simulation study is implemented. Therefore, although we are presented with results from GSEA we have no indication to the performance of the method in identifying significant pathways of known effect.

2.2.1 GSEA in practice

The website `http://www.broadinstitute.org/gsea/` provides many freely available resources to aid researchers in performing GSEA, along with tutorials, guides and data on which to practice the method. There are several ways provided for the implementation of GSEA. These are grouped under two main headings GSEA-P which refers to applications using java and GSEA-R which refers to the application using R. They do essentially the same thing using the same methods, each with differing amounts of freedom and control. The GSEA software is distributed in four different ways:

- GSEA desktop application, which is a menu driven application that is relatively straightforward to use;
- GSEA java jar file, which allows command line usage;
- GSEA java source code, which allows the user to incorporate GSEA straight into their analysis;
- R-GSEA, which is an R code that can be incorporated straight into the analysis and is relatively simple to change/ modify.

Each of these methods have their advantages over the others. In particular the desktop application and the R code are very useful.The desktop application has several additional features, such as:

- Ability to choose gene sets from a gene set browser, connected to MSigDB;
- Easy to load data by menu;
- Produces a table which includes columns for ES, FDR and p-values along with enrichment plots and annotated reports of enrichment results;
- 'Leading edge analysis' which allows the user, after performing GSEA, to select top scoring gene sets and compare the genes within these sets that occur before the maximum of the running ES (i.e. the genes in a gene set which gives the set its high ES). The idea behind this is that it allows the user to group leading edge subsets and thus categorize high scoring gene sets into biological pathways.

A downside of the desktop application, which is a feature of many menu driven applications, is its lack of flexibility. The R-code is more adaptable, but requires a good working knowledge of R. As mentioned on the website the R-code was developed as a prototype for the method and is now generally used by researchers wanting to adapt the method of GSEA.

GSEA is presented as a useful method for identifying pathways and processes of groups of genes, which at individual gene level would be difficult to identify and interpret. It is an adequate tool for dealing with high dimensional data of this type, and with the freely available software and information it is a relatively straightforward method for researchers to use. However, we must consider the usefulness, limitations and possible shortcomings of such an ad-hoc method:

- The use of the Kolmogorov-Smirnov statistic is not necessarily the best choice of test statistic;
- The weighting of the Kolmogorov-Smirnov statistic is not justified. Therefore the weighting used is not necessarily the best;
- The method for construction of the null distribution is not unique. There is therefore room for investigation and perhaps improvement here;

- There are some statistical concerns about the method of GSEA as indicated by Damian and Gorfine (2004).

GSEA is a user friendly method that can allow the user to find real effects from groups of genes. However, there are papers which express concern over the method Damian and Gorfine (2004); Efron and Tibshirani (2006).

2.3 Gene Set analysis (GSA)

Efron and Tibshirani (2006) give a more general overview of the methodologies involved in testing for relationships between groups of genes and some outcome. There are two particular areas on which the paper focuses:

- The discussion and assessment of the use of different gene set summary statistics;
- The construction of the null distribution.

2.3.1 The gene set summary statistic

There are several methods in common use for summarizing the test statistics $\mathbf{z}_s$ of a set of genes. These are:

- Mean;
- Absolute mean;
- Running sum Kolmogorov-Smirnov statistic (as in GSEA);
- Maxmean.

The maxmean test statistic is introduced in Efron and Tibshirani (2006) and is obtained as follows. Say we are interested in gene set s, for which we have the individual gene test statistics $\mathbf{z}_s = (z_1, z_2, \ldots, z_{m_s})$ then

$$S(z_i) = (S^+(z_i), S^-(z_i)) \qquad i = 1, 2, \ldots, m_s \tag{2.2}$$

where

$$\begin{aligned} S^{+}(z_i) &= max(z_i, 0) \qquad i = 1, 2, \ldots, m_s \\ S^{-}(z_i) &= -min(z_i, 0) \qquad i = 1, 2, \ldots, m_s \end{aligned} \tag{2.3}$$

And the maxmean test statistic is then given by

$$S_{max} = max\big(\overline{S}^{(+)}, \overline{S}^{(-)}\big) \tag{2.4}$$

The maxmean test statistic is designed to detect large values in either or both negative and positive directions and is shown by Efron and Tibshirani (2006) to be consistently more reliable as a gene set summary statistics than the other methods mentioned above.

2.3.2 The null distribution and Restandardization

The other main theme of this paper is the construction of the null distribution for the gene set summary statistics. Two methods for the construction of the null distribution are discussed by Efron and Tibshirani (2006). These are Randomization and Permutation, each of these sampling methods having its own associated advantages and disadvantages:

- Permutation model - Under the permutation model the null hypothesis H_{perm} is that the n rows of the expression matrix for gene set s are independently and identically distributed vectors.
  - Advantages of the Permutation model: It keeps (approximately) the same correlation structure as the original data.
  - Disadvantages of the Permutation model: It does not take into account the parameters of the whole distribution $(mean_z, sd_z)$ and so can lack relevance.
- Randomization model - Under the randomization model the null hypothesis H_{rand} is that the gene sets are chosen by a random selection of m genes.
  - Advantages of the Randomization model: Operates conditionally on the full set $\mathbf{z} = (z_1, z_2, \ldots, z_N)$ of z values, i.e. the randomization model tests the null

hypothesis that given $\mathbf{z}$, the observed gene set summary statistic $S(\mathbf{z}_s)$ from gene set s is no different than if the gene set s were chosen by a random selection of genes.

– Disadvantages of the Randomization model: The randomization model destroys gene-wide correlations, on which these methods lend much power, also it is the interest in this correlation structure that aided in the development of pathway analysis methodologies.

The method of GSA combines the randomization and permutation techniques to form a method named restandardization. This method takes advantage of the desirable properties of each of the two methods. Essentially restandardization uses permutation to create the null distribution and then corrects the permutation values to take account of the real observed scores (as would be used in the randomization). The method of restandardization is as follows

- Obtain the observed N z-scores for each gene $\mathbf{z} = (z_1, z_2, \ldots, z_{N)}$ and compute the mean and standard deviation (m_{obs}, sd_{obs}).
- Compute the observed gene set summary statistic $S_{obs}(\mathbf{z}_s)$.
- Permute the data n times and compute z-scores $\mathbf{z}_{perm}$ and corresponding gene set summary statistics $S_{perm}(\mathbf{z}_s)$ for each gene and gene set in each permutation. Compute the mean and standard deviation over all permuted z-values (m_{perm}, sd_{perm}).
- Compute the p-value for each gene set using standardized gene set summary statistics by

$$p_s = \frac{\sum I\left(\frac{S_{perm}(\mathbf{z}_s) - m_{perm}}{sd_{perm}} > \frac{S_{obs}(\mathbf{z}_s) - m_{obs}}{sd_{obs}}\right)}{n} \tag{2.5}$$

Where $I()$ is the indicator function.

This method of restandardization does not need to be applied to GSEA as the use of the Kolmogorov-Smirnov statistic automatically fulfills the criteria met by restandardization. The GSEA test compares the observed cumulative distribution of z-values in s with the cumulative distribution of all other z-values, thus including information from the whole distribution.

Efron and Tibshirani (2006) present analyses of two real data sets, one being the $p53$ data. A simulation study was also conducted whereby 1000 $N(0,1)$ gene expressions were generated across 50 samples in each of 2 classes. Genes were assigned to gene sets where each non-overlapping block of 20 genes comprised a gene set (this simulation model will be discussed in detail in Chapter 3). Five different scenarios were looked at, these being:

- All 20 genes in set 1 are 0.2 units higher in class 2;
- The first 15 genes in set 1 are 0.3 units higher in class 2;
- The first 10 genes in set 1 are 0.4 units higher in class 2;
- The first 5 genes in set 1 are 0.6 units higher in class 2;
- The first 10 genes in set 1 are 0.4 units higher in class 2 and the remaining 10 genes in set 1 are 0.4 units lower in class 2.

The GSA procedure performs well at identifying the active gene set in all five cases and the maxmean statistic proves to have better performance than the mean, absolute mean, GSEA or absolute GSEA in identifying the active set. However, the simulation procedure presented is not entirely realistic, as will be discussed in Chapter 3, and therefore a more thorough simulation study would be useful.

The R library GSA has been introduced by Efron and Tibshirani to implement gene set analysis with restandardization using the gene set summary statistics maxmean, mean and mean absolute z-scores.

2.4 Bayesian gene set analysis (BGSA)

Bayesian gene set analysis (BGSA) as proposed by Shahbaba et al. (2011) utilizes within gene set hierarchical Bayesian models to asses the significance of pathways with regard to phenotypes of two levels. This model offers a more practical, knowledge based approach to the analysis of gene sets, utilizing probability distributions to describe the various layers and mechanisms within gene sets that act as pathways to phenotype.

2.4.1 The Model

A within gene set hierarchical model is defined as

$$y_{sgi} = \alpha_{sg} + \beta_{sg} x_i + \varepsilon_{sgi} \quad g = 1,2,\ldots,m_s \quad i = 1,2,\ldots n \quad s = 1,2,\ldots,K$$
$$\varepsilon \sim N(0,\sigma_{sg}^2) \tag{2.6}$$

where y_{sgi} denotes the i^{th} observed expression of the g^{th} gene in gene set s, x_i the phenotype (binary) of the i^{th} subject, α_{sg} the mean expression of gene g in set s and β_{sg} the change in expression of gene g in set s between the two phenotypes. The underlying theme to the identification of active gene sets in this method is in the posterior distribution of β_{sg}, or, more specifically it is the posterior distribution of the variance of β_{sg}, τ_s that is used as a measure of activeness of gene set. Note that τ_s is common to all effects within gene set s. For example if in set s all effects are very near to zero, then τ_s will be small. On the other hand, if in another set s there are some relatively large negative effects, some relatively large positive effects and some zero effects then τ_s will be large, representing a more active gene set.

2.4.2 Prior model Specification

The model uses the following prior distributions on its parameters

$$\begin{aligned}
\sigma_{sg}^2|\xi,\eta &\sim Inv-\chi^2(\xi,\eta^2)\\
\alpha_{sg}|\gamma &\sim N(0,\gamma^2)\\
\beta_{sg} &\sim N(0,\tau_s^2)
\end{aligned}$$

whereby non-informative priors are used for the hyperparameters ξ, η and γ. The prior of β_{sg} has associated hyperprior τ_s^2 with its own mixture prior, such that

$$\tau_s^2 \sim (1-\lambda)F_0 + \lambda F_1$$

Where

$$F_0 = Inv - \chi^2(\nu, \phi_0^2) \qquad \text{and} \qquad F_1 = Inv - \chi^2(\nu, \phi_0^2 + \phi_1^2) \tag{2.7}$$

Where F_0 is the distribution of τ_s^2 under the null hypothesis of no set effect and F_1 the distribution of τ_s^2 under the alternative hypothesis of a set effect. This is facilitated by a relatively small scale parameter ϕ_0^2 for the null part of the distribution and a relatively large scale parameter $\phi_0^2 + \phi_1^2$ for the alternative.The parameters in this prior are assumed

$$\begin{aligned} \phi_0^2, \phi_1^2 &\sim gamma(1,1) \\ \nu &\sim gamma(1,1) \\ \lambda &\sim beta(a,b) \end{aligned}$$

Shahbaba et al. (2011) comment that in their analyses $a = b = 1$, however, in practice only a small number of gene sets would be expected to be significant and a more informative prior such as $beta(1,10)$ might be used. In order to allow for sampling for λ, a binary latent variable v_s is introduced whereby

$$v_s \sim Bernoulli(p)$$

such that

$$\tau_s^2 | v_s, \phi_0, \phi_1 \sim (1-v_s)F_0 + v_s F_1$$

It is the posterior expectation of v_s given data D, $E[v_s|D]$ that allows the calculation of the significance of a gene set. The p-value $\hat{p}_0$ is given by

$$\hat{p}_0 = 1 - E[v_s|D] \tag{2.8}$$

2.4.3 Posterior Sampling

The priors used in this model, as specified previously, are all conditionally conjugate other than ν, ϕ_0^2, ϕ_1^2. Therefore the Gibbs sampler is used to obtain posterior samples of the parameters. Posteriors for ν, ϕ_0^2 and ϕ_1^2 are sampled using single variable slice sampling because the priors for ν, ϕ_0^2 and ϕ_1^2 are not conditionally conjugate. All of the `R` code for the implementation of this model is freely available at `http://www.ics.uci.edu/babaks/Homepage/Codes.html` and is straightforward to implement on a given data set.

Alongside analyses of the *p*53 data set, a simulation study is implemented to asses the performance of BGSA and to compare its performance with that of GSEA and GSA. Strictly speaking, data are not simulated in the data simulation procedure presented by Shahbaba et al. (2011). The data simulation procedure presented is as follows:

- Individual gene analysis (IGA) is performed on the full set of genes and the top 20 genes are identified;

- Genes from the *p*53 data set are randomly reallocated amongst the 522 gene sets in the molecular signatures data base C1 curation of gene sets. Therefore gene set should be found significant by chance only;

- Five gene sets from the molecular signatures data base C1 curation of gene sets are randomly selected to be made active;

- The 20 selected genes are randomly allocated to these five sets such that there are five active gene sets.

Following this procedure, three scenarios are studied:

1. Gene sets are assumed mutually exclusive i.e. each gene belongs to only one gene set;

2. Only the top 20 genes are assumed mutually exclusive with regards to gene set, whereas other genes can belong to more than one set;

3. All gene sets can share common genes (including the top 20).

The Receiver operator characteristic (ROC) curve is used to asses the performance of GSEA, GSA and BGSA in determining active gene sets from the three above scenarios. The ROC curve is a great way to evaluate model performance as it allows for simultaneous consideration of power and type 1 error rate without specifying an arbitrary cut off. In this case the five gene sets selected as active are taken to be true positives and the remaining gene sets are taken to be true negative. It is shown that BGSA out performs all other methods on such simulated data, with a consistently larger area under the curve (AUC) in all three cases. It can also be seen that in such situations GSA is consistently more succesful in identifying active gene sets than GSEA with a larger AUC in every case.

2.5 Appraisal of the methods

The three methods presented previously are some of the more prominent methods for the analysis of gene sets.

Looking into results presented by the three papers from analyses on the *p*53 data set, it can be seen that the methods find common results. However, this gives little insight into the accuracy, dependence or performance of the methods as the truth behind such results is unknown. To this note, a thorough simulation study should be carried out, whereby data are simulated such that the truth of the significance of pathways are known. The data should be simulated such that each methods ability to identify an active gene set is tested and the sensitivity of the method is revealed.

A notable feature of the work presented by Subramanian et al. (2005) is that the methodology is based on the application of ad-hoc analytical methods with no underlying statistical model and little justification for the application of these ad-hoc analytical methods. It is discussed and demonstrated by both Damian and Gorfine (2004) and Efron and Tibshirani (2006) that there are some quite simple cases where GSEA can be shown to fail.

To a lesser degree, it could also be said that GSA, as presented by Efron and Tibshirani (2006) is based on the application of ad-hoc analytical methods with no underlying statistical model. However, Efron and Tibshirani (2006) do give some justification to the analytical methods that are used by GSA and a thorough simulation study is implemented, testing the methods presented and comparing GSA to GSEA using the maxmean test statistic alongside other gene set summaries (mean, absolute mean). The maxmean test statistic and GSA are shown to be consistently more reliable in identifying active gene sets. It is mentioned by Shahbaba et al. (2011), however, that there are cases, such as when there are both positive and negative gene effects within a gene set, where the maxmean statistic fails to detect real gene set effects.

BGSA, as proposed by Shahbaba et al. (2011) is a fully Bayesian hierarchical modelling approach to the analysis of gene sets. Again, a simulation study is presented, showing that this model is more successful at identifying active gene sets than GSA and GSEA. As mentioned in Section 2.4, the data in the simulation study presented by Shahbaba et al. (2011) is not strictly simulated. In fact, real data are used, but reorganized such that genes found to be significantly related to phenotype (by IGA) are put into gene sets selected to be significant. This could be misleading in some ways as it has been discussed and examples shown (Chapter 1) that IGA can be unreliable in identifying significant genes.

The simulation procedures presented by Efron and Tibshirani (2006) and Shahbaba et al. (2011) both leave room for improvement. The conditioning in the simulation model proposed by Efron and Tibshirani (2006), whereby differential gene expression is modelled conditional on outcome, could possibly be reversed to represent a more realistic situation, i.e. outcome to be simulated conditionally on expression. The fact that the data used in the simulation study presented by Shahbaba et al. (2011) are real, essentially with the columns shuffled according to an IGA could lead to too much noise and an unfair and possibly misleading comparison of the methods. Both presented simulation studies agree that generally GSEA is the more unreliable method, they also show that their proposed method is the most successful in identifying active gene sets. A larger, more realistic and impartial simulation study would be useful to show the relative merits of the three methods. It would be useful to implement a more in depth analysis of the methods, investigating in which situations the three methods work well and which cases the methods fail. From such an analysis adaptations to existing methods/models or new methods/-

models could be defined that can deal with a more broad spectrum of problem. A more knowledge based practical appraisal of the methods could also be made after a simulation study.

All three of the described methods propose univariate analyses; GSEA and GSA use some test statistic for each gene and BGSA uses a univariate regression model for each of the genes, clearly the relationship between phenotype and gene is not univariate as is a theme of this group of methodologies. However, it must be noted that both GSEA and GSA operate conditionally on the full set of test statistics. Another argument against these univariate analyses is the inability to account for other descriptive variables, for example age, weight or smoking status.

Of more concern with these methods is the conditionality. The methods condition gene expression on phenotype, i.e. $data|outcome$ whereas the aim of this class of model is to model the pathway to phenotype, i.e. to model $outcome|data$. The distinction between the two ways of conditioning such modelling is an important one. In general, we are attempting to relate how the expression of genes or groups of genes relate to outcome, or in other words we are attempting to determine the pathway to phenotype. This implies a causal relationship whereby phenotype depends on gene expression. Hence it makes sense to model conditionally on expression, rather than the other way around as in GSEA, GSA and BGSA. Clearly, there is room for the construction of multivariate models for this problem, whereby phenotype is modelled conditional on gene expression.

The following Chapter assesses simulation procedures and provides an in depth simulation study presenting applications of GSEA, GSA and BGSA to simulated data.

Chapter 3

Comparison of existing methodologies for gene set analysis

3.1 Introduction

As discussed in Chapter 2 there is the need to compare and evaluate the performance of the previously described methods in identifying significant pathways. Possible limitations with existing methods have been discussed in Chapter 2, in papers proposing new methods, such as Efron and Tibshirani (2006) and Shahbaba et al. (2011) and in review papers, for example Nam and Kim (2008) and Song and Black (2008) amongst others. Several of these papers present simulation studies.

There are several papers gathering together ideas about the analysis of gene sets, for example Irizarry et al. (2009) and Nam and Kim (2008) provide an extensive list of many available methods. Some papers focus solely upon simulation studies contrasting and comparing methods, for example Goeman and Buhlmann (2007) and Song and Black (2008). Nam and Kim (2008) also present a simulation study comparing some number of current methods for the analysis of gene sets based upon the null hypothesis that they test. The simulation studies presented in these publications follow a very similar vein to that presented by Efron and Tibshirani (2006), in that gene expression is simulated conditionally on phenotype. This does not necessarily result in the most realistic data. Song and Black (2008) take the simulation procedure one step further by adding a correlation structure to their simulated gene expressions. However, there is still an issue of the conditionality in the data simulation procedure.

It would be beneficial to consolidate ideas of data simulation models in order to produce a simulation study that follows a consistent theme and such that data are as realistic as possible. A wide range of gene set scenarios should be examined such that a good practical understanding of the methods discussed (GSEA, GSA and BGSA) can be obtained.

The papers presenting GSA (Efron and Tibshirani (2006)) and BGSA (Shahbaba et al. (2011)) provide simulation studies comparing the three methods, yet these simulation studies are fairly limited, as discussed in Chapter 2 and below. A larger, more realistic and independent simulation study should be carried out.

This chapter proceeds as follows; Section 3.2 introduces and discusses the relative merits and weaknesses of two data simulation models. Section 3.3 outlines the simulation study that will be carried out, defining several scenarios. Section 3.3 also presents results from applying GSEA, GSA and BGSA to the simulated data sets. Finally, Section 3.4 discusses the results obtained from the analyses.

3.2 Data Simulation

There are essentially two approaches to modelling gene effect.

1. Condition on phenotype and model the effect of expression given phenotype.

2. Condition on expression and model phenotype conditional on expression.

3.2.1 Approach 1

The first approach is that adopted by Efron and Tibshirani (2006), whereby data is generated on N genes across n samples, and the expression of the g^{th} gene in set s in the i^{th} subject is given by

$$x_{sgi} \sim N(0,1)$$

This represents the general practice of gene expression studies where gene expression is transformed such that it follows a standard normal distribution. The first $n/2$ samples are then taken to be of phenotype 1 and the second $n/2$ samples are taken to be of phenotype

2. Each consecutive, non-overlapping block of m_s genes is considered to be a gene set. A gene set is then made active by adding or subtracting some constant from a number of genes within that gene set in a given phenotypic condition. Nam and Kim (2008) and Shahbaba et al. (2011) amongst many others follow this data simulation procedure.

This simulation model conditions on outcome and therefore assumes a univariate relationship between gene expression and outcome. This is somewhat of an over simplification, and it is unreasonable to assume such relationships.

An interesting feature of this approach is that it means that the marginal distribution of expressions of active genes is no longer normally distributed with mean zero and variance one. Assuming we desire the expression of all genes to follow this standard normal distribution, it is useful to look at this in more depth. Say we have gene $x_g \sim N(0,1)$ which we desire to be active, where half of the values in x_g correspond to $y = 0$ and half of the values in x_g correspond to $y = 1$. Then following Approach 1 a constant $c = 2$ is added to x_g when $y = 1$. Now, if we look at the conditional distributions of x_g in the two phenotypic conditions, $y = 0$ and $y = 1$, as shown in Figure 3.1, it can be seen that the conditional distributions of $x_g|y = 0$ and $x_g|y = 1$ both follow a normal distribution with

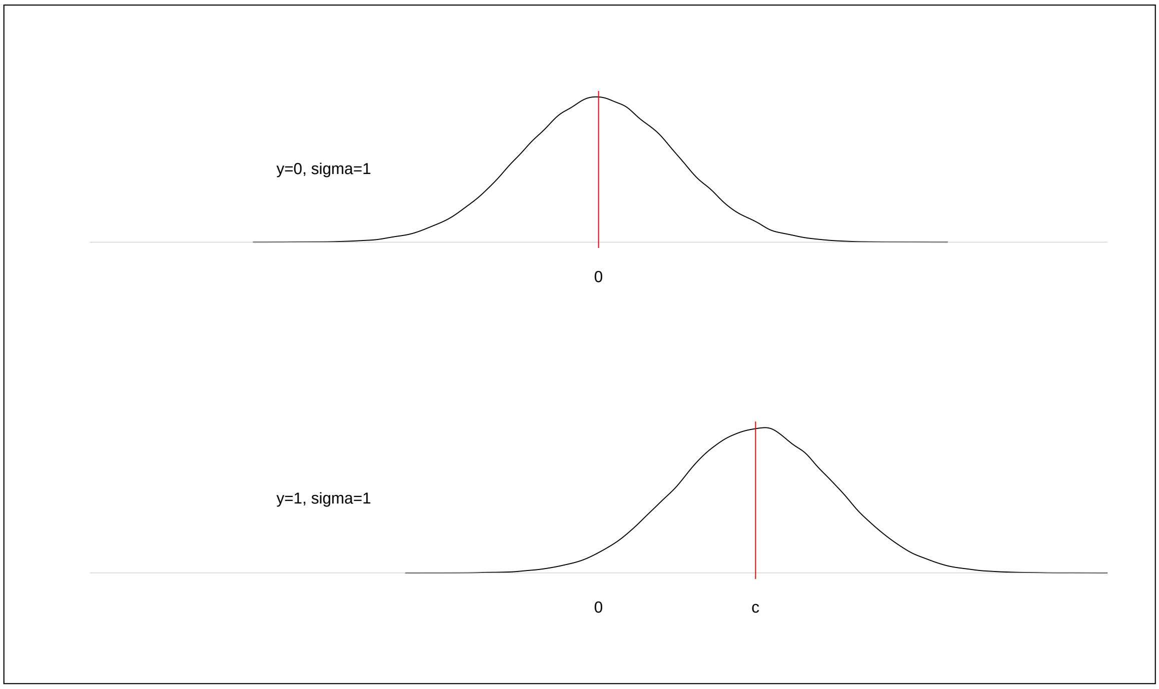


Figure 3.1: Conditional distribution of $x_g|y$ where a constant $c = 2$ is added to x_g when $y = 1$. Vertical line showing the mean of the distribution.

means of 0 and c respectively and have variance one.

However, if we look at the marginal distribution of x_g, as shown in Figure 3.2 then clearly x no longer follows a standard normal distribution. This can also be described by the conditional expectation and conditional variance theorems, where

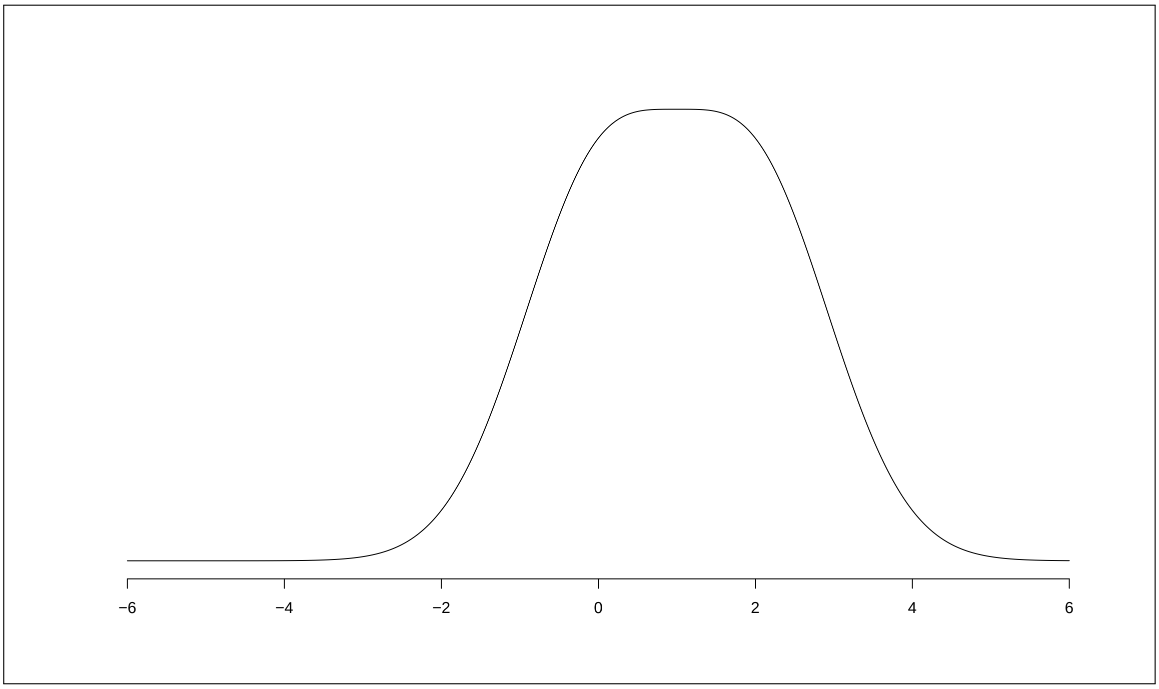


Figure 3.2: Marginal distribution of x_g. Vertical line showing the mean of the distribution.

$$E[X] = E\left[E[X|Y]\right] \tag{3.1}$$

and

$$V[X] = V\left[E[X|Y]\right] + E\left[V[X|Y]\right] \tag{3.2}$$

then the expected value and the variance of x is given by

$$E[X] = \frac{c}{2}$$

$$V[X] = 1 + \frac{c^2}{2}$$

So after the constant c is added to x, we have

$$f(x) = \frac{1}{\sqrt{8\pi}}(e^{-x^2/2} + e^{-(x-2)^2/2})$$

Clearly, this is not a desirable property, and data resulting from such a simulation no longer look like data from a typical genomics study. A partial solution to such problems could be to re-standardize x_g such that

$$x_g' = \frac{x_g - \frac{c}{2}}{\sqrt{1 + \frac{c^2}{2}}} \tag{3.3}$$

This produces a distribution with mean zero and variance one, which, unless c is large will be close to a normal distribution.

Song and Black (2008) take this simulation procedure one step further by allowing for a correlation structure in the data, simulating gene expression

$$\boldsymbol{x}_i \sim MVN(0, \Sigma)$$

where Σ is defined by pairwise (gene by gene) correlation and $\boldsymbol{x}_i$ is the expression of genes $g = 1, \ldots, N$ in subject i, $i = 1, \ldots n$. It seems odd to define a correlation structure for simulating data when a univariate relationship that conditions gene expression on phenotype is assumed.

3.2.2 Approach 2

In this approach we model phenotype conditionally on gene expression. A logistic model is assumed for the relationship between phenotype for the i^{th} subject y_i and gene expression $\boldsymbol{x}_i$, such that

$$P[y_i = 1] = \frac{1}{1 + e^{-\eta_i}} \qquad i = 1, \ldots, n \tag{3.4}$$

where

$$\eta_i = \boldsymbol{x}_i \boldsymbol{\beta} \qquad i = 1, \ldots, n \tag{3.5}$$

Data are to be simulated whereby the expression of the $g = 1, \dots, N$ genes in the i^{th} subject follow a multivariate normal distribution, i.e.

$$\boldsymbol{x}_i \sim MVN(0, \Sigma), \qquad i = 1, \dots, n$$

where Σ is an $N \times N$ matrix with all diagonal elements equal to 1, therefore all off-diagonal elements define pairwise correlations between genes. This, again, reflects the general practice in gene expression studies of standardizing the expression of all genes to follow a standard normal distribution, however we can now also define a correlation structure. Gene set can be simply defined as consecutive non-overlapping blocks of $m_s = N/K$ genes whereby there are K gene sets in total with s denoting gene set ($s = 1, 2, \dots, K$). Gene set effect for gene set s, can be denoted as z_s and specified as

$$z_s = \begin{cases} 0 & \text{No gene set effect} \\ 1 & \text{Gene set effect} \end{cases} \tag{3.6}$$

Gene effect for the g^{th} gene in gene set s, denoted β_{sg} is specified in a similar way, whereby

$$\beta_{sg}|(z_s = 1) = \begin{cases} -\delta & \text{Negative gene effect} \\ 0 & \text{No gene effect} \\ \delta & \text{Positive Gene effect} \end{cases} \tag{3.7}$$

and

$$\beta_{sg}|(z_s = 0) = 0 \tag{3.8}$$

So if a gene set is simulated as inactive then all genes within that gene set are considered inactive and if a gene set is simulated as active, some genes within that set will have a relatively large effect in either a positive or a negative direction. Figure 3.3 shows a conceptual diagram of the simulation of gene and gene set effects. This simulation of gene effect size on a linear predictor for a logistic model reflects ideas that some genes will have no effect on phenotype, whilst differential expression of some genes will have a positive effect on phenotype and differential expression of other genes will have a negative

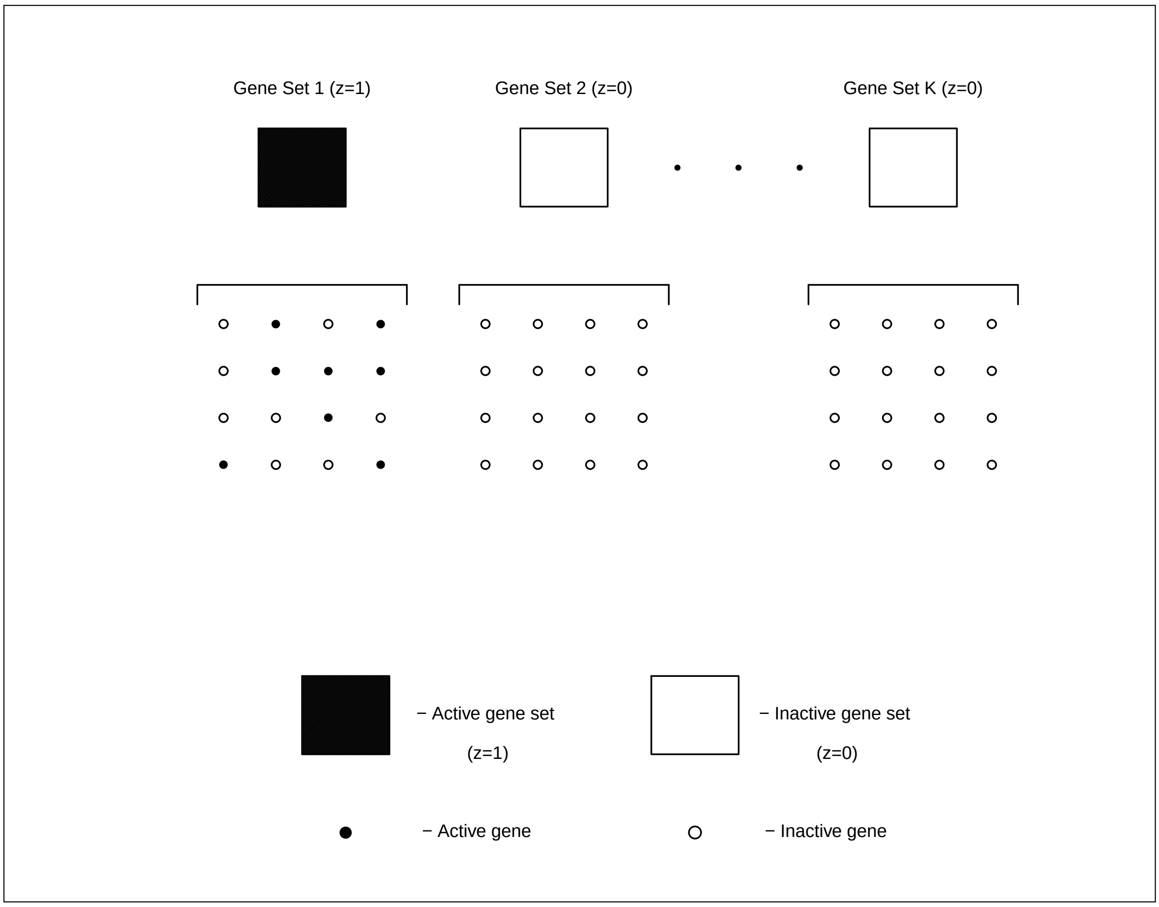


Figure 3.3: Conceptual diagram of the simulation of gene and gene set effects.

effect on phenotype. This is represented by the effects δ and $-\delta$. From these simulated gene effects and gene expressions a binary outcome (or phenotype) for the i^{th} subject $(i = 1,2,\ldots,n)$ is then given by

$$y_i \sim \text{Bernoulli}\left(\frac{1}{1+\exp\left(-\boldsymbol{x}_i\boldsymbol{\beta}\right)}\right)$$

This data simulation procedure results in a data set whereby there is some large number (N) of genes with normally distributed expression levels that are partitioned into N/K gene sets. Some specified number of genes have an effect of zero $(z_s = 0)$ on phenotype, while some specified number of genes inside active gene sets $(z_s = 1)$ will have some relatively large known effect. More importantly, however, is that this procedure results in a data set where outcome is modelled conditional on expression. In contrast to data resulting from Approach 1, the marginal distribution of $\boldsymbol{x}_g$ is the same for an active gene and an inactive gene, i.e. as x has not been altered in any way, then x still follows a standard normal distribution. Figure 3.4 shows the conditional densities of $x|y=0$ and $x|y=1$. As can be seen the conditional densities have different means, yet still follow a normal distribution with variance one. It is the specification of the effects $(\boldsymbol{\beta})$ that take the smaller values from x that translate to $y=0$ and the larger values from x to $y=1$.

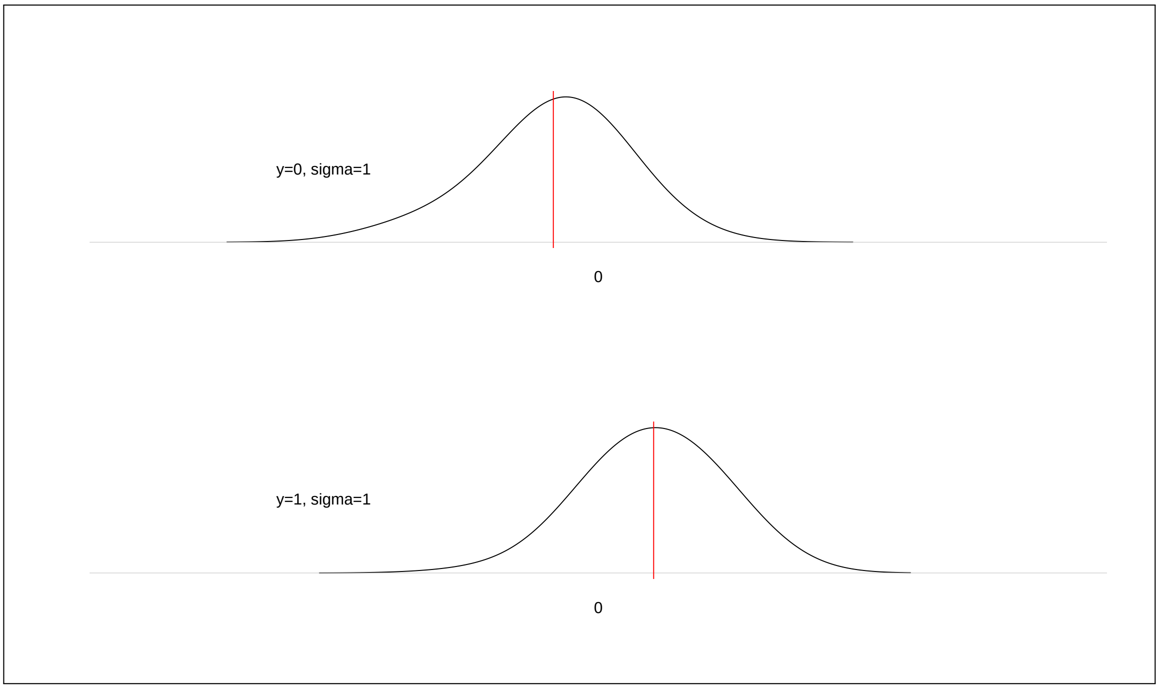


Figure 3.4: Conditional distribution of $x|y$ from proposed simulation procedure.

When looking at cases where a single gene affects phenotype, as shown here, there is a clear distinction between the conditional distributions of x. However, with increasing numbers of uncorrelated active genes we see a dilution of effect as outcome is described by several effects. If we were to look at plots as in Figure 3.4 where we have even as few as five genes affecting phenotype then we would see very little difference in the conditional distributions. This is described by Figure 3.5. As can be seen in Panel A, where there is only one gene affecting y, there is a clear distinction between the conditional distribution of $x|y$. This distinction is blurred somewhat when there are five genes affecting y, as shown in the boxplot of one of these genes in Panel B. In Panel C, where there are 20 genes affecting phenotype, there is little visible evidence of a difference in the conditional distributions of one of these genes. Due to the variation in outcome being caused by many factors, it can be difficult to attribute this variation or proportion of variation to single explanatory variables. This presents a problem in the analysis of such data, which closely mimics a realistic situation. For the reasons presented, the simulation model from the second approach will be used in the following simulation study.

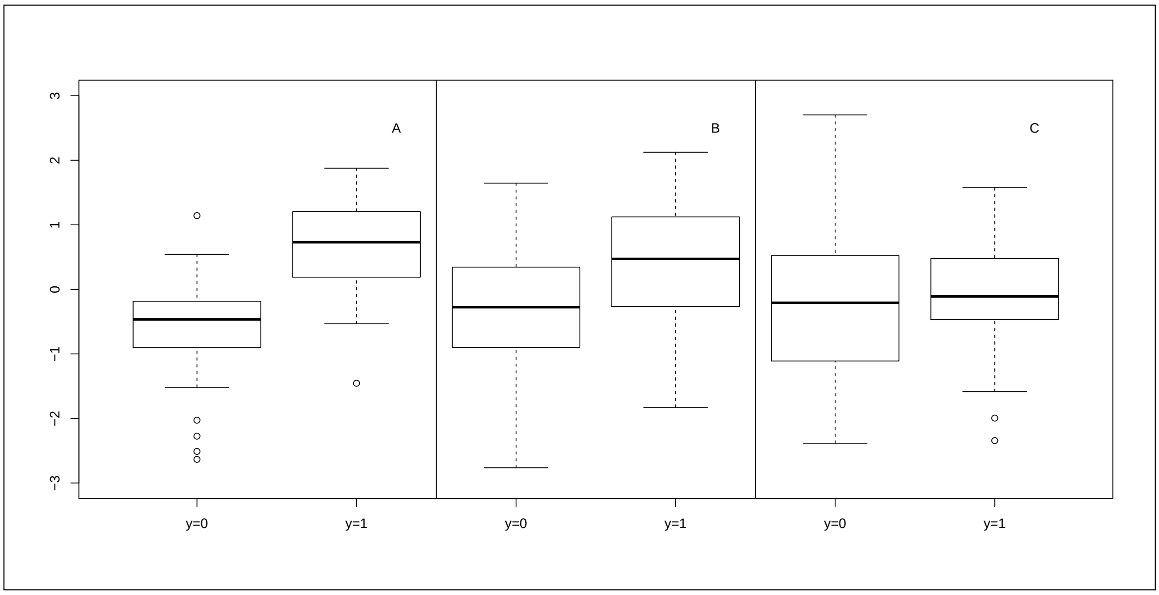


Figure 3.5: Box plots of conditionals of an active gene. Panel A shows a situation when there is only one active gene. Panel B shows conditionals of one gene when there are 5 active genes. Panel C shows conditionals of one gene when there are 20 active genes.

Looking at equivalent situations where data are correlated, this dilution of effects is not so evident and the larger the correlation the less the dilution of effect.

3.3 Simulation Study

Following a similar thread to the majority of published simulation studies in this area, the simulation study begins by focusing upon scenarios where gene expression is assumed independent of other genes. This is followed by carrying out a second study, which is not so common, where correlation structures are defined.

The area under the curve (AUC) of the receiver operator characteristic (ROC) curve, as described in Chapter 1, will be used to summarize the performance of each method in identifying the active gene set. The set defined as active being true positive and a non-active (or null) set from each analysis being true negative. The ROC curve allows for the simultaneous consideration of power and type 1 error rate. This is particularly useful as each method defines its own cut off; GSA suggests a level of 0.05 or 0.01, BGSA a level of 0.1 and GSEA uses a cut off on the FDR of 0.25. Not only does the use of the AUC remove scale, but no definition of p-value/ q-value cut off is required which allows us to asses the true performance of each method without the need to define or use some arbitrarily cut off.

3.3.1 Uncorrelated data

Data are simulated according to Approach 2 for $n = 100$ subjects across $N = 1000$ genes, where $\Sigma = I$ and each consecutive non-overlapping block of $m_s = 50$ genes constituting a gene set. Twelve scenarios are defined by the number of active genes within an active set. As discussed in the previous section, a gene is considered active when it has a non-zero effect (β) on phenotype. In each of the twelve scenarios outlined below gene effect will range from 0.5 to 5, thus giving 10 ‘sub-scenarios’ to each scenario. Every simulated data set will be analyzed using GSEA, GSA and BGSA. The scenarios are:

1. One gene in set 1 is active;
2. Two genes in set 1 are active with effects of the same size in the same direction;
3. Two genes in set 1 are active with one positive effect and one negative effect;
4. Four genes in set 1 are active with effects of the same size in the same direction;
5. Four genes in set 1 are active with two positive effects and two negative effects;
6. Five genes in set 1 are active all with effects of the same size in the same direction;
7. Ten genes in set 1 are active all with effects of the same size in the same direction;
8. Ten genes in set 1 are active with five positive effects and five negative effects. All effects are of the same size;
9. Twenty genes in set 1 are active all with effects of the same size in the same direction;

10. Twenty genes in set 1 are active with ten positive effects and ten negative effects. All effects are of the same size;

11. Forty genes in set1 are active with all effects in the same direction;

12. Forty genes in set 1 are active with twenty positive effects and twenty negative effects. All effects are of the same size.

The three methods of interest will be applied to the data resulting from these 120 situations. This analysis will be replicated 100 times with newly simulated data (according to the same parameters) and the AUC calculated over each of these 100 replicates. Essentially there will be 12000 sets of simulated expression data looked at.

3.3.1.1 Results

Figures 3.6 to 3.17 show plots of the AUC for each of the twelve scenarios for increasing gene effect ($\boldsymbol{\beta}$). There are several consistent themes that can be seen from the AUC plots for the 12 scenarios:

- In every case it can be seen that with increasing effect size, the AUC also increases. I.e. the bigger the effect size the better the methods are at identifying the set as active;
- When we have effect sizes in both positive and negative directions the performance of GSA and particularly GSEA suffers;
- BGSA consistently performs the best out of the three methods;
- GSEA generally shows inferior performance to GSA and BGSA;
- The AUC for GSEA, particularly in scenarios where there are both positive and negative effects, shows much variability with increasing effect size, for example see Figure 3.17.

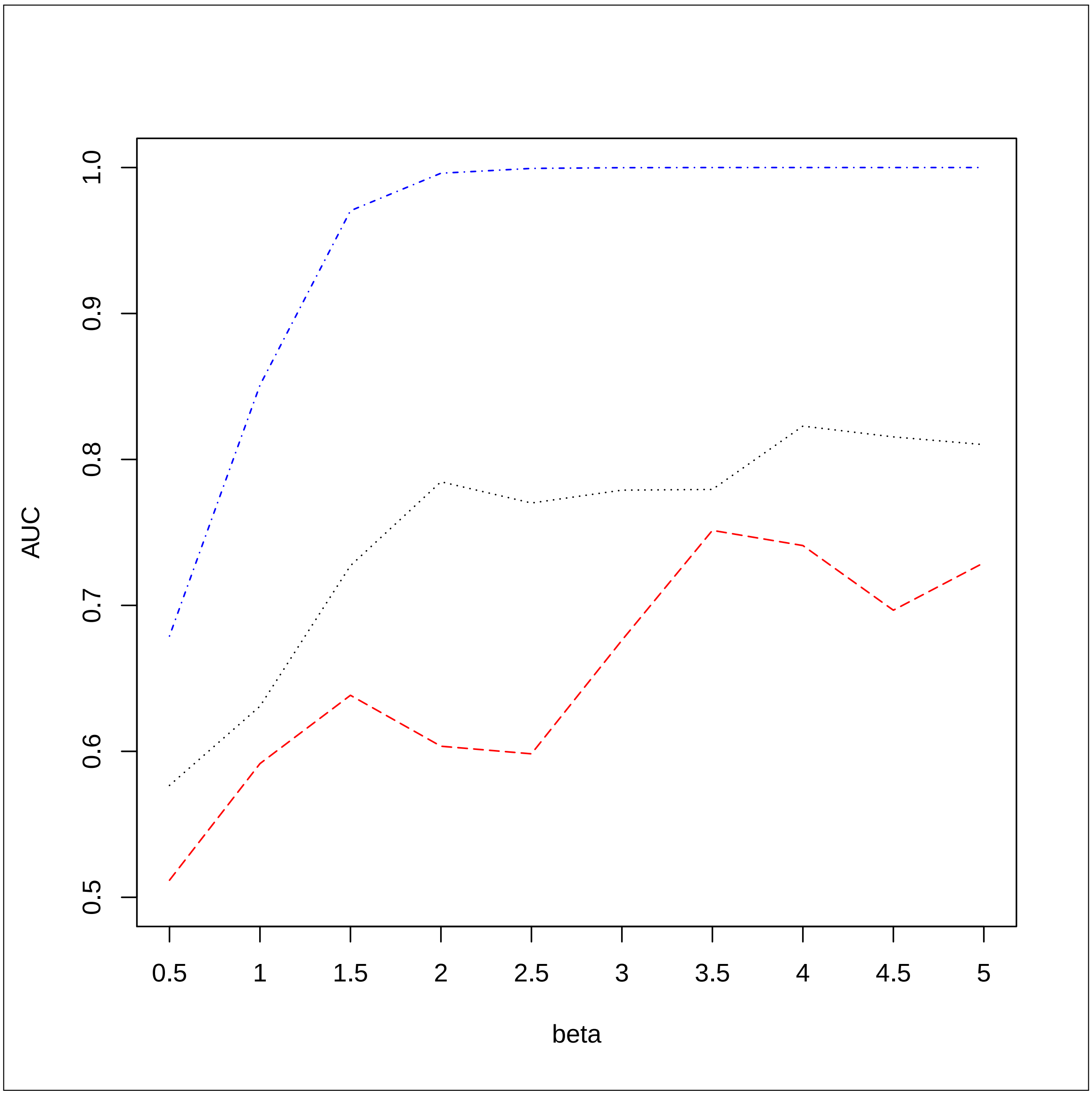


Figure 3.6: Plot of AUCs for increasing β in scenario 1. GSA-Dotted curve, BGSA-Dot-dash curve, GSEA-Dashed curve

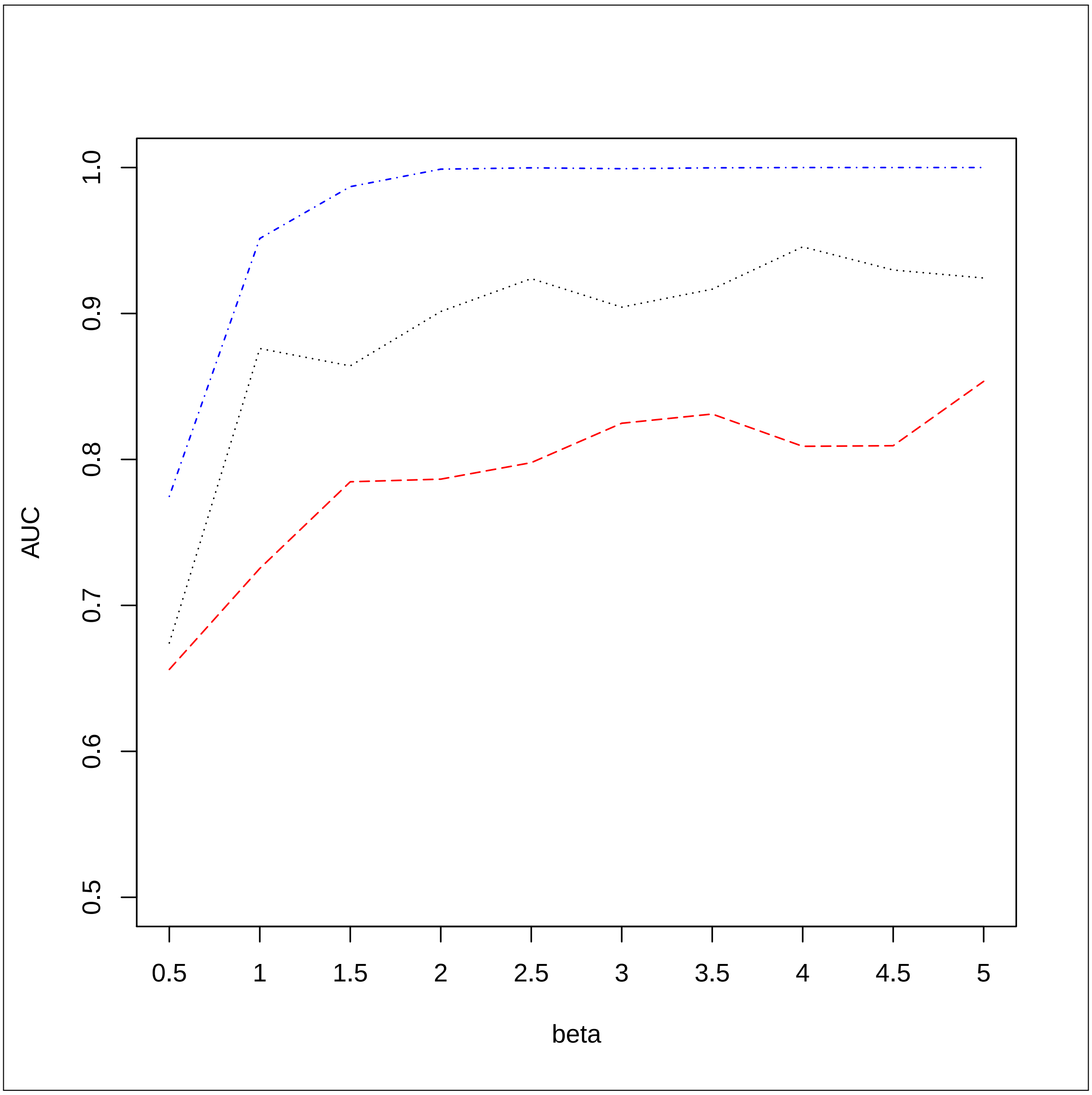


Figure 3.7: Plot of AUCs for increasing β in scenario 2. GSA-Dotted curve, BGSA-Dot-dash curve, GSEA-Dashed curve

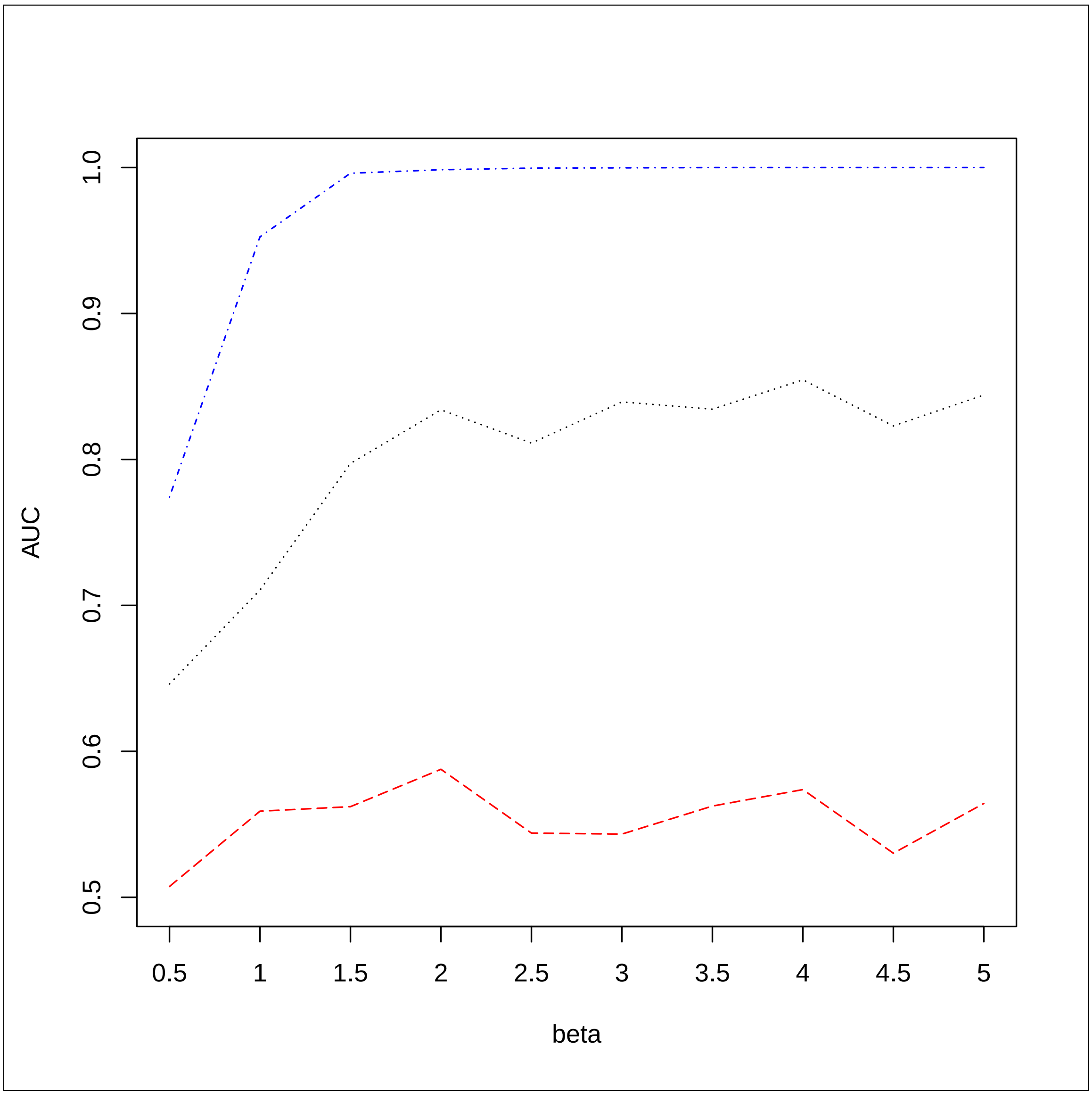


Figure 3.8: Plot of AUCs for increasing β in scenario 3. GSA-Dotted curve, BGSA-Dot-dash curve, GSEA-Dashed curve

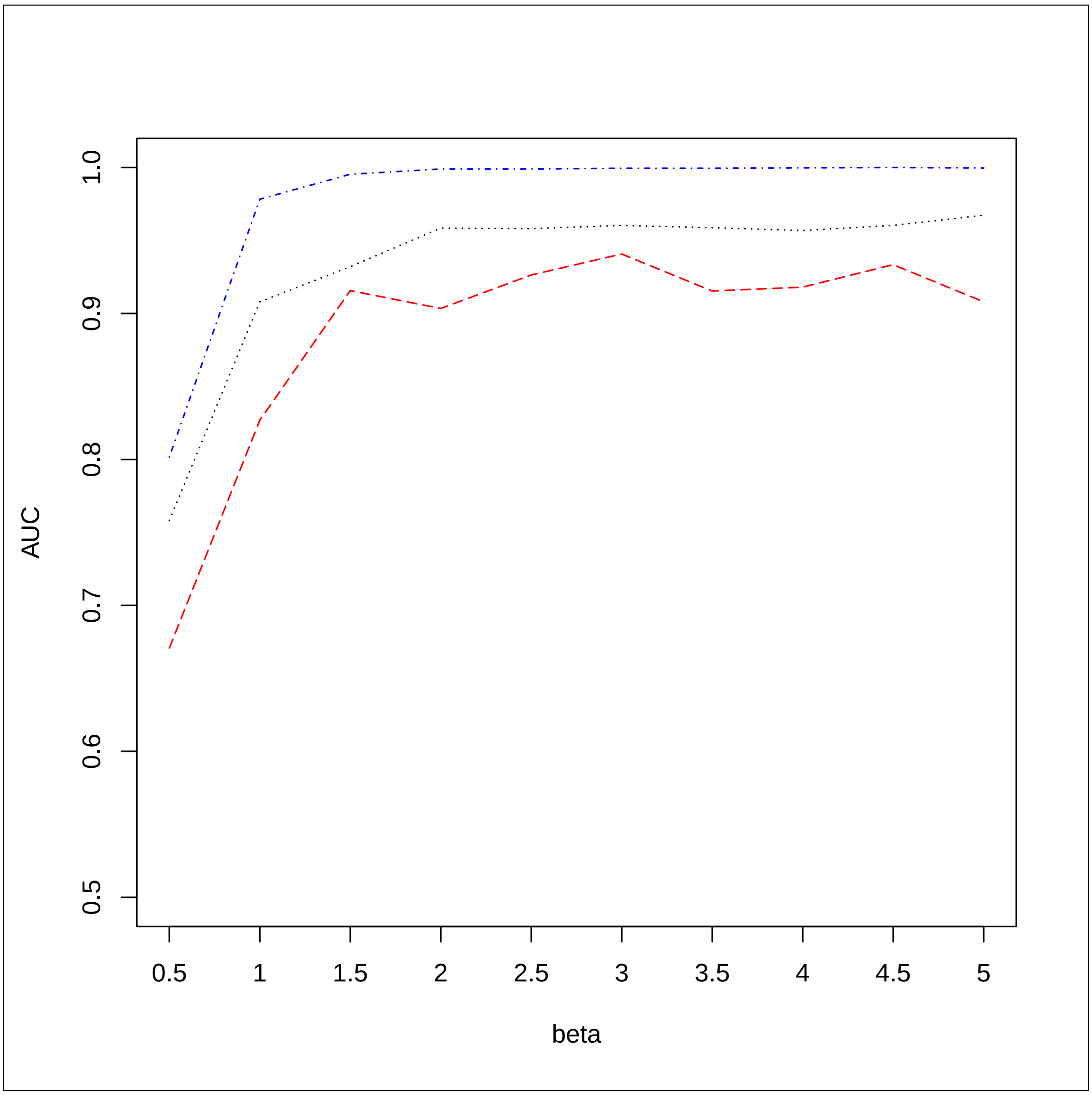


Figure 3.9: Plot of AUCs for increasing β in scenario 4. GSA-Dotted curve, BGSA-Dot-dash curve, GSEA-Dashed curve

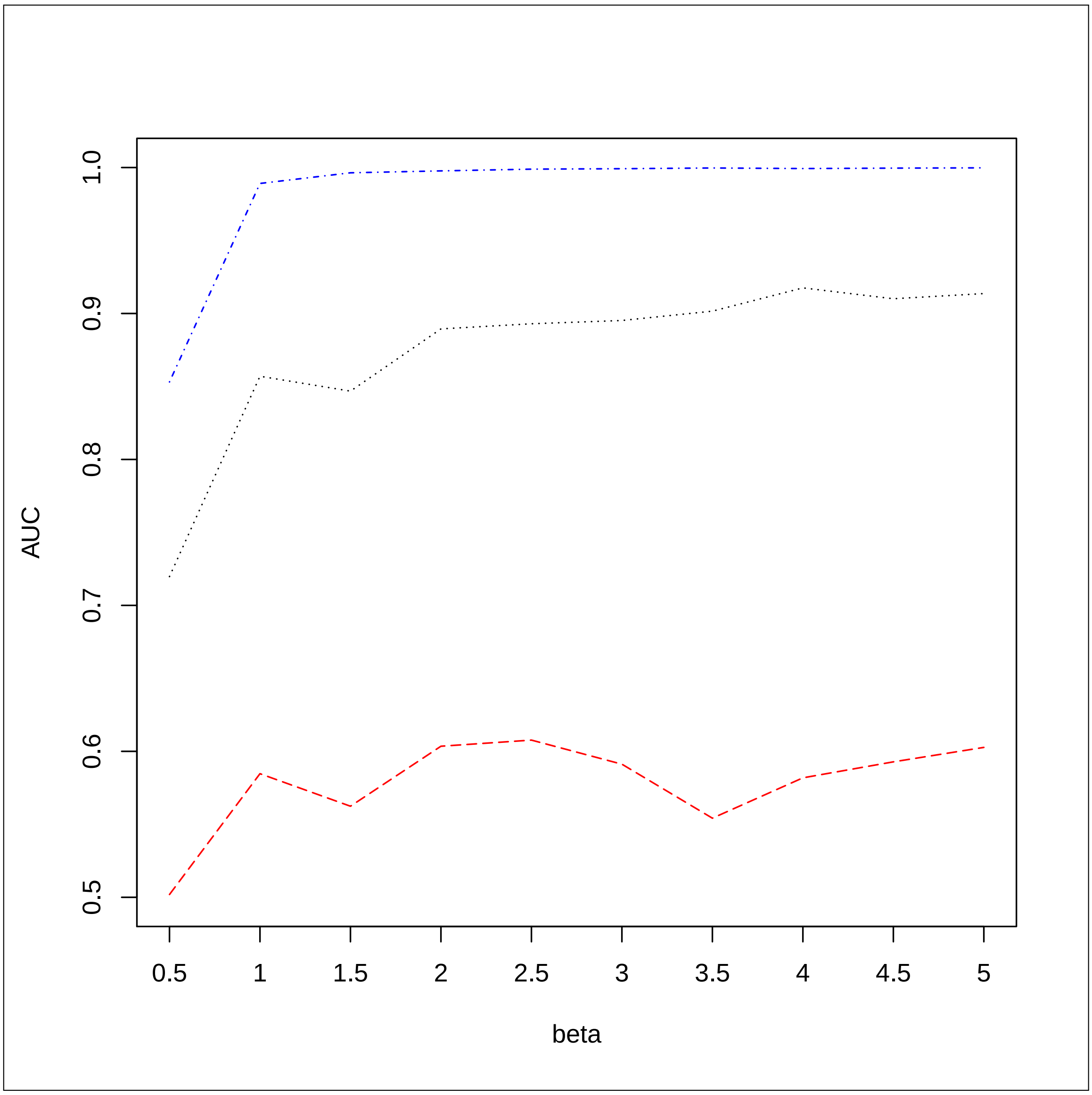


Figure 3.10: Plot of AUCs for increasing β in scenario 5. GSA-Dotted curve, BGSA-Dot-dash curve, GSEA-Dashed curve

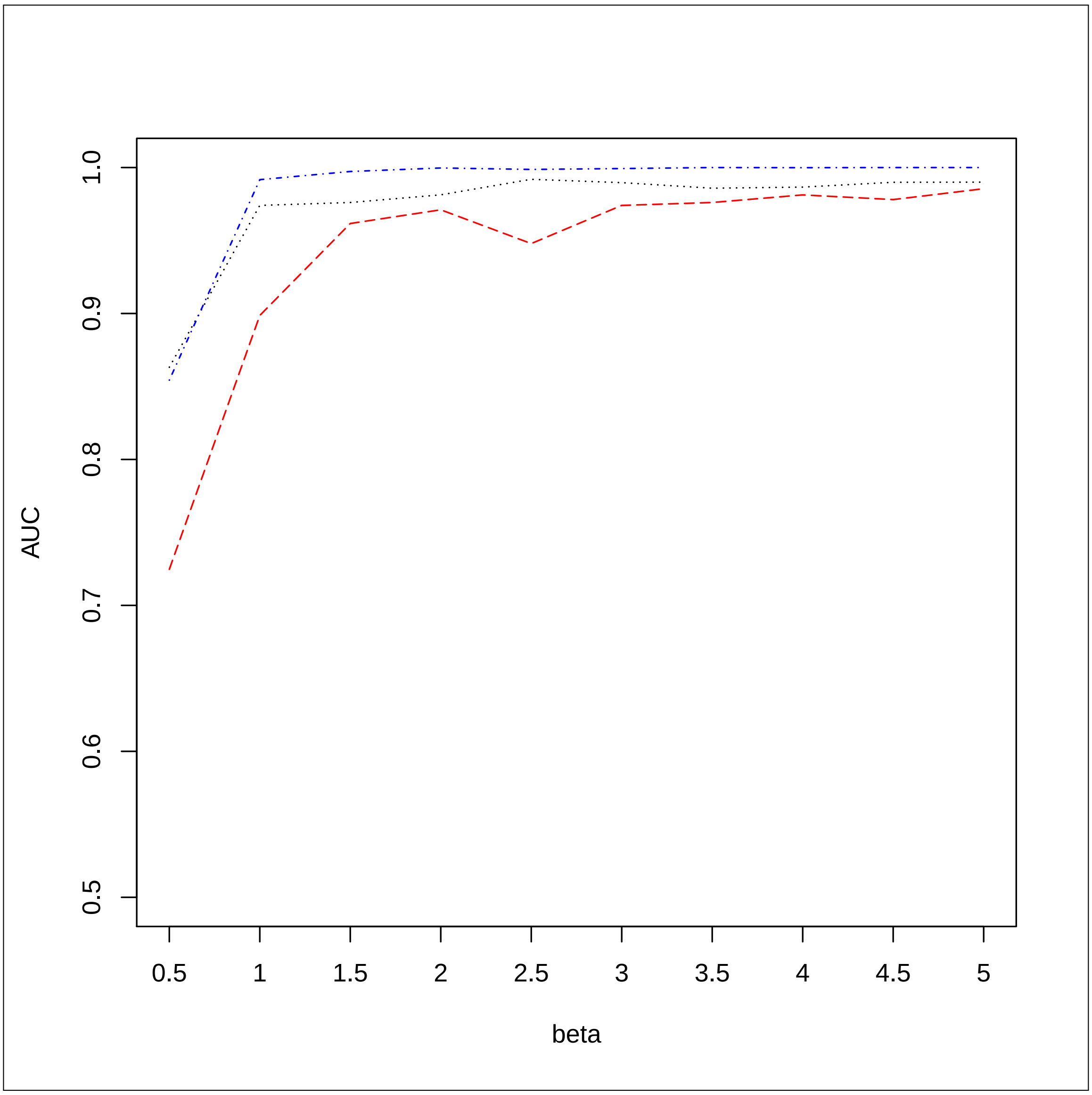


Figure 3.11: Plot of AUCs for increasing β in scenario 6. GSA-Dotted curve, BGSA-Dot-dash curve, GSEA-Dashed curve

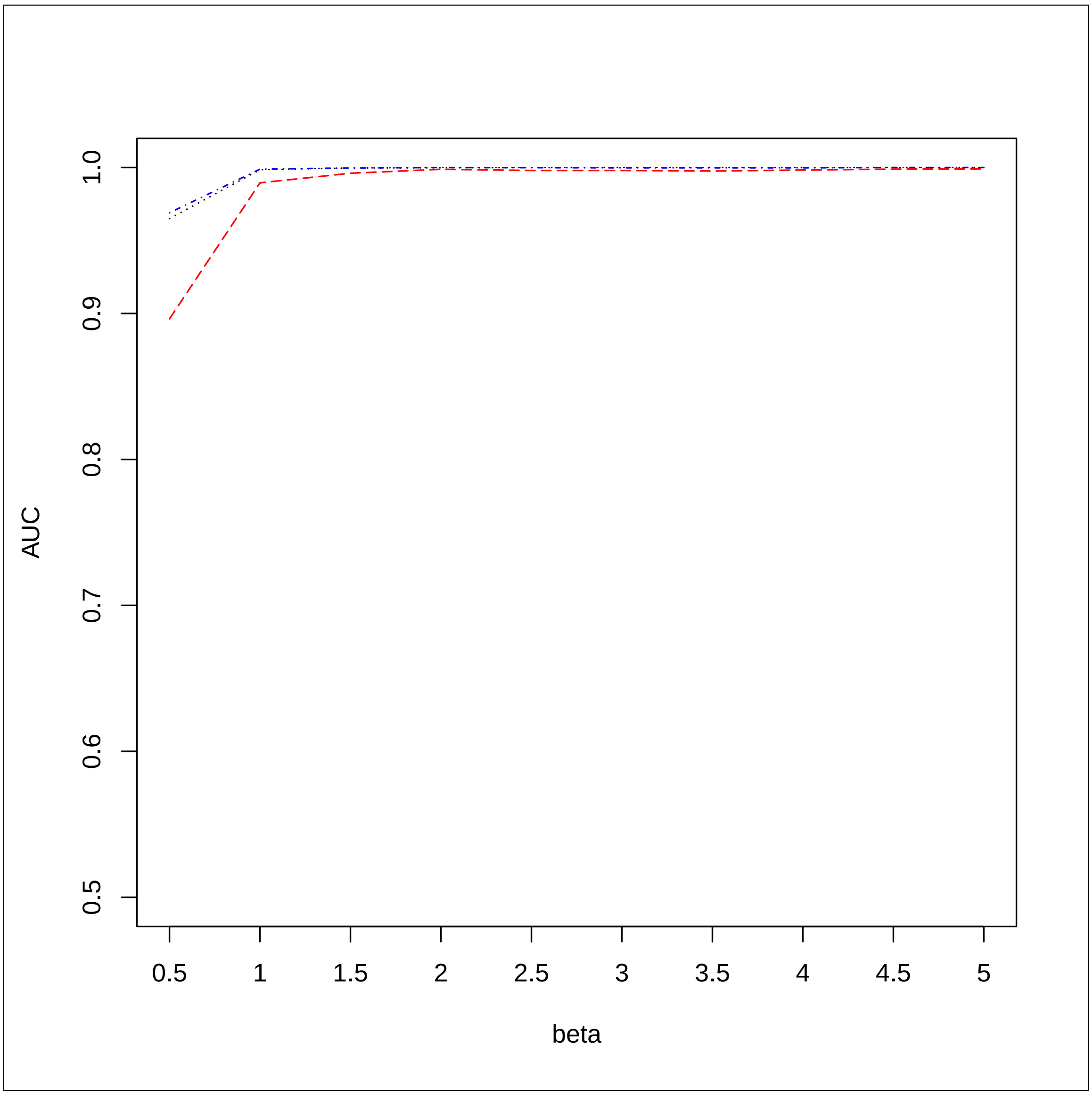


Figure 3.12: Plot of AUCs for increasing β in scenario 7. GSA-Dotted curve, BGSA-Dot-dash curve, GSEA-Dashed curve

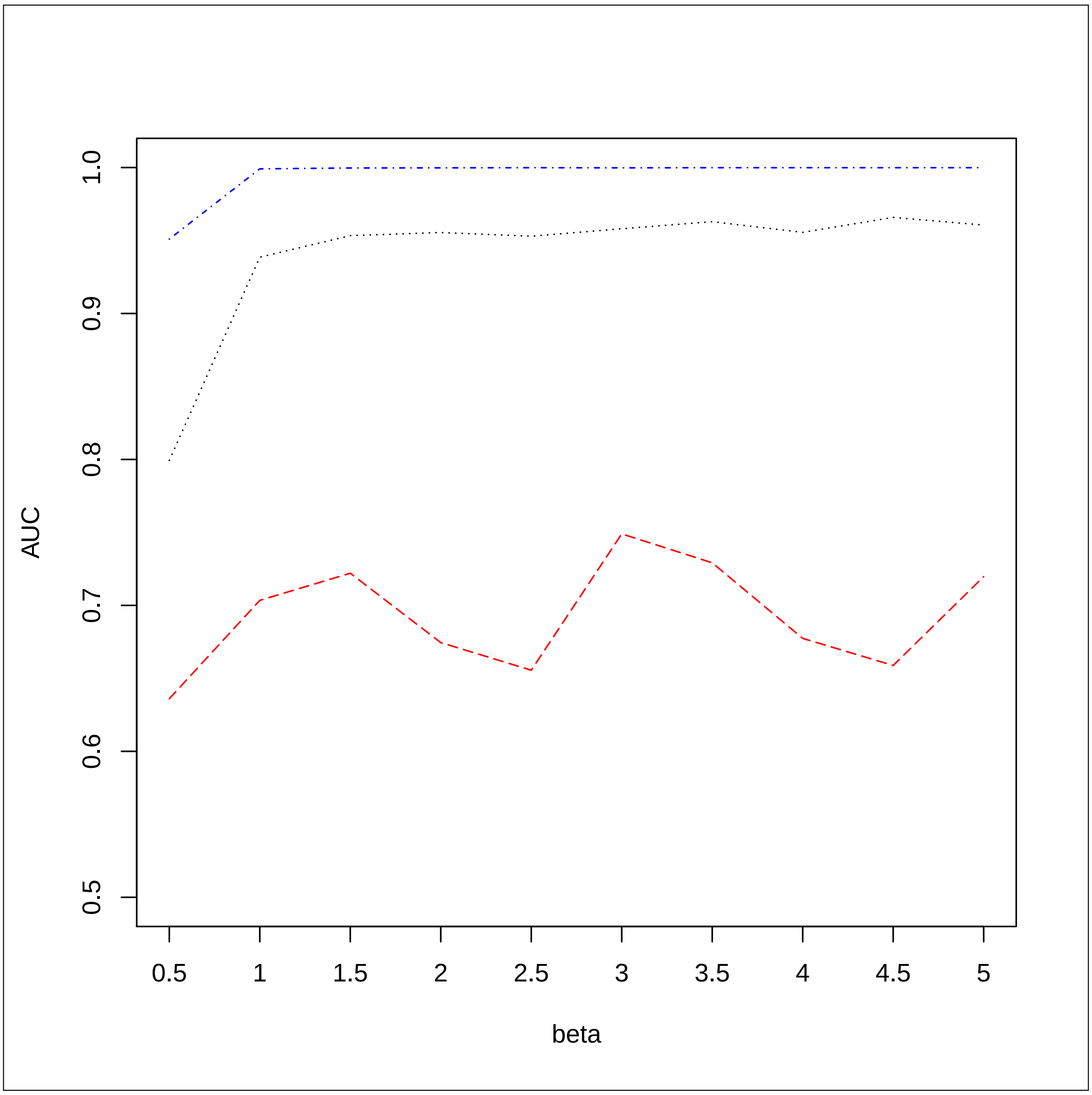


Figure 3.13: Plot of AUCs for increasing β in scenario 8. GSA-Dotted curve, BGSA-Dot-dash curve, GSEA-Dashed curve

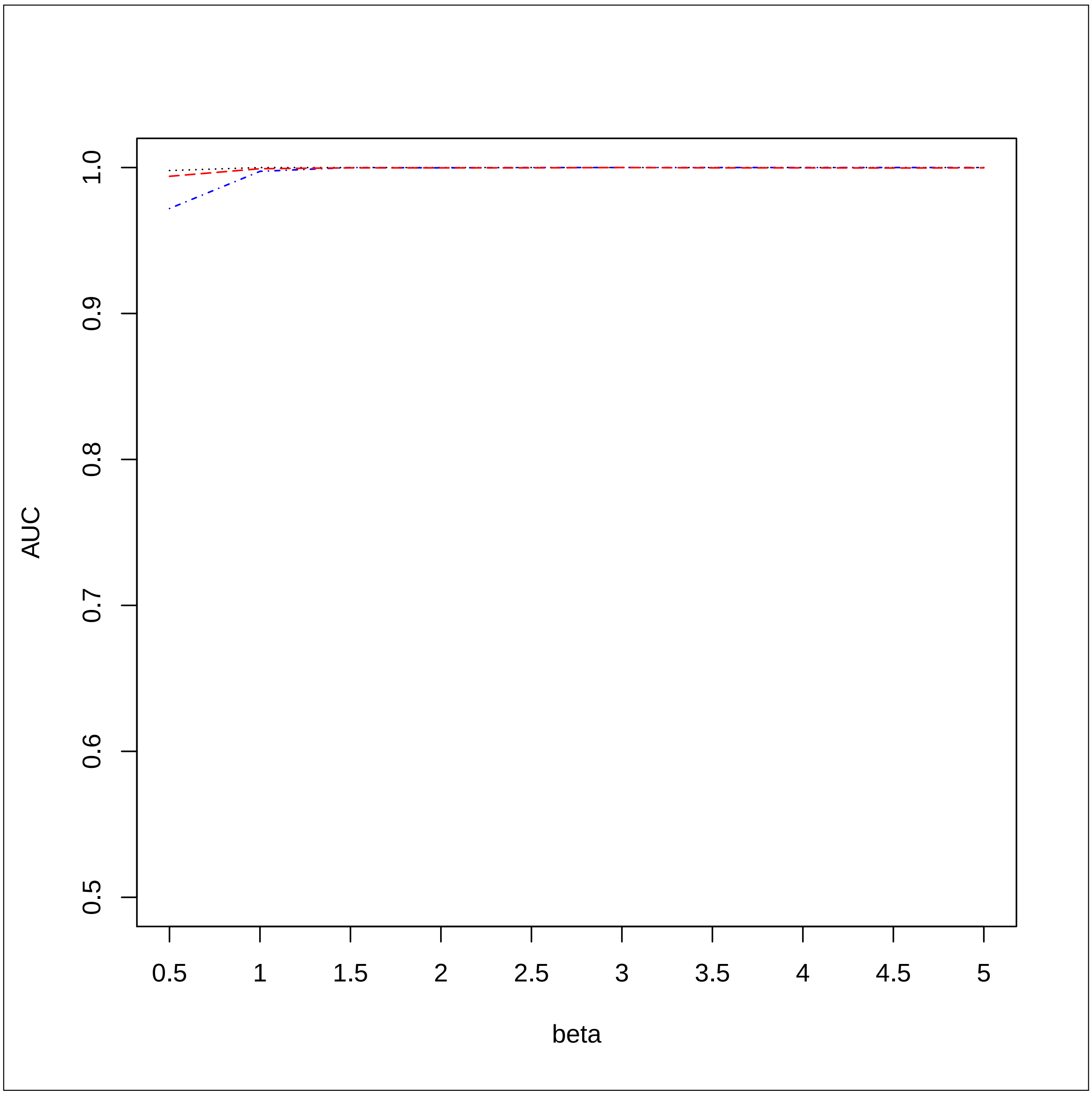


Figure 3.14: Plot of AUCs for increasing β in scenario 9. GSA-Dotted curve, BGSA-Dot-dash curve, GSEA-Dashed curve

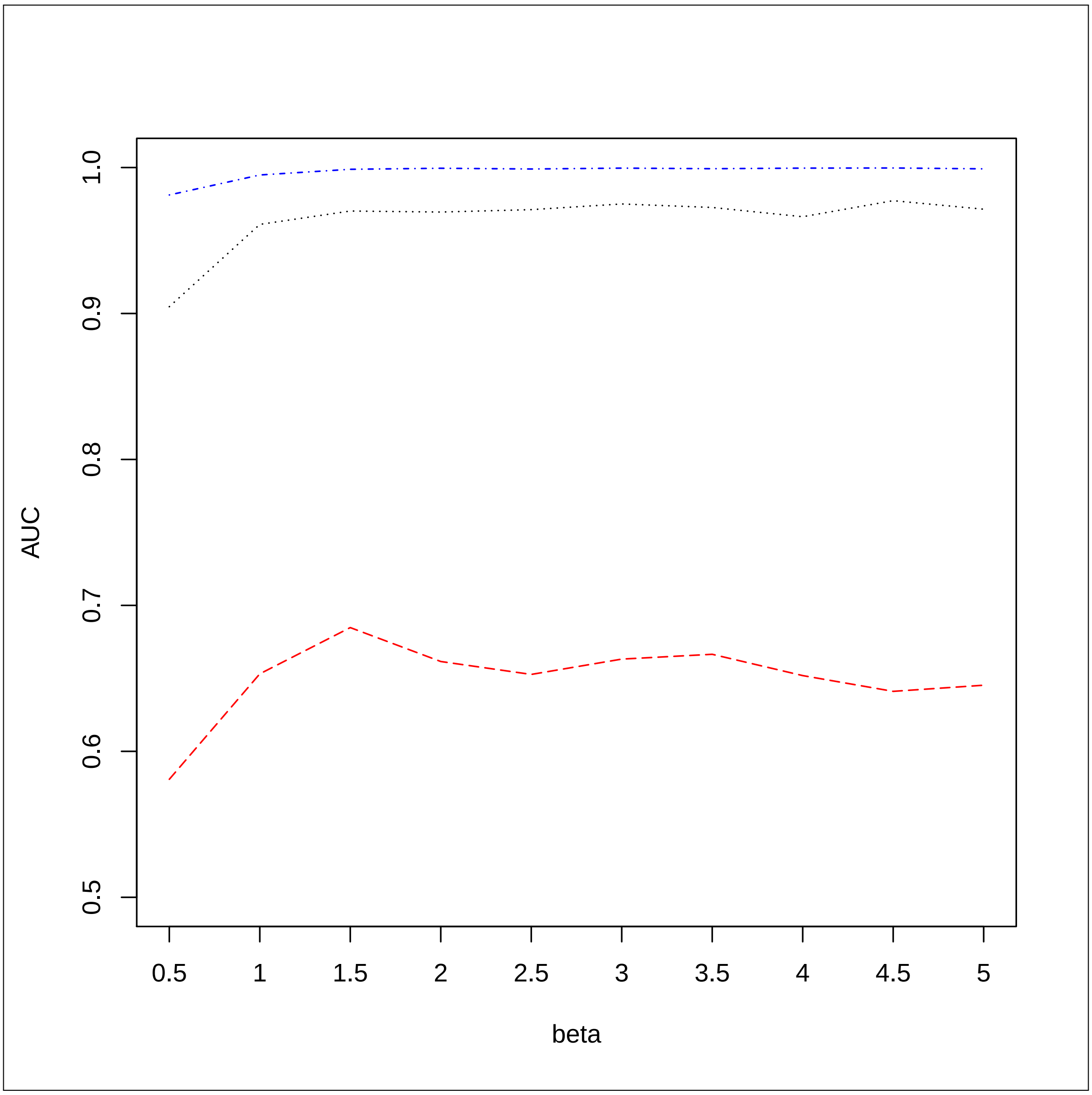


Figure 3.15: Plot of AUCs for increasing β in scenario 10. GSA-Dotted curve, BGSA-Dot-dash curve, GSEA-Dashed curve

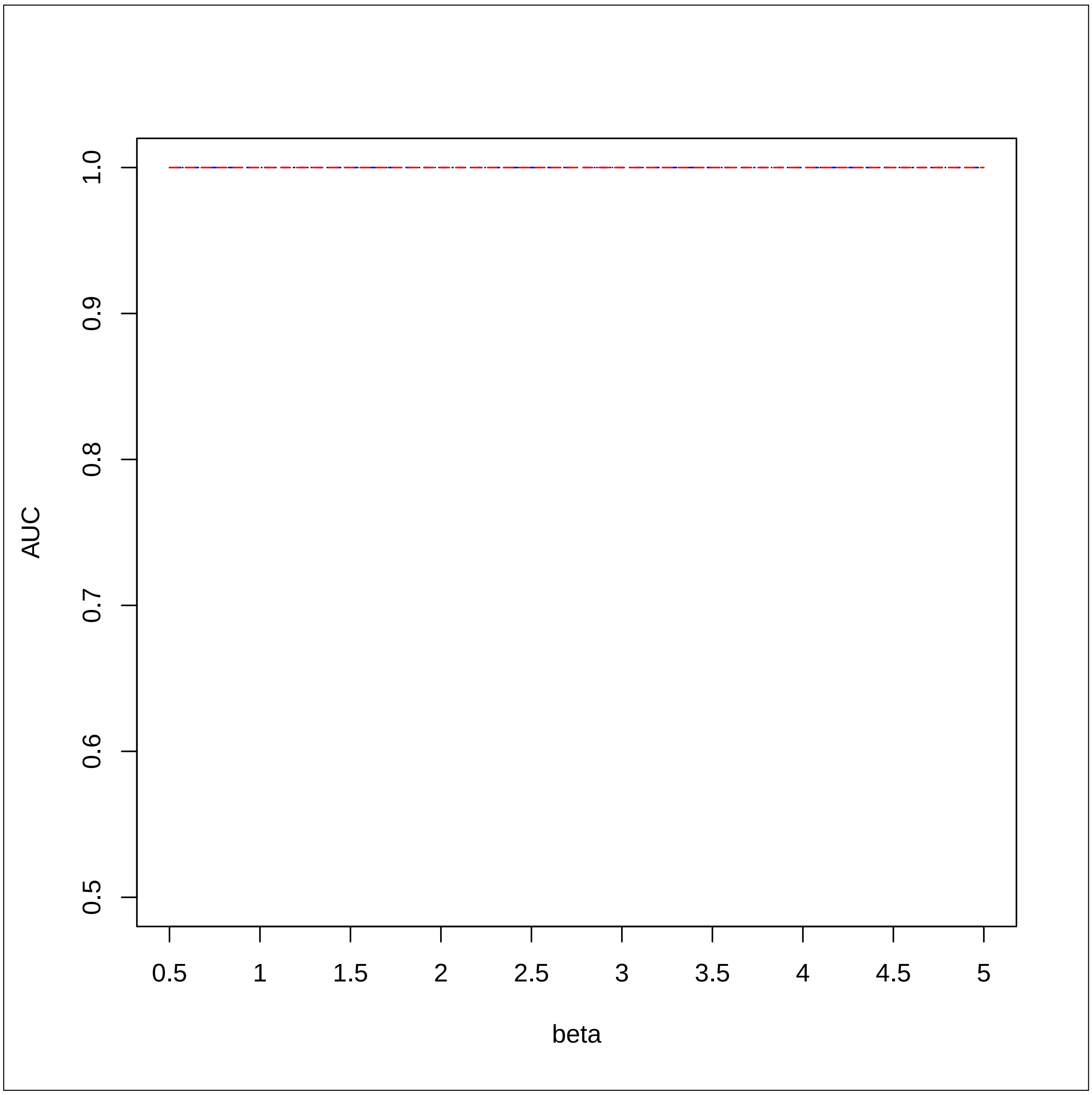


Figure 3.16: Plot of AUCs for increasing β in scenario 11. GSA-Dotted curve, BGSA-Dot-dash curve, GSEA-Dashed curve

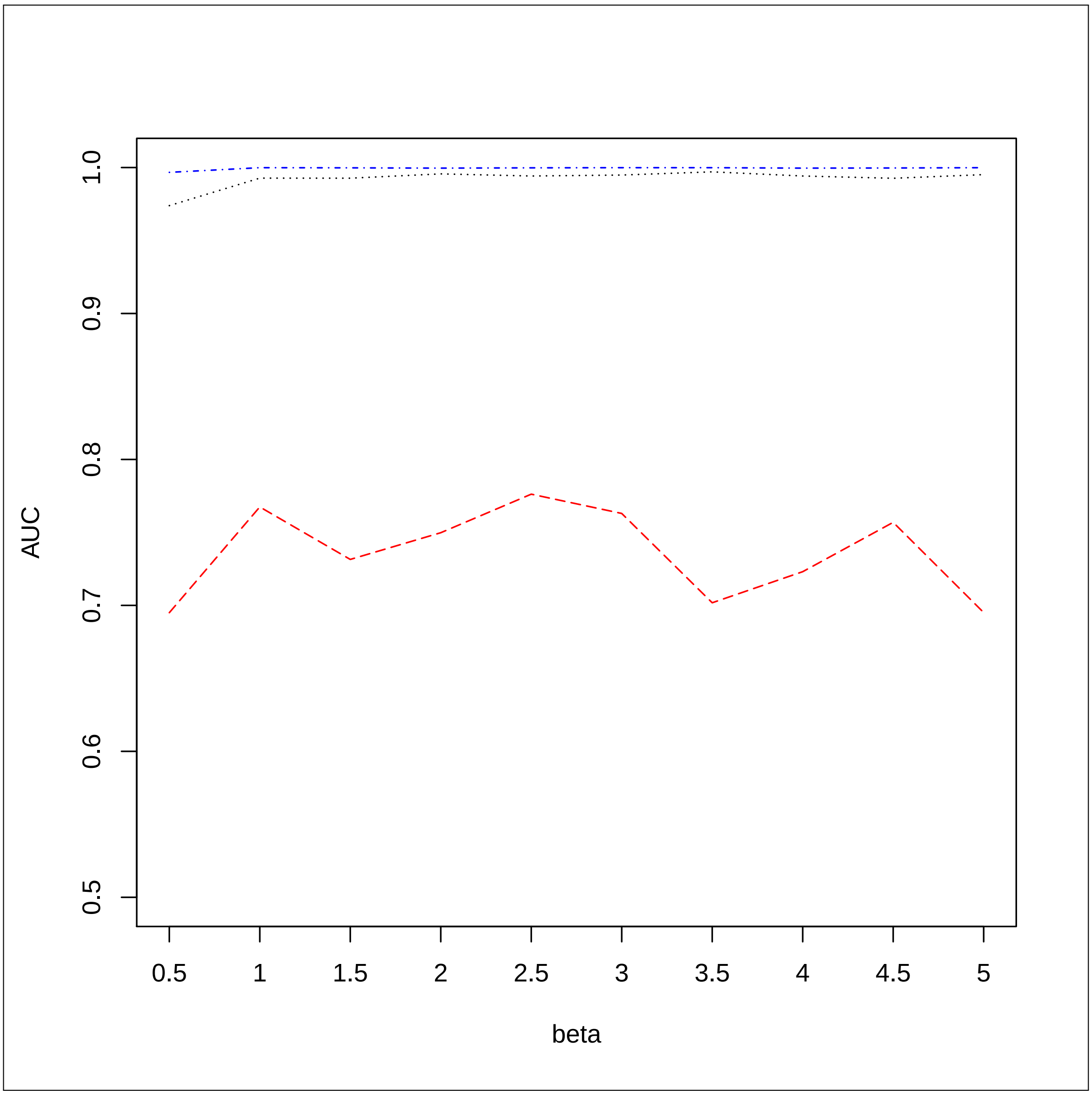


Figure 3.17: Plot of AUCs for increasing β in scenario 12. GSA-Dotted curve, BGSA-Dot-dash curve, GSEA-Dashed curve

3.3.2 Correlated data

Again, data are simulated according to Approach 2 for $n = 100$ subjects across $N = 1000$ genes, with each consecutive non-overlapping block of $m_s = 50$ genes constituting a gene set.

Here, however, it is the construction of Σ or more specifically the effect that it has upon the performance of the three methods that is of interest. For our $K = 20$ gene sets, Σ will be defined,

$$\Sigma = \begin{pmatrix} \Sigma_1 & 0 & 0 & 0 \\ 0 & \Sigma_2 & 0 & 0 \\ 0 & 0 & \ddots & 0 \\ 0 & 0 & 0 & \Sigma_K \end{pmatrix}$$

such that there is no correlation between gene sets, yet within gene set $s = 1, \ldots, K$ we have

$$\Sigma_s = \begin{pmatrix} 1 & \rho & \cdots & \rho \\ \rho & 1 & \cdots & \rho \\ \vdots & \vdots & \ddots & \vdots \\ \rho & \rho & \cdots & 1 \end{pmatrix}$$

i.e. we define within gene set pairwise correlations of ρ for all genes within the set.

Clearly this correlation structure is an over-simplification, yet it is as realistic as possible whilst keeping a simple structure that is easily controlled. By definition gene expression within a gene set will be correlated, regardless of whether the gene set is related to phenotype. Although real data would possibly show correlations between gene sets, there is no reason to define gene set to gene set correlation in this case and in doing so there are many considerations, such as:

- How do we define gene set to gene set correlation whilst keeping full control and a realistic correlation structure?

- What is the rationale behind the between gene set correlation structure?

- In defining a full correlation structure methods may fail to work at all due to the complexity of the data.

Ten scenarios are defined by numbers of active genes, size of gene effects and most interestingly by correlation of gene expression. We define one gene set as active where:

1. Twenty genes within the set are active, with effects on the linear predictor for the logistic model of 0.5;
2. Twenty genes within the set are active, with ten effects on the linear predictor for the logistic model of 0.5 and ten effects of -0.5;
3. Twenty genes within the set are active, with effects on the linear predictor for the logistic model of 1;
4. Twenty genes within the set are active, with ten effects on the linear predictor for the logistic model of 1 and ten effects of -1;
5. Twenty genes within the set are active, with effects on the linear predictor for the logistic model of 3;
6. All fifty genes within the set are active with effects on the linear predictor for the logistic model of 0.5;
7. All fifty genes within the set are active with twenty five effects on the linear predictor for the logistic model of 0.5 and twenty five effects of -0.5;
8. All fifty genes within the set are active with effects on the linear predictor for the logistic model of 1;
9. All fifty genes within the set are active with twenty five effects on the linear predictor for the logistic model of 1 and twenty five effects of -1;
10. All fifty genes within the set are active with effects on the linear predictor for the logistic model of 3.

To form sub-scenarios we vary ρ 0(0.05)0.5. We therefore end up with $11 \times 10 = 110$ different data simulation criteria. To each of these 110 data simulation criteria 100 data sets will be simulated and the three methods applied, such that for every sub-scenario we have 100 repetitions of each method.

3.3.2.1 Results

Figures 3.18 to 3.23 show plots of the AUC for each of the ten scenarios for increasing within gene set pairwise gene correlation (ρ). As can be seen in all ten scenarios the performance of each method decreases with increasing pairwise correlation. In every case the performance of BGSA is the best and that of GSEA is the worst. Similar to the uncorrelated scenarios as illustrated in the previous section, the more active genes there are in the set and the larger the gene effect is, the better the methods perform.

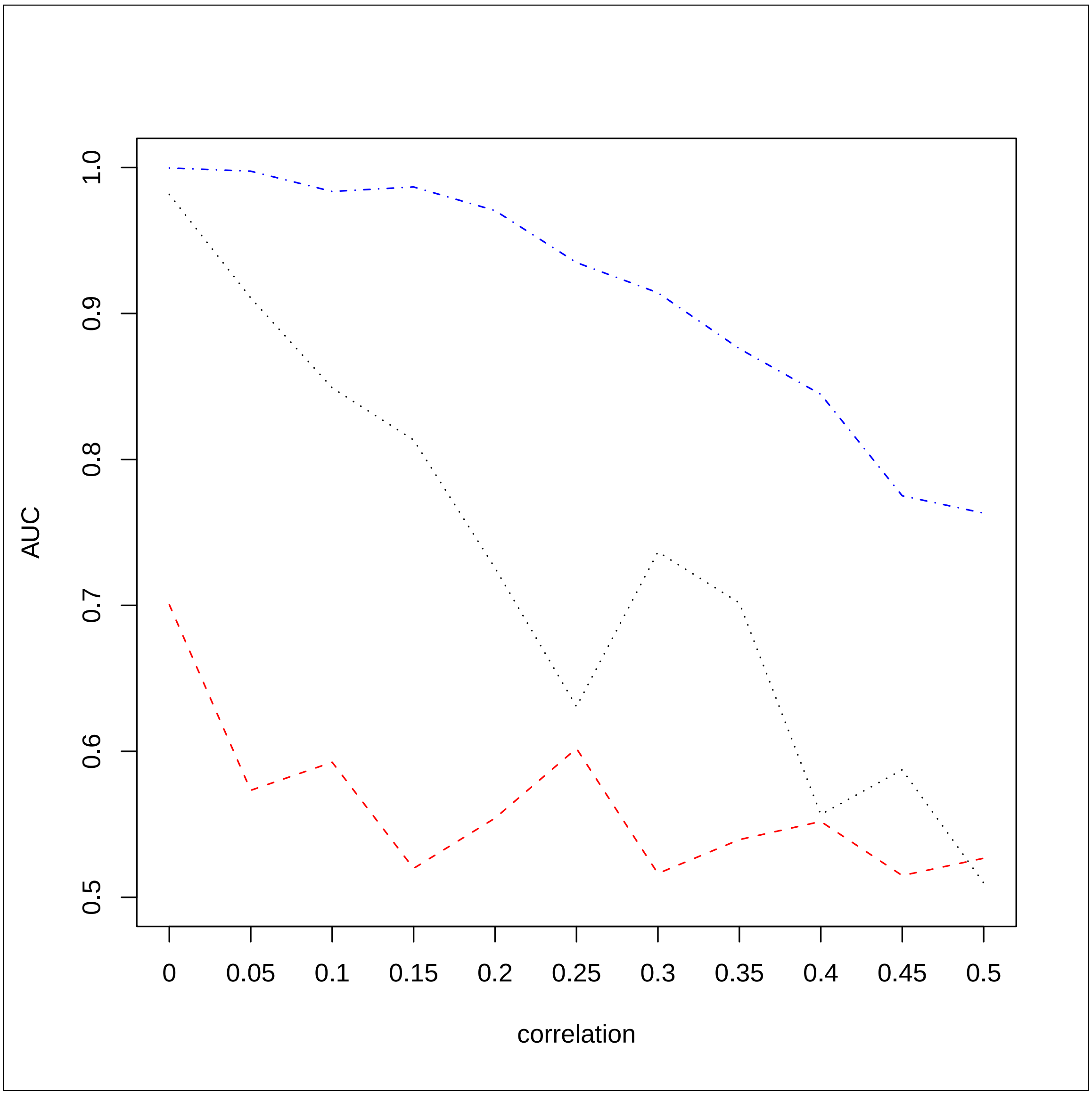


Figure 3.18: Plot of AUCs for increasing ρ in scenario 1. GSA-Dotted curve, BGSA-Dot-dash curve, GSEA-Dashed curve

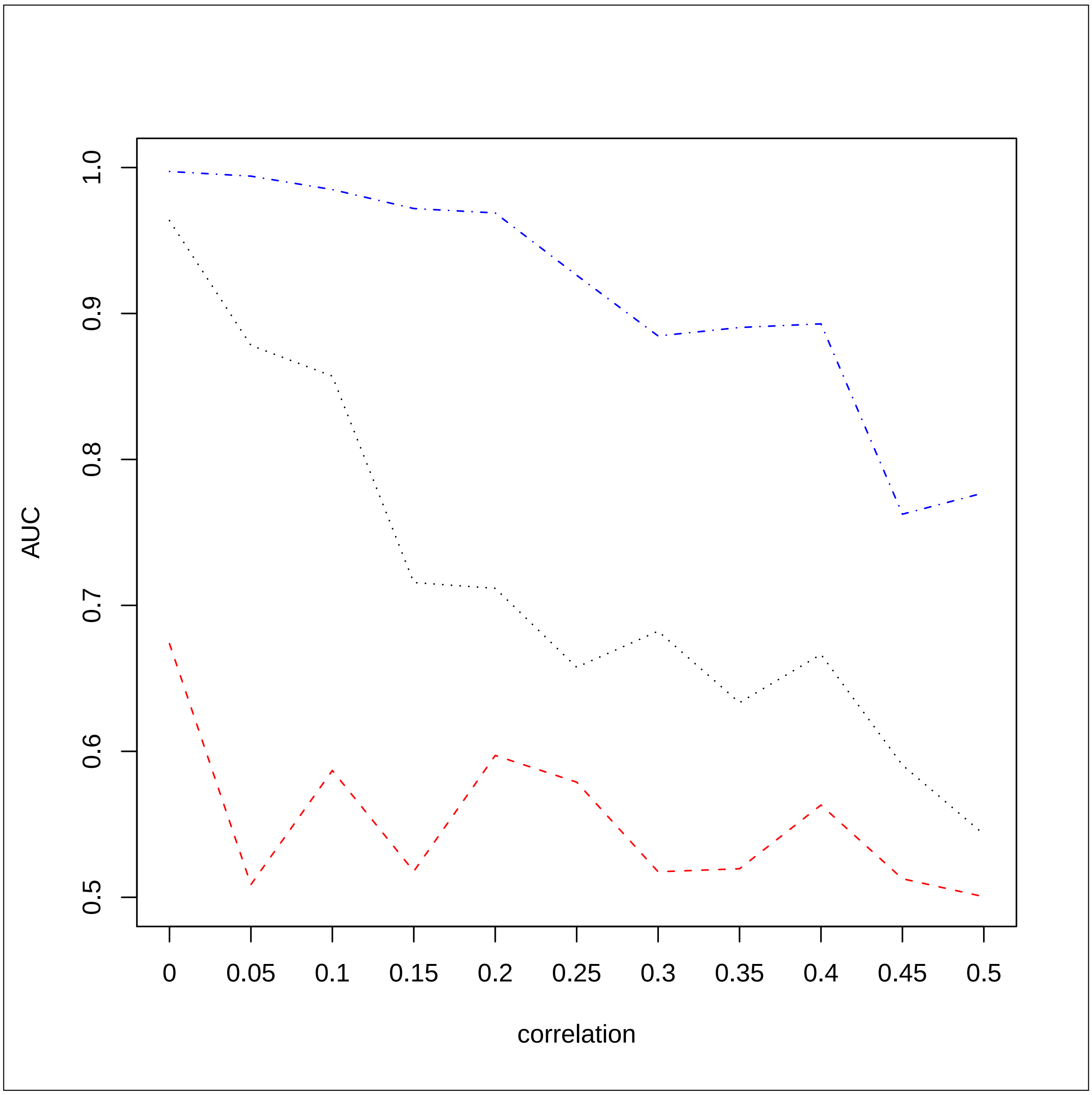


Figure 3.19: Plot of AUCs for increasing ρ in scenario 2. GSA-Dotted curve, BGSA-Dot-dash curve, GSEA-Dashed curve

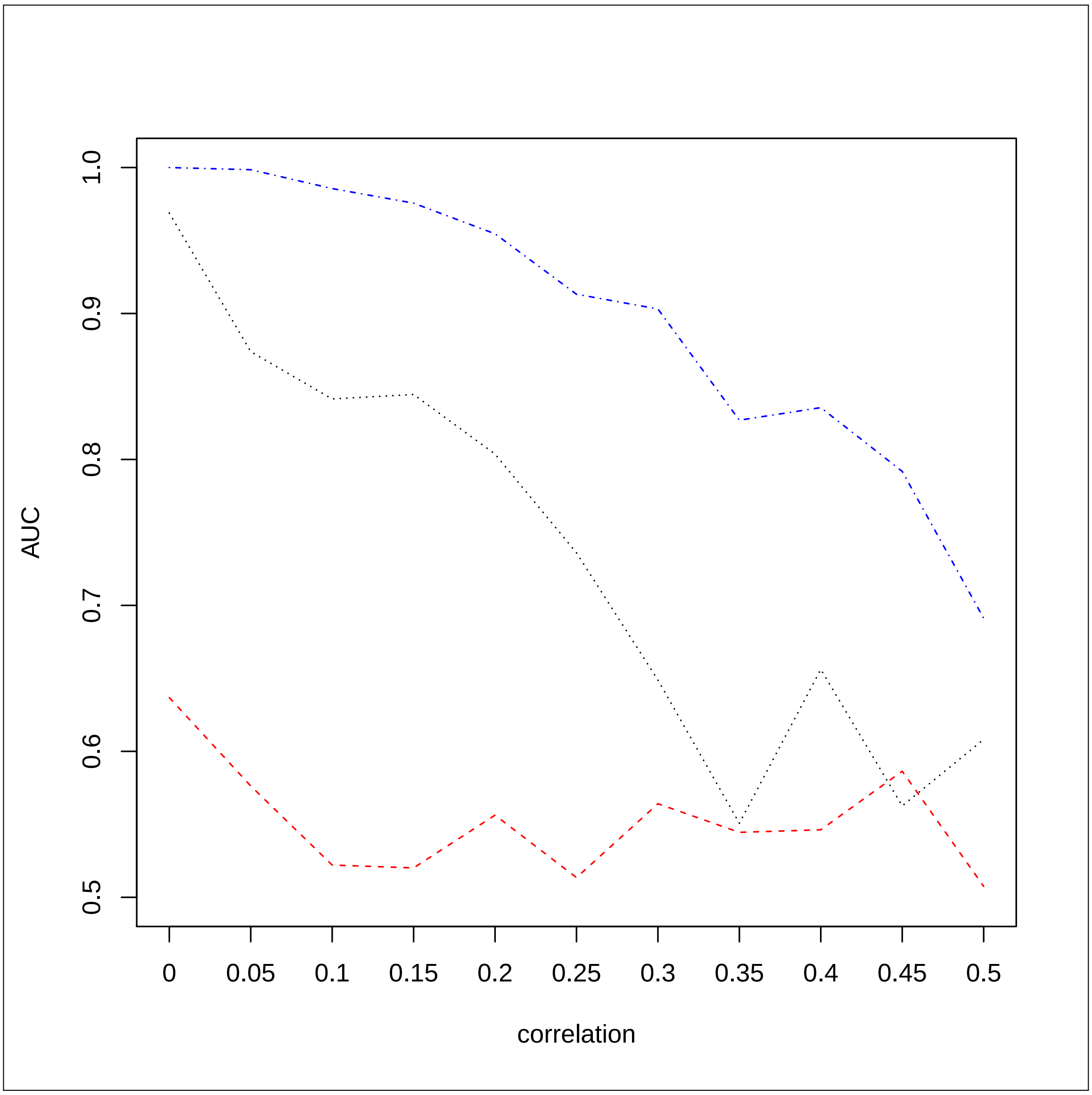


Figure 3.20: Plot of AUCs for increasing ρ in scenario 3. GSA-Dotted curve, BGSA-Dot-dash curve, GSEA-Dashed curve

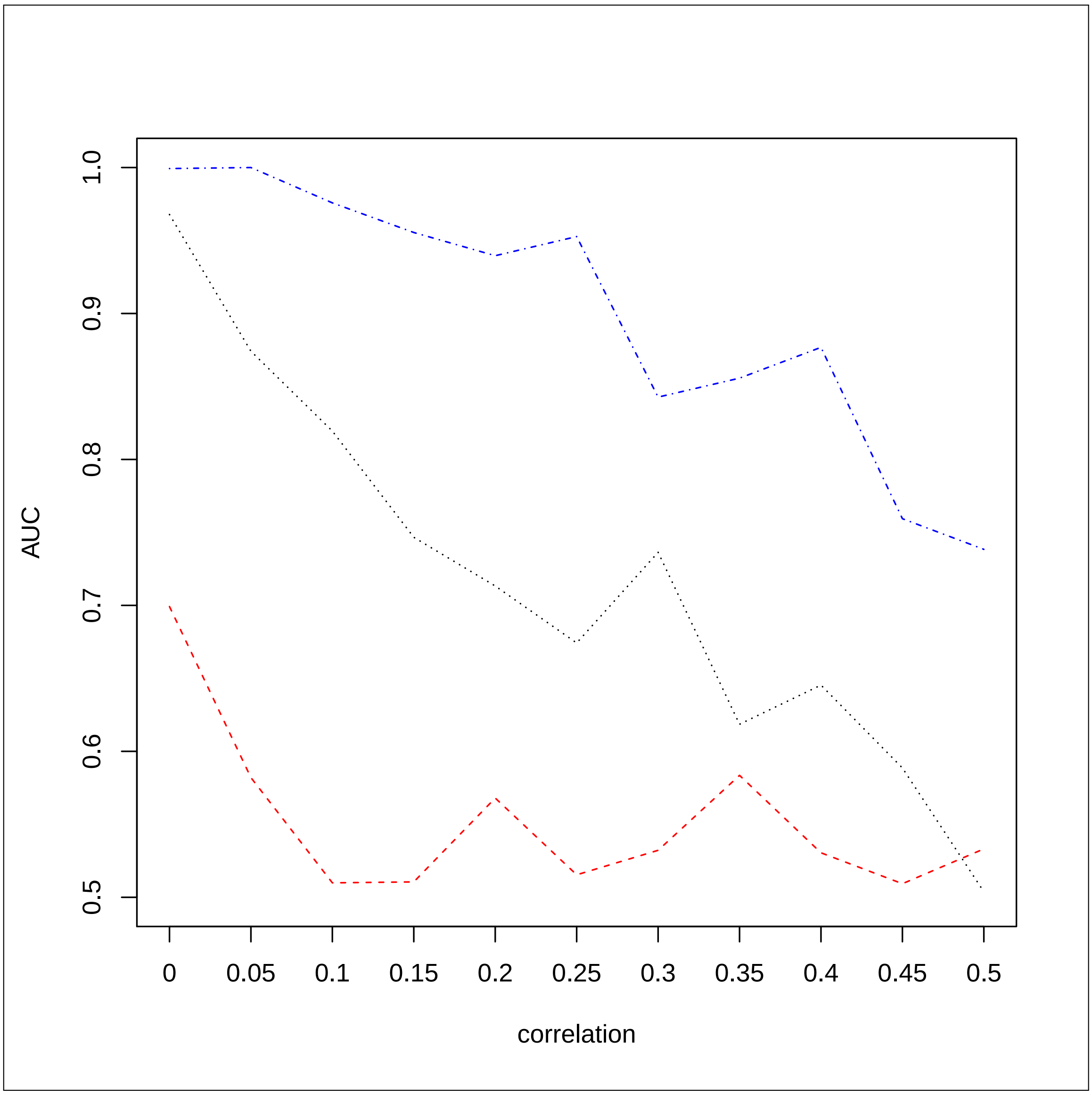


Figure 3.21: Plot of AUCs for increasing ρ in scenario 4. GSA-Dotted curve, BGSA-Dot-dash curve, GSEA-Dashed curve

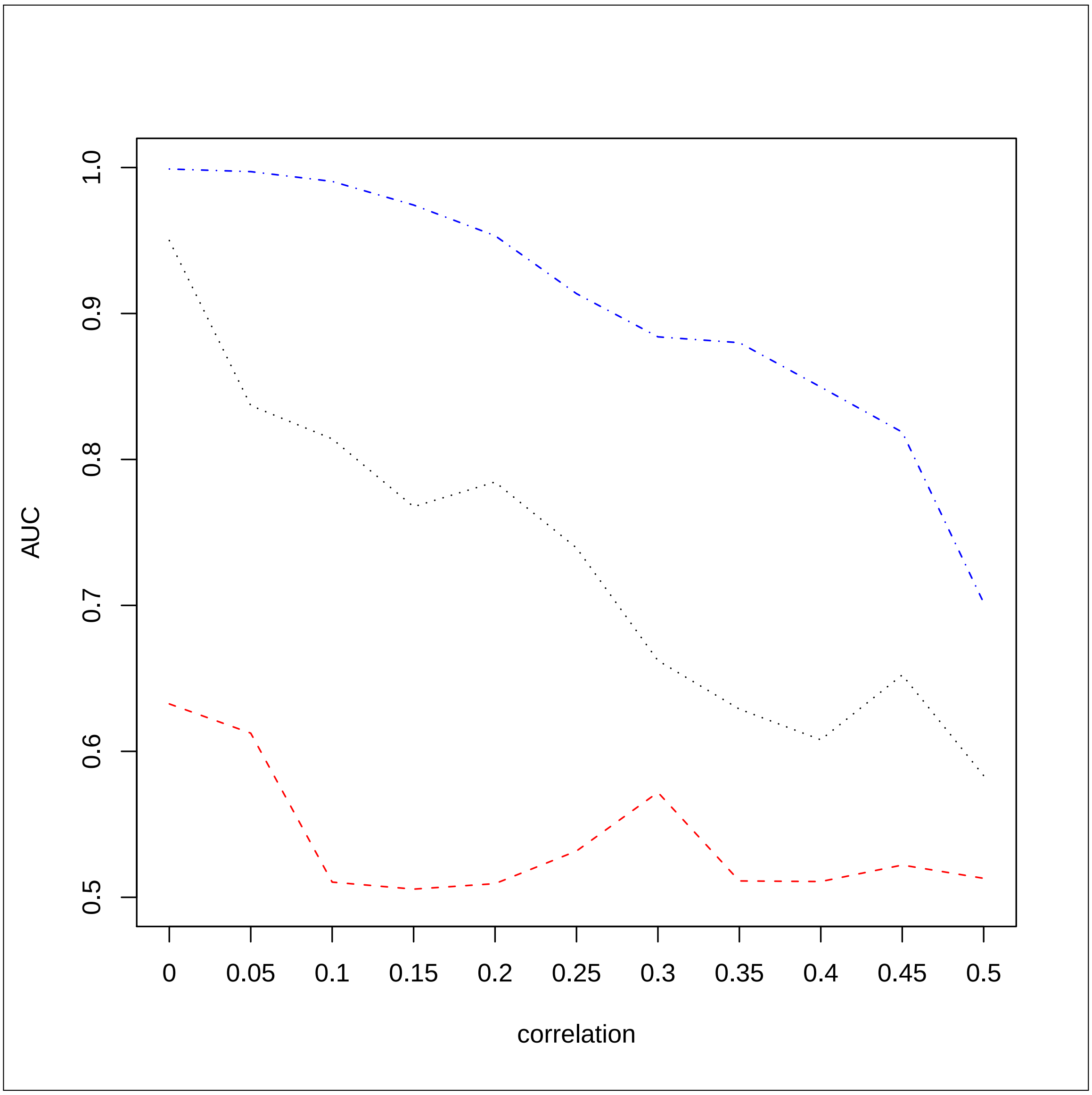


Figure 3.22: Plot of AUCs for increasing ρ in scenario 5. GSA-Dotted curve, BGSA-Dot-dash curve, GSEA-Dashed curve

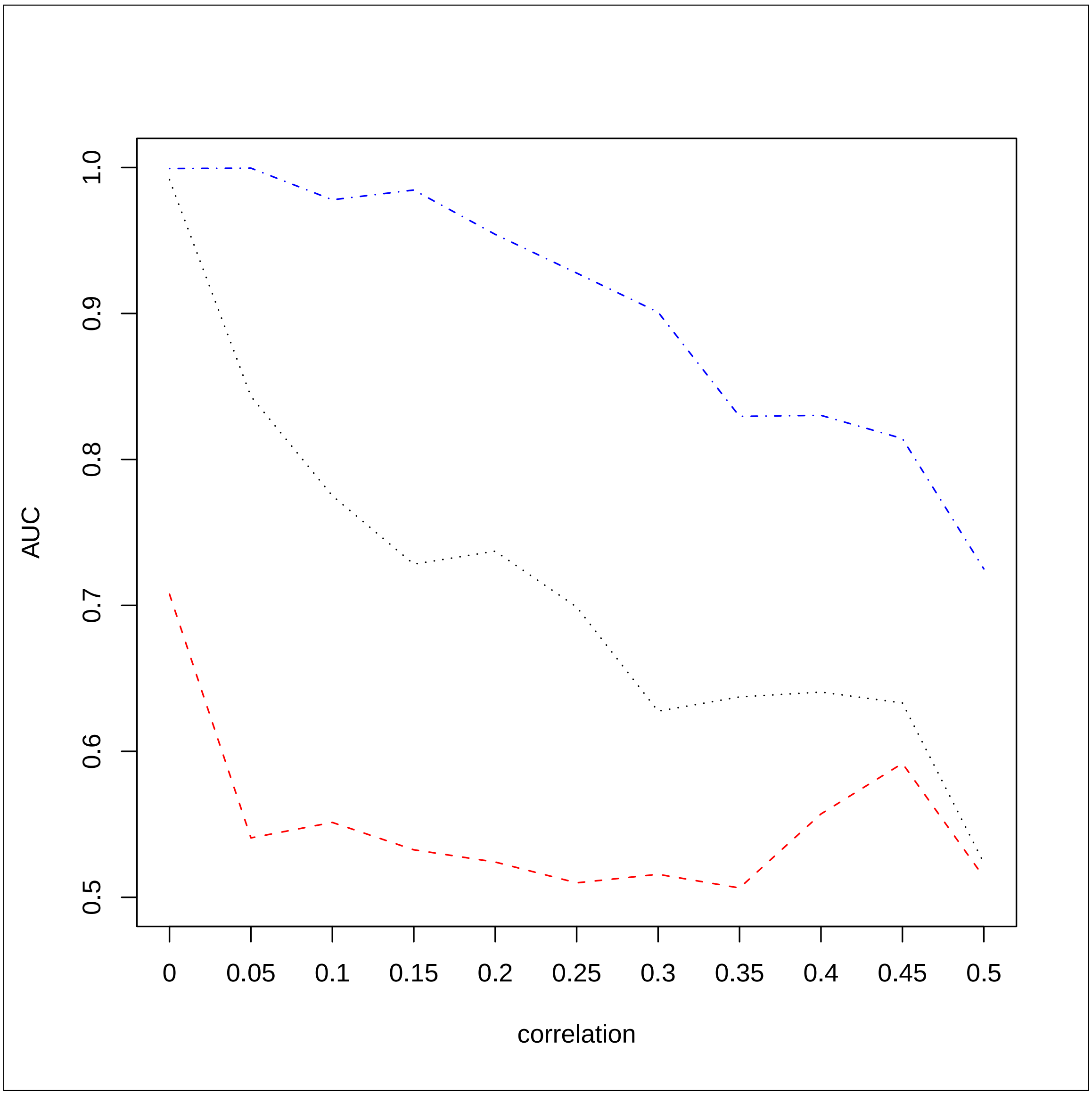


Figure 3.23: Plot of AUCs for increasing ρ in scenario 6. GSA-Dotted curve, BGSA-Dot-dash curve, GSEA-Dashed curve

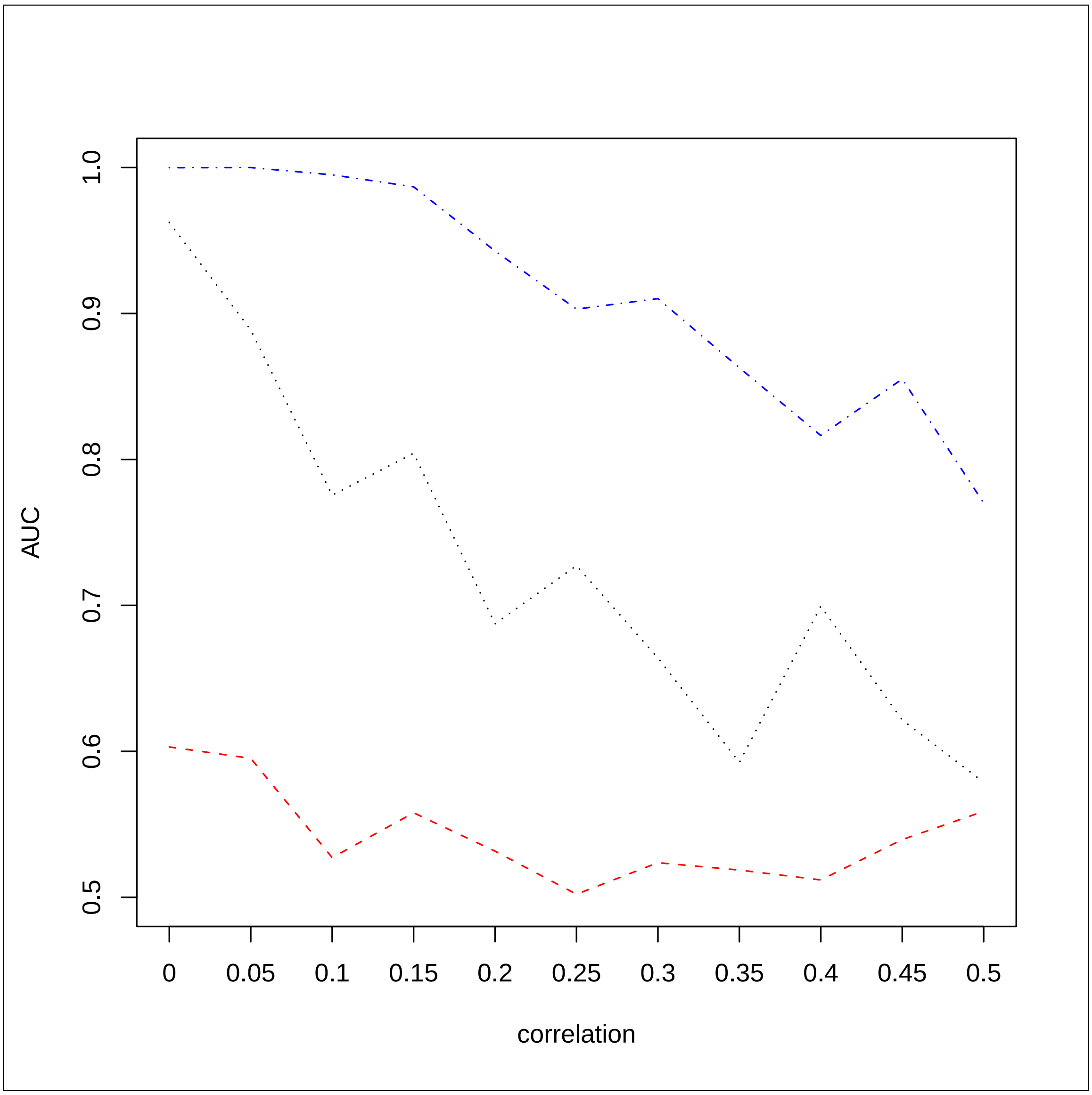


Figure 3.24: Plot of AUCs for increasing ρ in scenario 7. GSA-Dotted curve, BGSA-Dot-dash curve, GSEA-Dashed curve

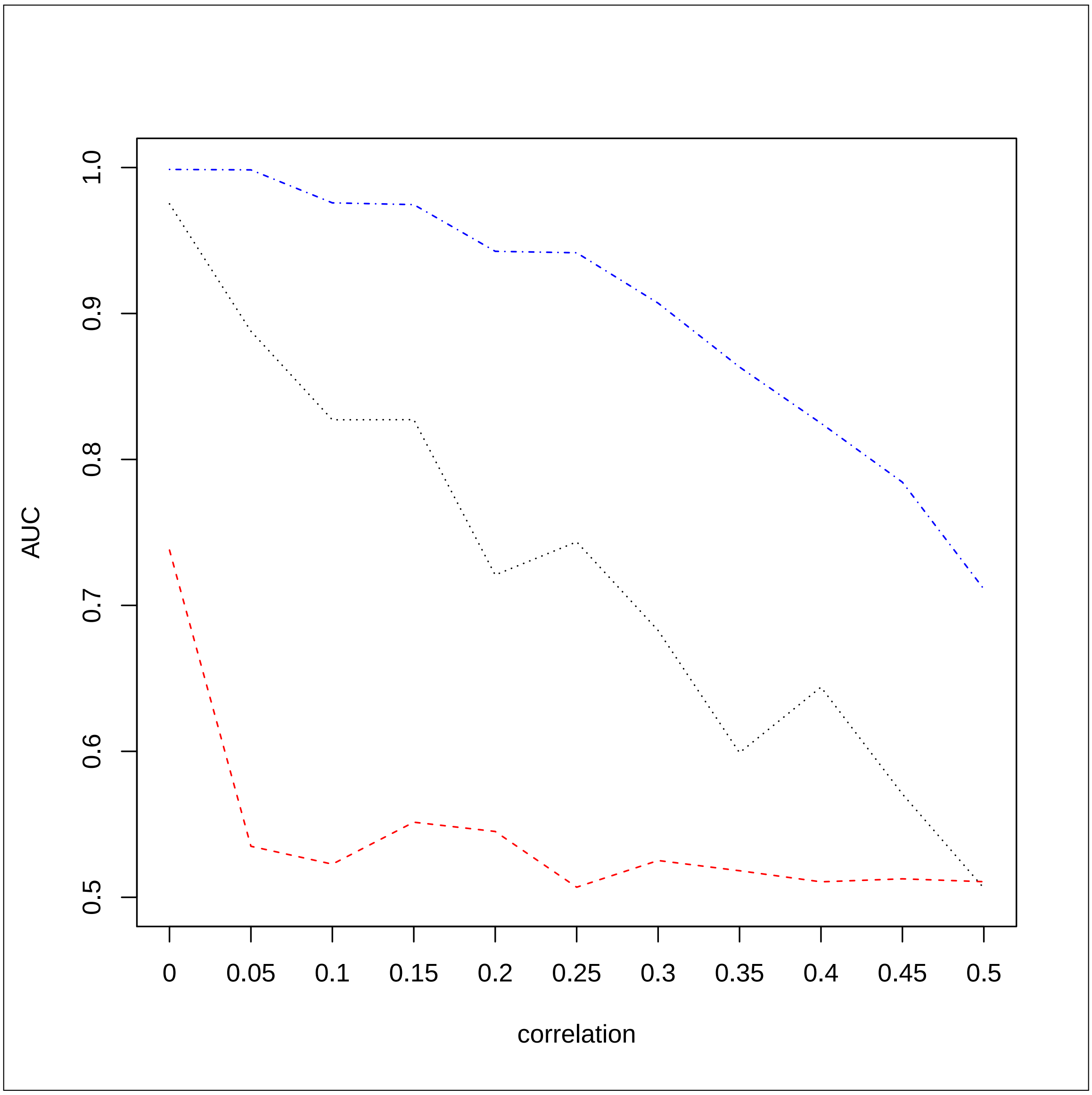


Figure 3.25: Plot of AUCs for increasing ρ in scenario 8. GSA-Dotted curve, BGSA-Dot-dash curve, GSEA-Dashed curve

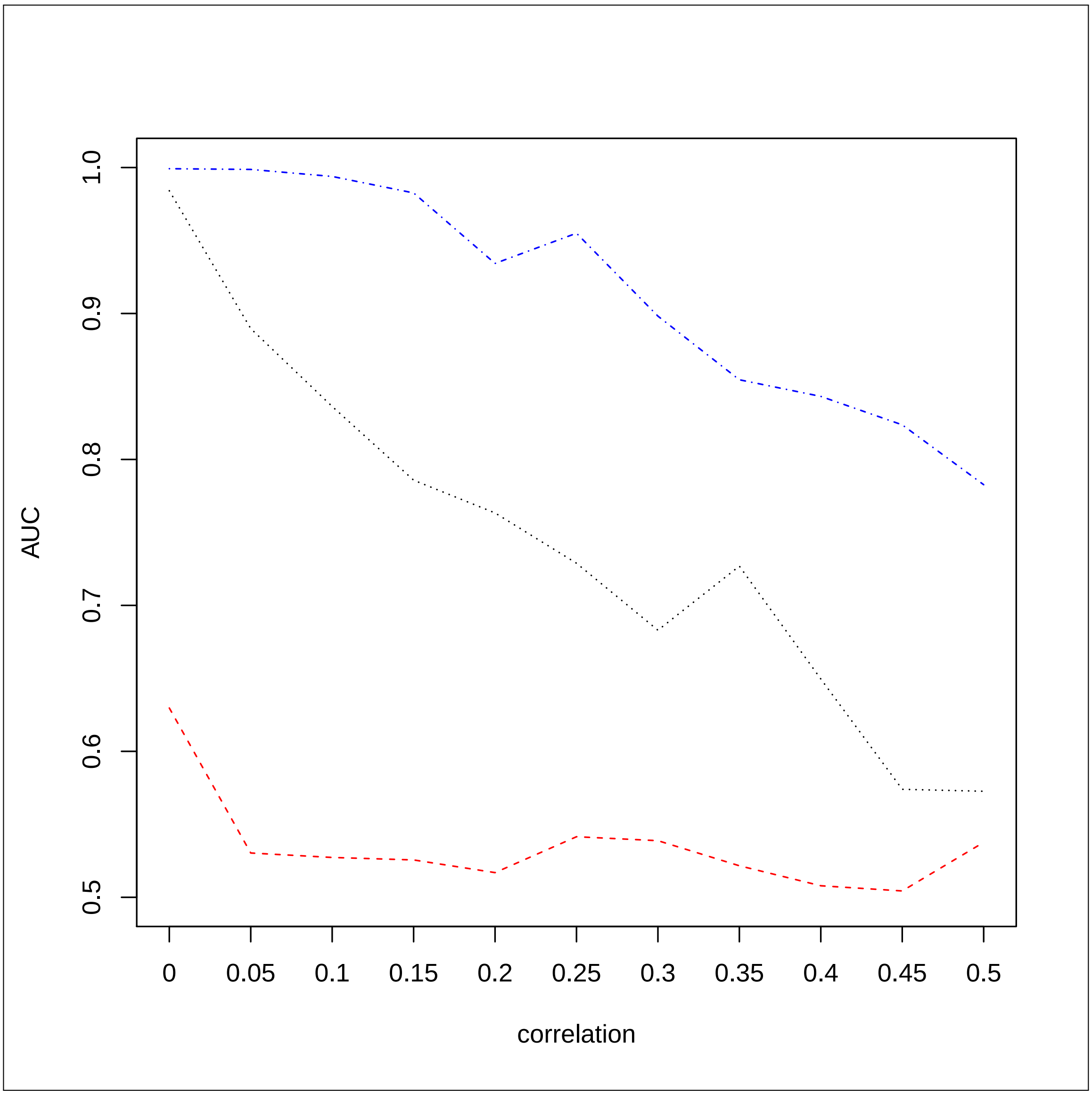


Figure 3.26: Plot of AUCs for increasing ρ in scenario 9. GSA-Dotted curve, BGSA-Dot-dash curve, GSEA-Dashed curve

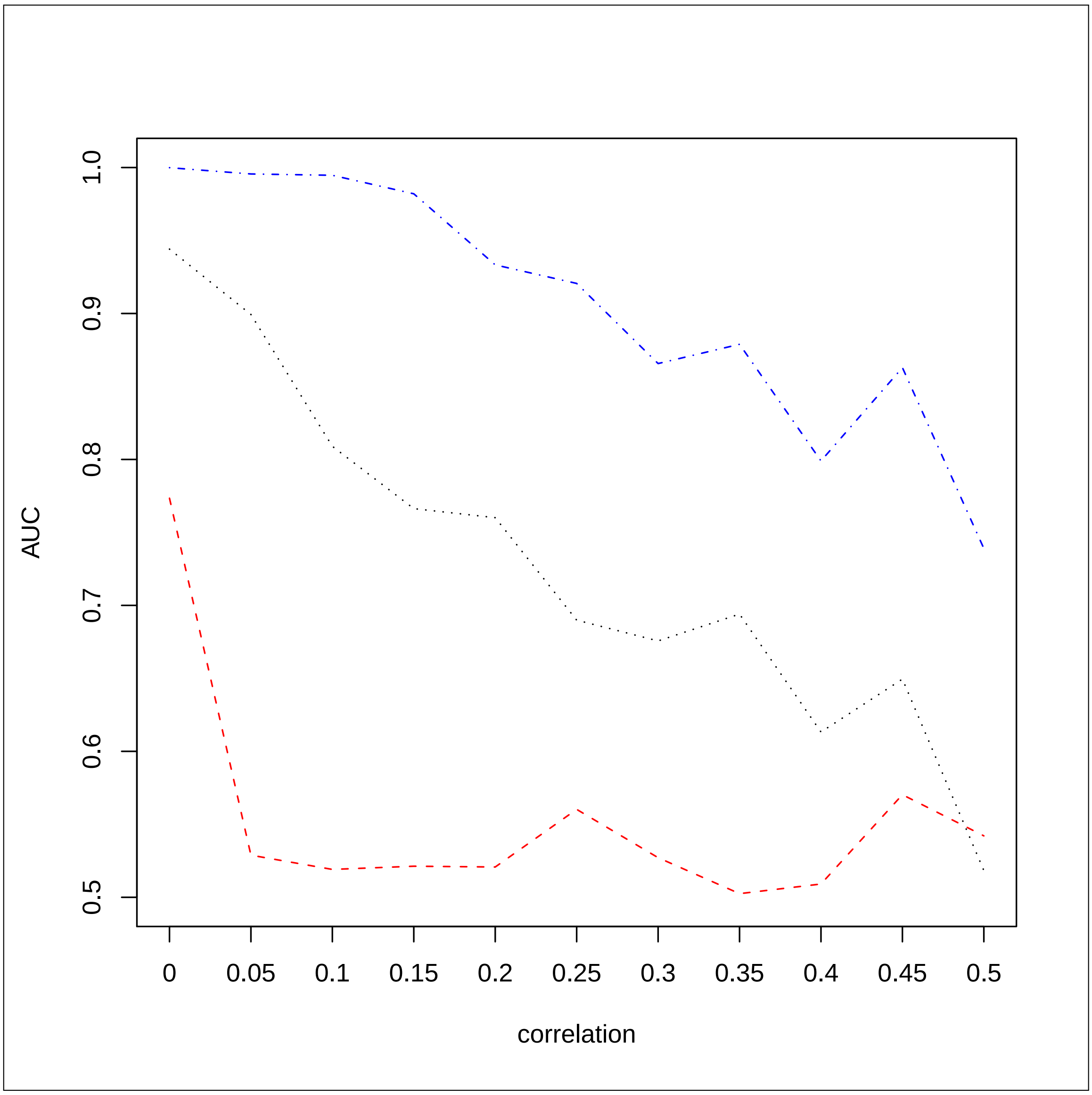


Figure 3.27: Plot of AUCs for increasing ρ in scenario 10. GSA-Dotted curve, BGSA-Dot-dash curve, GSEA-Dashed curve

3.4 Discussion and Conclusions

The data simulation approach as introduced above (Approach 2) produces realistic data whereby gene expression directly affects phenotype, where phenotype is modelled conditionally on gene expression and can be influenced simultaneously by any number of genes. The vast majority of published simulation studies use some close variant of Approach 1 as a data simulation procedure, such as Efron and Tibshirani (2006), Nam and Kim (2008) and Song and Black (2008).

Song and Black (2008) bring a new level to simulation Approach 1 in adding a correlation structure. However, in their simulation procedure it is only active gene sets that are given a correlation structure, whilst null gene sets are assumed a collection of independent samples. Approach 2 defines a correlation structure for all gene sets, regardless of gene expression. This is because a gene set is constructed due to some functional relationship between its member genes regardless of differential expression. It is therefore safe to assume that the expression of the member genes would be correlated to some degree. It could be argued as to why between gene set correlation structure is not also included, as it is reasonable to believe that in real data there will be between gene set correlation. For the purpose of the simulation study presented here this added level of complexity would not add enough to the study and brings to bear the question of how to define such a complex structure in a realistic manner.

Song and Black (2008) represent a realistic situation, whereby they measure the mean expression and the variance matrix of real data, and from such simulate data according to these measures. In future work it would be interesting to carry this out, but using simulation Approach 2 such that we can also model phenotype rather than using fixed values.

It can be seen that BGSA performs very well for scenarios where data is uncorrelated. In general BGSA performs the best out of the three methods in identifying the active gene set, with consistently higher AUCs than the other methods. Due to the construction of the model, when there are both positive and negative gene effects within the active gene set BGSA actually performs better than when all effects are in the same direction. This is because the model relies upon the variance of gene effects within the gene set to be a measure of activeness of a gene set.

It can also be seen that GSEA shows the worst performance. In cases where there are both positive and negative gene effects GSEA performs poorly, with little better than random prediction of the active gene set. The simulation study as outlined in Section 3.3 clearly demonstrates that GSEA is not the most reliable method to use in real life applications as it is likely to miss any gene sets that combine both up- and down-regulated genes, which are common.

GSA performs well on uncorrelated data. However, to a lesser degree, the method does show the same characteristic drop in performance when there are both positive and negative gene effects as in the case of GSEA. Concerns with GSA's tendency to demonstrate such behavior are expressed by Shahbaba et al. (2011) and are clearly shown here. However, GSA is still a method to be rated and with the computational cost of running analyses being so low it is useful.

It could be argued, that in the case of uncorrelated data, BGSA is somewhat oversensitive and when there is only one active gene within a set, once the effect becomes moderately large ($\beta = 2$) the set is identified every time. GSEA does seem to be somewhat better calibrated than BGSA and GSA whereby larger percentages of active genes with larger effects within a set are needed before the set is judged significantly enriched. This brings us to the argument of how an active gene set is defined. This is a difficult question and answers must be somewhat fluid, one solution to the above question would be to use probabilities rather than p-values. This would allow uncertainty about the above question to be expressed within the measure of activeness of the gene set.

It can be seen that the three methods show a characteristic drop in performance as the within gene-set pairwise correlation increases. This is because the three methods assume that a gene set is a collection of independent samples from the entire list of genes. The majority of biologically defined gene sets will be a collection of correlated genes, i.e. gene sets will not be made up of a random collection of independent genes. This assumption of independence boosts gene set scores when there is in fact a within gene set correlation structure. This will therefore increase the incidence of false positives, thus decreasing the performance of the methods as gene-gene correlations increase.

As an illustration of the above remark, say that we have gene sets s_k, $k = 1, 2, K$, each containing m_k genes, where s_1 is related to phenotype but $s_2, \ldots, s_K$ are not. Furthermore say that genes within a gene set have pairwise correlations of ρ and that there is no between gene set correlation.

GSEA ranks these genes by their effect upon phenotype (using t-test statistics) and calculates a running sum statistic according to gene set membership. Permutation of class label is used to generate an empirical null distribution and t-test statistics calculated for each gene within each permutation. Due to correlation between genes the t-statistics from a given gene set will be ranked closely together in the list. There will therefore be gene sets with no effect on phenotype that have many t-statistics near the top of the list and will therefore be judged enriched.

Similarly, GSA, which uses the maxmean statistic to detect enrichment within a gene set also gives many false positive results when dealing with correlated data. If gene expression within a null set is highly correlated, then the t-test statistics will also be correlated, and so instead of the case with uncorrelated data, where we may see the occasional false positive result for a single gene, this will be exaggerated to the whole or the majority of the gene set for correlated data. The maxmean statistic will therefore, in some cases, be particularly high for some null gene sets which will lead to false positive results.

In their paper Efron and Tibshirani (2006) show GSA to be superior to GSEA in detecting active gene sets on data simulated according to Approach 1 as described previously. Shahbaba et al. (2011) also show GSA to be superior to GSEA on their simulated data (simulation described in Chapter 2) and show BGSA to be superior to both GSEA and GSA on the same simulated data. The simulation study presented here covers many more scenarios than presented in previous papers and simulated data represent more realistic situations than those presented by Efron and Tibshirani (2006) due to the conditioning of the phenotype on expression. The simulation study presented by Shahbaba et al. (2011) uses real data and so the truth behind whether the genes are related to phenotype, and how, is unknown. The simulation study presented here clearly demonstrates when and where the given methods work and when they fail.

All three of the methods studied determine whether a gene set is active due to some arbitrary p-value or FDR cut-off. Their performance can be assessed independently of these cut-offs using ROC curves and the AUC as a summary of the ROC. However, where the aim is to identify groups of genes worthy of further investigation these cut-off based methods are somewhat black and white. To say ‘this gene set is related to pathway’ or ‘this gene set is not related to pathway’ is a little too certain and a more fluid definition is required.

A demonstrative example of this is in scenario 1 as can be seen in Section 3.3 where there is one active gene in the gene set, once the effect gets larger than 2 then BGSA perfectly classifies the gene set as active, whilst GSA and GSEA show reasonable success in classifying the set as significant. Now, ignoring the argument that a gene set with only one active gene should not be classed as active, the three methods investigated here- particularly BGSA- would class this set as active in many instances. To put this set in the same bracket, for example, as a set with 10 active genes is very misleading. However, if we had a probability that this set were active, it is much more interpretable and useful. In defining a fully Bayesian model for the analysis of gene sets posterior probabilities of ‘activeness’ of gene set could be determined, which would allow the investigator a means of identifying which sets are worthy of further investigation with regards to how likely it is that the set is related to phenotype. The univariate nature of the three methods is also something that should be addressed. If a model were designed such that phenotype is modelled conditionally on gene expression we would be able to discard the assumption of independence, thus constructing a model that can take correlation structure into account.

Chapter 4

A Bayesian/frequentist hybrid model: A model comparison approach to Pathway analysis

As discussed previously, many current methods for the analysis of sets of genes make use of ad-hoc analytical methods. In the development of an original model it is useful to go back to first principles in order to structure a model that makes practical sense.

Some problems with current methods have been highlighted in previous chapters here and in several papers, such as Nam and Kim (2008), Goeman and Buhlmann (2007) and Song and Black (2008). There seem to be three major drawbacks to the majority of gene set analyses:

- Gene sets tend to be assumed a collection of independent genes, i.e. uncorrelated. This results in a drop in performance of the majority of methods as within gene set correlation increases;
- Many methods struggle to deal with both up- and down-regulated genes;
- The majority of methods assume a univariate relationship in many cases conditioning gene expression on phenotype, rather than phenotype on condition.

We are presented with the problem of identifying the presence of a relationship between a pre-defined set of genes and a dichotomous phenotype. Some gene sets will have an effect upon phenotype and some will not. Within a gene set that has an effect upon phenotype there will be genes whose expression directly effects phenotype and some genes whose expression will not affect phenotype. By definition the expression of genes within a curated gene set will be correlated to some degree, whether those genes are differentially expressed or not.

A method should be developed that overcomes the above outlined problems and relates to the biology behind the process of a pathway affect on phenotype. A possible way of doing this would be to use a model comparison approach where we define models that fit a gene set that has no effect on phenotype and models that fit a gene set that does have an effect on phenotype. It would be useful to define two models, one of which would best fit a gene set that has no effect upon phenotype and one which would best fit a gene set that does have an effect upon phenotype. The adequacy of the fit of these models could be quantified and from this the significance of a relationship could be estimated.

The proposed approach utilizes Bayesian model fitting to fit within gene set logistic regression models. There will be two models, defined by their priors. These will be a null model, where it is believed that there is no gene set effect, and an active model, where it is believed that there will be some gene effect. A model is defined such that there is a slope parameter for every gene within a gene set on a linear predictor for both logistic models. The adequacy of the fit of these models can be quantified and compared, and from this a frequentist style p-value computed.

Section 4.1 proceeds to introduce the model as mentioned above. For the Bayesian modelling part of the procedure (to be outlined), the posterior distribution for the slope parameters is intractable algebraically, hence Markov chain Monte Carlo methods must be employed to obtain samples from the posterior distribution. Section 4.2 presents several possible MCMC algorithms and discusses which algorithm is the most suitable by way of computer simulations.

4.1 The Model

The following proceeds to introduce a model comparison approach to gene set analysis. In order for a model comparison two models are defined:

1. Null model - Designed to best fit gene sets that have no effect on phenotype;

2. Active model - Designed to best fit gene sets that have a relationship with phenotype.

Using MCMC techniques these two models are fitted to each gene set simultaneously. It should be noted that we do not define a fully multivariate model, yet neither is a univariate relationship assumed. A within gene set multivariate relationship is assumed.

Essentially there will be two stages to the modelling. The first stage will consist of the fitting of two logistic regression models to each gene set, the null model whereby we assume there is no gene set effect, and the active model whereby we assume that there are some gene set effects. The second stage is a comparison of the two models. The null and active models can be contrasted by comparing the fit of the models, large, consistent differences indicating a gene set effect.

4.1.1 Stage one - Logistic regression modelling

As mentioned above two logistic models are to be defined, however both models will have the same likelihood. Looking within gene set s, containing m_s genes, the likelihood for such a model is given by

$$f(x|\beta) = \prod_{i=1}^{n} \pi_i^{y_i}(1-\pi_i)^{1-y_i} \tag{4.1}$$

where

$$\pi_i = \frac{1}{1+\exp\{-\eta_i\}} \tag{4.2}$$

and

$$\eta_i = \sum_{g=1}^{m_s} x_{ig}\beta_{sg} \tag{4.3}$$

where x_{ig} is the expression of the g^{th} gene in gene set s for the i^{th} subject and β_{sg} is the slope parameter for the g^{th} gene in gene set s. The difference in the two models is defined by their respective prior distributions, $f(\boldsymbol{\beta})$.

4.1.1.1 The null model

A prior distribution for the slopes, $\boldsymbol{\beta}$, for a null logistic model should represent a situation where the genes in the set have no effect on phenotype. It is unreasonable to suggest that all slopes within a null gene set are identically zero, as there will be noise in expression measurements and some genes within an inactive gene set may have a small effect on phenotype. However, it is reasonable to suggest that effects will be small and centered about zero. Therefore a normal prior of the form

$$f(\boldsymbol{\beta}) = \frac{1}{\sqrt{2\pi}\sigma} \exp\left\{ \frac{-\boldsymbol{\beta}^2}{2\sigma^2} \right\} \tag{4.4}$$

is used, where σ is small. This represents prior beliefs that for a gene set that has no relationship with phenotype and the majority of gene effects will be close to zero.

4.1.1.2 The active model

Within a gene set that has a relationship with phenotype it would be expected that there are a whole range of gene effects. There will be some genes that have no effect upon phenotype and some that have relatively large effects upon phenotype, both in negative and positive directions. The prior distribution for the slopes, $\boldsymbol{\beta}$, for an active logistic model should represent this. Therefore a prior of the form

$$f(\boldsymbol{\beta}) = \frac{1}{\sqrt{2\pi}} \left(\frac{(1-p)}{2\sigma_1} \exp\left\{ \frac{-(\boldsymbol{\beta}-\mu)^2}{2\sigma_1^2} \right\} + \frac{p}{\sigma_2} \exp\left\{ \frac{-\boldsymbol{\beta}^2}{2\sigma_2^2} \right\} + \frac{(1-p)}{2\sigma_1} \exp\left\{ \frac{-(\boldsymbol{\beta}+\mu)^2}{2\sigma_1^2} \right\} \right) \tag{4.5}$$

is used. Figure 4.1 shows a conceptual picture of prior distribution for $\boldsymbol{\beta}$ for an active gene set. As can be seen this allows for very small effects, as it is believed that within an active pathway not all genes will be up- or down-regulated. The broad peaks in the extremes represent the belief that the up- or down-regulation of certain genes will have a positive or negative effect upon the linear predictor for the logistic model The posterior distribution of $\boldsymbol{\beta}$ is intractable algebraically, hence MCMC techniques should be employed. Section 4.2 discusses possible MCMC samplers and algorithms.

4.1.2 Stage two - Model comparison

The second stage of modelling relies heavily upon measures of posterior deviance and so it is useful to first define deviance. Deviance (D) can be used as a measure of model fit

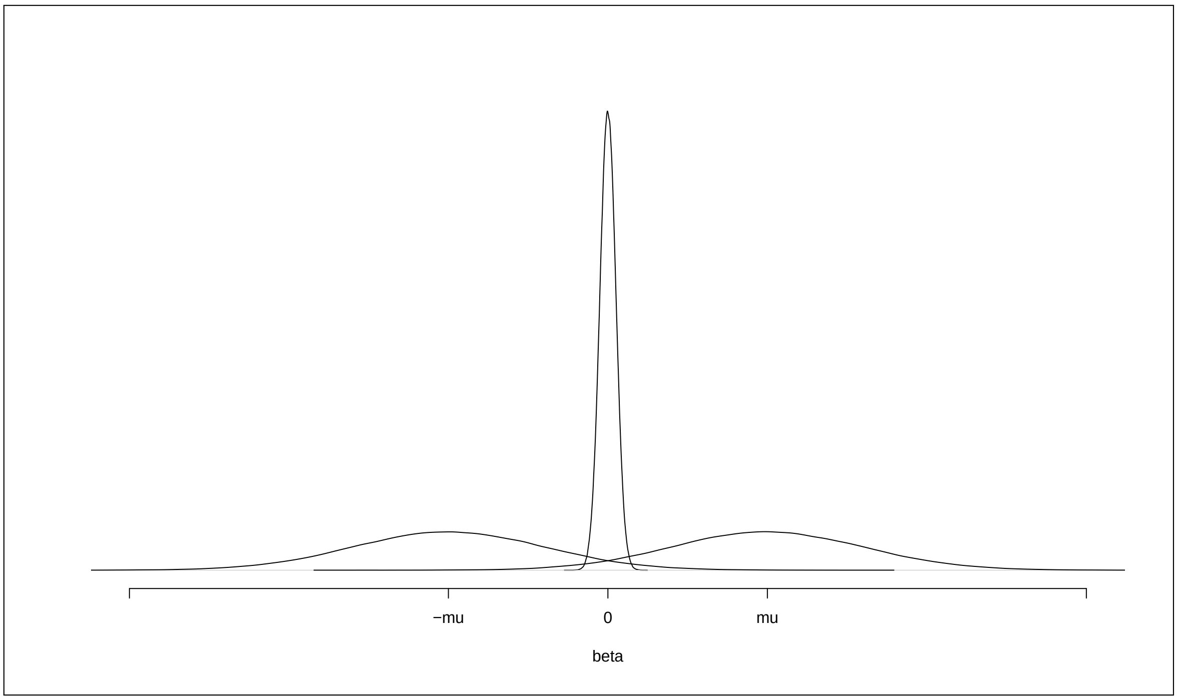


Figure 4.1: Conceptual picture of prior distribution for β's for an active gene set

and can be defined for the likelihood $f(x|\beta)$, as

$$D(\beta) = -2 \log f(x|\beta) \tag{4.6}$$

where x are data and β are unknown parameters. The mean posterior deviance is often used as a measure of fit for Bayesian models, where the mean posterior deviance $\bar{D}$ is given by

$$\bar{D} = E[D(\beta)] \tag{4.7}$$

Both deviance and the mean posterior deviance are particularly straightforward quantities to calculate within MCMC steps and provide an accurate measure of model fit.

The deviance from both the null model and the active model, call them $\boldsymbol{D}^{null}$ and $\boldsymbol{D}^{active}$, are calculated within each MCMC step. From these vectors of null and active model deviances an empirical distribution of deviances can be calculated. Then a frequentist style p-value can be calculated

$$p_s = \text{mean}\big(I(\boldsymbol{D}_s^{null} > \boldsymbol{D}_s^{active})\big) \tag{4.8}$$

where $I()$ is the indicator function.

It could be argued that using the posterior deviance of a model is not necessarily the best measure of model fit, as it does not take into account model complexity, and that maybe the Deviance information criterion (DIC), the Akaike information criterion (AIC) or the Bayesian information criterion (BIC) would be a more suitable measure. However, the two models include the same number of parameters (the same as the number of genes within the given set) and so a measure that takes account of model complexity is not essential and using the deviance as a measure suffices.

Figure 4.2 shows a simplified diagram of the general modelling procedure within gene set s, as outlined above.

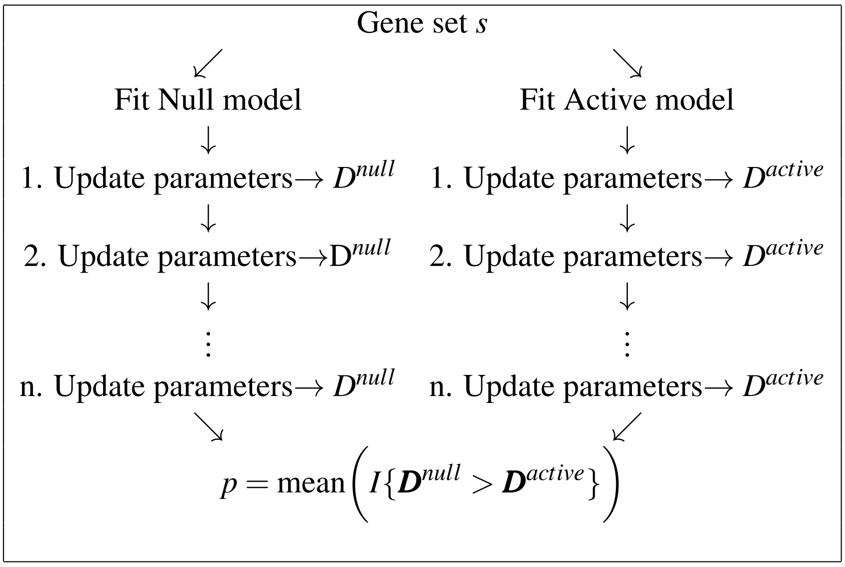

Gene set s
Fit Null model
Fit Active model
1. Update parameters→ D^{null}
1. Update parameters→ D^{active}
2. Update parameters→D^{null}
2. Update parameters→ D^{active}
n. Update parameters→ D^{null}
n. Update parameters→ D^{active}
$p = \text{mean}\left(I\{\boldsymbol{D}^{null} > \boldsymbol{D}^{active}\}\right)$

Figure 4.2: Conceptual picture of calculation of significance of the relationship between gene set s and phenotype

4.2 The algorithm

As mentioned above the posterior distribution for β is intractable algebraically and therefore variates will be sampled from the posterior distribution using MCMC methods. A suitable MCMC algorithm should be designed, along with the use of an effective and efficient proposal density. It should be investigated whether to update all parameters at once, known as block updating, or singly, which is often referred to as single component updating.

Two bespoke algorithms have been written, one using block updating the other using single component updating. The following outlines the two algorithms.

A single component Metropolis-Hastings algorithm has been designed such that within a gene set parameters are updated one-by-one, conditional on all other parameters. The order of this one-by-one update is randomized to speed convergence and aid the mixing of chains. The algorithm is as follows

Algorithm 4.1 Single parameter update

1. Say we are at iteration t of the algorithm. Randomly select the order of updating (without replacement), say we want to update β_g
2. Propose β_g^{prop} from $q(\beta_g^{prop}|\beta_g^{t-1})$.
3. Calculate the acceptance probability

$$\alpha(\beta_g^{t-1}, \beta_g^{prop}) = min\left(1, \frac{f(x|\beta_g^{prop}, \boldsymbol{\beta}_{-g}) f(\beta_g^{prop}) q(\beta_g^{t-1}|\beta_g^{prop})}{f(x|\beta_g^{t-1}, \boldsymbol{\beta}_{-g}) f(\beta_g^{t-1}) q(\beta_g^{prop}|\beta_g^{t-1})}\right) \tag{4.9}$$

   where $\boldsymbol{\beta}_{-g}$ denotes all parameters other than β_g at the current stage of updating.
4. Sample $u \sim U(0,1)$, if $u < \alpha$ then set $\beta_g^t = \beta_g^{prop}$ else $\beta_g^t = \beta_g^{t-1}$
5. Run through all β_g, $g = 1, 2, \ldots, m_s$
6. Iterate t

A standard Metropolis-Hastings algorithm to block update all parameters in the given gene set has been designed. This can be seen in Algorithm 4.2.

There are many proposal distributions that could be used as part of these algorithms, as many proposal distributions (subject to certain conditions) will ultimately deliver samples from the target distribution, Gilks et al. (1996). However, proposal densities that give rapid convergence to the stationary distribution and that allow the chain to mix well are more difficult to come by. To this end several proposal densities will be discussed and implemented as part of the above algorithms, in order to find the best MCMC sampler for the task in hand.

We are presented with the problem that within a given gene set, β can come from

Algorithm 4.2 Block update

```
1. Say we are at iteration $t$ of the algorithm.
2. Propose $\boldsymbol{\beta}^{prop}$ from $q(\boldsymbol{\beta}^{prop}|\boldsymbol{\beta}^{t-1})$
3. Calculate the acceptance probability
   $$\alpha(\boldsymbol{\beta}^{t-1},\boldsymbol{\beta}^{prop}) = min\left(1, \frac{f(x|\boldsymbol{\beta}^{prop})f(\boldsymbol{\beta}^{prop})q(\boldsymbol{\beta}^{t-1}|\boldsymbol{\beta}^{prop})}{f(x|\boldsymbol{\beta}^{t-1})f(\boldsymbol{\beta}^{t-1})q(\boldsymbol{\beta}^{prop}|\boldsymbol{\beta}^{t-1})}\right) \quad (4.10)$$
4. Sample $u \sim U(0,1)$, if $u < \alpha$ then set $\boldsymbol{\beta}^{t} = \boldsymbol{\beta}^{prop}$, else $\boldsymbol{\beta}^{t} = \boldsymbol{\beta}^{t-1}$
5. Iterate $t$
```

a tri-modal distribution, as shown in Figure 4.1. This means that a sampler could have problems in determining each of the three modes, be these problems getting stuck in one particular mode and therefore not converging correctly or not having enough coverage and therefore underestimating extreme modes. Some proposal distributions that will be considered here are described below.

1. **Independence sampler**

   A proposal density of the form

   $$q(\beta_{prop}|\beta) = N(0,\tau^2) \tag{4.11}$$

   could be used to sample β. This proposal density is considered as it is hypothesized that many of the β's will be approximately zero. A relatively large value for τ should be used such that large positive and negative values for β will be covered.

2. **Simple random walk**

   A proposal density of the form

   $$q(\beta_{prop}|\beta) = N(\beta,\tau^2) \tag{4.12}$$

   could be used to sample β. A simple random walk is considered as all modes of the target distribution can be covered relatively quickly.

3. **Bi-modal Independence sampler**

   A proposal density of the form

$$q(\beta_{prop}|\beta) = qN(0,\tau_1^2) + (1-q)N(0,\tau_2^2) \qquad 0 < q < 1 \tag{4.13}$$

   is considered. The reason for this proposal density is that it is believed that there will be many β's that are approximately zero, therefore setting τ_1 to some small value we will be sampling points close to zero. There will also be large values of β in both positive and negative directions, setting τ_2 to some relatively large value, there will be coverage for both large positive and negative β's.

4. **Tri-modal independence sampler**

$$q(\beta_{prop}|\beta) = \frac{1}{2}(1-q)N(-\upsilon,\tau_1^2) + qN(0,\tau_2^2) + \frac{1}{2}(1-q)N(\upsilon,\tau_1^2) \qquad 0 < q < 1 \tag{4.14}$$

   A proposal density of this form is considered as it is an approximation of the posterior distribution of β and therefore should allow the algorithm to explore the full parameter space. To avoid low acceptance rates υ and τ_1 should be chosen such that $q(\cdot|\cdot)$ is heavier tailed than the posterior distribution for β. This is because if the proposal were not heavier tailed than the posterior and the chain was in the tail of the posterior then $\pi(\beta|x)q(\beta^{prop}|\beta)$ will be larger than $\pi(\beta^{prop}|x)q(\beta|\beta^{prop})$ thus giving a low acceptance probability. Heavy tailed independence proposals help to prevent long periods stuck in the tails, therefore help to speed up mixing.

5. **Tri-modal random walk**

   A proposal density

$$q(\beta_{prop}|\beta) = \frac{1}{2}(1-q)N(-\upsilon,\tau_1^2) + qN(\beta,\tau_2^2) + \frac{1}{2}(1-q)N(\upsilon,\tau_1^2) \qquad 0 < q < 1 \tag{4.15}$$

   has been designed to allow for coverage of the whole parameter space with the fixed modes of the proposal, with a random walk to help speed up convergence. We define υ and τ_1 such that the posterior distribution is approximately covered, yet with q relatively large such that candidates generally come from the random walk mode of the proposal. The rationale behind such a proposal is as follows. We are updating parameters from a tri modal distribution, this comes with the associated problems of getting stuck in the wrong mode and slow mixing. The random walk part of the proposal allows us to generally sample candidates conditionally on the previous state of the chain and the fixed parts of the proposal allow for large jumps into any area of the parameter space, thus giving the chains an opportunity to converge to the correct mode quickly and stay there.

In order to identify the best algorithm with the best proposal distribution for the problem in hand a small simulation study will be implemented.

4.2.1 A simulation study

To decide upon the most appropriate MCMC algorithm and sampler both algorithms will be applied, with each of the 5 proposal distributions, to a small data set simulated from the prior for the active model. In order to judge the performance of the ten MCMC samplers Gelman-Rubin statistics will be calculated, Gelman-Rubin-Brooks plots will be drawn, trace and ACF plots will be produced and acceptance rates will be calculated. Gelman-Rubin statistics and Gelman-Rubin-Brooks plots will be based upon five parallel chains initiated from different starting points. Based upon the five indicators described above the best MCMC sampler for this problem will be chosen.

4.2.1.1 Data

Data will be simulated according to the data simulation procedure as outlined in Chapter 3. Data will be simulated for 100 subjects for only one gene set containing 10 genes. Gene effects will be directly sampled from the prior

$$f(\beta) = \frac{1}{\sqrt{2\pi}}\left(\frac{0.1}{0.6}exp\left\{\frac{-(\beta-1)^2}{2\times 0.6^2}\right\}+\frac{0.8}{0.05}exp\left\{\frac{-\beta^2}{2\times 0.05^2}\right\}+\frac{0.1}{0.6}exp\left\{\frac{-(\beta+1)^2}{2\times 0.6^2}\right\}\right)$$

and simplified, such that we have

$$\boldsymbol{\beta} = (1, -1, 0, 0, 0, 0, 0, 0, 0, 0)^t$$

For simplicity gene expression is simulated with no correlation structure

$$X \sim MVN(0, I)$$

and hence

$$y_i \sim \text{Bernoulli}\left(\frac{1}{1 + \exp(-\boldsymbol{x}_i\boldsymbol{\beta})}\right)$$

this results in 52 subjects being of phenotype 1 and 48 subjects being of phenotype 2.

4.2.1.2 Results

The samplers described above have been implemented within the above described algorithms. Table 4.1 shows the parameters used within the different samplers as part of the different algorithms. These parameters were chosen on the basis of previous study and familiarity with the algorithms. Notice that no independence proposals have been implemented as part of the block updating algorithm. This is because acceptance rates for such samplers are so low as to make the sampler useless. Acceptance rates are extremely low because in independently block sampling proposals it is unlikely that a proposal will cover all three modes of the parameter space in the correct position of the vector of parameters.

Proposal	Algorithm	$q(\beta^{prop} \mid \beta)$
1	Single component	$N(0, 1)$
2	Single component	$N(\beta, 0.06^2)$
3	Single component	$\frac{4}{5}N(0, 0.06^2) + \frac{1}{5}N(0, 1.8^2)$
4	Single component	$\frac{1}{3}N(-1.5, 1.75^2) + \frac{1}{3}N(0, 0.05^2) + \frac{1}{3}N(1.5, 1.75^2)$
5	Single component	$\frac{1}{10}N(-1.5, 0.8^2) + \frac{4}{5}N(\beta, 0.05^2) + \frac{1}{10}(1-p)N(1.5, 0.8^2)$
2	Block	$N(\boldsymbol{\beta}, 0.025^2)$
5	Block	$\frac{1}{10}N(-1.5, 0.7^2) + \frac{4}{5}N(\boldsymbol{\beta}, 0.025^2) + \frac{1}{10}N(1.5, 0.7^2)$

Table 4.1: Table showing proposal distributions as implemented within the two algorithms. Proposal numbering refers to that where the proposal distributions are introduced.

Appendices A.1 to A.7 show tables of point estimates of potential scale reduction factor (Gelman-Rubin statistic), corresponding upper confidence limits and acceptance rates alongside trace plots, acf plots and Gelman-Rubin-Brooks plots for the ten logistic slope parameters.

It can be seen from trace plots, Gelman-Rubin-Brooks plots and Gelman-Rubin statistics that in all cases chains converge. Appendices A.1 to A.5 show that there seems to be a compromise between mixing for the zero parameters $(\beta_3,\ldots,\beta_{10})$ and the mixing for the non-zero parameters (β_1,β_2) in the single component updating algorithm. Either the zero parameters mix a little slowly and the non-zero parameters mix well or vice versa, a particularly extreme case of this can be seen in Appendix C, however, generally this is not a problem and a reasonable compromise can be found.

Looking at Appendices A.3 and A.4 it can be seen that the chains for the zero parameters, in particular $\beta_5,\beta_6,\beta_7,\beta_8$ and β_{10} tend to jump out to relatively extreme values, instantly returning to the mode at zero. This behavior drags down or pulls up the mean for these parameters drastically.

The block updating algorithm overestimates β_5 in both cases, as can be seen in Appendices A.6 and A.7.

4.2.2 Discussion: Which algorithm to use

It can be immediately seen that independent block updating samplers should not be used. The independent proposals, used as part of a block updating algorithm, give very low acceptance rates. We are attempting to update parameters sampled from a tri-modal distribution, updating all parameters at once independent of their last state, clearly the larger the number of parameters we wish to update the smaller the acceptance probability. These samplers are therefore impractical to use in this case. It should be noted that acceptance rates are so low as to make convergence near impossible in some reasonable number of iterations.

Independent proposal distributions as used as part of the single component updating algorithm result in inconsistent and somewhat low acceptance rates, for example in the case of the independence sampler the chains for the parameters with value zero mix poorly, whereas the chains for non-zero parameters mix reasonably well. Gelman-Rubin statistics as presented in Appendices A.1, A.3 and A.4 indicate that chains resulting from all three single component independent updating algorithms converge. However, looking at trace plots as shown in these Appendices it can be seen that in several cases chains will jump out to rather extreme values in either a positive or a negative direction returning to its stationary distribution in the next move. There is also the problem that if there are any parameters with very extreme stationary distributions, independent samplers will take a long time to converge and would mix very slowly, if not miss these modes completely. A random walk proposal would converge much more quickly and could explore any sample space.

The two block updating random walk samplers work well, with acceptance rates of 0.1828 and 0.5139 for the tri-modal random walk sampler and the simple random walk sampler respectively. Looking more closely, it can be seen that with the simple random walk sampler all chains mix well (Appendix A.6), converging quickly to the stationary distribution and exploring the full space. Gelman-Rubin statistics as shown in Appendix A.6 indicate convergence, with point estimates consistently close to one. Similarly, chains for the tri-modal random walk sampler mix reasonably well, if a little slowly, with consistently low Gelman-Rubin statistics. One major disadvantage with these (or any) block updating algorithms is that as the number of parameters increases, acceptance rates correspondingly decrease. As discussed in previous chapters gene sets can be of varying size from very small to very large, this could lead to mixing problems when block updating large gene sets. These potential mixing problems for large gene sets could be countered by splitting up gene sets into smaller blocks or by altering variances of proposal distributions, however, this can only solve the problem to some degree. Splitting gene sets into smaller blocks in some ways defeats the object of block-updating and in a real life situation could lead to problems both practically and computationally. Altering variances is also an impractical solution, as altering the proposal distribution gene-set by gene-set would be time consuming and somewhat contrived.

The single component updating algorithm does not suffer from this problem, as it updates parameters one by one. Looking first at the simple random walk sampler it can be seen that chains mix well for all of the parameters. Gelman-Rubin statistics are consistently low, indicating convergence. Chains for the tri-modal random walk sampler mix well and also have correspondingly low Gelman-Rubin statistics.

All things considered, we should not use independent samplers or the block-updating algorithms. The best performing and most versatile samplers are the single component updating algorithm with the simple random walk and tri-modal random walk. There is not much between the two samplers, yet it is believed that the tri-modal sampler is more versatile and slightly better performing. The tri-modal random walk single component updating sampler will therefore be used in future development of this model.

4.3 Discussion

The above mentioned Bayesian/ frequentist hybrid model for the analysis of gene sets takes into account the biological processes behind a pathway effect upon phenotype. However, we are still working with frequentist p-values, and with this comes the associated problems of where and how to define a p-value cut off. It would be preferable to work with proper probabilities. There are few fully Bayesian methods for the analysis of gene sets that provide probabilities for activeness of gene set with regard to pathway to phenotype. The following Chapter introduces two such models.

Chapter 5

Application of RJMCMC to a Bayesian gene set analysis

To date the majority of methods/ models for the analysis of gene sets focus upon producing some frequentist p-value against some null hypothesis relating to no gene set effect, the construction of the method defining the null hypothesis, of which there are several, for example:

- $H_0^{competitive}$: The genes within gene set s are at most as often differentially expressed as genes in $s_{compliment}$;
- $H_0^{selfcontained}$: No genes in gene set s are differentially expressed.

As discussed through Chapters 2,3 and 4 there are many associated problems with using frequentist methodologies that employ a cut-off on a p-value relating to some gene set score to determine activeness/enrichment of a gene set. Bayesian gene set analysis (BGSA) is a fully Bayesian model, yet a frequentist style p-value is computed as a summary of the relationship between gene set and phenotype, conforming to the general gene set analysis method.

When using a method where we must state if a gene set is active based upon whether an associated p-value is below some arbitrarily defined cut-off, we have the problem of defining that cut-off. Nam and Kim (2008), amongst others discuss problems with IGA, one of these problems being the use of cut-offs on p-values. Methods for gene set analysis were designed to overcome these problems, but in the case of p-value cut-offs the problem has simply been scaled down, rather than solved. Another drawback to be considered when using a method that uses a cut-off on p-values is that we are limited to concluding a gene set active or not which is a little black and white. It would be beneficial to use a probability, whereby we could say 'gene set s is related to phenotype with probability p' rather than 'gene set s is related to phenotype' or 'there is no evidence that gene set s is related to phenotype'.

There is of course the difficulty of bridging the gap between fitting a model to calculating a probability relating to the fitted model that indicates the degree to which a gene set is active. Chapter 4 demonstrates this with the hybrid model, whereby Bayesian models are fitted but a frequentist style p-value calculated based upon the fit of the models.

A development of the hybrid model, as introduced in Chapter 4, would be to instead of simultaneously fit two models and compare, allow for a reversible jump step between the two models. In doing so the hybrid model of Chapter 4 is developed into a fully Bayesian model utilizing Reversible jump Markov chain Monte Carlo (RJMCMC). In such a framework we can define an active model and a null model as in Chapter 4 and then allow the reversible jump moves to define a probability for activeness of set.

The above mentioned model can be seen to work well (Chapter 6) as it takes a multivariate modelling approach to all genes within a gene set, where phenotype is modelled conditionally on gene expression. Within gene set correlation structure is also accounted for due to the within gene set model. However, it is of interest how gene sets affect the behavior of other gene sets in the pathway to phenotype and the impact that this has on phenotype. This is implemented by a fully multivariate model where all genes within the data set are accounted for in a logistic model. Using RJMCMC we can propose to discard gene sets from the model, or include gene sets in the model conditionally on the state of all other gene sets. This results in probabilities referring to a gene sets contribution to phenotype for each gene set and allows the user a much more in-depth view of the behavior exhibited by genes and gene sets

This chapter is set out as follows; Section 5.1 introduces reversible jump Markov chain Monte Carlo (RJMCMC) and provides an illustrative example of the algorithm in practice. Section 5.2 describes in detail the first of the above introduced models, Section 5.3 introduces and describes the second of the above introduced models, and finally Section 5.4 discusses the two models.

5.1 Reversible jump Markov chain Monte Carlo (RJMCMC)

The most commonly considered MCMC algorithm moves around the parameter space of a single model. The RJMCMC algorithm moves around the parameter space of a collection of models, generally finite in number. These models can be nested or arbitrary. The algorithm is a combination of the standard MCMC algorithm for a given model, with an additional step which moves between the different models. Suppose that there are $k = 1, 2, \ldots, K$ models, call them M_k, under consideration. Each of the M_k models have unknown parameter vector of length n_k, $\boldsymbol{\theta}_k \in \mathbb{R}^{n_k}$.

The joint posterior distribution of $(k, \boldsymbol{\theta}_k)$, $\pi(k, \boldsymbol{\theta}_k|x)$, given data x, is obtained by multiplying the joint prior of $(k, \boldsymbol{\theta}_k)$, $f(k, \boldsymbol{\theta}_k) = f(\boldsymbol{\theta}_k|k)f(k)$ by the likelihood $L(x|k, \boldsymbol{\theta}_k)$. The joint posterior distribution is therefore given by

$$\pi(k, \theta_k|x) = \frac{f(\boldsymbol{\theta}_k|k)f(k)L(x|k, \boldsymbol{\theta}_k)}{\sum_{k' \in K} \int_{\mathbb{R}^{n_{k'}}} f(\boldsymbol{\theta}_{k'}|k')f(k')L(x|k', \boldsymbol{\theta}_{k'})d\boldsymbol{\theta}_{k'}} \tag{5.1}$$

The reversible jump algorithm takes the above joint posterior distribution as the target of an MCMC sampler, where the states of the Markov chain are of the form $(k, \boldsymbol{\theta}_k)$. The output from such an algorithm provides us with not only posterior distributions of parameters from each individual model, but also posterior probabilities of each model given data x. The general RJMCMC algorithm is defined by Algorithm 5.1

5.1.1 Implementation of RJMCMC

Suppose we are currently residing in state $(k, \boldsymbol{\theta}_k)$ in model M_k and we want to propose a move to state $(k', \boldsymbol{\theta}_{k'})$ in model $M_{k'}$, which is of a higher dimension such that $n_{k'} > n_k$. In order to match dimensions between the model states the random vector of auxiliary variables $\boldsymbol{u}$ of length $d_{kk'} = n_{k'} - n_k$ is generated from a known density $q_{d_{kk'}}(\boldsymbol{u})$. The

Algorithm 5.1 Simplified RJMCMC algorithm

```
1  At $t = 1$ initialize $k$ and $\boldsymbol{\theta}_k$
2  At iteration $t$
    2.1  Say we are currently residing in model $k$. Update the model parameters $\boldsymbol{\theta}_k$.
    2.2  Update model indicator $k$ and corresponding parameters $\boldsymbol{\theta}_k$
3  Iterate $t$.
```

current state θ_k and the auxiliary variables $\boldsymbol{u}$ are then mapped to the new state $\boldsymbol{\theta}_{k'}$ by a one-to-one mapping function $g_{kk'}(\boldsymbol{\theta}_k, \boldsymbol{u})$. The acceptance probability of such a move is given by

$$\alpha[(k, \boldsymbol{\theta}_k), (k', \boldsymbol{\theta}_{k'})] = \min\left\{1, \frac{\pi(k', \boldsymbol{\theta}_{k'}|x)q(k', k)}{\pi(k, \theta_k|x)q(k, k')q_{d_{kk'}}(\boldsymbol{u})} \left|\frac{\partial g_{kk'}(\boldsymbol{\theta}_k, \boldsymbol{u})}{\partial(\boldsymbol{\theta}_k, \boldsymbol{u})}\right|\right\} \tag{5.2}$$

where $q(k', k)$ denotes the probability of proposing a move from model M_k to model $M_{k'}$ and

$$J = \left|\frac{\partial g_{kk'}(\boldsymbol{\theta}_k, \boldsymbol{u})}{\partial(\boldsymbol{\theta}_k, \boldsymbol{u})}\right| \tag{5.3}$$

is the determinant of the Jacobian matrix. This term is required due to the change of variables function $g_{kk'}$. The reverse move proposal, from model $M_{k'}$ to model M_k is made deterministically and is accepted with probability

$$\alpha[(k', \boldsymbol{\theta}_{k'}), (k, \boldsymbol{\theta}_k)] = \alpha[(k, \boldsymbol{\theta}_k), (k', \boldsymbol{\theta}_{k'})]^{-1} \tag{5.4}$$

Note that as with standard MCMC a normalization constant is not needed to evaluate $\alpha[(k, \boldsymbol{\theta}_k), (k', \boldsymbol{\theta}_{k'})]$.

5.1.2 A simple example

The survival times in years of 20 patients following major surgery are

0.8	8.8	1.4	3.8	6.7	0.8	0.3	2.1	3.7	1.2
2.7	2.7	4.5	1.5	10.1	1.4	10.3	1.2	8.7	5.3

Two models are proposed for the distribution of the lifetime X of these patients.

- **Model 1** (M_1): $X \sim \text{Gamma}(\alpha, \beta)$
- **Model 2** (M_1): $X \sim \text{Weibull}(\gamma, \delta)$

The one to one mapping between the parameters from Model 1 to Model 2 can be defined as

$$\gamma = \frac{1}{\alpha} \qquad \delta = \frac{1}{\beta}$$

and hence the Jacobian for the transformation J is

$$J = \frac{1}{\alpha^2 \beta^2}$$

There are many one-to-one mappings to choose from, yet a mapping should be chosen such that

- Transformation of variables is straightforward.
- The Jacobian can be (relatively) easily calculated.
- The mapping allows for good mixing between models.

The above chosen bijection conforms to such criteria.

A RJMCMC algorithm has been written and implemented to fit the above models. The following presents a small investigation into the behavior of this algorithm as applied to the above data.

After a burn-in of 1000 iterations, 10,000 iterations were saved. Table 5.1 shows the posterior means and variances of the parameters α, β, δ, and γ. Figure 5.1 shows the trace plot of the reversible jump move between the two models. As can be seen the reversible

Parameter	Mean	Variance
α	1.6090	0.1790
β	0.4284	0.0171
δ	1.2166	0.0434
γ	4.4031	0.6918

Table 5.1: Posterior means and variances of parameters from the two models

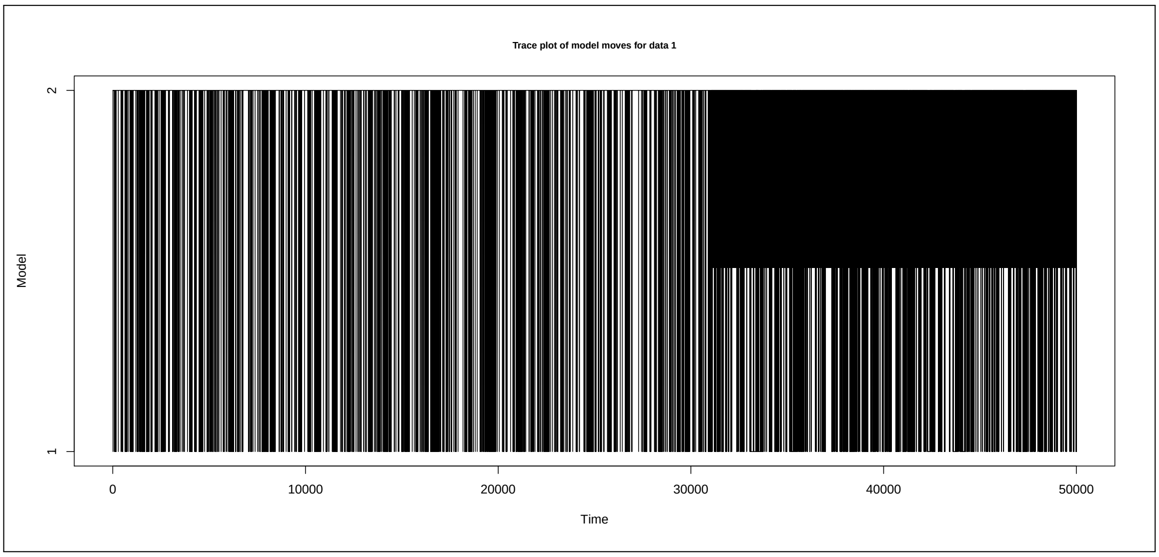


Figure 5.1: Trace plot of the reversible jump between Models 1 and 2

jump moves between the two parameter spaces very often, with $P[M_1] = 0.12604$ and $P[M_2] = 0.87396$ so clearly Model 2 is a more adequate fit for the data.

As a matter of interest and to aid in the understanding of the behavior of the reversible jump algorithm, the above modelling will be repeated twice. The first repetition will be implemented on a data set (data set 2) containing 100 measurements rather than 20, whereby each measurement in the existing data is replicated five times. The second repetition will be implemented on a data set (data set 3) containing 200 measurements rather than 20, whereby each measurement in the existing data is replicated ten times. Figure 5.2 shows the trace plot of the reversible jump move between the two models for data set 2. As can be seen, with more data and therefore more support for Model 2 the algorithm makes far fewer jumps between the models. The posterior probabilities of the two models are $P[M_1] = 0.21166$ and $P[M_2] = 0.78834$

Figure 5.3 shows the trace plot of the reversible jump move between the two models for data set 3. As can be seen, with again, more data the algorithm makes even fewer jumps between the models. The posterior probabilities of the two models are $P[M_1] = 0.17094$ and $P[M_2] = 0.82906$

These examples show that with more data the algorithm makes far fewer jumps between the two models due to the added support for the models by the extra data, however,

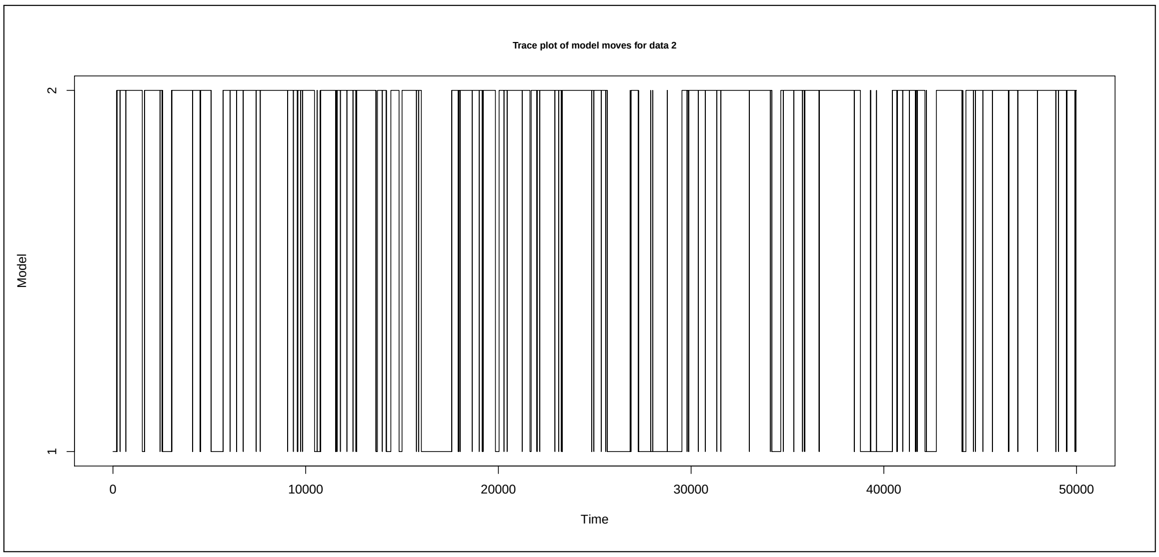


Figure 5.2: Trace plot of the reversible jump between Models 1 and 2 for data set 2

in this case, the posterior probabilities for the models are similar in all three cases. Repeated modelling of data 3 shows a lack of convergence, with widely varying results for the model jumping move. This is due to such low acceptance rates and a much greater number of iterations would be required to reach convergence. This tendency to 'get stuck'

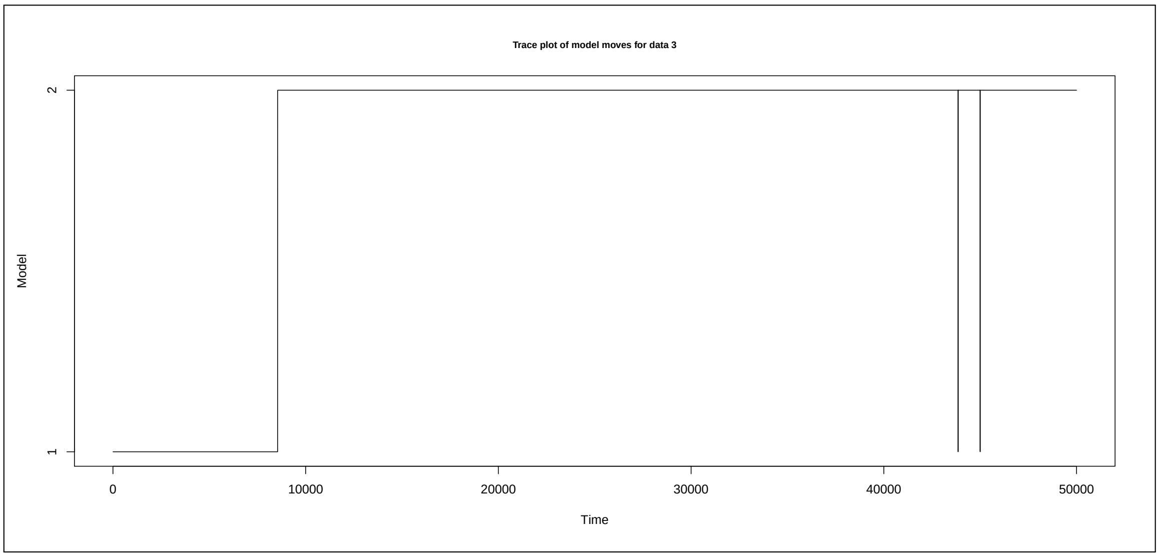


Figure 5.3: Trace plot of the reversible jump between Models 1 and 2 for data set 3

in a parameter space when there is much support for a model is common to many RJMCMC problems, and it is not good to have such a low acceptance rate of model switching moves. Rather than running an algorithm for a very large number of iterations, it would be more efficient to improve the mixing of the reversible jump and therefore speed up convergence. A common technique used to improve acceptance rates of model switching moves when there is a simple bijection, as in this case, is to add a zero mean random variable to current parameters when proposing a model switching move. For example, say we have two parameter vectors of the same length $\boldsymbol{\theta}_1$ and $\boldsymbol{\theta}_2$ corresponding to two models. Then when proposing a move from model 1 to model 2 we propose $\boldsymbol{\theta}_2 = \boldsymbol{\theta}_1 + \boldsymbol{\xi}$, where $\boldsymbol{\xi}$ is a zero mean random variable. The variance of $\boldsymbol{\xi}$ can be calibrated to allow for a reasonable acceptance rate of model switching moves. There is no such simple method for variable dimensional problems, however, there are many recognized techniques to improve the acceptance rate of model switching moves in variable dimension problems such as those presented by Brooks et al. (2003) and Al-Awadhi et al. (2004).

5.2 The application of RJMCMC in a Bayesian analysis of gene sets (BAGS)

The proposed approach to pathway analysis, Bayesian Analysis of Gene Sets (BAGS), utilizes RJMCMC techniques in order to produce posterior probabilities of 'activeness' of gene set. This approach enables the implementer to define active/ inactive gene sets with respect to gene effect on phenotype within gene set. Essentially we define two models; a null model for a null or inactive gene set and an active model, for a gene set that is active in the pathway to phenotype. RJMCMC is then used to jump between the parameter spaces of the two models. The posterior probability for the active model can then be used as a probability of activeness of the gene set.

The null and active within gene set logistic models are defined as in Chapter 4 whereby we define two logistic regression models by their priors.

5.2.1 A logistic regression model

As mentioned above two logistic models are to be defined, however both models will have the same likelihood. Looking within gene set s, containing m_s genes, the likelihood for such a model is given by

$$f(x|\beta) = \prod_{i=1}^{n} \pi_i^{y_i}(1-\pi_i)^{1-y_i} \tag{5.5}$$

where

$$\pi_i = \frac{1}{1+\exp\{-\eta_i\}} \tag{5.6}$$

and

$$\eta_i = \sum_{g=1}^{m_s} x_{ig}\beta_{sg} \tag{5.7}$$

where x_{ig} is the expression of the g^{th} gene in gene set s for the i^{th} subject and β_{sg} is the slope parameter for the g^{th} gene in gene set s. The difference in the two models is defined by their respective prior distributions, $f(\beta)$.

5.2.1.1 The null model

A prior distribution for the slopes, $\boldsymbol{\beta}_s$, for a null logistic model should represent a situation where the genes in the set have no effect on phenotype. It is unreasonable to suggest that all slopes within a null gene set are identically zero, as there will be noise in expression measurements and some genes within an inactive gene set may have a small effect on phenotype. However, it is reasonable to suggest that effects will be small and centered about zero. Therefore a normal prior of the form

$$f(\boldsymbol{\beta}_s) = \frac{1}{\sqrt{2\pi}\sigma}\exp\left\{\frac{-\boldsymbol{\beta}_s^2}{2\sigma^2}\right\} \tag{5.8}$$

is used, where σ is small. This represents prior beliefs that for a gene set that has no relationship with phenotype and the majority of gene effects will be close to zero.

5.2.1.2 The active model

Within a gene set that has a relationship with phenotype it would be expected that there are a whole range of gene effects. There will be some genes that have no effect upon phenotype and some that have relatively large effects upon phenotype, both in negative and positive directions. The prior distribution for the slopes, $\boldsymbol{\beta}_s$, for an active logistic model should represent this. Therefore a prior of the form

$$f(\boldsymbol{\beta}_s) = \frac{1}{\sqrt{2\pi}}\left(\frac{(1-p)}{2\sigma_1}\exp\left\{\frac{-(\boldsymbol{\beta}_s-\mu)^2}{2\sigma_1^2}\right\}+\frac{p}{\sigma_2}\exp\left\{\frac{-\boldsymbol{\beta}_s^2}{2\sigma_2^2}\right\}+\frac{(1-p)}{2\sigma_1}\exp\left\{\frac{-(\boldsymbol{\beta}_s+\mu)^2}{2\sigma_1^2}\right\}\right) \tag{5.9}$$

is used, where $0 < p < 1$ is a fixed value that is specified by the user. This allows for both very small effects and large effects, as it is believed that within an active pathway not all genes will be up- or down-regulated, the broad peaks in the extremes represent the belief that the up- or down-regulation of certain genes will have a positive or negative effect upon the linear predictor for the logistic model.

We put these models in a reversible jump MCMC framework, whereby not only do we propose new parameters ($\boldsymbol{\beta}$) for a model and accept with probability $\alpha(\boldsymbol{\beta},\boldsymbol{\beta}^{prop})$ but we also propose to move from the null model (M_1) to the active model (M_2) with probability $\alpha(M_1,M_2)$ and vice versa.

The algorithm is outlined by Algorithm 5.2, note that as gene sets are considered independent in the above model, steps 3 and 4 are interchangeable.

Before any application to real data, or subjecting the model to extensive simulation study, it is necessary to observe the behaviour of the model on simulated data and to calibrate proposal distributions in this new setting.

The above described model has been programmed in R and applied to a simulated data set which has been simulated according to approach 2 in Chapter 3 complying to criteria set out by scenario 7, whereby we have ten active genes with effects of 1 within one active set, all other sets being inactive and $\rho = 0$. The model has been run for $10,000$ iterations after a burn in of 1000. Figure 5.4 shows trace plots of the model indicator for an active set and an inactive set.

This results in posterior model probabilities for the active set of $P[\text{active}] = 0.999$ and $P[\text{active}] = 0$ for the null set. As can be seen in Figure 5.4, mixing between models is very poor and needs to be improved in order for the full model space to be explored. One recognized technique for improving the mixing of reversible jump moves between models of equal dimension is to add a vector of zero mean random variables to the parameter

Algorithm 5.2 RJMCMC algorithm for gene set analysis

1 Say we are modelling gene set s, containing $g = 1, \ldots, m_s$ genes. Also say we are in model M_m, $m = (1,2)$, with parameters $\boldsymbol{\beta}$ at iteration t. Using a random order, single component Metropolis-Hastings move, a within model parameter update is made, according to Algorithm 4.2 in Chapter 4.

2 Propose to move from model M_m with parameters $\boldsymbol{\beta}$ to model M_{m^c}, where M_{m^c} denotes the compliment of M_m, with parameters $\boldsymbol{\beta}'$. More generally, we could have > 2 models and would have to include some model proposal scheme to the algorithm, and add a model proposal density term to the acceptance probability (as shown below). However, in this case such is not needed as if we have only two models then if we are in model M_m then with probability 1 we propose to move to model M_{m^c} and vice-versa.

2.1 We must first define a one-to-one mapping between $\boldsymbol{\beta}_s$ and $\boldsymbol{\beta}'_s$. This can be done simply in this case, due to the fact that for every β_g $(g = 1, \ldots, m_s)$ in M_m there is an equivalent β_g in M_{m^c}. We therefore define the one-to-one mapping as

$$\boldsymbol{\beta}'_s = \boldsymbol{\beta}_s \tag{5.10}$$

2.2 Calculate the acceptance probability for the move from M_m to M_{m^c}

$$\begin{aligned} \alpha_{mm^c}(\boldsymbol{\beta}_s, \boldsymbol{\beta}'_s) &= \min\left\{1, \frac{\pi_{m^c}(\boldsymbol{\beta}'_s|x)}{\pi_m(\boldsymbol{\beta}_s|x)}\right\} \\ &= \min\left\{1, \frac{f(\boldsymbol{\beta}'_s)}{f(\boldsymbol{\beta}_s)}\right\} \qquad (5.11) \end{aligned}$$

as with the above mapping the likelihood is the same for both the numerator and the denominator.

2.3 Sample $u \sim U(0,1)$, if $u < \alpha$ then move to M_{m^c} with parameters $\boldsymbol{\beta}'_s$ else stay in M_m with parameters $\boldsymbol{\beta}_s$.

3 Increment t.

4 Move through all $s = 1, \ldots, K$ gene sets.

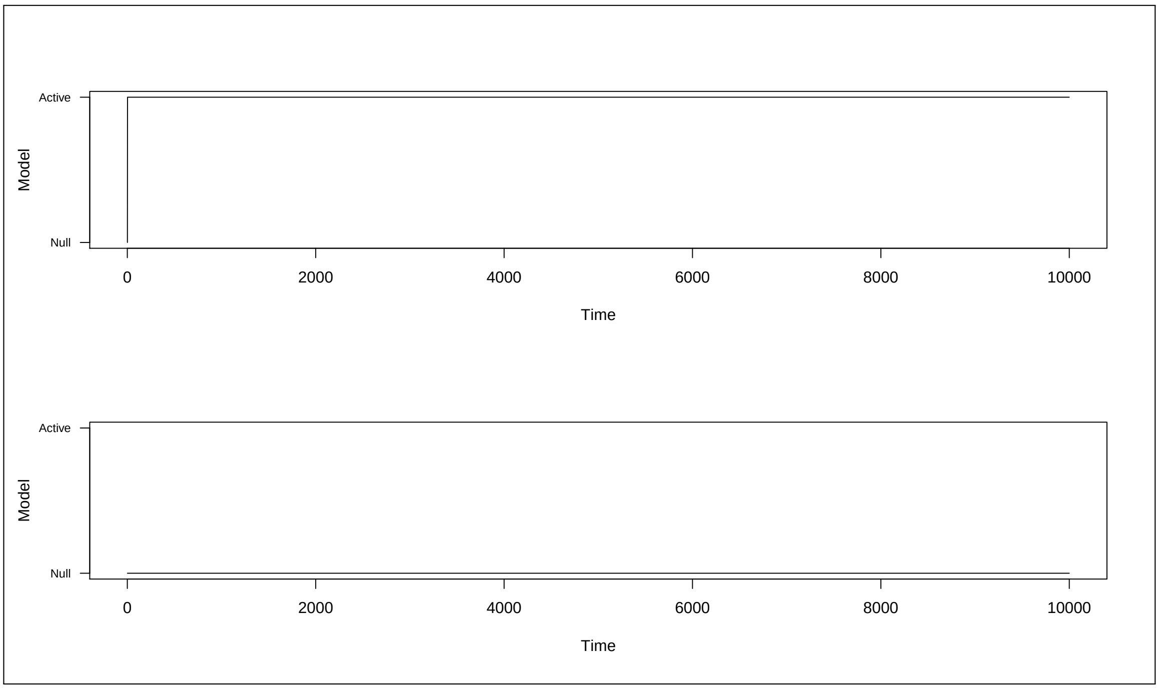


Figure 5.4: Trace plot of the reversible jump between null and active models for simulated data. Top: active gene set. Bottom: null gene set.

vector when proposing a model move, i.e. when moving from model M_m with parameters $\boldsymbol{\beta}$ to model M_{m^c} propose new parameters

$$\boldsymbol{\beta}' = \boldsymbol{\beta} + \boldsymbol{\xi} \tag{5.12}$$

where $\boldsymbol{\xi}$ is a zero mean random variable. Clearly, the likelihoods of the two models will no longer cancel and so the posterior distributions for both models should be used in the model move acceptance probability. This has been integrated into the above algorithm whereby we set

$$\boldsymbol{\beta}' = \boldsymbol{\beta} + N(0, \sigma^2_{mix}) \tag{5.13}$$

On running the algorithm and systematically exploring values for σ^2_{mix} it is found that the best $\sigma_{mix} = 0.0675$. This altered algorithm has been applied to the above referred to simulated data set. Figure 5.5 shows trace plots of the model indicator corresponding to those shown in Figure 5.4.

As can be seen we now move around the parameter space much more freely, this

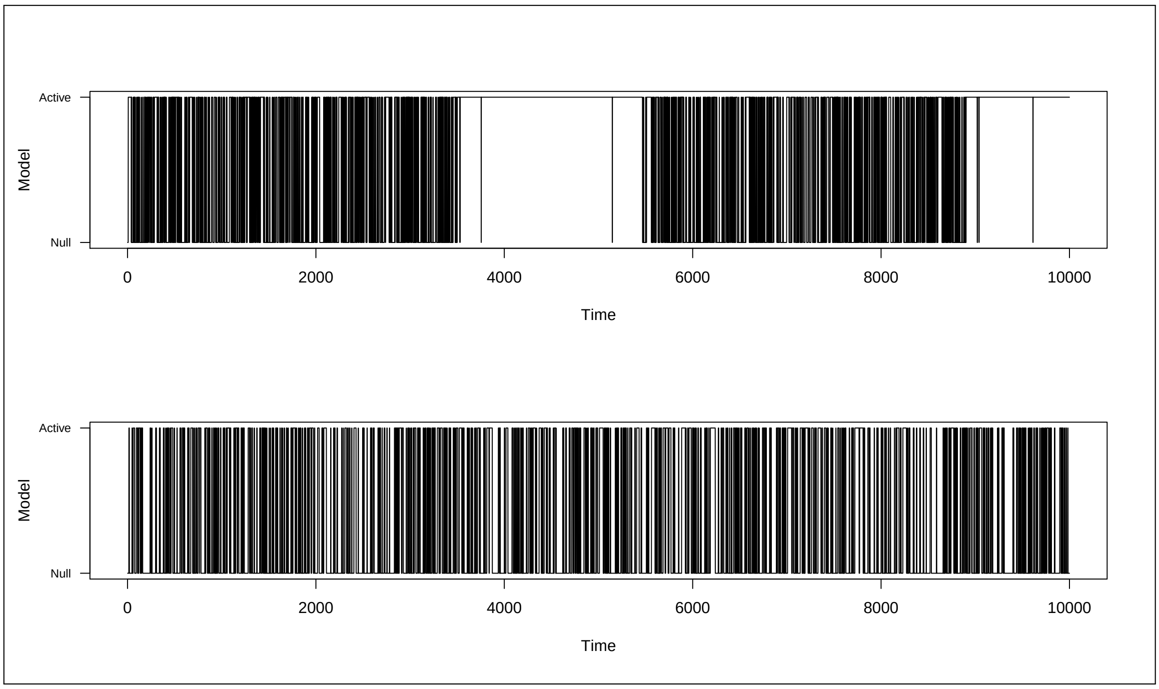


Figure 5.5: Trace plot of the reversible jump between null and active models for simulated data. Top: active gene set. Bottom: null gene set.

results in the much more realistic posterior model probabilities of $P[\text{active}] = 0.6924$ for the active set and $P[\text{active}] = 0.2377$ for the null set.

In fitting such a model, not only do we produce posterior probabilities of activeness of gene set, but we also have posterior samples for each of the gene effects on a linear predictor for a logistic model on phenotype. Typically in a gene set analysis, once per gene set p-values are calculated and gene sets of interest (with p-values below an arbitrary cut-off) are selected, further analysis takes place whereby gene by gene statistics are looked at, or some modelling procedure is implemented on genes from these gene sets of interest to find the most contributing genes. In the case of the above outlined model, when fitted, gene sets will have an associated probability of activeness and genes of interest within these sets can be found by simple summaries of the posterior samples of the genes, for example plotting histograms, and calculating means and variances of the parameters.

It could be questioned as to why multi-level models have not been used. For example, hyperpriors could be put on the means and variances for the outlying normal distributions in the active model. Consider adding such complexity to the active model, and fitting this to a null gene set. Both the means and variances of the outlying normal distributions will shrink towards zero and the active model will become near identical to the null model, hence rendering the reversible jump step useless. This would defeat the reasoning behind such a procedure, where we rely on the reversible jump step between the models to determine a probability for activeness of gene set.

The above outlined proposed model brings to bear many of the suggested improvements to existing methodologies, whereby phenotype is modelled conditionally on gene expression and within a gene set the expression of all genes is taken into account at once, thus accounting for correlations between genes within gene set.

5.3 The application of RJMCMC to a Multivariate Bayesian analysis of gene sets (MVBAGS)

In this approach to gene set analysis a fully multivariate approach is proposed, Multivariate Bayesian Analysis of Gene Sets (MVBAGS), whereby we allow for a variable dimension logistic regression model for the full compliment of genes within a dataset. The contribution to phenotype of each gene set and hence gene is taken into account in such a model, allowing for gene set to gene set interaction. This therefore takes into account inter-gene set correlation and dependence. This will provide a more rounded view of the processes involved in the pathway to phenotype and will provide posterior probabilities of activeness of gene set conditionally on the expression of other genes and the behavior of other gene sets.

To begin with we simply fit a logistic regression model for all genes

$$\eta_i = \boldsymbol{x}_i \boldsymbol{\beta} \tag{5.14}$$

where

$$P[y_i = 1] = \frac{1}{1+\exp(-\eta_i)} \tag{5.15}$$

where $\boldsymbol{x}_i$ is the expression of N genes in subject i, $i = 1, \ldots, n$ and $\boldsymbol{\beta}$ are the effects of these genes upon a linear predictor (η) for phenotype y. Making use of RJMCMC we propose, conditionally on all other gene sets, to remove gene set s, $s = 1, \ldots, K$, or to update its parameters $\boldsymbol{\beta}_s$. This is carried out systematically running through all gene sets, gene set by gene set for some large number of iterations. The following goes on to describe the model in detail.

5.3.1 The model

As mentioned above a logistic model will be fitted to the full compliment of genes, whereby the likelihood is given by

$$f(\boldsymbol{X}|\boldsymbol{\beta}) = \prod_{i=1}^{n} \pi_i^{y_i}(1-\pi_i)^{1-y_i} \tag{5.16}$$

where

$$\pi_i = \frac{1}{1+\exp\{-\eta_i\}} \tag{5.17}$$

and

$$\eta_i = \boldsymbol{x}_i\boldsymbol{\beta} \tag{5.18}$$

where $\boldsymbol{x}_i$ is the expression of N genes in subject i, $i = 1,\ldots,n$ and $\boldsymbol{\beta}$ are the effects of these genes upon a linear predictor (η) for phenotype y. A prior of the form

$$f(\boldsymbol{\beta}) = \frac{1}{\sqrt{2\pi}}\left(\frac{(1-p)}{2\sigma_1}\exp\left\{\frac{-(\boldsymbol{\beta}-\mu)^2}{2\sigma_1^2}\right\} + \frac{p}{\sigma_2}\exp\left\{\frac{-\boldsymbol{\beta}^2}{2\sigma_2^2}\right\} + \frac{(1-p)}{2\sigma_1}\exp\left\{\frac{-(\boldsymbol{\beta}+\mu)^2}{2\sigma_1^2}\right\}\right) \tag{5.19}$$

is used, where $0 < p < 1$. This allows for both very small effects and large effects, the peaks in the extremes represent the belief that the up- or down-regulation of certain genes will have a positive or negative effect upon the linear predictor for the logistic model and the distribution centered about zero represents beliefs that there will be many genes with no effect on phenotype. The posterior distribution for the set of parameters $\boldsymbol{\beta}$ is given by

$$f(\boldsymbol{\beta}|x) = \frac{f(\boldsymbol{\beta})f(x|\boldsymbol{\beta})}{\int f(\boldsymbol{\beta})f(x|\boldsymbol{\beta})d\boldsymbol{\beta}} \tag{5.20}$$

which is intractable algebraically. Therefore MCMC techniques will be employed to obtain samples from the posterior distribution of $\boldsymbol{\beta}$, this will be described in further detail below.

The parameter vector $\boldsymbol{\beta}$ is a vector of variable length. The model will start with the full compliment of genes, i.e. the vector $\boldsymbol{\beta}$ will start at length N. We will then partition $\boldsymbol{\beta}$ into gene sets (for now assuming that gene sets do not overlap) $s = 1,\ldots,K$. Say we are looking at gene set s, let $\boldsymbol{\beta}_s$ denote the subset of gene effects corresponding to the genes within gene set s and let $\boldsymbol{\beta}_{-s}$ denote all other parameters, then we proceed to propose to either remove the parameters for gene set s, i.e. set $\boldsymbol{\beta}_s = 0$ and thus move into a new model space where $\boldsymbol{\beta} = \boldsymbol{\beta}_{-s}$, or update the parameters belonging to gene set s conditionally on $\boldsymbol{\beta}_s$ and $\boldsymbol{\beta}_{-s}$ using a standard MCMC step. So, at any point during this procedure we can either include or exclude a gene set from the model. This will result in a fully multivariate logistic regression model for phenotype whereby all genes are accounted for and each gene set has an associated probability for its relationship with phenotype. This will also result in model probabilities where we can find the most likely model which will consist of gene sets that have an effect on phenotype.

Clearly in such a modelling procedure, a bijection between parameters is not so simple as in the model outlined in the previous section, as rather than moving between just two models we will be moving about 2^K possible models, where K is the total number of gene sets. For each update of all N parameters there are 2^K possible models and therefore many possible lengths for $\boldsymbol{\beta}$, we therefore need to define auxiliary variables $\boldsymbol{u}$ in each case such that $(\boldsymbol{\beta},\boldsymbol{u})$ always has length N. Once again, this is relatively simple within the RJMCMC step, say we remove parameters for $\boldsymbol{\beta}_s$ then

$$\boldsymbol{u}_s = \boldsymbol{\beta}_s \tag{5.21}$$

and in the next update we update $\boldsymbol{u}_s$ to the same recipe as we would update $\boldsymbol{\beta}_s$. The following goes on to describe the practical workings of the model by way of a simplified algorithm. An algorithm for this a model is outlined in Algorithm 5.3.

From such a modelling procedure we have probabilities for a number of models each containing some number of gene sets, probabilities for activeness of each gene set and posterior samples from the full compliment of genes that both take into account effects from other genes and gene sets. From this output and the use of some simple summaries

Algorithm 5.3 RJMCMC algorithm for gene set analysis

1 Say we are at time $t = 1$. Update the full compliment of N gene effects $\boldsymbol{\beta}$. From here on $\boldsymbol{\beta}$ is of variable length, with maximum length N, minimum length 0.

2 Now for any $t > 1$, we systematically run through all gene sets for each iteration of t, say we are now in gene set s. Let k_s^t denote a model indicator for gene set s at time t, where $k_s^t = 1$ if the parameters corresponding to gene set s, $\boldsymbol{\beta}_s$, are included in the model and $k_s^t = 0$ if not.

2.1 If $k_s^t = 1$ then $\boldsymbol{\beta}_s$ is included in the model. Therefore propose to remove $\boldsymbol{\beta}_s$ with probability

$$\alpha(\boldsymbol{\beta}, \boldsymbol{\beta}_{-s}) = \min\left\{1, \frac{\pi(\boldsymbol{\beta}_{-s}|x)}{\pi(\boldsymbol{\beta}|x)}\right\} \tag{5.22}$$

sample $u \sim U(0,1)$. If $u < \alpha$ then accept the move and set $k_s^{t+1} = 0$, $\boldsymbol{\beta} = \boldsymbol{\beta}_{-s}$ and update auxiliary variables $\boldsymbol{u}_s$ according to Algorithm 4.1, else $k_s^{t+1} = 1$, update $\boldsymbol{\beta}_s$ according to Algorithm 4.1 and move to next set.

2.2 If $k_s^t = 0$ then $\boldsymbol{\beta}_s$ is not included in the model. Propose to include $\boldsymbol{\beta}_s$ with probability

$$\alpha(\boldsymbol{\beta}, \boldsymbol{\beta}^{prop}) = \min\left\{1, \frac{\pi(\boldsymbol{\beta}^{prop}|x)}{\pi(\boldsymbol{\beta}|x)q(\boldsymbol{u}_s)}\right\} \tag{5.23}$$

Where $\boldsymbol{\beta}^{prop} = (\boldsymbol{\beta}, \boldsymbol{u}_s)$. Sample $u \sim U(0,1)$. If $u < \alpha$ then accept the move and set $k_s^{t+1} = 1$ and set $\boldsymbol{\beta} = \boldsymbol{\beta}^{prop}$, else $k_s^{t+1} = 0$, update $\boldsymbol{u}_s$ according to Algorithm 4.1 and move to next set.

3 Move through the full set of $s = 1, \ldots, K$ gene sets.

4 Iterate t.

such as means, variances and credible intervals and plots such as histograms we are provided with the information to make a detailed interpretation of the underlying biological features of the data set. This interpretation can be based upon which gene sets have a relationship with phenotype, the strength of the gene set's relationship with phenotype and the exact relationship that the gene set has with phenotype with regards to the expression of the genes within the set. In contrast to the majority of other methods, all of the above information provided is fully multivariate and it therefore takes into account correlations between genes both within and between gene sets. It also conditions phenotype on gene expression, which seems to be a somewhat overlooked feature in many of the current existing methods for the analysis of gene sets.

Chapter 6 applies BAGS and MVBAGS to simulated data. Results from the simulation study will be used to assess the performance of each of the proposed models and to compare and contrast the relative strengths and weaknesses of the two proposed models with not only the existing methods as described in Chapter 2 but also with each other. Chapter 7 then goes on to apply BAGS to a real data set.

5.4 Discussion

The two introduced models (BAGS and MVBAGS) hold many advantages over the recognized field of methods and models for the analysis of gene sets to date. The models are defined from a biological point of view, whereby distributions are used to represent how gene expression affects a linear predictor on a dichotomous phenotype, in such a way that phenotype is modelled conditionally on gene expression. The modelling of phenotype conditionally on gene expression is a different approach to all of the methods or models that have been studied, however, from a modelling point of view this makes more sense as we are interested in the pathway to phenotype, i.e. how the behavior a gene set affects phenotype.

BAGS allows for multivariate effects within a gene set. The use of such a model takes into account all genes within the set simultaneously, thus taking into account within gene set correlations. MVBAGS allows for a fully multivariate model, such that not only within gene set correlations are taken into account, but also between gene set correlations are accounted for. The ability to simultaneously consider all genes, either within a gene set or the full compliment of genes, is advantageous as it allows us to recognize active gene sets when single gene effects are small, yet there is a concerted effect from all genes within a set, or in the case of the second model, if several gene sets are working together to affect phenotype.

Thirdly both models produce posterior probabilities of activeness of gene set, rather than p-values against some null hypothesis. As mentioned previously this is advantageous as it allows for a more fluid definition of activeness, and allows us to associate a probability to our statement of activeness.

These models appear to have many advantages over the current methods for the analysis of gene sets. However, it would be very useful to determine how these models work practically. The following chapter goes on to apply these methods to several sets of simulated data.

Chapter 6

Some case studies based upon simulated data

Chapter 5 presented two new Bayesian models for the analysis of gene sets. It is claimed that the methods are superior to existing methods, the main advantages being

- Phenotype is modelled conditional on gene expression.
- Gene expression is modelled simultaneously either within a gene set or for the entire set of genes.
- Posterior probabilities of activeness are computed, rather than p-values.

BAGS and MVBAGS are computationally intensive. Due to the time and computing constraints of this project, it is unfeasible to implement a large scale simulation study. Access to a powerful computer, or a lot of time for running repetitive simulations are beyond the reach of this project and so several small case studies will be designed to allow us to explore some of the properties of BAGS and MVBAGS and to allow for a comparison with GSEA, GSA and BGSA.

The presented models make sense from both a biological and a statistical modelling point of view. However, it must be checked not only that the proposed models work, but how well they work, how they compare to one another and how they compare to the existing methods discussed in Chapters 2 and 3. In applying the five methods to simulated data whereby the truth of any effects within the data are known, we can fairly and independently judge the performance of the proposed methods compared to the existing methods.

The remainder of this chapter is arranged as follows: Section 6.1 outlines the investigation and introduces the scenarios according to which data will be simulated. Section 6.2 presents results from applying GSEA, GSA, BGSA, BAGS and MVBAGS to the simulated data. Finally, Section 6.3 discusses results.

6.1 Outline

As mentioned above, the resources for computationally intensive, repetitive simulations are not available. Due to this, a repetitive simulation study will not be implemented.

Eight data simulation scenarios will be defined, whereby GSEA, GSA, BGSA, BAGS and MVBAGS will be applied to each of the resulting eight data sets. Data will be simulated according to Approach 2, as outlined in Chapter 3, whereby we simulate an $n \times N$ matrix of gene expressions, X, where $n = 100$ is the number of subjects and $N = 200$ is the number of genes. We define gene effects, $\boldsymbol{\beta}$, due to the criteria laid out by our data simulation scenarios. Phenotype, y, is simulated as a random binomial with probability $1/(1+\exp(-X\boldsymbol{\beta}))$ and finally each non-overlapping block of 20 genes constitutes a gene set, such that we have 10 gene sets.

The eight scenarios will therefore be defined by the following criteria:

- The number of active gene sets;
- The number of active genes within an active gene set;
- The size of the effect of the active genes within the active gene set(s) on the linear predictor for phenotype;
- The pairwise correlation between genes within gene sets;

In defining data simulation scenarios we have the problem that there is an infinitely large space of scenarios. We need to define some realistic scenarios that will reveal the behaviour of the five methods in different but realistic situations. As there is no hard and fast definition for an active gene set, we should look at some different ways in which a set could be active and assess each method's ability to detect these active sets. It would be interesting to look at both small and large gene effects with differing numbers of active genes, and with different correlation structures. It would also be of interest to observe the performance of the five methods when there are several active gene sets. The following defines eight data simulation criteria that attempts to cover these points.

The scenarios are defined such that we have situations where a single active gene set with either small numbers of active genes within the active set having relatively large effects, situations where we have many small gene effects and situations where there are a middling number of active genes with middling effect sizes. There should also be scenarios that look at a number of active sets. This is implemented where an active gene set is defined to be the set with middling effects and repeated such that we have some number > 1 of active gene sets within a data set. In each case data are simulated as above with two different pairwise correlations for genes within gene set, these being $\rho = 0$ and $\rho = 0.25$. Table 6.1 summarizes these data simulation criteria.

Scenario	No of active sets	Gene effects in active sets	Correlation
1	1	$\beta_1 = -\beta_2 = 3$	0
2	1	$\beta_1 : \beta_{10} = 1$	0
3	1	$\beta_1 : \beta_{10} = -(\beta_{11} : \beta_{20}) = 0.5$	0
4	3	$\beta_1 : \beta_{10} = 1$	0
5	1	$\beta_1 = -\beta_2 = 3$	0.25
6	1	$\beta_1 : \beta_{10} = 1$	0.25
7	1	$\beta_1 : \beta_{10} = -(\beta_{11} : \beta_{20}) = 0.5$	0.25
8	3	$\beta_1 : \beta_{10} = 1$	0.25

Table 6.1: Simulation criteria for the eight data scenarios

In the next section we present results from the application of GSEA, GSA, BGSA, BAGS and MVBAGS to the simulated data.

6.2 Results

The five methods of interest have been applied to the eight simulated data sets.

- GSEA is implemented with the recommended 1000 permutations.

- GSA is implemented with the 200 recommended permutations.

- BGSA obtains posterior samples and bases its p-values upon a recommended 2000 iterations after a burn-in of 200.

- Probabilities from BAGS and MVBAGS are based upon 20,000 posterior samples after a burn-in of 10,000

Figures 6.1 to 6.8 show bar charts of the results from scenarios 1 to 8. In the case of GSA and BGSA the bars in the bar charts show p-values against their respective null hypotheses. The bars for GSEA represent fdr q-values and in the case of the Bayesian methods as presented in Chapter 5 the bars show $1 - P[active]$ or in other words $P[null]$. It has been chosen to show $1 - P[active]$ for ease of comparison with frequentist p-values when presented on a bar chart. The horizontal lines in each of the plots show p-value cut-offs corresponding to GSEA, GSA and BGSA depending upon the method. It is recommended by Subramanian et al. (2005) when using GSEA to use a cut-off on the FDR q-values of 0.25. Efron and Tibshirani (2006) define a p-value cut-off of 0.05 when using GSA. Shahbaba et al. (2011) employ a cut-off of 0.1 in their paper presenting BGSA, this will be used here. BAGS and MVBAGS produce probabilities of activeness of gene set (shown below as $1 - P[active]$), there is therefore no strict cut-off, however, gene sets with $P[active] \geq 0.5$ will generally be considered interesting.

Table 6.2 provides p-values, q-values and $P[active]$ for the active gene sets in each of the eight scenarios. The underlined values highlight where an active gene set has not been identified by the given method.

As can be seen in Table 6.1 the data simulation criteria for Scenarios 1 and 5, Scenarios 2 and 6, Scenarios 3 and 7 and Scenarios 4 and 8 are identical, apart from the fact that a correlation structure is defined for Scenarios 5, 6, 7 and 8. Comparing Scenarios 1 and 5 (Figures 6.1 and 6.5) it can be seen that with the addition of a correlation structure the performance of GSEA, GSA and BGSA is worse, whereas BAGS and MVBAGS give slightly higher probabilities of activeness. This can also be seen when comparing Scenarios 3 and 7, results of which can be seen in Figures 6.3 and 6.7. Comparing results from

Scenarios 4 and 8, no patterns can be seen moving from uncorrelated data to correlated data. It would be interesting to repeat these analyses in a repetitive simulation study to see exactly how this addition of a correlation structure affects results.

Active set(s)	Method				
identified	GSEA	GSA	BGSA	BAGS	MVBAGS
Scenario 1	0.6826743	0.0700000	0.4389764	0.9502333	0.9997333
Scenario 2	0.0000000	0.0000000	0.1525789	0.9051000	0.9799000
Scenario 3	0.7706575	0.0100000	0.1689010	0.9118333	0.9991333
Scenario 4	0.0099000	0.0050000	0.4757358	0.7581000	0.7338333
	0.0340071	0.0000000	0.3907052	0.9908667	0.9768000
	0.2590949	0.0800000	0.4767546	0.8600000	0.7588000
Scenario 5	1.0000000	0.1350000	0.2118809	0.9994000	0.9997667
Scenario 6	0.0000000	0.0000000	0.1351847	0.9991000	0.9999333
Scenario 7	0.8549705	0.3000000	0.4248656	0.9921333	0.8064667
Scenario 8	0.0245833	0.0050000	0.1323388	0.9998333	0.9999667
	0.1803993	0.0600000	0.1998703	0.8432667	0.9970667
	0.2087529	0.0550000	0.1882647	0.9864000	0.9998667

Table 6.2: P-values,q-values and P[active] for active gene sets. P-values,q-values and P[active] that miss a defined cut off are underlined.

BGSA consistently misses active gene sets as can be seen in Table 6.2, however, this can clearly be seen to be down to the choice of cut-off in the majority of cases. It can be seen that in the majority of scenarios the p-value is much smaller for active sets than for inactive sets (this is not so for Scenarios 4 and 7). This is down to the rather complicated construction of a frequentist style p-value from a Bayesian model, as discussed in Chapter 2. Either a more relaxed cut-off is needed, a slightly different interpretation of the p-value should be made or the calculation of the p-value should be better calibrated. In a real life situation where the truth behind which gene sets are active is unknown, the above comments could not be made and therefore real effects could and would be missed. This possible problem is not highlighted by the repetitive simulation study in Chapter 3 because the AUC is used.

GSA performs reasonably in identifying the active sets, six out of twelve active sets are not identified. GSA fails to identify the active sets in Scenarios 1 and 7, misses one of the three active sets in Scenario 4 and fails to identify two of three active sets in Scenario 8. GSA provides several false positive results, in fact Scenarios 2, 4 and 5 are the only cases where GSA does not give at least one false positive.

GSEA fails to identify the active gene sets in Scenarios 1, 3, 5 and 7, fails to identify one out of three active gene sets in Scenario 4 and fails to identify two out of three active gene sets in Scenario 8. It should be noted that gene sets in Scenarios 1, 3, 5 and 7 are active with gene effects in both a positive and a negative direction. GSEA gives several false positives, these are in Scenarios 1, 3, 6, and 7. Interestingly, in the case of Scenario 1 the active gene set is missed by both GSEA and GSA, yet both methods classify gene set 6 as active when in fact it is inactive.

BAGS can be seen to perform well in all cases. Every active gene set is identified successfully. All inactive sets have $P[active] < 0.5$, the model therefore gives no false positive results for the eight scenarios.

MVBAGS successfully identifies all active gene sets and there are no false positive results.

To summarize the results from BAGS and MVBAGS it can be said that they both appear successful in identifying active gene sets in a range of scenarios.

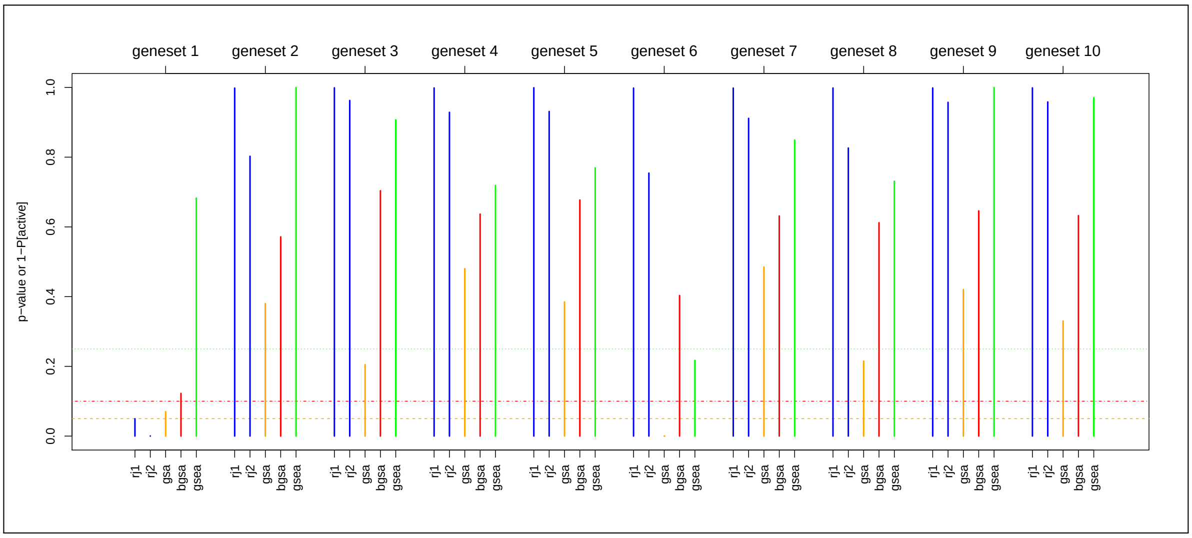


Figure 6.1: Bar chart for Scenario 1 (1 active gene set; $\beta_1 = -\beta_2 = 3$; correlation=0). RJ1 denoting BAGS and RJ2 denoting MVBAGS. Vertical bars showing q-values for GSEA, p-values for GSA and BGSA and $1 - P[active]$ for BAGS and MVBAGS. Dashed horizontal line showing a p-value cut-off of 0.05 for GSA, dot-dash line showing a p-value cut-off of 0.1 for BGSA and dotted line showing an FDR cut-off for GSEA of 0.25.

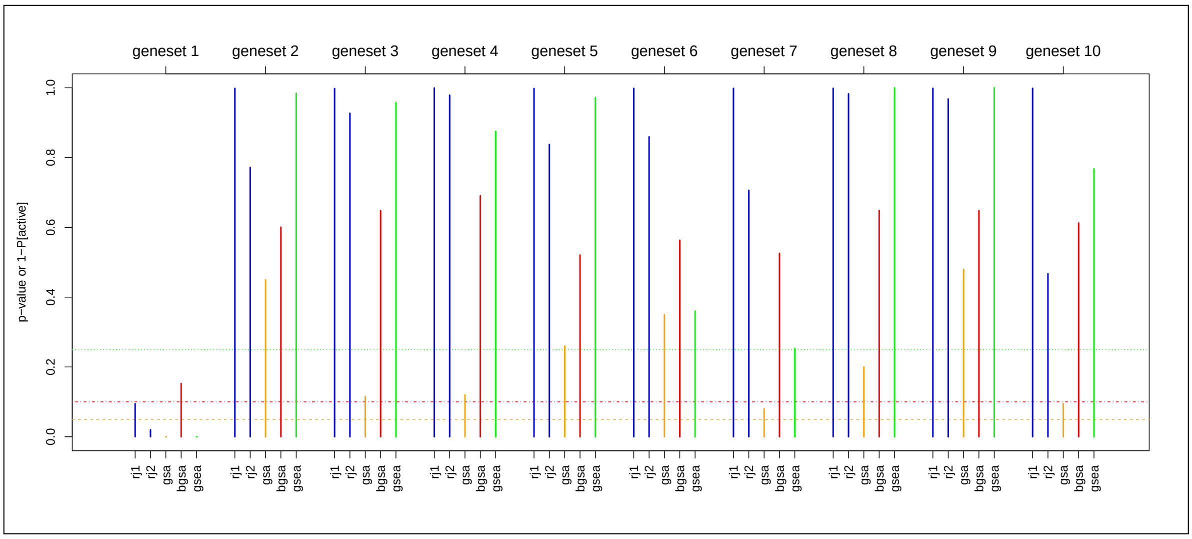


Figure 6.2: Bar chart for Scenario 2 (1 active set; $\beta_1 : \beta_{10} = 1$; correlation=0). RJ1 denoting BAGS and RJ2 denoting MVBAGS. Vertical bars showing q-values for GSEA, p-values for GSA and BGSA and $1 - P[active]$ for BAGS and MVBAGS. Dashed horizontal line showing a p-value cut-off of 0.05 for GSA, dot-dash line showing a p-value cut-off of 0.1 for BGSA and dotted line showing the FDR cut-off for GSEA of 0.25.

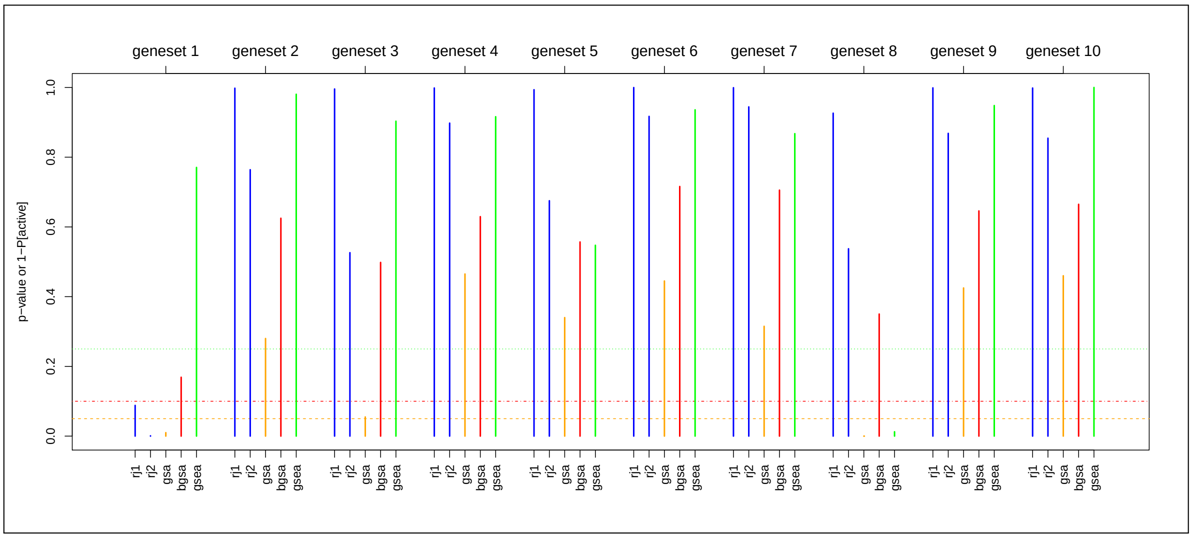


Figure 6.3: Bar chart for Scenario 3 (1 active set; $\beta_1 : \beta_{10} = -(\beta_{11} : \beta_{20}) = 0.5$; correlation=0). RJ1 denoting BAGS and RJ2 denoting MVBAGS. Vertical bars showing q-values for GSEA, p-values for GSA and BGSA and $1 - P[active]$ for BAGS and MVBAGS. Dashed horizontal line showing a p-value cut-off of 0.05 for GSA, dot-dash line showing a p-value cut-off of 0.1 for BGSA and dotted line showing the FDR cut-off for GSEA of 0.25.

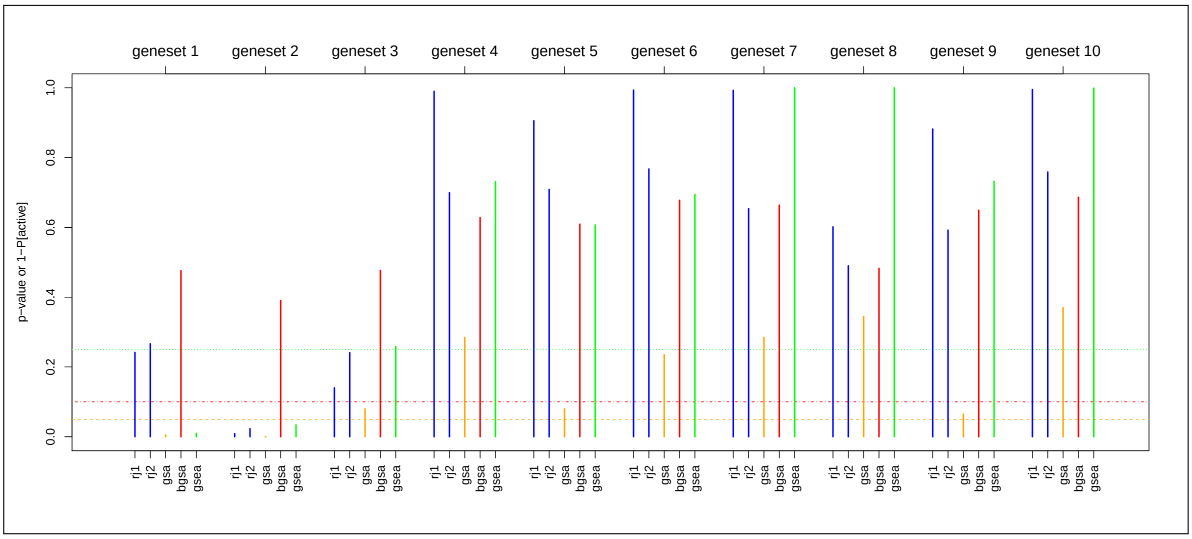


Figure 6.4: Bar chart for Scenario 4 (3 active sets; $\beta_1 : \beta_{10} = 1$; correlation=0). RJ1 denoting BAGS and RJ2 denoting MVBAGS. Vertical bars showing q-values for GSEA, p-values for GSA and BGSA and $1 - P[active]$ for BAGS and MVBAGS. Dashed horizontal line showing a p-value cut-off of 0.05 for GSA, dot-dash line showing a p-value cut-off of 0.1 for BGSA and dotted line showing the FDR cut-off for GSEA of 0.25.

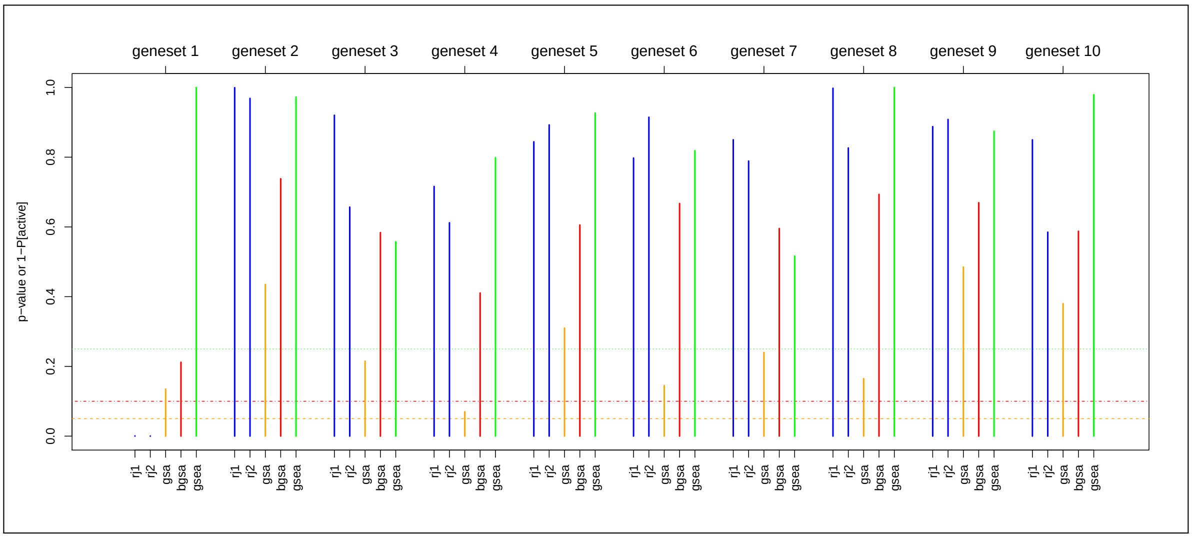


Figure 6.5: Bar chart for Scenario 5 (1 active gene set; $\beta_1 = -\beta_2 = 3$; correlation=0.25). RJ1 denoting BAGS and RJ2 denoting MVBAGS. Vertical bars showing q-values for GSEA, p-values for GSA and BGSA and $1 - P[active]$ for BAGS and MVBAGS. Dashed horizontal line showing a p-value cut-off of 0.05 for GSA, dot-dash line showing a p-value cut-off of 0.1 for BGSA and dotted line showing the FDR cut-off for GSEA of 0.25.

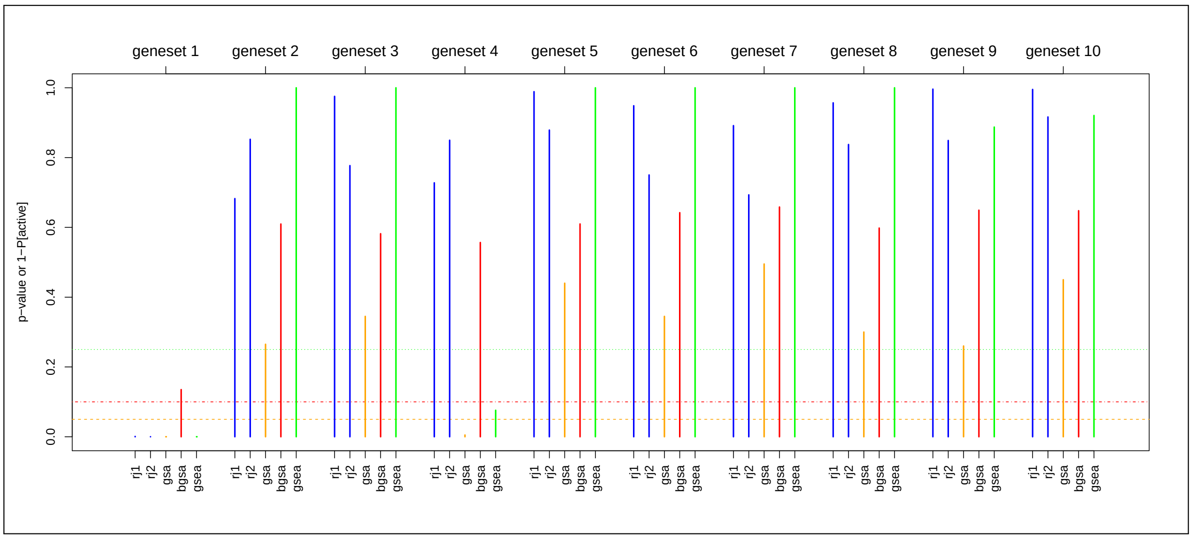


Figure 6.6: Bar chart for Scenario 6 (1 active set; $\beta_1 : \beta_{10} = 1$; correlation=0.25). RJ1 denoting BAGS and RJ2 denoting MVBAGS. Vertical bars showing q-values for GSEA, p-values for GSA and BGSA and $1 - P[active]$ for BAGS and MVBAGS. Dashed horizontal line showing a p-value cut-off of 0.05 for GSA, dot-dash line showing a p-value cut-off of 0.1 for BGSA and dotted line showing the FDR cut-off for GSEA of 0.25.

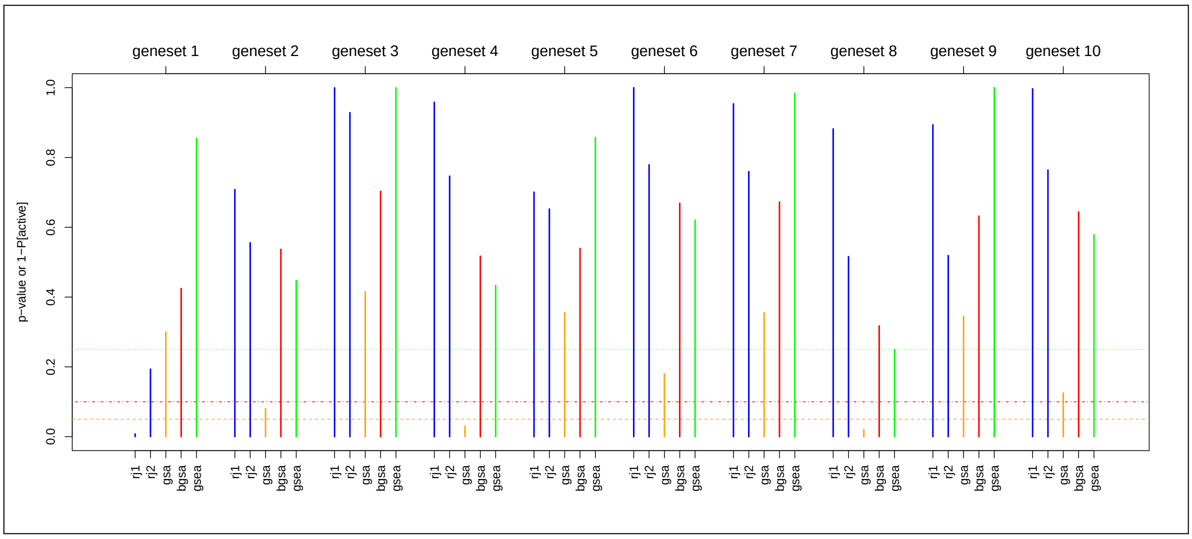


Figure 6.7: Bar chart for Scenario 7 (1 active set; $\beta_1 : \beta_{10} = -(\beta_{11} : \beta_{20}) = 0.5$; correlation=0.25). RJ1 denoting BAGS and RJ2 denoting MVBAGS. Vertical bars showing q-values for GSEA, p-values for GSA and BGSA and $1 - P[active]$ for BAGS and MVBAGS. Dashed horizontal line showing a p-value cut-off of 0.05 for GSA, dot-dash line showing a p-value cut-off of 0.1 for BGSA and dotted line showing the FDR cut-off for GSEA of 0.25.

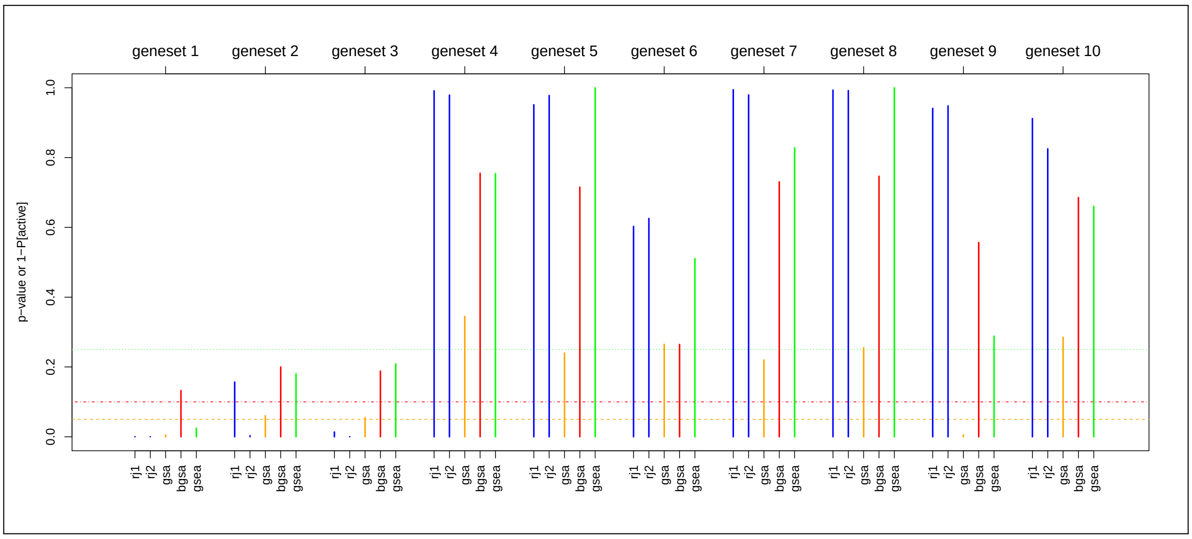


Figure 6.8: Bar chart for Scenario 8 (3 active sets; $\beta_1 : \beta_{10} = 1$; correlation=0.25). RJ1 denoting BAGS and RJ2 denoting MVBAGS. Vertical bars showing q-values for GSEA, p-values for GSA and BGSA and $1 - P[active]$ for BAGS and MVBAGS. Dashed horizontal line showing a p-value cut-off of 0.05 for GSA, dot-dash line showing a p-value cut-off of 0.1 for BGSA and dotted line showing the FDR cut-off for GSEA of 0.25.

6.3 Discussion

The eight Scenarios presented above show a range of simplified possible situations that may be encountered when performing a gene set analysis. It can be seen that BAGS and MVBAGS perform better and more consistently than either GSEA, GSA or BGSA.

The problem of having a p-value cut-off has been highlighted in this chapter. It can be seen that with the recommended p-value cut-off of 0.1 for BGSA many active sets would be missed. However, ignoring the cut-off it can clearly be seen that BGSA does in fact identify several of the active gene sets correctly. This is supported by findings in Chapter 3 where BGSA is proven to be more reliable at identifying more active gene sets than GSEA or GSA, by using the AUC as a measure of performance. The AUC measures the performance of a method across all cut-offs and so the 0.1 cut-off was not an issue there. BAGS and MVBAGS cannot exhibit such problems as no cut-off is necessary.

The performance of GSEA and GSA is consistent with the findings in Chapter 3. However, here GSEA performs a little better in identifying the active gene set than GSA.

GSA also gives more false positive results than GSEA.

BAGS and MVBAGS can be seen to be superior to the current methods that they are compared to here in the scenarios presented. It would be useful to implement a large scale repetitive simulation study on BAGS and MVBAGS. In implementing a larger scale simulation study a wider range of scenarios could be studied and the consistency of the models could be highlighted.

BAGS and MVBAGS are advantageous not only in their performance at identifying active gene sets in this simulation study. Rather than a p-value against some null hypothesis with some arbitrarily defined cut-off, we produce Bayesian posterior probabilities. These probabilities allow us to assess the 'interestingness' of a gene set and to express our uncertainty of this gene set being active, rather than to say 'yes this gene set is related to phenotype' or 'no this gene set is not related to phenotype'.

The advantage of using such modelling techniques as part of a Bayesian model is that we can easily access behind the scenes. For example if we have a gene set where $P[active] = 0.5$ we can quickly and easily look at histograms to determine which genes within the set are having an affect on phenotype and asses trace plots of individual parameters for convergence issues. In a real life situation when we have borderline cases such as in the example above, simply looking at trace plots of individual genes would give a good indication whether to pursue investigation into the gene set or not.

Another advantage is that once an active set has been identified, posterior samples for the gene effect parameters within the gene set can be accessed and we also already have a model that relates the gene set to phenotype.

In conclusion, from the results gained here, it would be recommend to use either BAGS or MVBAGS over the other methods studied in this thesis.

Chapter 7

p53 Data set

7.1 Introduction and Background

The two models introduced in Chapter 5 have been shown to work very well when applied to simulated data in Chapter 6, however there is the need to apply the models to real data. There are several suitable, freely available datasets at `http://www.broadinstitute.org/gsea/datasets.jsp`, for, example there are data on diabetes, leukemia, lung cancer and p53 mutation status amongst others.

The p53 data is somewhat of a benchmark dataset when proposing new gene set analysis methodology with analyses presented in the publications of GSEA, GSA and BGSA. For this reason it has been chosen to apply the first of the two models from Chapter 5 to the p53 data. The second model will not be applied to the data as further work is required in programming the model before it can be applied to data where gene sets overlap. This will be discussed further in Chapter 8.

p53 is a gene that codes for proteins that regulate cell cycle and hence functions as a tumor suppressor. Damage to DNA and other stress signals from the cell, for example ionizing radiation, abnormal growth signals from oncogenes, hypoxic stress, the reaction to hot- or cold-shock and inflammation in tissues can prompt the increase in p53 proteins. These p53 proteins have three major functions:

- Growth arrest - Stops the progression of the cell cycle, thus preventing the replication of damaged DNA;

- DNA repair - During growth arrest p53 may activate the transcription of proteins involved in DNA repair;

- Apoptosis (cell death) - the ‘last resort’ to avoid proliferation of cells containing abnormal DNA.

The different types of stress that are responded to by the p53 protein have one thing in common: they all have the potential to disrupt the efficient and authentic replication of the cell, resulting in an enhanced mutation rate or aneuploidy during cell division.

The cellular concentration of p53 must be tightly regulated. While it can suppress tumors, high levels of p53 may accelerate the aging process by excessive apoptosis. The major regulator of p53 is Mdm2, which can trigger the degradation of p53 by the ubiquitin system. Figure 7.1 shows normal p53 activity pictorially. It is acknowledged that in many cancerous tumors the *p*53 gene can be found to be mutated, and hence not performing its job of tumor-suppressing. In fact over half of all tumors can be seen to have mutated p53 genes. It is therefore of great interest to determine pathways related to the mutation of the *p*53 gene.

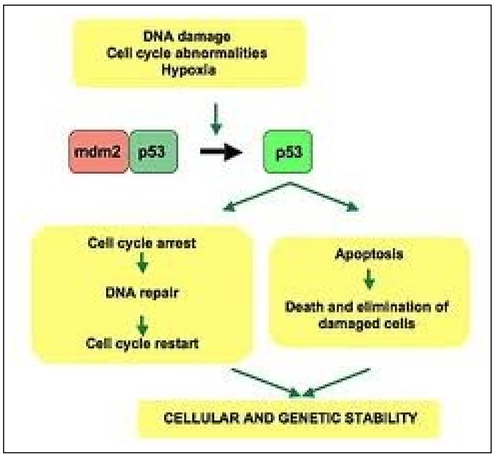


Figure 7.1: Simplified p53 activity.

As mentioned above the p53 dataset has become something of a benchmark when proposing new gene set analysis methodology, with analyses presented by Subramanian et al. (2005), Efron and Tibshirani (2006) and Shahbaba et al. (2011) in the publications of GSEA, GSA and BGSA respectively. The aim of these analyses was to identify pathways that are differentially expressed between cell lines with normal and mutated p53 genes. The same group of gene sets were used by each group, these being the C1 and C2 gene set catalogues, both freely available at `http://w ww.broadinstitute.org/gsea/msigdb/index.jsp`. By combining the C1 and C2 gene set catalogues genes were resultantly allocated to 522 overlapping gene sets.

In analyzing the p53 data using the same 522 gene sets, all three groups (Subramanian et al. (2005), Efron and Tibshirani (2006) and Shahbaba et al. (2011)) found positive results relating gene sets to p53 mutation status. There are some gene sets that all three methods agree are related to p53 mutation status and perhaps more interestingly some gene sets over which the methods disagree. These results will be discussed and compared later in the chapter.

The rest of the chapter is as follows; Section 7.2 outlines the p53 dataset in detail and further discusses the allocation of genes to gene sets. Section 7.3 discusses model fitting. Section 7.4 presents results from applying the first of the two models from Chapter 5 to the p53 data, Section 7.5 compares these results to those presented by Subramanian et al. (2005), Efron and Tibshirani (2006) and Shahbaba et al. (2011). Finally Section 7.6 discusses results produced and comparisons made between the four methods.

7.2 Data

Original data were obtained on interrogating the mutation status of *p*53 in cancer cell lines from the NCI-60 (National Cancer Institute-60) collection (Ross et al. (2000)) which was created to explore gene expression over 60 subjects diagnosed with various cancers. Data from 50 of these cell lines are freely available at `http://www.broadinstitute.org/gsea/datasets.jsp` whereby the expression of 4486 genes is measured over 50 cancer cell lines, 33 of the cell lines having a *p*53 mutation and the remaining 17 being wild type.

Min.	1st Qu.	Median	3rd Qu.	Max.
2.00	10.00	17.00	28.00	358.00

Table 7.1: Summaries of the sizes of the 522 gene sets.

Genes are allocated into 522 gene sets, derived from the C1 and C2 gene set catalogues as mentioned previously. The C1 gene set catalogue, known as the positional gene set collection, is a collection of gene sets corresponding to each human chromosome and each cytogenetic band that has at least one gene. These gene sets are useful for identifying effects related to chromosomal amplifications or deletions, dosage compensation, epigenetic silencing, and other locational effects. The C2 gene set catalogue, known as the curated gene set collection, consists of gene sets collected from sources such as online pathway databases, publications in PubMed, and knowledge of experts in the area. In combining the C1 and C2 gene set catalogues, 522 gene sets were used in each analysis.

No repetitions	Frequency
1	1734
2	961
3	573
4	338
5	260
6-10	410
11-20	157
21-40	45
>40	17
Total	15059

Table 7.2: Summary of the number of repetitions of the 4486 genes in the 522 gene sets.

Table 7.1 shows summary statistics of the sizes of these gene sets. As can be seen gene set sizes are generally spread about 17, however there are some gene sets containing very few genes and some gene sets containing very large numbers of genes. It can be seen from the five number summary in Table 7.1 that the distribution of gene set sizes is very positively skewed. The majority of these gene sets are overlapping, in particular some of the larger gene sets consist almost entirely of smaller gene sets.

Table 7.2 shows a frequency table of the number of times genes have been represented within the 522 gene sets. As can be seen many genes are represented several times, for example the gene known as NFKB1 (NF-kappa-B) is represented 51 times in the 522 gene sets. NF-kappa-B is a gene which influences multiple phenotypic traits, it is present in almost all cell types and is involved in many biological processes such as inflammation, immunity, differentiation, cell growth, tumorigenesis and apoptosis. It would therefore be expected that such a gene would appear many times.

The goal is to identify pathways that are differentially expressed between cell lines with mutated and normal p53 genes.

7.3 Model fitting

There are several sets of parameters that must be provided in order to fit the model. Before defining these parameters it is useful to refresh the definition of the model. As mentioned in Chapter 5, two logistic models are defined, both models having the same likelihood. Looking within gene set s, containing m_s genes, the likelihood is given by

$$f(x|\boldsymbol{\beta}) = \prod_{i=1}^{n} \pi_i^{y_i}(1-\pi_i)^{1-y_i} \tag{7.1}$$

where

$$\pi_i = \frac{1}{1+\exp\{-\eta_i\}} \tag{7.2}$$

and

$$\eta_i = \boldsymbol{x}_i\boldsymbol{\beta} \tag{7.3}$$

where $\boldsymbol{x}_i$ is the vector of m_s gene expressions corresponding to gene set s in the i^{th} subject and $\boldsymbol{\beta}$ is the vector of slope parameters for gene set s.

The null and active models are distinguished by their respective prior distributions. The prior for the null model (M_1) is given by

$$f(\boldsymbol{\beta}) = \frac{1}{\sqrt{2\pi}\sigma}\exp\left\{\frac{-\boldsymbol{\beta}^2}{2\sigma^2}\right\} \tag{7.4}$$

and the prior for the active model (M_2) is given by

$$f(\boldsymbol{\beta}) = \frac{1}{\sqrt{2\pi}}\left(\frac{(1-p)}{2\sigma_1}\exp\left\{\frac{-(\boldsymbol{\beta}-\mu)^2}{2\sigma_1^2}\right\} + \frac{p}{\sigma_2}\exp\left\{\frac{-\boldsymbol{\beta}^2}{2\sigma_2^2}\right\} + \frac{(1-p)}{2\sigma_1}\exp\left\{\frac{-(\boldsymbol{\beta}+\mu)^2}{2\sigma_1^2}\right\}\right) \tag{7.5}$$

The prior information for the model move is specified to represent the belief that relatively few gene sets will have an effect on phenotype. The prior information for the active model is specified to represent beliefs that relatively few genes within an active gene set will have a large (either positive or negative) effect upon phenotype, whilst most of the genes will have little effect on phenotype. The prior distribution for the null model is specified to represent the belief that within an inactive gene set gene effects will be small and centered about zero. We therefore define the prior parameters as follows:

1 Null model prior parameters:

   - $\sigma = 0.1$

2 Active model prior parameters:

   - $\sigma_1 = 0.05$
   - $\sigma_2 = 0.6$
   - $\mu = 1$
   - $p = 0.8$

3 Reversible jump prior:

   - $P[M_2] = 0.8$

We put these models in a RJMCMC framework, whereby not only do we propose new parameters ($\boldsymbol{\beta}$) for a model and accept with probability $\alpha(\boldsymbol{\beta}, \boldsymbol{\beta}^{prop})$ but we also propose to move from the null model (M_1) to the active model (M_2) with probability $\alpha(M_1, M_2)$ and vice versa.

The within model proposal distribution, which is used in Algorithm 4.1, is given by

$$q(\beta_{prop}|\beta) = \frac{1}{2}(1-q)N(-\delta, \tau_1^2) + qN(\beta, \tau_2^2) + \frac{1}{2}(1-q)N(\delta, \tau_1^2) \qquad 0 < q < 1 \quad (7.6)$$

The parameters for the within model proposal distribution have been chosen to speed up convergence of chains, based on the experimentation presented in Chapter 4. These are:

- $\tau_1 = 0.05$
- $\tau_2 = 0.6$
- $\delta = 2$

- $q = 0.8$

Chapter 5 also describes the necessity for a mixing aid within the reversible jump move of the model. This mixing aid is simply the addition of a zero mean random variable to the parameters when moving from one model to the other. We define this as

$$\boldsymbol{\beta}' = \boldsymbol{\beta} + N(0, \sigma_{mix}^2) \tag{7.7}$$

On running the algorithm and systematically exploring values for σ_{mix} it is found that the best $\sigma_{mix} = 0.0675$.

The model is applied to the p53 dataset with Markov chain Monte Carlo simulations for 50,000 iterations, of which the first 20,000 are discarded as burn-in. Convergence has been assessed by checking trace plots. Due to the number of parameters the full set of trace plots are not provided. However, Figure 7.2 shows two typical trace plots obtained. Figure 7.2 panel A shows a trace plot for the model moves for a randomly selected gene set, this being *parkinsonsPathway*. Figure 7.2 panel B shows a trace plot of the posterior samples for the parameter of a gene in this set, this being PARK. As can be seen in both cases chains mix well. The chain for the model move explores the space of models well, with moves to both models accepted frequently. The chain for the gene parameter can be seen to converge about zero.

7.4 Results

Figure 7.3 shows a plot of the density of the posterior probabilities of activeness for the 522 gene sets. A distinct separation between active and in-active gene sets can be seen in the bi-modality of the posterior distribution. This shows the ability of the model to distinguish between active and inactive sets.

Figure 7.4 shows a plot of the density of 1-(p-value) from the BGSA of the 522 gene sets. These are the results reported by Shahbaba et al. (2011) and were obtained from `http://www.ics.uci.edu/ babaks/Site/Home.html`. This plot gives a very different picture from that produced by the proposed model. Here can be seen a roughly bell

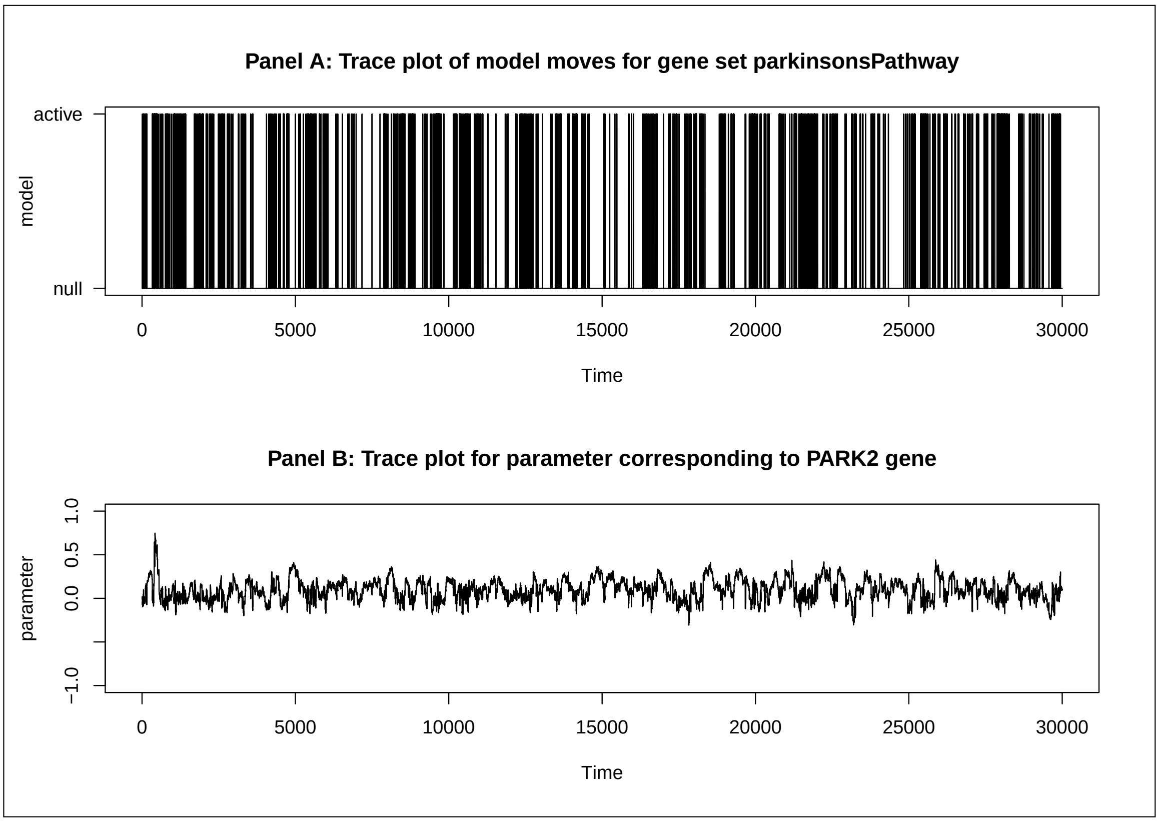


Figure 7.2: Panel A shows trace plot of model moves for gene set *parkinsonsPathway* with $P[active] = 0.03347$. Panel B shows a trace plot of the posterior samples for the parameter corresponding to PARK gene within this gene set.

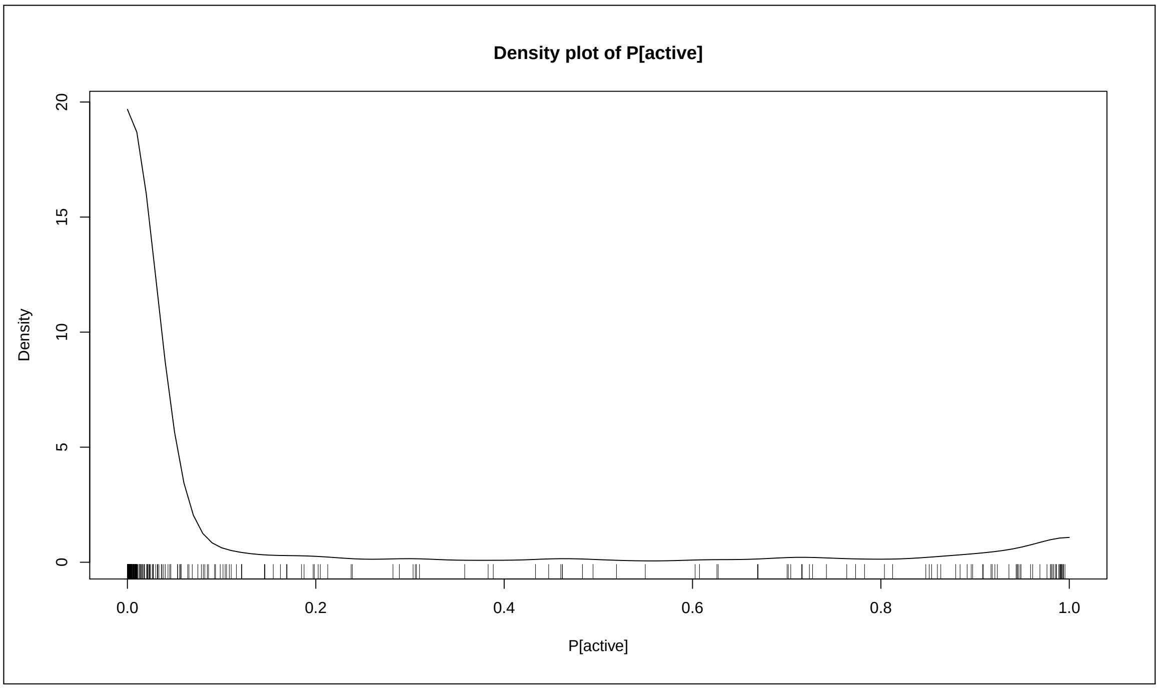


Figure 7.3: Plot showing posterior distribution of $P[active]$

shaped curve with a mode at approximately 0.5. There can be seen no separation between active and in-active gene sets, instead we see the curve flattening off in either extreme. The

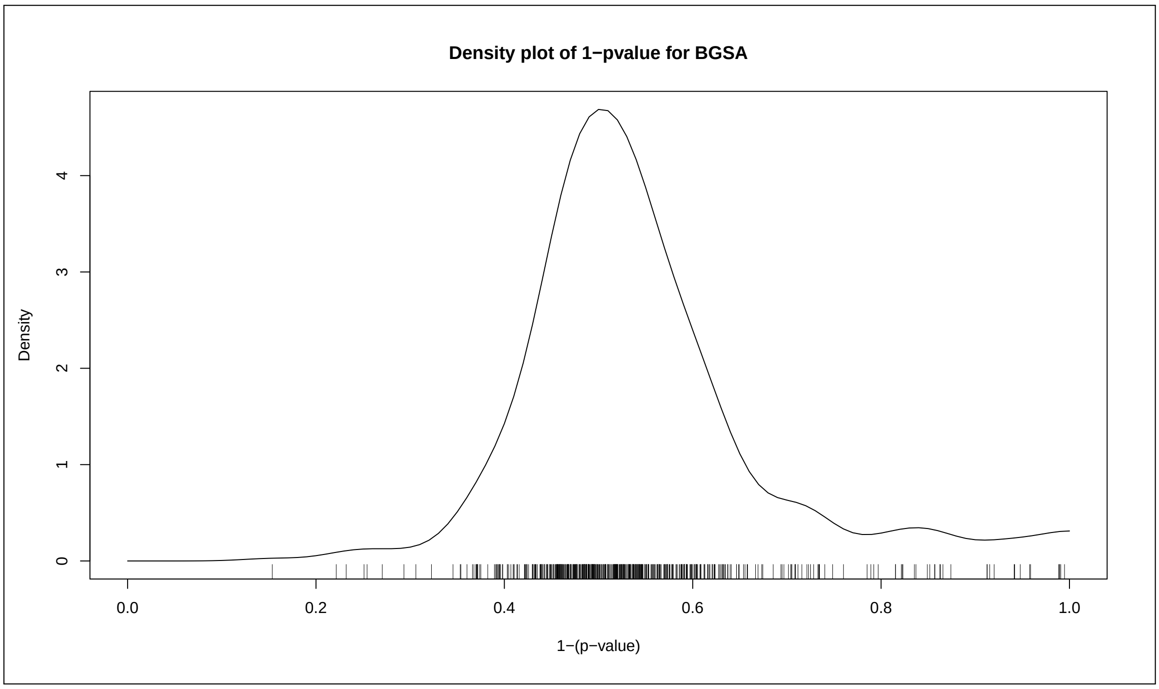


Figure 7.4: Plot showing posterior distribution of $1-(p-value)$ for BGSA

p-values other than those in the right hand tail of the plot in Figure 7.4 tell us little. However, in comparison all probabilities shown in Figure 7.3 have a clear and interpretable meaning:

- Those in the mode at zero tell us that these corresponding gene sets are very unlikely to have a relationship with p53 mutation status;
- The relatively few probabilities between this mode and 0.5 tells us that these gene sets are unlikely to have a relationship with p53 mutation status;
- The probabilities between 0.5 and the right hand mode tell us that the corresponding gene sets could have a relationship with p53 mutation and the further to the right we go the more worthy these gene sets are of further investigation;
- The probabilities that make up the mode near 1 indicate that the corresponding gene sets are very likely to have a relationship with p53 mutation status and investigation and interpretation would certainly be worthwhile.

Figure 7.4 also highlights the problem of arbitrarily setting a p-value cut off. A cut off of $p \leq 0.1$ is recommended in Shahbaba et al. (2011), this translates to $(1-p) \geq 0.9$ in the plot in Figure 7.4. It could be that all gene sets not contained in the central mode are interesting and in setting this cut off we discard more than half of them. Conversely, we see a clear group of interesting gene sets from the plot in Figure 7.3, there is no cut off on the probabilities and so none of these potentially interesting and active gene sets are overlooked.

From a practical point of view the results shown in Figure 7.3 would be much better to deal with than those shown in Figure 7.4. This is because the proposed model not only clearly distinguishes the active gene sets from all others but also provides probabilities of activeness, thus making all results meaningful and interpretable.

Figure 7.5 shows all positive results gained from applying the first of the two models from Chapter 5 to the p53 data, where a positive result is defined as $P[active] \geq 0.5$. Clearly we would not class all of these gene sets as particularly influential on p53 mutation status, however, it is of interest to identify all gene sets with some positive result.

Of greater interest are the gene sets where the probability of activeness is very high. Table 7.3 shows gene sets where $P[active] \geq 0.95$, of which there are 22. Of these 22 gene sets, many are large gene sets, with 12 out of the 22 gene sets containing more than 100 genes. There can be seen to be three main themes to the gene sets that are found to be active. These are:

1 Cell cycle, cell growth and cell death - Associated with tumor growth and suppression, which is governed by p53;

2 Hypoxia and respiratory system - Hypoxic stress, like DNA damage, induces the accumulation of p53 proteins;

3 Energy production and transportation - p53 mediates metabolic changes in cells through the regulation of energy metabolism and oxidative stress.

These three themes are interconnecting and extremely complex. These themes are well documented in their relationships with p53, for example Vogelstein et al. (2000), Harris and Levine (2005) and Puzio-Kuter (2011).

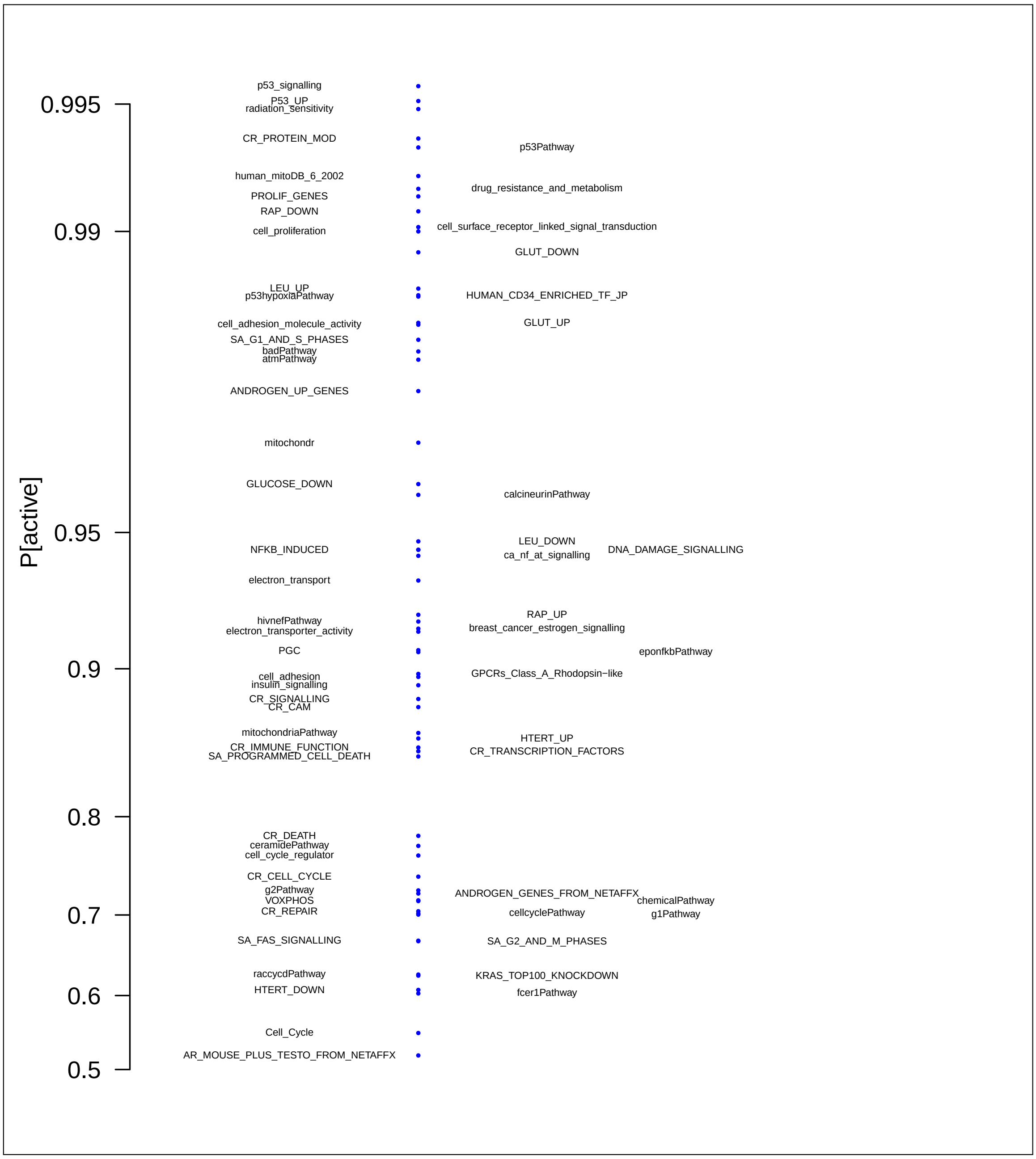


Figure 7.5: Plot showing probabilities for gene sets with $P[active] \geq 0.5$ on a logit scale

Pathway	$P[active]$	No genes	Pathway description
p53 signaling	0.9955	87	Signaling pathway which turns p53 on.
CR PROTEIN MOD	0.9940	146	Predictor of cardiovascular disease (hypoxia).
p53Pathway	0.9937	16	Network of genes related to the expression and functioning of p53.
human mitoDB 6 2002	0.9926	326	Mitochondrial genes. Related to the production of energy within the cell.
drug resistance and metabolism	0.9921	95	Produce proteins for drug metabolizing enzymes and transporters.
PROLIF GENES	0.9917	358	Cell proliferation. Cell growth used in the context of cell development and cell division.
RAP DOWN	0.9910	211	RAP members aid in the regulation of cell proliferation, differentiation, apoptosis and cell adhesion mechanisms.
cell surface receptor linked signal transduction	0.9902	123	Produce proteins involved in communication between the cell and the outside world.
GO 0005739	0.9900	144	Involved with respiration, signaling, differentiation, apoptosis and cell growth.
cell proliferation	0.9900	200	Cell proliferation. Cell growth used in the context of cell development and cell division.
GLUT DOWN	0.9888	285	Insulin regulated glucose transporter. Responsible for glucose translocation into the cell.
HUMAN CD34	0.9859	175	Involved in cell adhesion and aids T-cells entering lymph nodes.
p53 hypoxia Pathway	0.9858	20	Hypoxic stress, like DNA damage, induces p53 accumulation.
GLUT UP	0.9836	269	Insulin regulated glucose transporter. Responsible for glucose translocation into the cell.
cell adhesion molecule activity	0.9834	95	Play a critical role in hemostasis, immune response, inflammation, embryogenesis, and development of neuronal tissue.
SA G1 AND S PHASES	0.9820	14	Pathway involved in mitosis, DNA replication and synthesis and the cell proliferation cycle.
bad Pathway	0.9809	21	BAD physically interacts with cytoplasmic p53, thereby preventing p53 from entering the nucleus.
atm Pathway	0.9800	19	ATM encodes a protein kinase that acts as a tumor suppressor.
ANDROGEN UP GENES	0.9763	57	Androgens are steroids that constitute the male sex hormones. Upregulation of receptor activity has been implicated in prostate cancer.
mitochondr	0.9688	330	Mitochondrial genes. Related to the production of energy within the cell.
GLUCOSE DOWN	0.9612	145	Pathway related to the breakdown of glucose.
calcineurinPathway	0.9589	18	Activates T-cells particularly cytotoxic T cells, which kill cancer cells.

Table 7.3: Gene sets with $P[active] \geq 0.95$

7.5 Comparisons with GSEA, GSA and BGSA

Table 7.4 shows published results from Subramanian et al. (2005), Efron and Tibshirani (2006) and Shahbaba et al. (2011) from analyses of the p53 data using GSEA, GSA and BGSA respectively. Table 7.4 also shows the results gained from the analysis presented in this chapter, along with relevant gene set sizes. Also, three of the most probable gene sets identified by BAGS, but none of the other methods, are shown.

Pathway	Gene set size	GSEA	maxmean	BGSA	BAGS
p53hypoxia Pathway	20	< 0.0001	significant	0.011	0.9858
hsp27 Pathway	15	< 0.0001	significant	...	0.3101
p53 Pathway	16	< 0.0001	significant	0.004	0.9937
P53 UP	40	0.0013	significant	0.016	0.8039
radiation sensitivity	26	0.0078	significant	0.004	0.8125
ras Pathway	22	0.1710	significant	...	0.2015
SA G1 AND S PHASES	14	...	significant	...	0.9820
DNA DAMAGE SIGNALLING	90	...	...	0.047	0.9453
cell cycle regulator	23	...	...	0.023	0.7637
n.g.f pathway	19	...	significant	...	0.0667
ATM pathway	19	...	...	0.079	0.9800
g2 pathway	23	...	...	0.091	0.7275
BAD pathway	21	...	...	0.098	0.9809
p53signalling	87	...	...	...	0.9955
CR protien mod	146	...	...	...	0.9940
human mito db	326	...	...	...	0.9926

Table 7.4: Table showing q-values from GSEA with an FDR cut off of 0.25, indication of significant gene sets from GSA (with an FDR cut off of 0.1), p-values for BGSA with a cut off of 0.1 and $P[active]$ for BAGS. All cut offs are defined in the relevant publication of the methods.

It can be seen that for all gene sets found to be significantly related to p53 mutation status by BGSA, BAGS attaches high probabilities of activeness. In fact, it is only the SA G1 AND S PHASES pathway over which the method presented and BGSA disagree.

There are two cases where both GSEA and GSA identify gene sets as significantly related to p53 mutation status, whilst BGSA does not and BAGS gives low associated probabilities of activeness. These are the hsp27 pathway and the ras pathway.

GSA identifies SA G1 AND S PHASES as significant, which is in agreement with BAGS but disagreement with GSEA and BGSA. GSA also identifies the n.g.f pathway as significant which none of the other methods do.

In general BAGS gives consistent results with those gained by the three other methods.

There is also strong evidence that several additional gene sets are related to p53 mutation status as shown in Table 7.3. For example, the gene set given the highest probability of activeness by BAGS, *p53signalling* ($P[active] = 0.9955$) is not identified by the three existing methods. The reason for this is that of the 87 genes within the gene set there are only six genes with absolute effects of ≥ 0.5. The three existing methods miss this strong gene set effect because of their univariate nature. The relatively few gene effects within the set are diluted by all of the genes with little to no effect. It should be noted that all gene sets identified by GSEA and GSA are small and only one large gene set is identified by BGSA. This is due to the dilution of effects within the larger gene sets.

7.6 Discussion

BAGS can be seen to work well on the p53 data set. Several gene sets have been found to be strongly related to p53 mutation, many of these being previously unidentified by Subramanian et al. (2005), Efron and Tibshirani (2006) or Shahbaba et al. (2011). Looking at the biological literature, for example Vogelstein et al. (2000), all of these pathways have an interpretable connection with p53 mutation status.

Notably in comparison to BGSA (Figures 7.3 and 7.4) the proposed model clearly separates active and in-active gene sets. Moreover, by providing the posterior probabilities of activeness, our method does not rely on arbitrarily defined cut-offs unlike the three other methods studied in this thesis, who all suffer from this serious disadvantage.

Chapter 8

Discussion and future work

8.1 Summary

Genomewide expression analysis with DNA microarrays have become key in many genomics research projects over the past decade. The challenge of extracting meaningful biological insight and determining interpretable and reproducible results from such analyses has become an active area of research in itself. There is much interest in gene set analyses methods, which are a very popular tool for managing the above problem, with many researchers aiming to improve interpretation and reproducibility of microarray analyses through the inclusion of additional biological information.

This thesis focuses on methodology which attempts to relate biologically defined sets of genes with phenotype. A gene set comprises some number of genes that are related either by location on the chromosome or by function and methodology is based on the expectation that the effect on phenotype is restricted to a small number of gene sets. A typical data set upon which these methods would be employed would consist of gene expression measurements of thousands of genes across any number of subjects (generally small). Gene expression is typically transformed such that it follows a standard normal distribution, this transformation is generally relied upon by methodologies in this area.

This thesis begins by reviewing and comparing some of the existing methodology for the analysis of sets of genes. There are many methods from which to choose, yet here we focus upon three of the available methods: GSEA, GSA and BGSA. GSEA brought gene set analysis methods to the forefront and was chosen as it is the most widely cited gene set analysis method to date. One such paper citing GSEA is that by Efron and Tibshirani (2006), who generalize the GSEA framework and introduce a new methodology for the analysis of gene sets that is developed under this general framework, coined GSA. Finally, BGSA was chosen for two reasons, firstly as it represents the latest methodology; first published in 2011. Secondly BGSA was chosen as it is one of the few available Bayesian methodologies for the analysis of gene sets.

Chapter 2 describes these methods in detail, discusses the papers presenting them, and critically looks at the methods from a modelling perspective. A defining characteristic of these methods is that they are based on a construction in which the dependency between gene expression and outcome is defined in terms of the conditional distribution of gene expression|phenotype. One consequence of this, covered in Chapter 3, is that in published simulation studies where data are simulated assuming such a relationship expression levels for active genes no longer follow a standard normal distribution, thus removing any standardization. From both a biological and a statistical modelling perspective it could be interesting to model phenotype conditionally on gene expression. This way of conditioning would give much more justification to the term ‘pathway to phenotype’. Gene set analysis is based on the notion that a small minority of genes restricted to a small number of gene sets are substantially associated with phenotype but others have little or no association with phenotype. This notion can be captured formally in the prior distribution of a Bayesian analysis. Many of the remarks made in Chapter 2 about GSEA, GSA and BGSA are true for the majority of work in the area of gene set analysis methodologies.

Chapter 3 proceeds to study the behavior of these methods by way of an extensive simulation study. This chapter begins by focusing upon data simulation, in particular the conditionality between phenotype and gene expression when simulating data. A new data simulation procedure is introduced, whereby phenotype is simulated conditional on gene expression, the reverse of the general assumption of modelling expression|phenotype. This simulation procedure maintains the standard normal marginal distribution whilst allowing for associations between phenotype and gene expressions. This is an improvement over the generally accepted data simulation procedure, where for active genes data are simulated in such a way that removes the standardization. This new data simulation procedure also allows for the definition of a correlation structure.

All three of the methods can be seen to work well in some scenarios. However, when there are both up- and down-regulated genes within a gene set the performance of GSA and particularly GSEA suffers. It is also shown that as pairwise correlation between genes within gene set increases, the performance of all three methods drops. This can be attributed to increased numbers of false positive results with correlated data, which in turn is caused by the conditioning of the three methods. Finally, it is discussed that the necessity of a p-value cut-off to determine a gene set active/inactive can be somewhat restrictive when interpreting results.

The exploratory nature of these chapters allowed us to reflect upon possible improvements and areas of development for the models as proposed in this thesis. Chapters 4 to 6 focus upon methodology based on the following principles:

- Rationale: Rather than using ad-hoc analytical methods to determine differential expression, probability models should be used to represent biological mechanisms;

- Conditioning: Condition phenotype on gene expression rather than gene expression on phenotype;

- Model formulation: Use multivariate models rather than large scale univariate testing or modelling;

- Prior Information: The notion that a small minority of genes, restricted to a small number of gene sets are substantially associated with phenotype, whilst others have little or no association with phenotype, should be reflected in a prior model.

- Output: Produce posterior probabilities of activeness of gene set rather than a p-value with a corresponding arbitrarily defined cut-off.

In this thesis models in which phenotype is assumed conditional on gene expression have been explored. The comparison between such models and the existing methods is analogous to the comparisons between a series of t-tests and a logistic regression.

The models as introduced in Chapters 4 and 5 rely upon the definition of strong prior distributions. The model for an active gene set makes use of a strong prior for gene effects based on a mixture of three normal distributions. The motivation for such a prior was that we know there will be a range of gene effects: those with no effect upon phenotype, those with a relatively large positive effect on phenotype and finally those with a relatively large negative effect on phenotype. The model for an inactive (null) set uses a strong prior based on a normal distribution with mean zero and a small variance to represent beliefs that gene effects within a null gene set will be small and centered about zero.

Chapter 4 proposes a hybrid Bayesian/ frequentist model that employed the above points, except for producing a posterior probability of activeness. The problem of taking the information from a group of genes and reducing this into a probability proved to be problematic and therefore a frequentist style p-value was calculated thus rendering this a hybrid approach. Most importantly, Chapter 4 proposed Bayesian models for gene sets and introduced and calibrated an MCMC algorithm to obtain posterior samples of the parameters of these models, which gave a good base from which to design fully Bayesian approaches.

The null and active gene set models as introduced in Chapter 4 proved to fit simulated data well, however, the gap between fitting these models and producing a posterior probability of activeness needed to be bridged. In implementing a more general RJMCMC algorithm that allowed the previous MCMC algorithm to jump between the null and active models, posterior probabilities for activeness could be produced. Chapter 5 introduces and describes this model (BAGS) in detail. In fitting this model, some mixing problems were encountered in the reversible jump step. These mixing problems were overcome by the addition of a zero mean random variable in the between model variable transformation stage. Chapter 6 shows that this method works very well when applied to simulated data, and performs better than the existing methods.

One criticism that could be made of BAGS is that it considers gene sets as independent and therefore it is not a fully multivariate model. MVBGSA builds upon BAGS and overcomes this. A fully multivariate approach is proposed whereby all genes and therefore gene sets are simultaneously considered. This allows the full correlation structure of the data set to be accounted for. RJMCMC techniques are again employed to allow for the exclusion or inclusion of gene sets. This results in a final model where:

- Only gene sets that have an effect on phenotype are included;
- An associated probability for the model formulation is produced;
- Probabilities relating to every gene set's activeness in the pathway to phenotype are produced.

MVBAGS therefore not only does the job of a gene set analysis in determining an interesting subset of gene sets, but it also negates the necessity for the typical further analysis and/or modelling of this interesting subset of gene sets.

Chapter 6 applies and compares GSEA, GSA, BGSA, BAGS and MVBAGS. Extensive, repetitive simulations (as in Chapter 3) were not feasible within the constraints of this project due to the high computational cost of repeatedly running the models as proposed in Chapter 5. It is discussed how the computational cost of the models should not be considered detrimental to their practical applicability, due to the availability of relatively powerful machines to the typical user of such methods. The important point to take from Chapter 6 is that both of the models work very well in all scenarios and that their performance is superior to the three existing methods.

Chapter 7 presents the application of the first of the two models introduced in Chapter 5 to the p53 data set. We find strong evidence that 22 out of 522 gene sets are strongly related to p53 mutation status (For each of these gene sets $P[active] \geq 0.95$). These 22 gene sets act under one of three themes, these being:

1 Cell cycle, cell growth and cell death;

2 Hypoxia and respiratory system;

3 Mitochondrial function, energy production and energy transportation.

These themes are well documented in their relationships with p53, for example Vogelstein et al. (2000), Harris and Levine (2005) and Puzio-Kuter (2011). The results obtained in

Chapter 7 were also compared to the positive results presented by Subramanian et al. (2005), Efron and Tibshirani (2006) and Shahbaba et al. (2011). It is shown that amongst the 22 top hitting gene sets, the proposed method identifies the majority of the positive results found by GSEA, GSA and BGSA. In particular, the most probable two gene sets identified by BAGS (*p53signalling* and *CR PROTIEN MOD*) are not identified by GSEA, GSA or BGSA. It should also be noted that none of these three methods identify gene sets related to theme 3 (above).

We do not apply the second of the models from Chapter 5 to the p53 data, as further work to the programming of this model is required before it can be applied to overlapping gene sets. This will be discussed in further detail below.

8.2 Future work and further development

In this thesis the problem of analyzing the relationship between the expression of predefined, functionally related sets of genes and phenotype has been investigated. The models as presented in Chapter 5 can be seen to work well when applied to simulated data in Chapter 6 and when applied to real data in Chapter 7. The models prove to have many desirable properties and consistently exhibit superior performance over the existing methods.

There are many areas where further work is needed and it should be recognised that this thesis is a starting point for much further research in the area. The following goes on to discuss some of the possible future work that is beyond the scope of this thesis.

8.2.1 Computational aspects

The next steps in the development of the new models is to make them available in routine practice by improved implementation within a more efficient computing environment, such as C++, Fortran or Python. One such future direction for improving the computational efficiency of these models would be to re-program each model in parallel, such that when implemented on a multicore processor the job would be split between the available cores on the machine.

8.2.2 Simulation and application

The models presented in Chapter 5 are shown to be superior to the existing methods on the eight simulated data sets in Chapter 6. However, it needs to be shown whether the methods work consistently rather than in single instances. The models should also be shown to work across a broader range of scenarios than those presented in Chapter 6. A future simulation study should systematically vary

- the number of active genes within a gene set.
- gene effect size and direction.
- the number of active gene sets.
- pairwise correlation between genes.

We must be able to identify trends to how the models behave and when and where the models work well.

The second of the models presented in Chapter 5 needs further development before it can be applied to real data. It is for this reason that an application to real data is not provided in Chapter 7. In order to facilitate an application to real data the programs for the model should be modified to allow for overlapping gene sets.

Consider two gene sets, $s = 1$ and $s = 2$ each containing some number of different genes, but both containing gene g. Now, we cannot allow for two parameters for gene g, yet if we have a single parameter for gene g then when inevitably we want to remove one of the sets from the model, but keep the other, there is the question of how to deal with gene g. A way in which to avoid this problem is to have indicator terms for genes that are in several sets. In the above example the term for gene g in the likelihood would be

$$(I(\exists s : s = 1) + I(\exists s : s = 2))\beta_g x_g$$

where $I()$ is the indicator function such that $I(\exists s : s = 1) = 1$ if set $s = 1$ exists in the model and $I(\exists s : s = 1) = 0$ if set $s = 1$ does not exist in the model and so on for any other number of gene sets.

8.2.3 Modelling

The methodology for the two models introduced in Chapter 5 can be considered as a starting point for the two models and there are still many areas which need to be further investigated or in which they could be improved.

Both of the models rely heavily upon the definition of strong prior distributions. However, little formal investigation has been made into different prior distributions. There are infinately many distributions that could work for both the active and null models. For example an alternate prior distribution for the active model could be

$$f(\beta) = pU(-0.1, 0.1) + (1-p)U(-2, 2)$$

and an alternate prior for the null model could be

$$f(\beta) = U(-0.2, 0.2)$$

A thorough investigation into the prior distributions for the two models should be carried out. There are many ways in which we could alter the prior distributions for the first of two reversible jump models. For example we could relax the prior for the null model allowing for some active genes within the null model and strengthen the active model, allowing for fewer non-active genes within an active gene set. Clearly there are infinite possibilities for such an investigation, but it would be very useful to investigate some general properties of altering the prior distribution.

In strengthening the prior and allowing for fewer inactive genes within gene set for the second of the two reversible jump models we would widen the gap between the probability of inclusion/ exclusion of a gene set. This would increase the number of moves between models. It would be of interest to see how this might affect the posterior probabilities of activeness for gene sets.

For the second of the two presented models, it may be interesting to look at adapting the active model into a hierarchical model, whereby we could put hyperpriors on the means, variances and mixing proportions. Possibly this would involve changing the original prior to perhaps a bi-modal distribution representing a high proportion of active genes. Another interesting variation on the second model from Chapter 5 would be to move away from the idea of gene sets and go through the exclude/ include procedure by gene rather than gene set.

Covariates such as lifestyle and demographic factors that might be related to phenotype or can explain differential gene expression could, with little difficulty, be incorporated into the gene set analysis models as presented in Chapter 5. For example when exploring disease phenotypes age, BMI, smoking status, etc could be included into these models. There is no obvious way of achieving this when the conditioning is done on the phenotype.

8.2.4 The application of Bayesian networks to gene set analysis

Bayesian networks (BNs) could be applied to the identification of relationships between pre-defined functionally related sets of genes and phenotype.

A BN is a probabilistic graphical model that represents a set of random variables and their conditional dependencies via a directed acyclic graph (DAG). Formally, Bayesian networks are DAGs whose nodes represent random variables and edges represent conditional dependencies. Nodes which are not connected represent variables which are conditionally independent of each other. Each node is associated with a probability function that takes a particular set of values for the node's parent variables as input and gives the probability of the variable represented by the node.

BNs capture linear and non-linear interactions, handle stochastic events accounting for noise and focus upon local interactions, which can be related to causal inference. BNs are able to focus on local interactions, where each node is directly affected by a relatively small number of nodes. Such properties are widely observed in biological networks. This makes BNs great tools for modelling gene expression data and particularly pathway analysis.

There are several proposed methods utilizing BNs for gene set analysis. For example Isci et al. (2011) use pathway information to model each biological pathway as a BN and quantify the degree to which observed experimental data fit this BN using Bayesian Dirichlet equivalent (BDe) score calculation. They assess statistical significance for the BDe score of each pathway by testing it against datasets generated by applying randomization via bootstrapping. Some other examples of the use of BN's for pathway analysis can be seen in Bauer et al. (2010) and Zou and Conzen (2005) .

There are many ways in which BNs could be used in the context of gene set analysis. One such approach that follows on from the models presented in this thesis would be to allow for variable dimension BNs that could either be fully driven by the data, or begin as pre-defined gene sets. We could allow for the inclusion and exclusion of entire sets of genes and compute probabilities for relationships between gene sets and phenotype in this way. A further extension could be to allow for the inclusion and exclusion of genes from gene sets, such that the pre-defined pathways are used to represent our prior knowledge. It would also be interesting to look at BN's driven solely by data to see how (if at all) any pre-defined pathways are captured and their functional relationship(s) with phenotype.

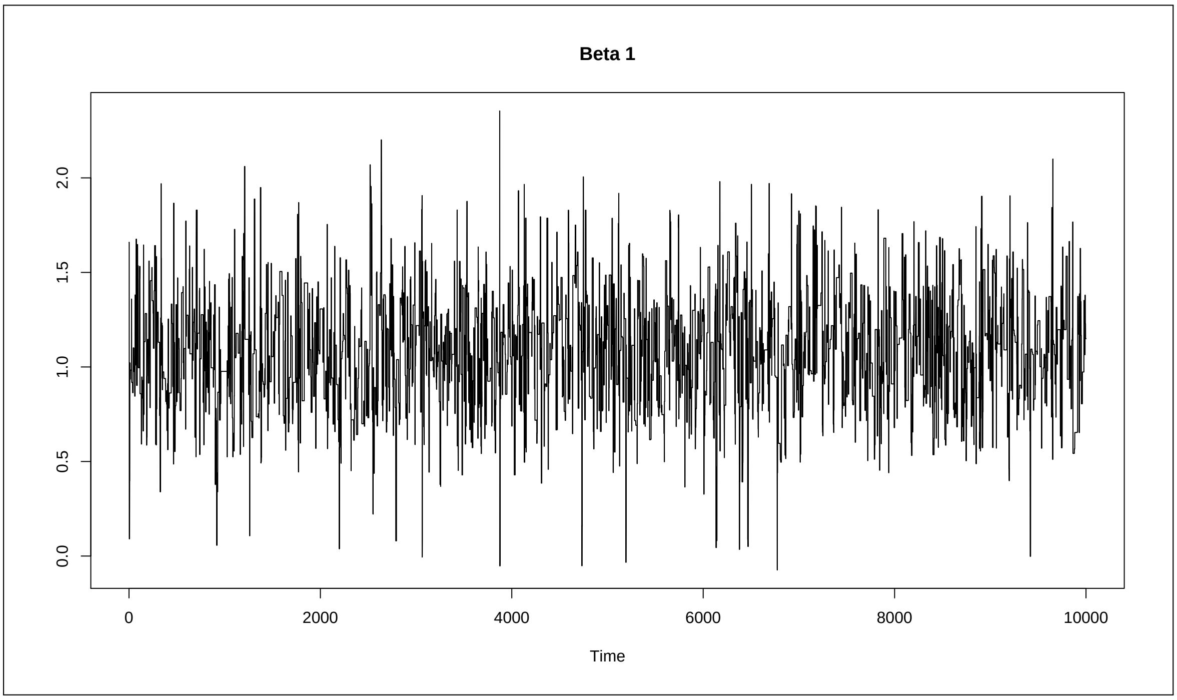
Appendix A

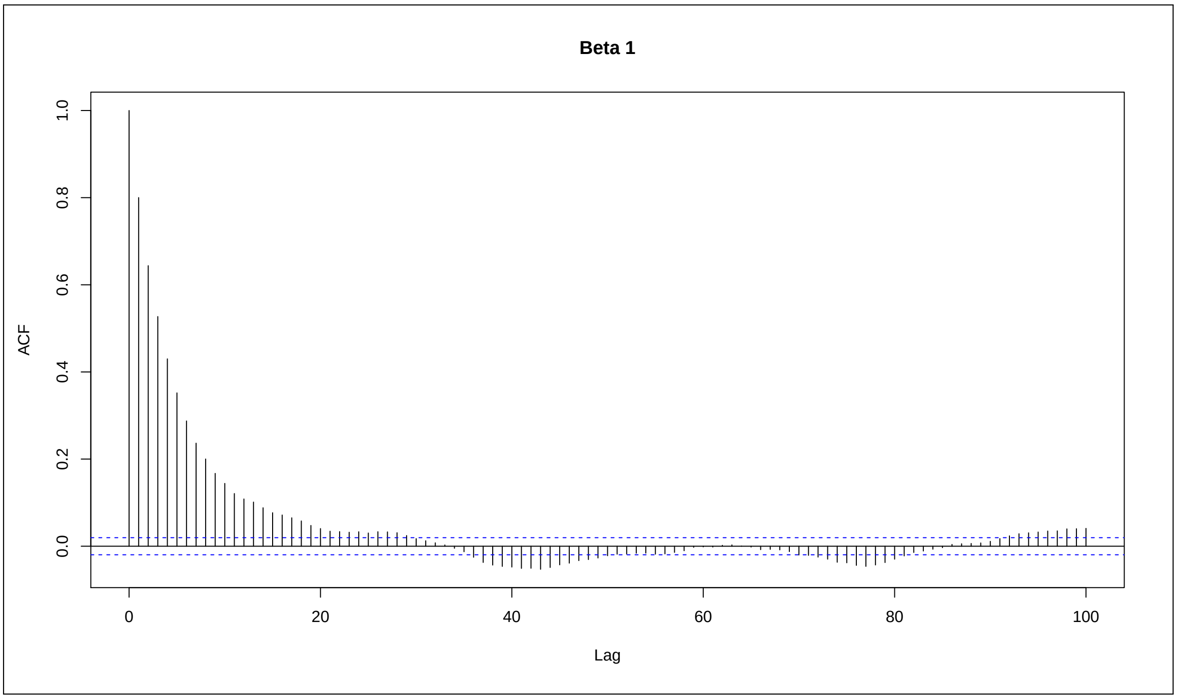
Appendix

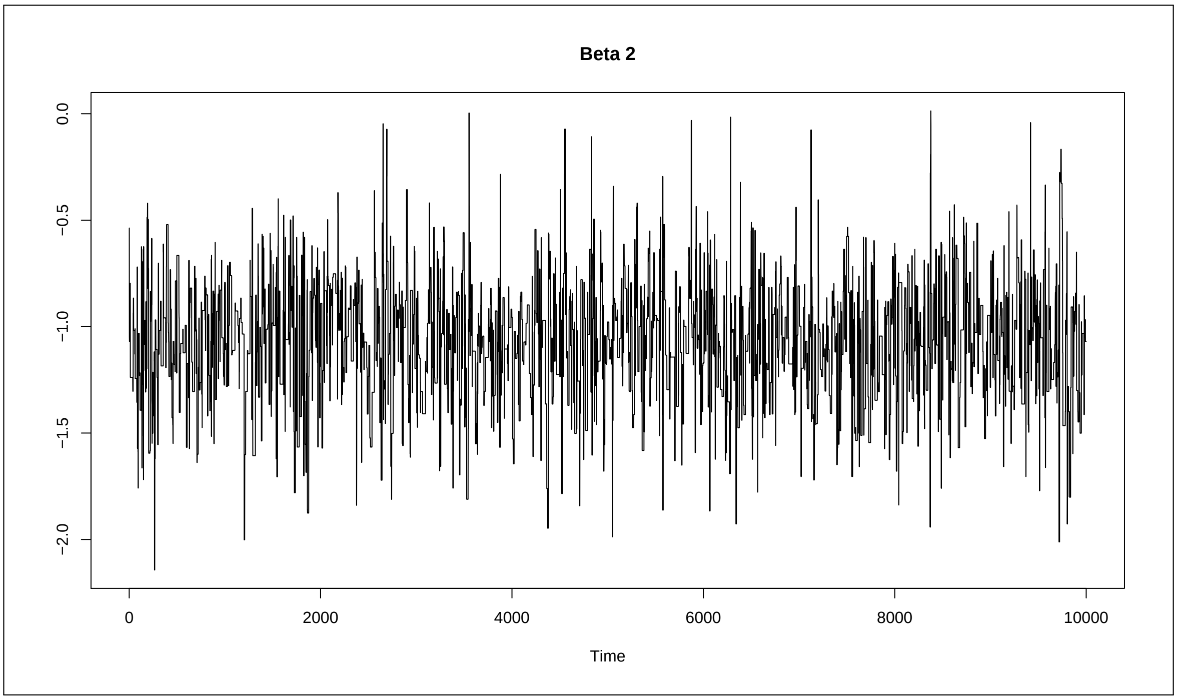
A.1 Single Componenent updating with Independence sampler

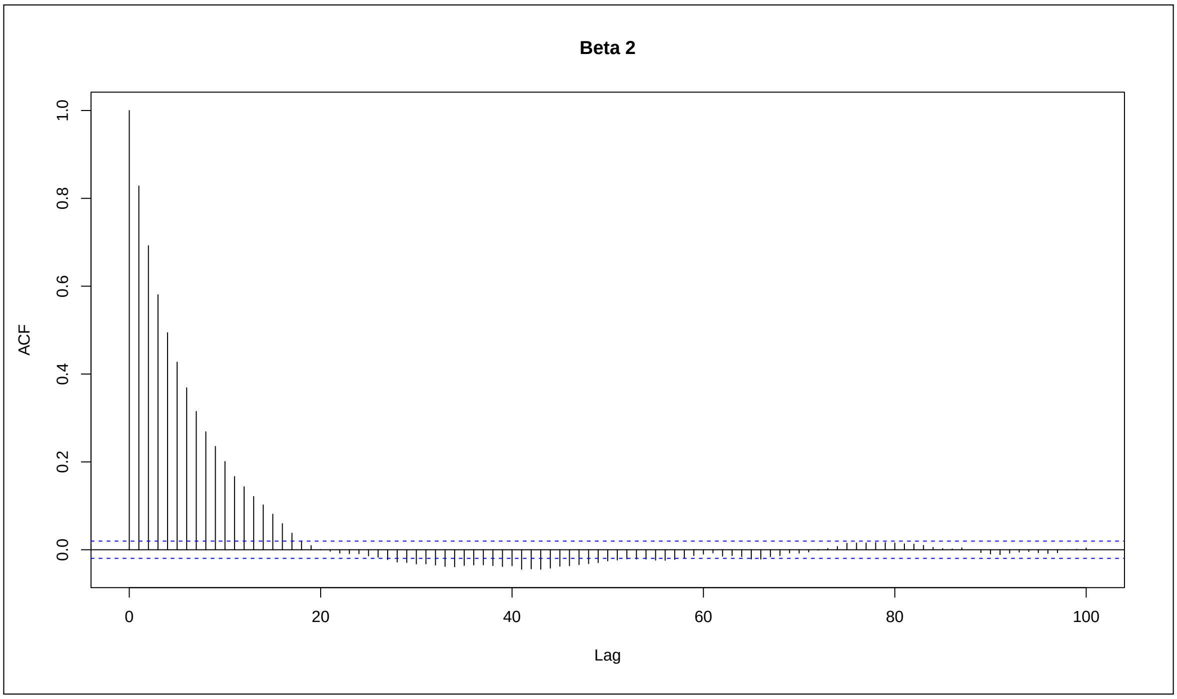
Parameter	G-R Point est.	G-R Upper C.I.	Acceptance rate
β_1	1.0007	1.0017	0.1203
β_2	1.0017	1.0038	0.1166
β_3	1.0036	1.0051	0.0493
β_4	1.0096	1.0111	0.0458
β_5	1.0071	1.0085	0.0584
β_6	1.0034	1.0046	0.0515
β_7	1.0067	1.0090	0.0493
β_8	1.0070	1.0084	0.0535
β_9	1.0063	1.0109	0.0485
β_{10}	1.0022	1.0040	0.0616

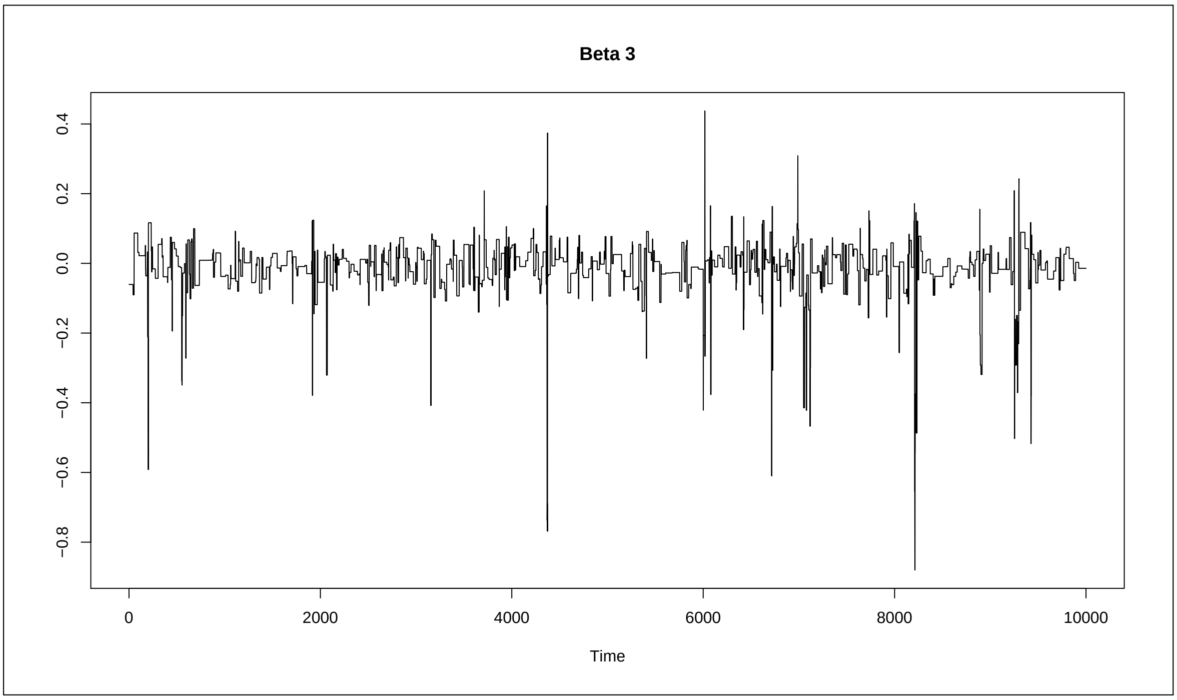
Table A.1: Point estimates of potential scale reduction factor, corresponding upper confidence limits and acceptance rates for the ten logistic slope parameters

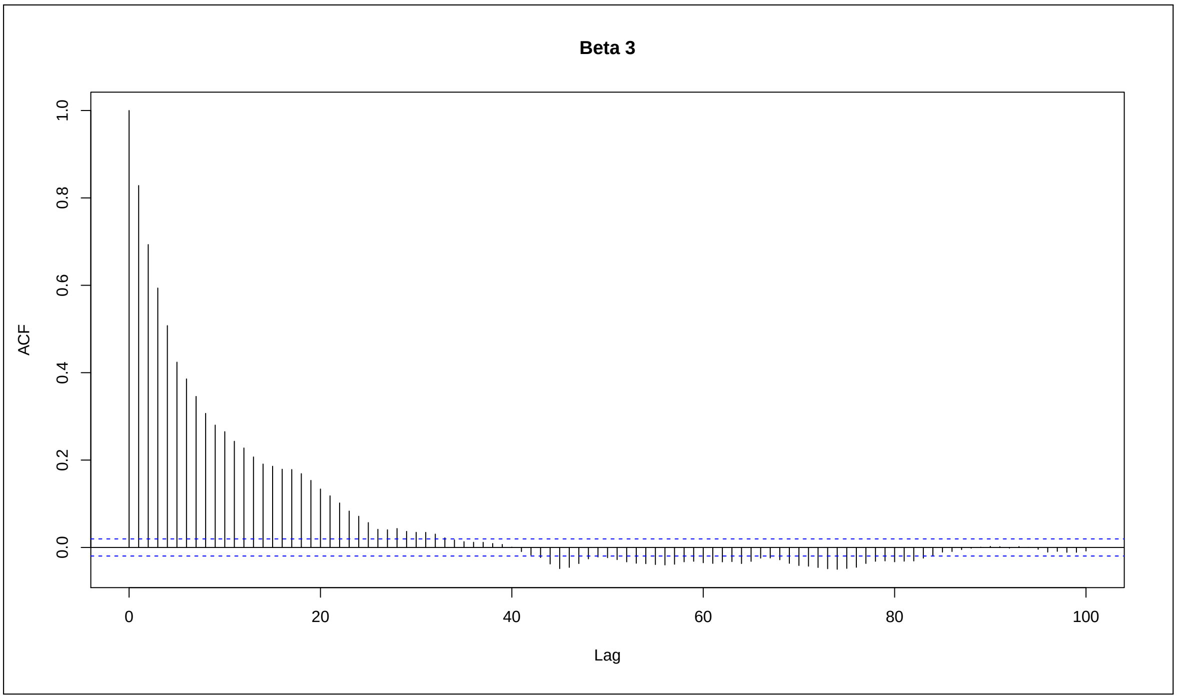

Beta 1
2.0
1.5
1.0
0.5
0.0
0
2000
4000
6000
8000
10000
Time

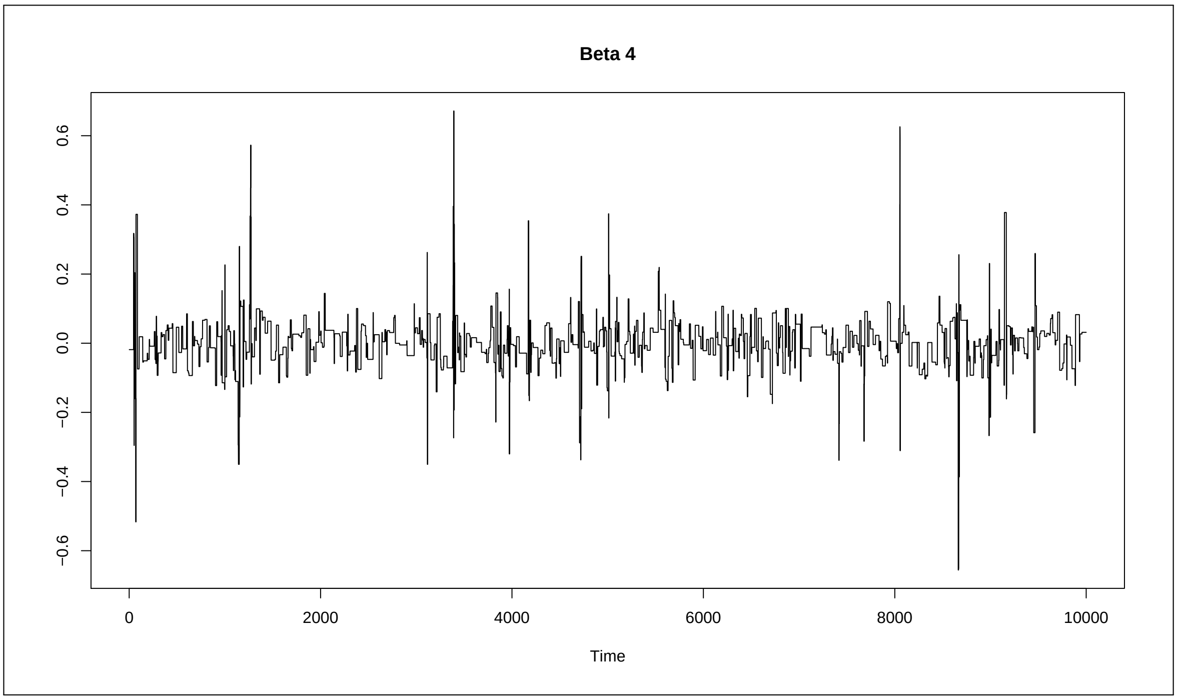

Beta 1
ACF
1.0
0.8
0.6
0.4
0.2
0.0
0
20
40
60
80
100
Lag

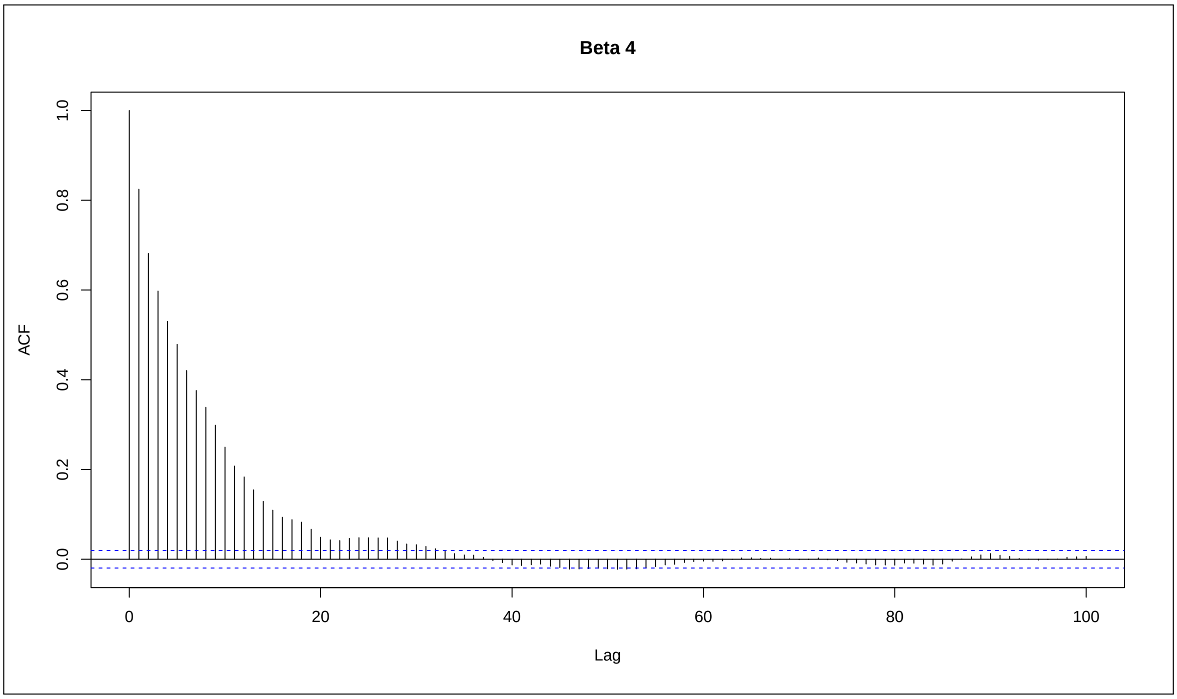

Beta 2
0.0
−0.5
−1.0
−1.5
−2.0
0
2000
4000
6000
8000
10000
Time

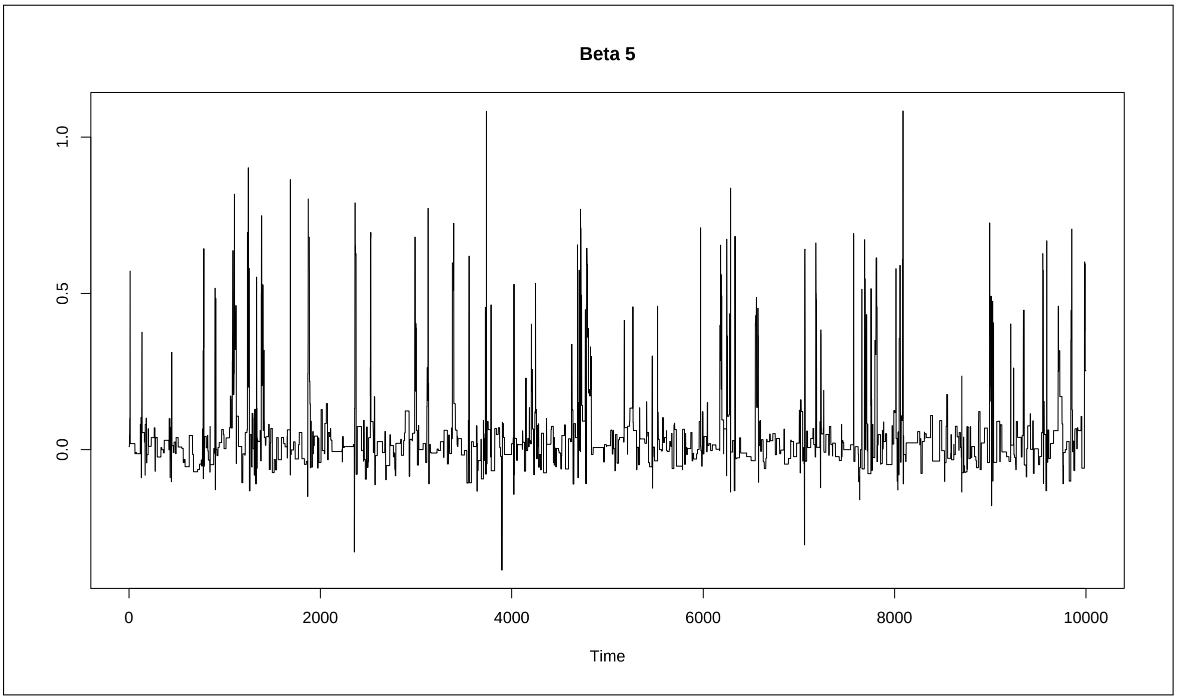

Beta 2
ACF
1.0
0.8
0.6
0.4
0.2
0.0
0
20
40
60
80
100
Lag

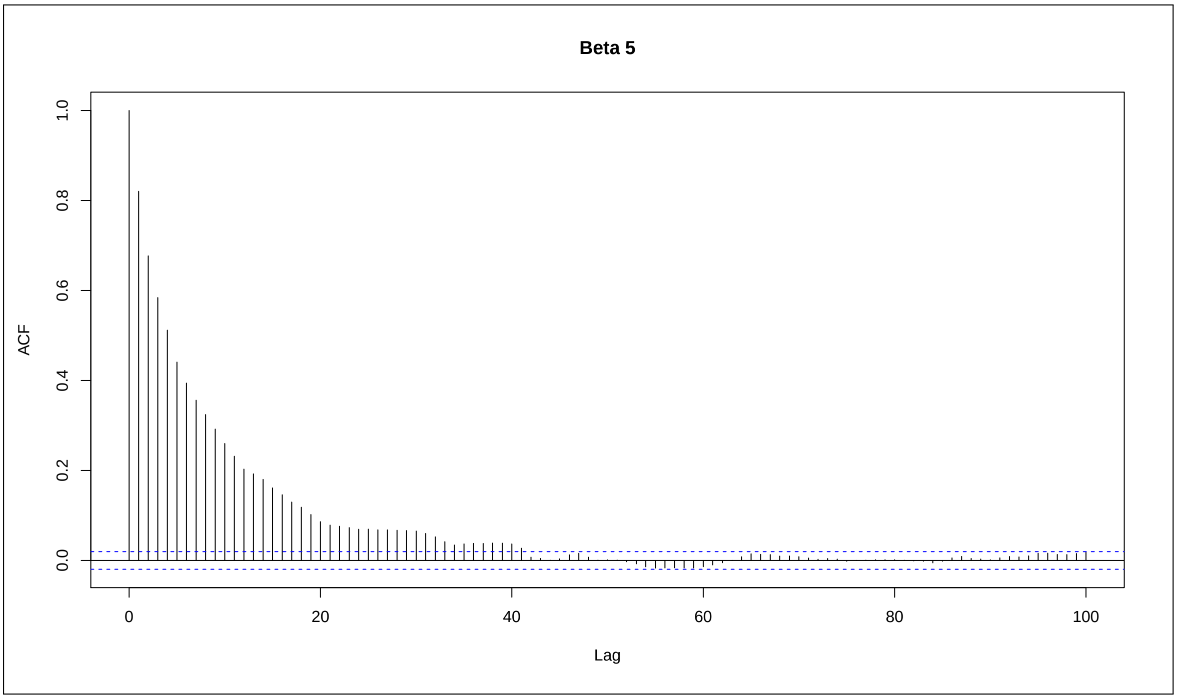

Beta 3
0.4
0.2
0.0
−0.2
−0.4
−0.6
−0.8
0
2000
4000
6000
8000
10000
Time

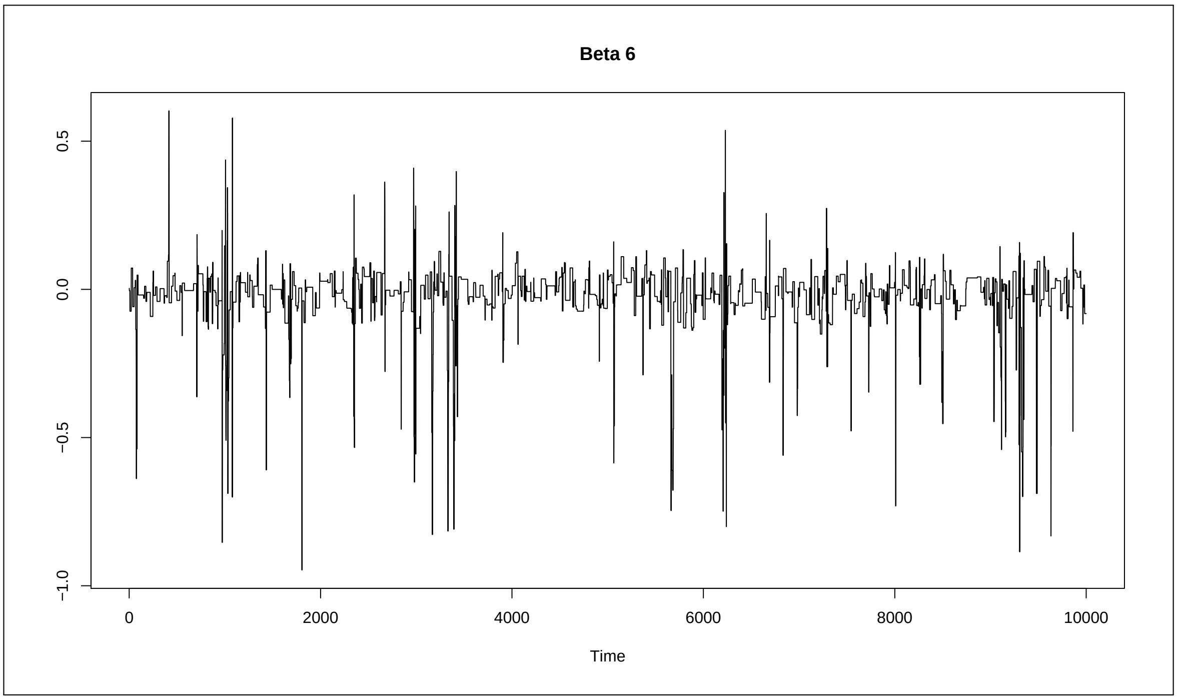

Beta 3
ACF
1.0
0.8
0.6
0.4
0.2
0.0
0
20
40
60
80
100
Lag


Beta 4
0.6
0.4
0.2
0.0
−0.2
−0.4
−0.6
0
2000
4000
6000
8000
10000
Time

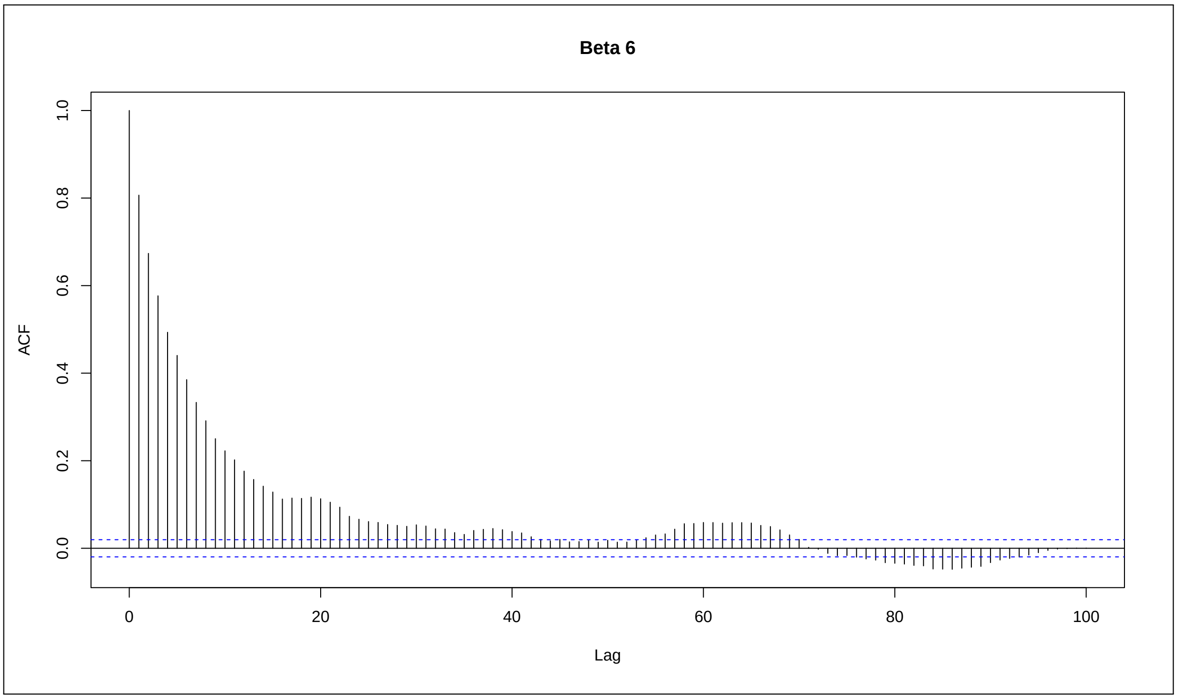

Beta 4
ACF
1.0
0.8
0.6
0.4
0.2
0.0
0
20
40
60
80
100
Lag


Beta 5
1.0
0.5
0.0
0
2000
4000
6000
8000
10000
Time

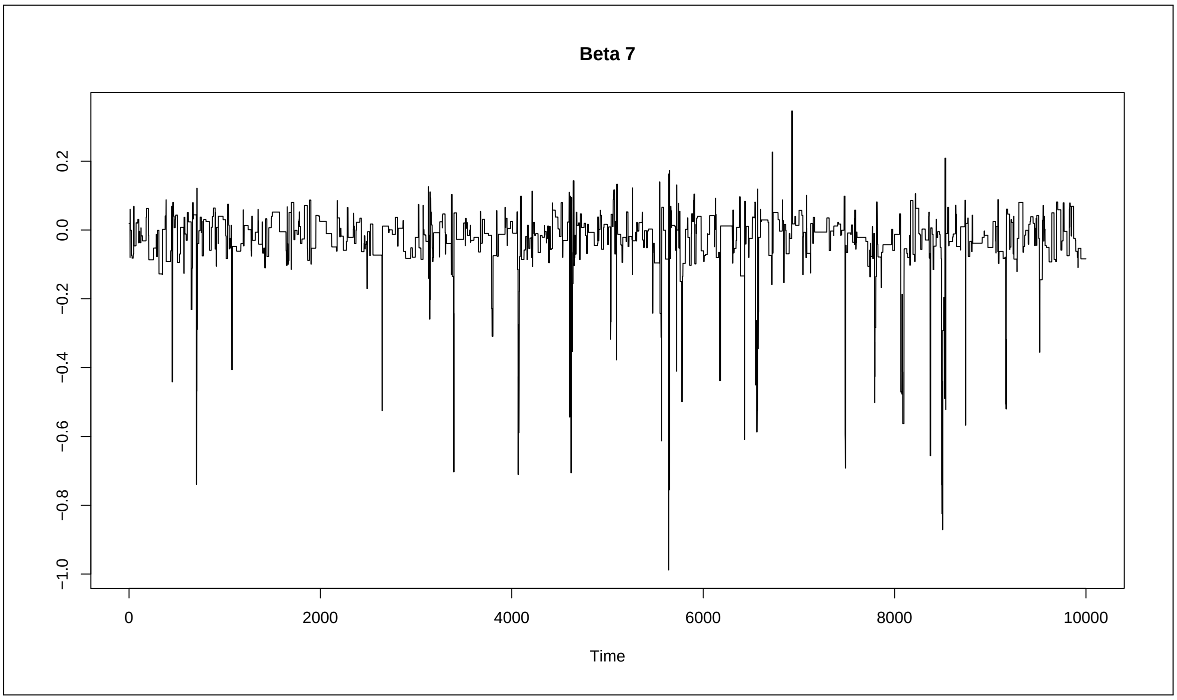

Beta 5
ACF
1.0
0.8
0.6
0.4
0.2
0.0
0
20
40
60
80
100
Lag

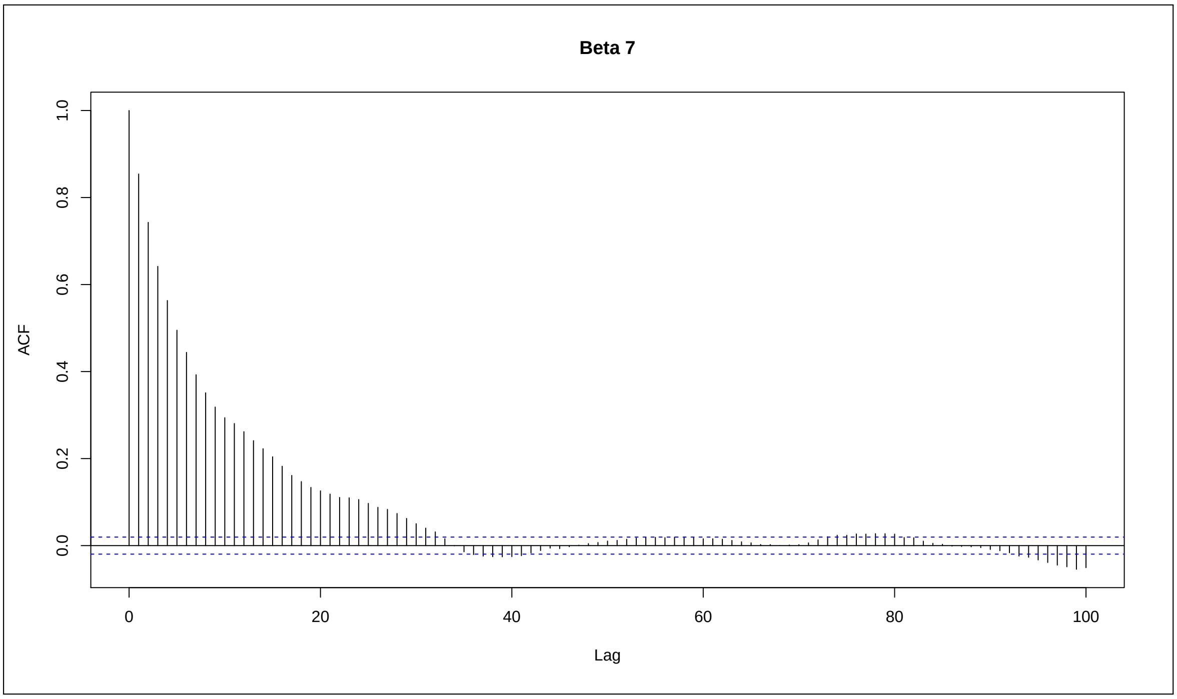


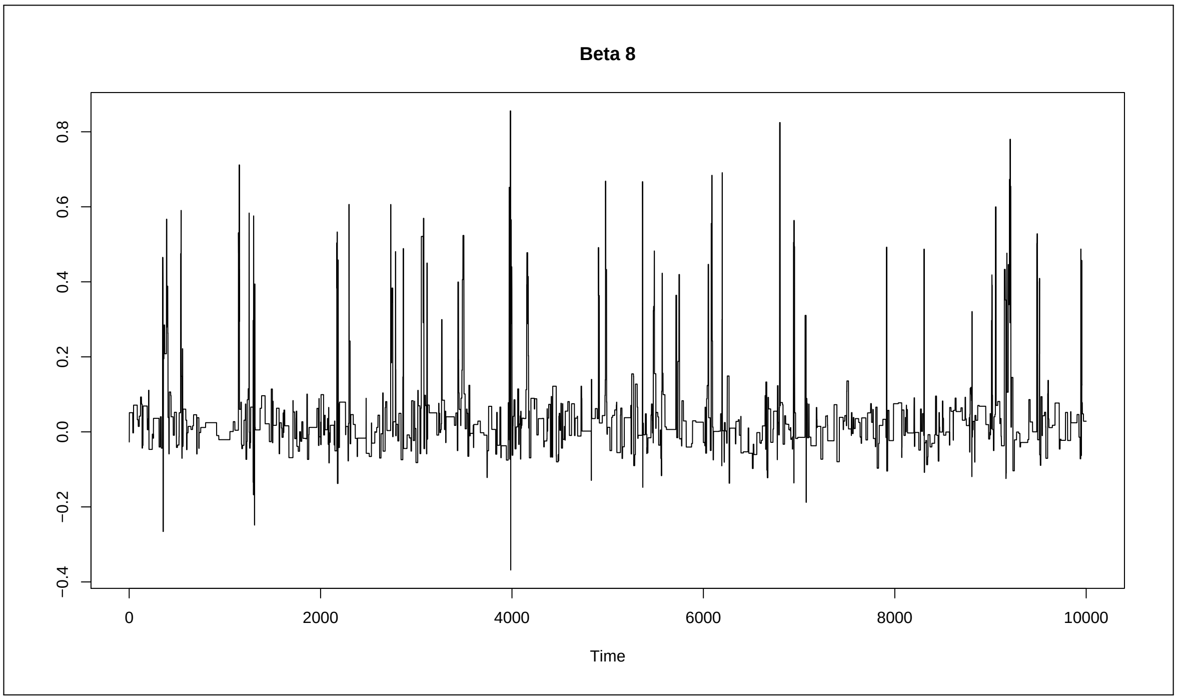
Beta 6
0.5
0.0
−0.5
−1.0
0
2000
4000
6000
8000
10000
Time

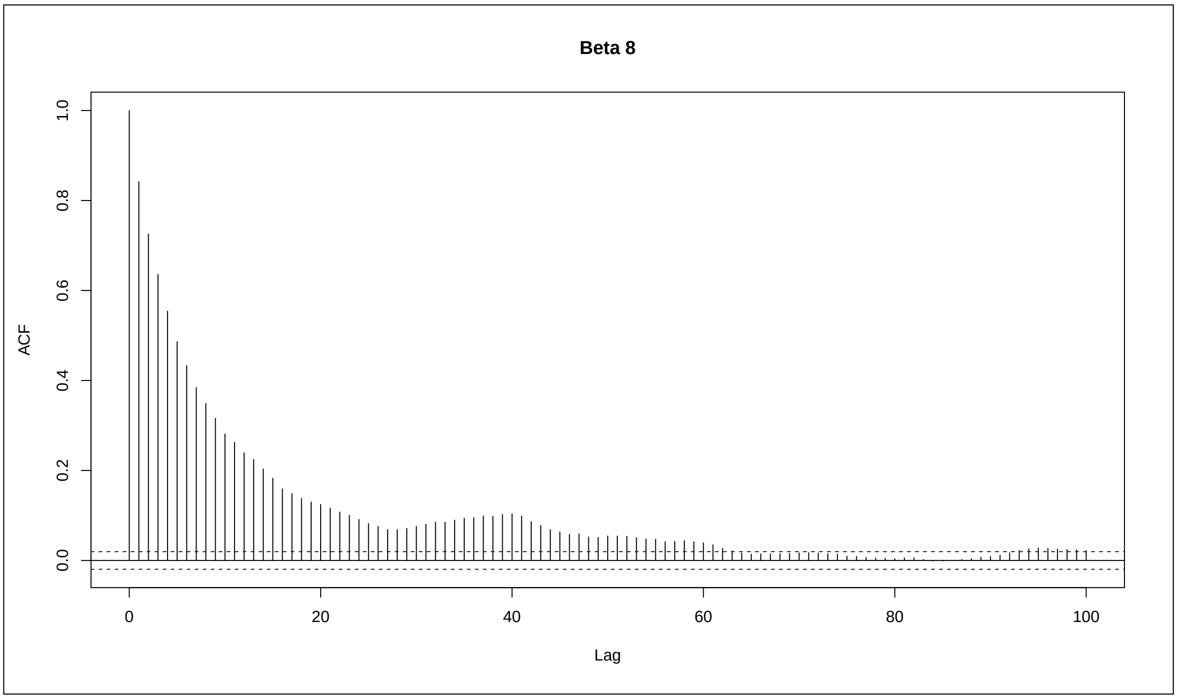


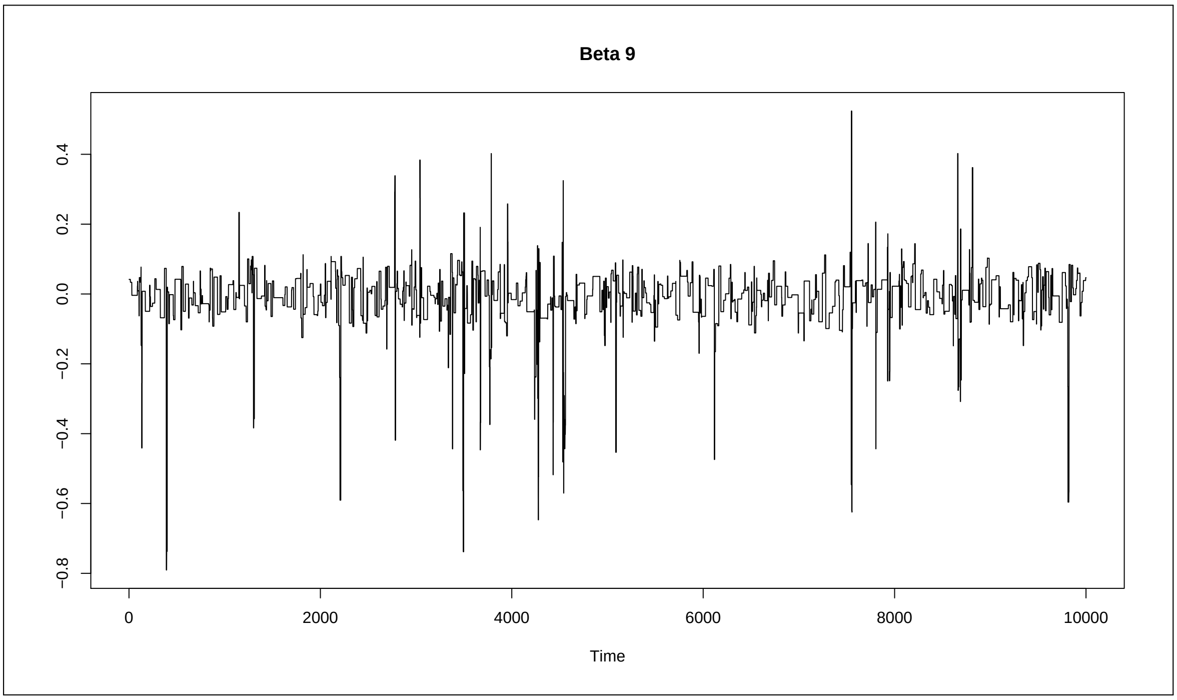
Beta 6
ACF
1.0
0.8
0.6
0.4
0.2
0.0
0
20
40
60
80
100
Lag

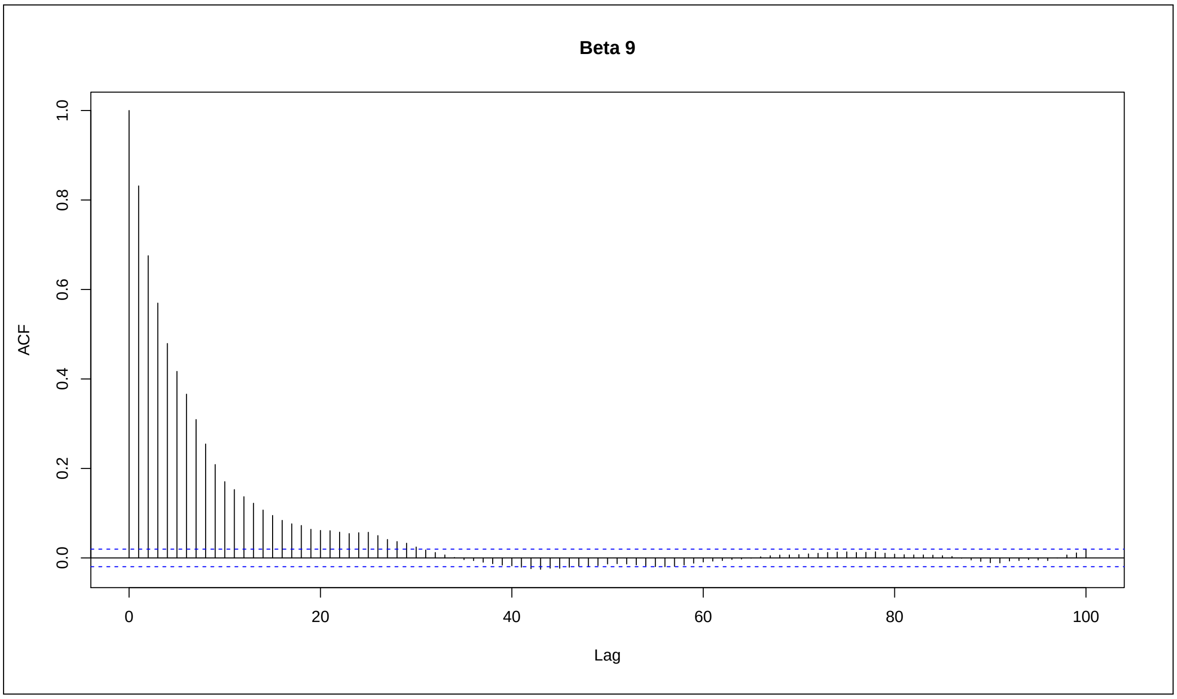

Beta 7
0.2
0.0
−0.2
−0.4
−0.6
−0.8
−1.0
0
2000
4000
6000
8000
10000
Time

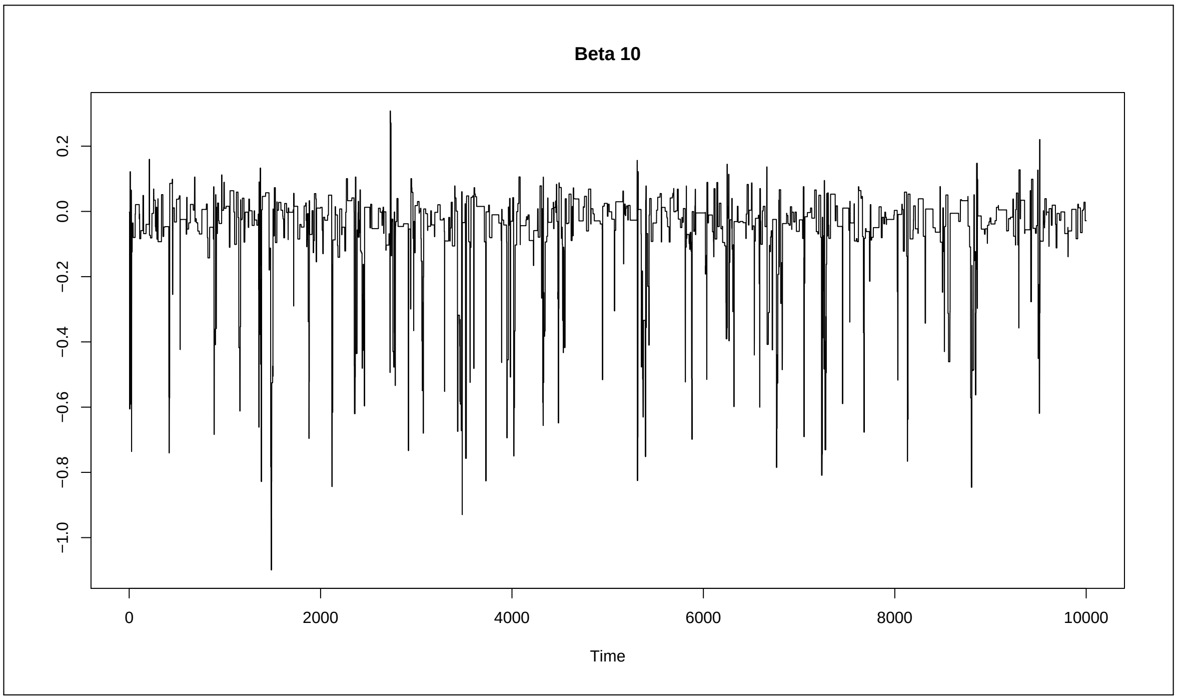

Beta 7
ACF
1.0
0.8
0.6
0.4
0.2
0.0
0
20
40
60
80
100
Lag

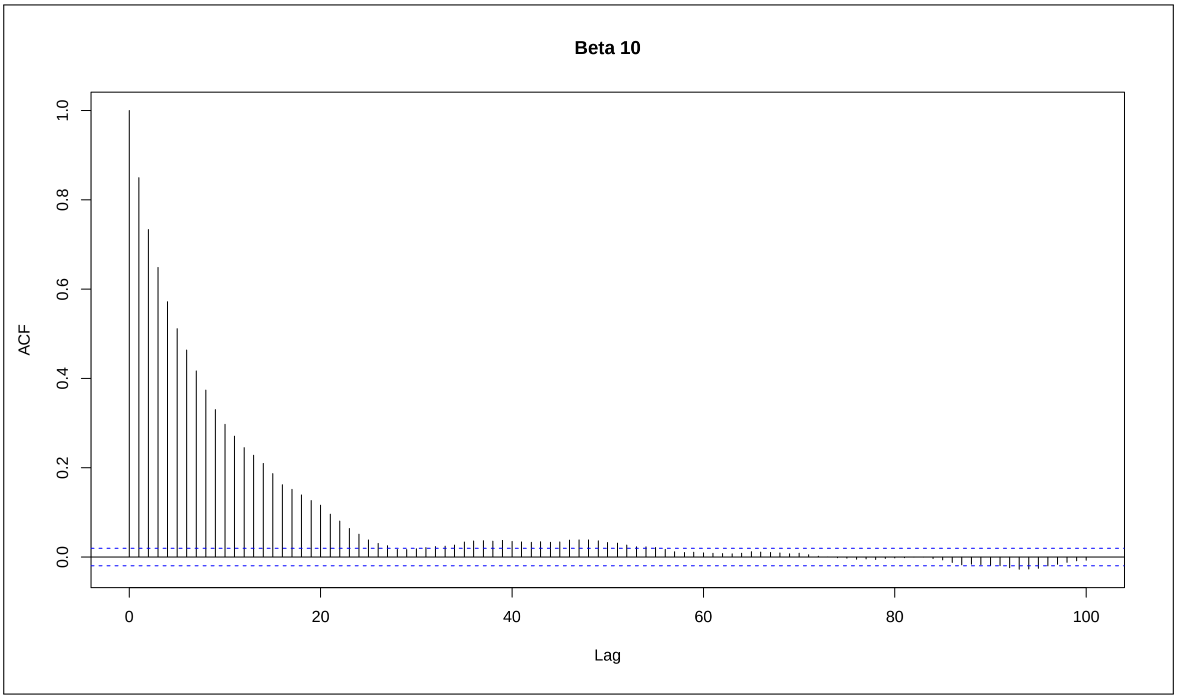

Beta 8
0.8
0.6
0.4
0.2
0.0
−0.2
−0.4
0
2000
4000
6000
8000
10000
Time

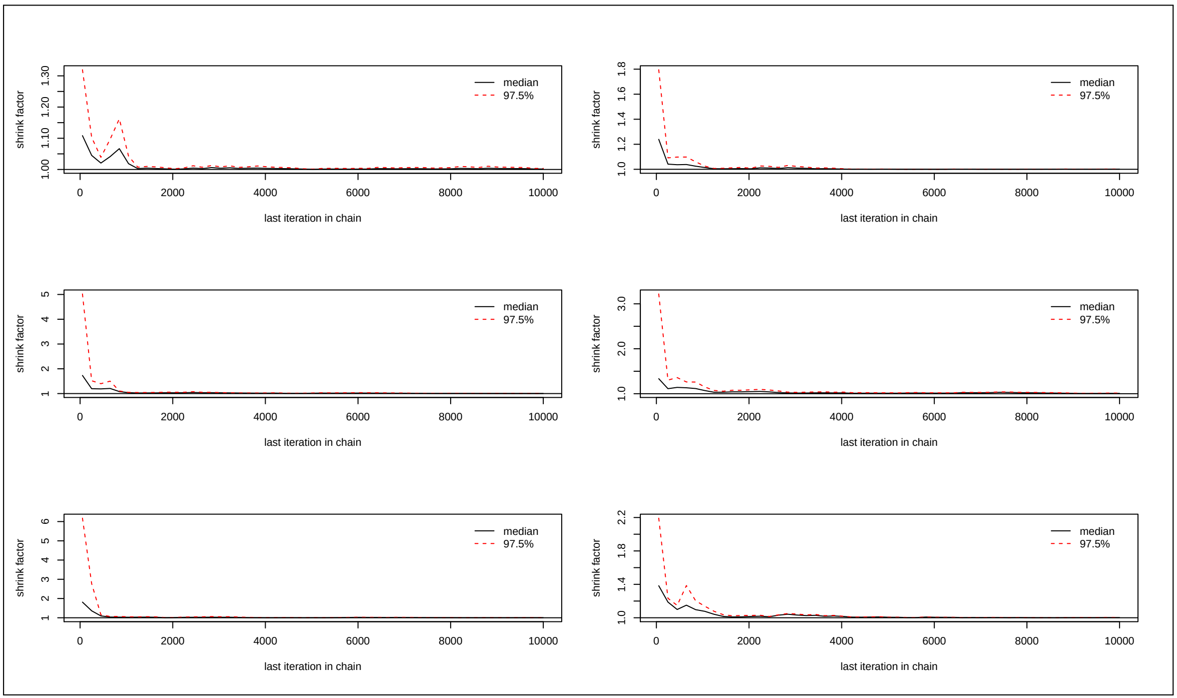

Beta 8
ACF
1.0
0.8
0.6
0.4
0.2
0.0
0
20
40
60
80
100
Lag

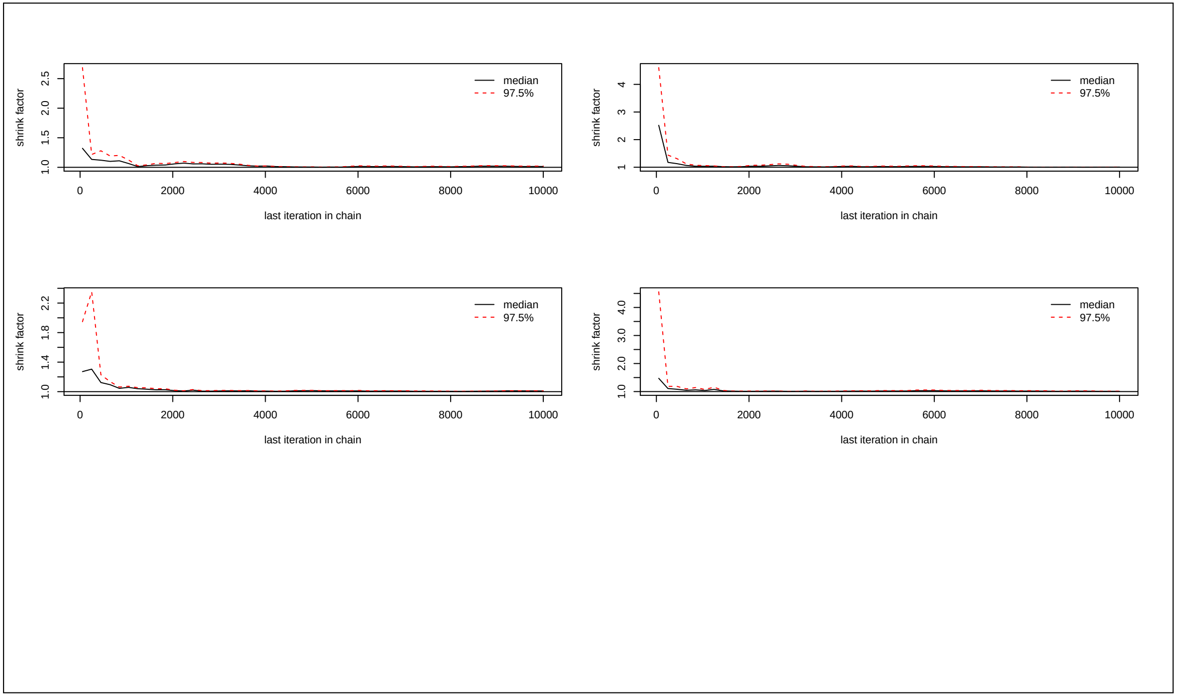

Beta 9
0.4
0.2
0.0
−0.2
−0.4
−0.6
−0.8
0
2000
4000
6000
8000
10000
Time


Beta 9
ACF
1.0
0.8
0.6
0.4
0.2
0.0
0
20
40
60
80
100
Lag


Beta 10
0.2
0.0
−0.2
−0.4
−0.6
−0.8
−1.0
0
2000
4000
6000
8000
10000
Time


Beta 10
ACF
1.0
0.8
0.6
0.4
0.2
0.0
0
20
40
60
80
100
Lag

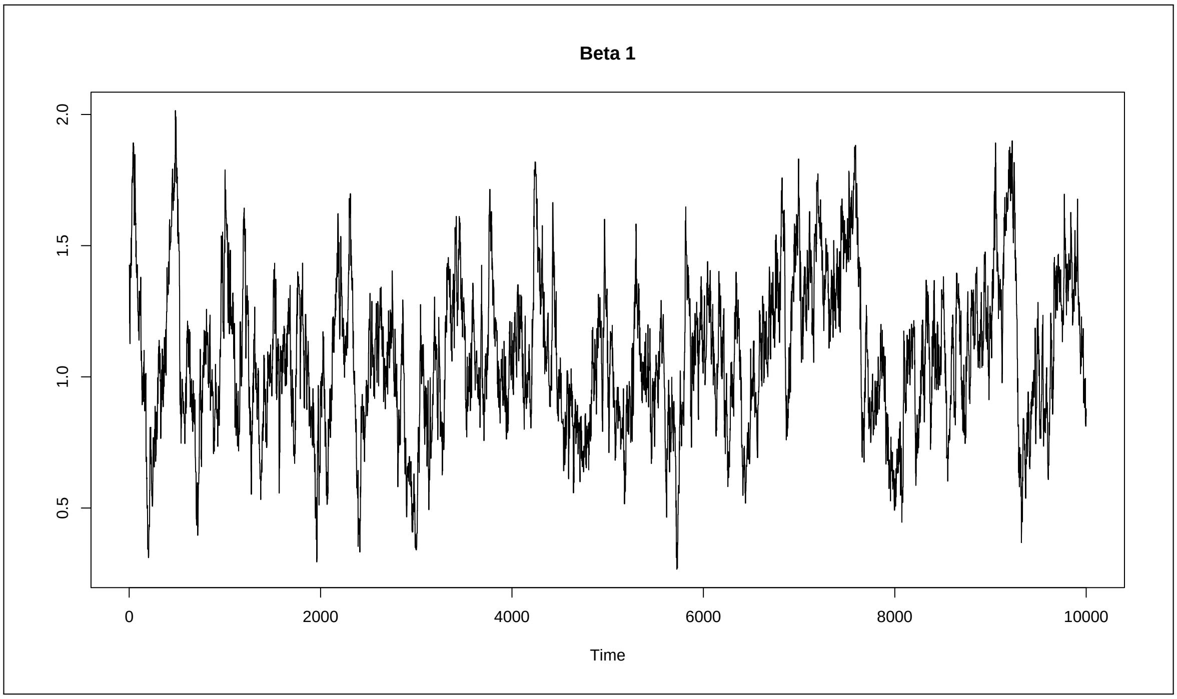


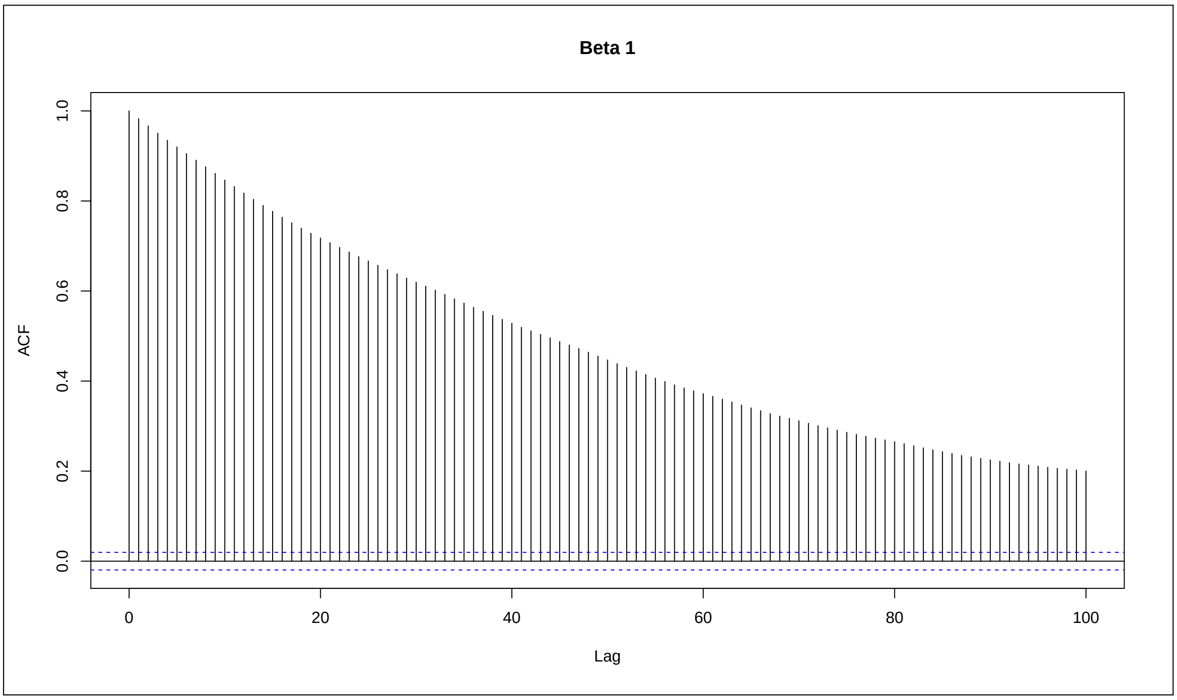


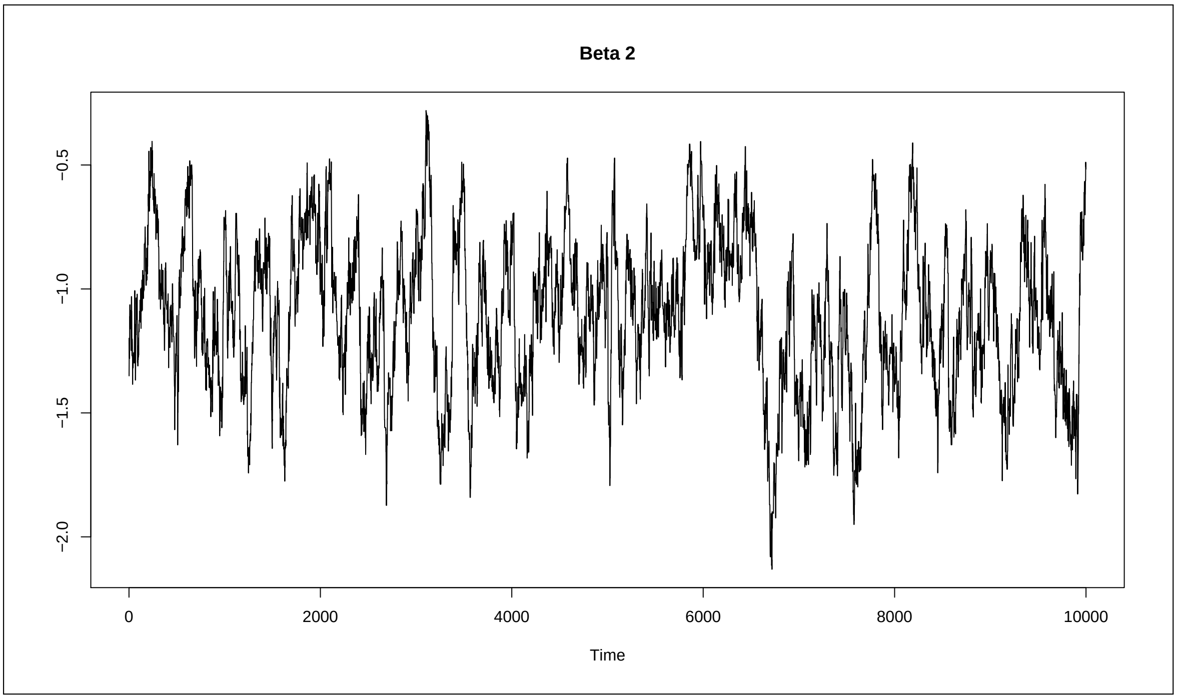
A.2 Single component updating random walk

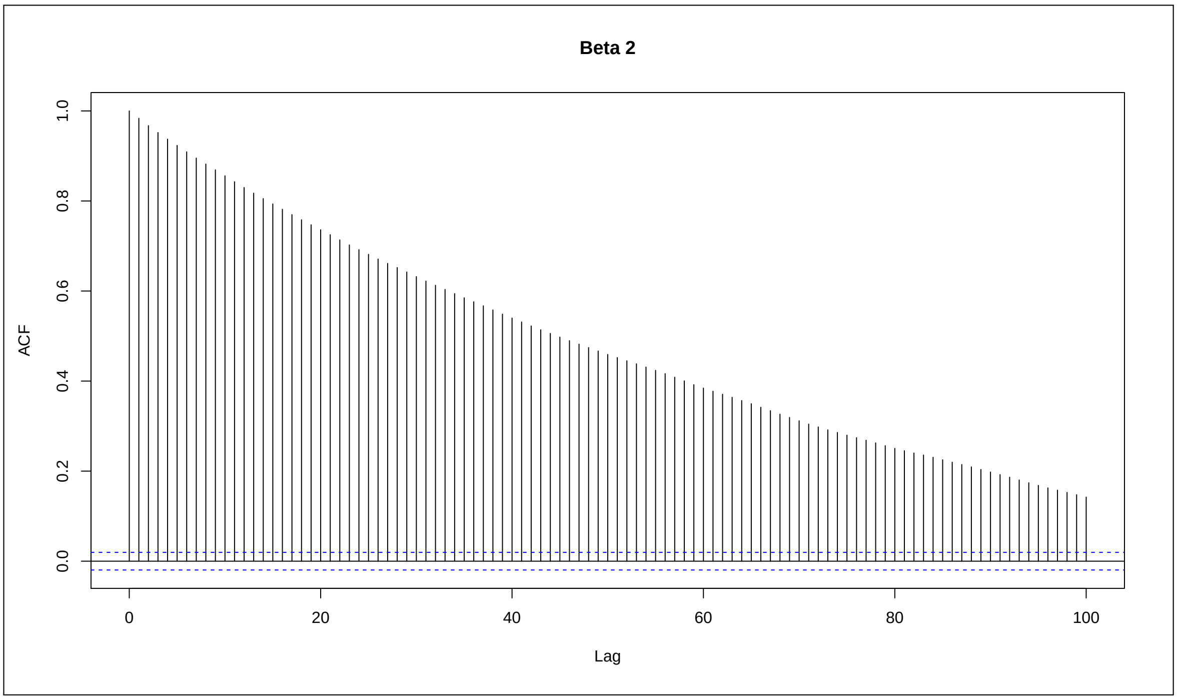
Parameter	G-R Point est.	G-R Upper C.I.	Acceptance rate
β_1	1.0057	1.0120	0.5258
β_2	1.0144	1.0347	0.5197
β_3	1.0857	1.1353	0.3678
β_4	1.0203	1.0319	0.3663
β_5	1.0867	1.1518	0.3810
β_6	1.0494	1.0767	0.3740
β_7	1.0875	1.1607	0.3726
β_8	1.0166	1.0272	0.3722
β_9	1.0156	1.0221	0.3711
β_{10}	1.0468	1.0981	0.3808

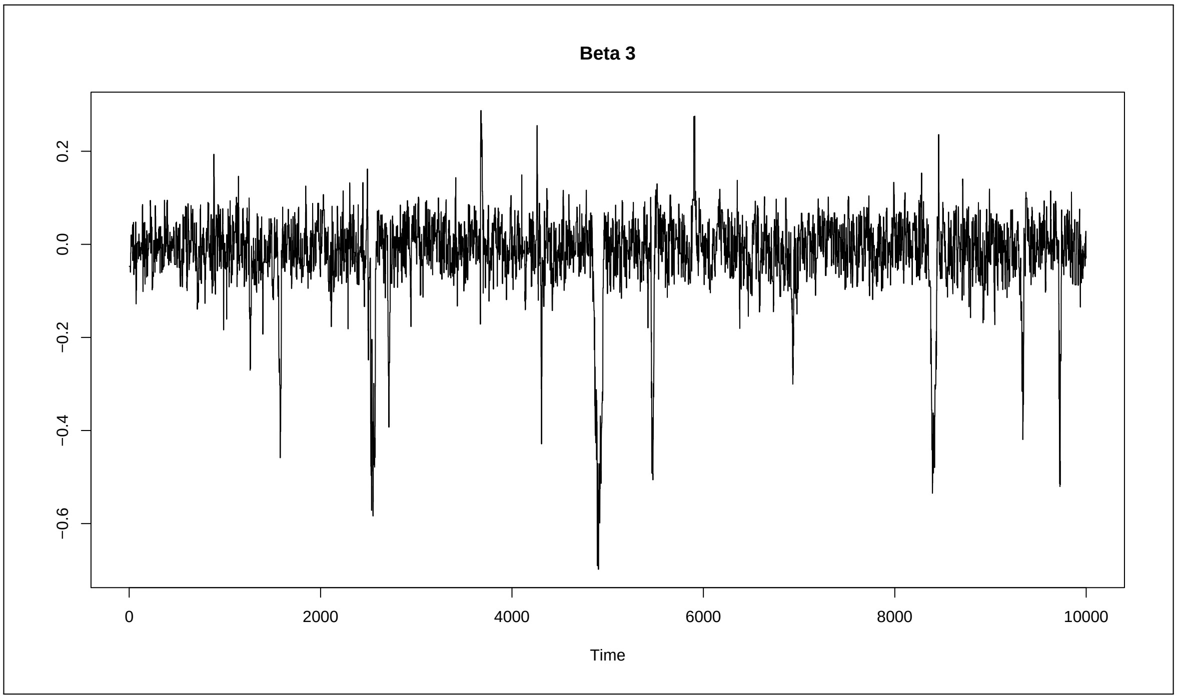
Table A.2: Point estimates of potential scale reduction factor, corresponding upper confidence limits and acceptance rates for the ten logistic slope parameters.

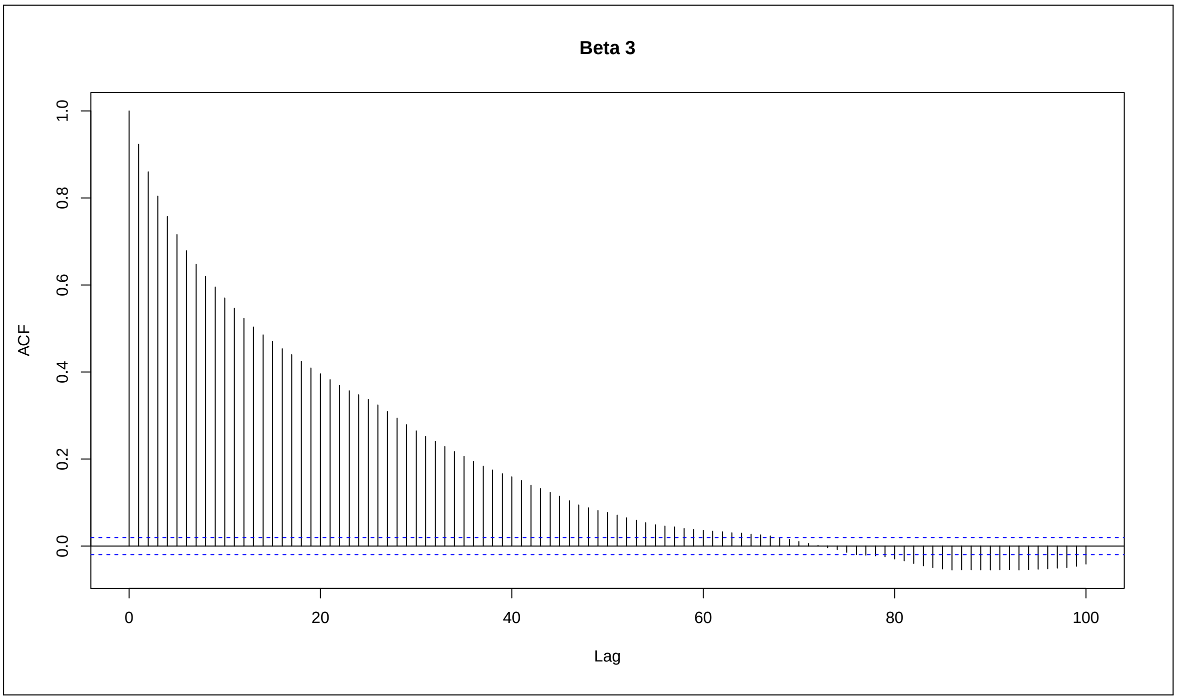


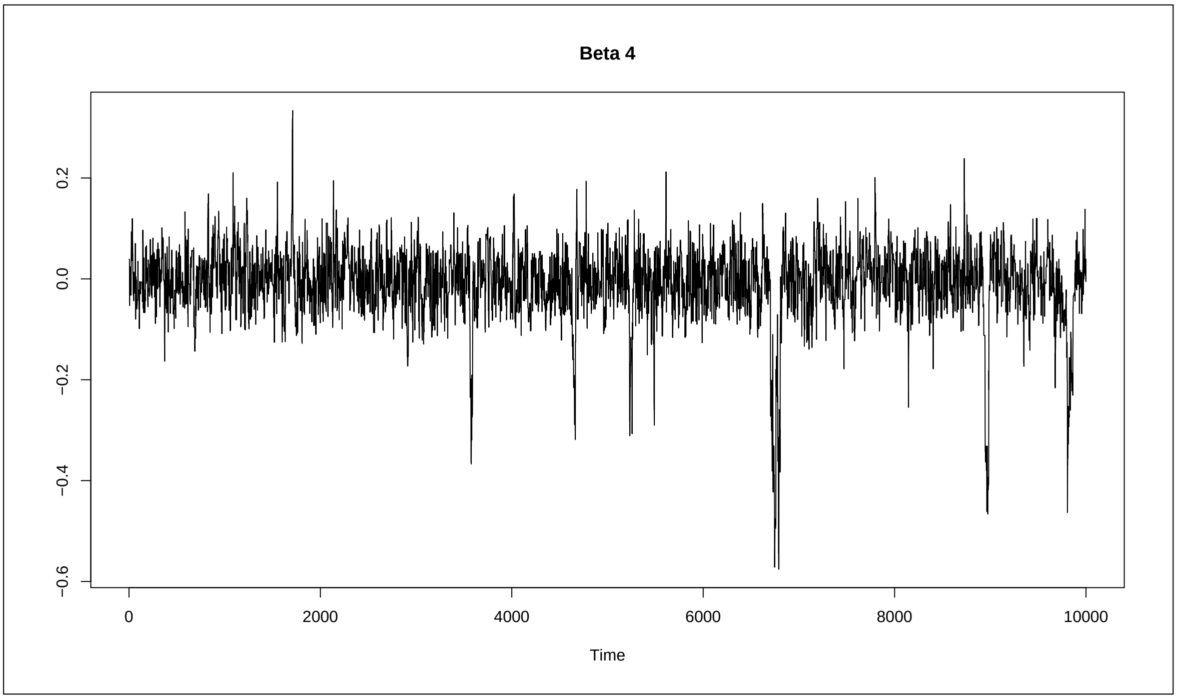

Beta 1
ACF
1.0
0.8
0.6
0.4
0.2
0.0
0
20
40
60
80
100
Lag

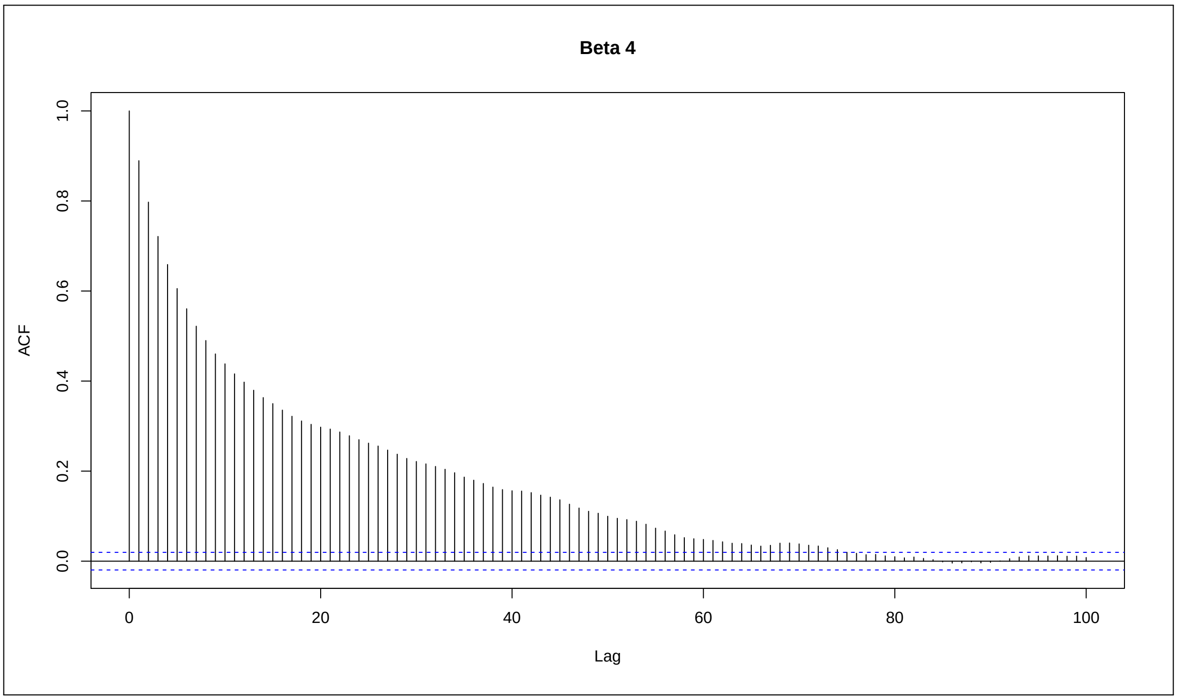

Beta 2
−0.5
−1.0
−1.5
−2.0
0
2000
4000
6000
8000
10000
Time

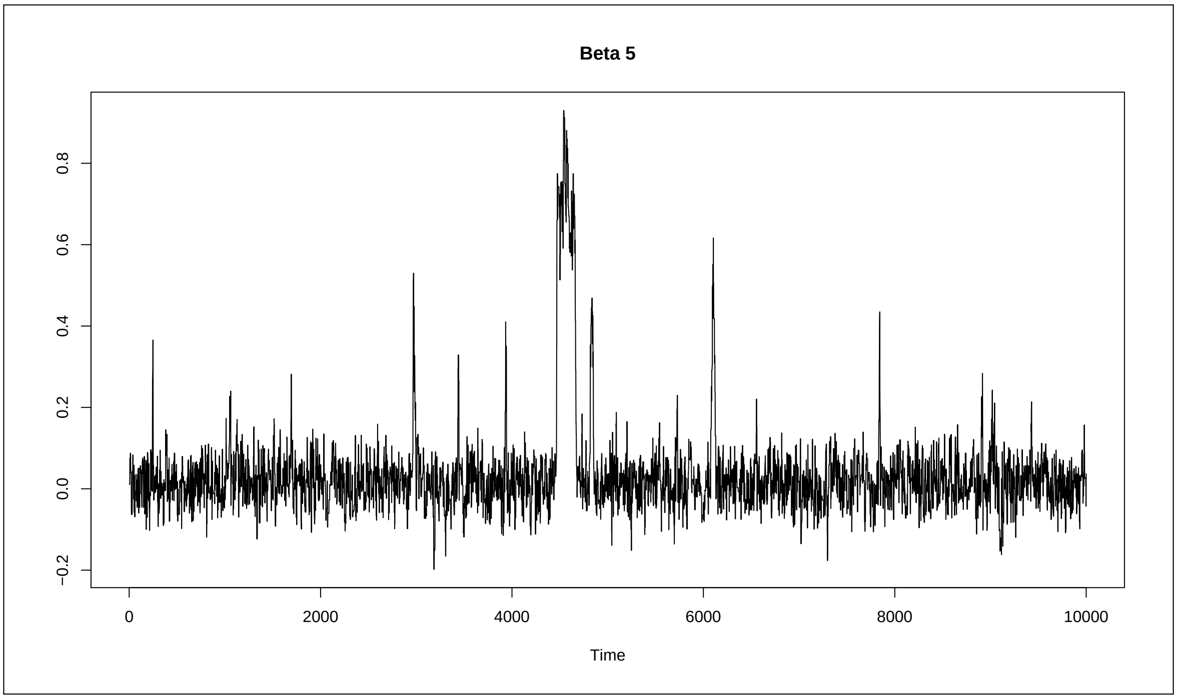

Beta 2
ACF
1.0
0.8
0.6
0.4
0.2
0.0
0
20
40
60
80
100
Lag

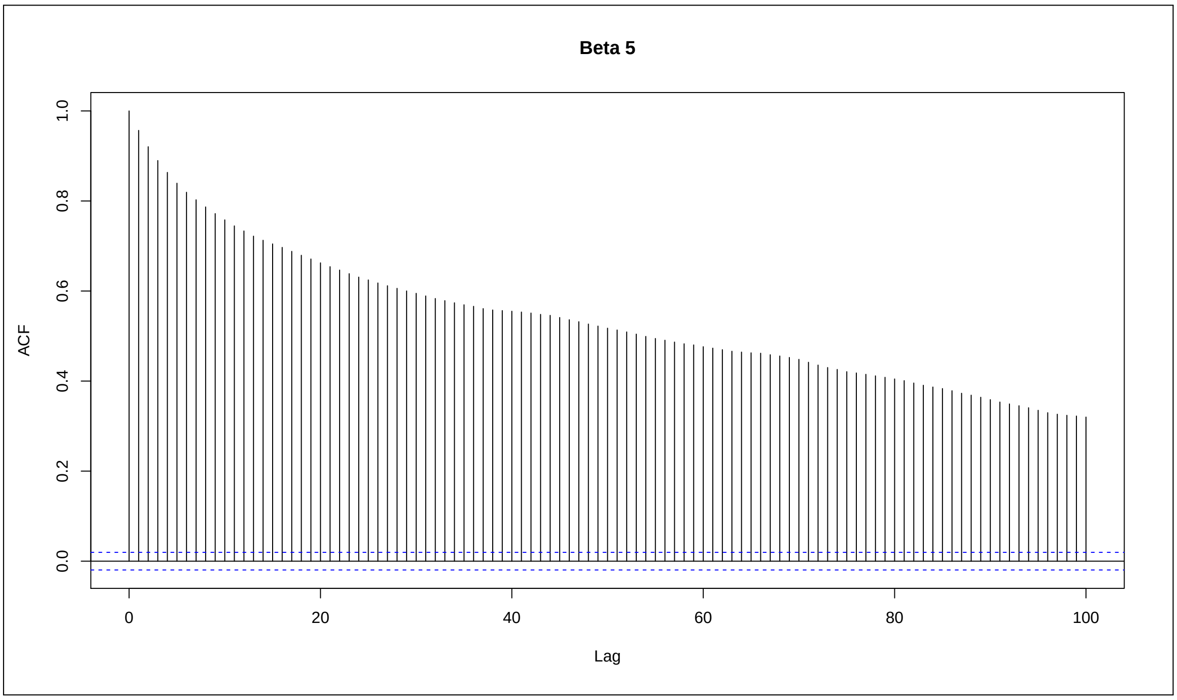

Beta 3
0.2
0.0
−0.2
−0.4
−0.6
0
2000
4000
6000
8000
10000
Time

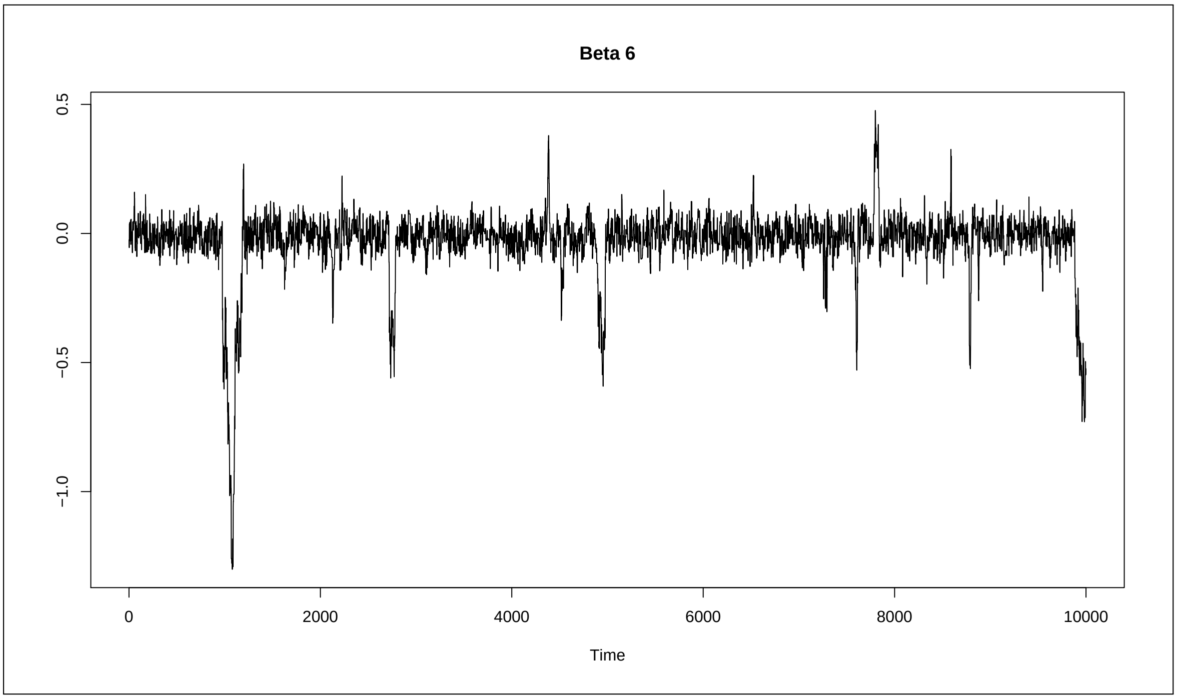

Beta 3
ACF
1.0
0.8
0.6
0.4
0.2
0.0
0
20
40
60
80
100
Lag

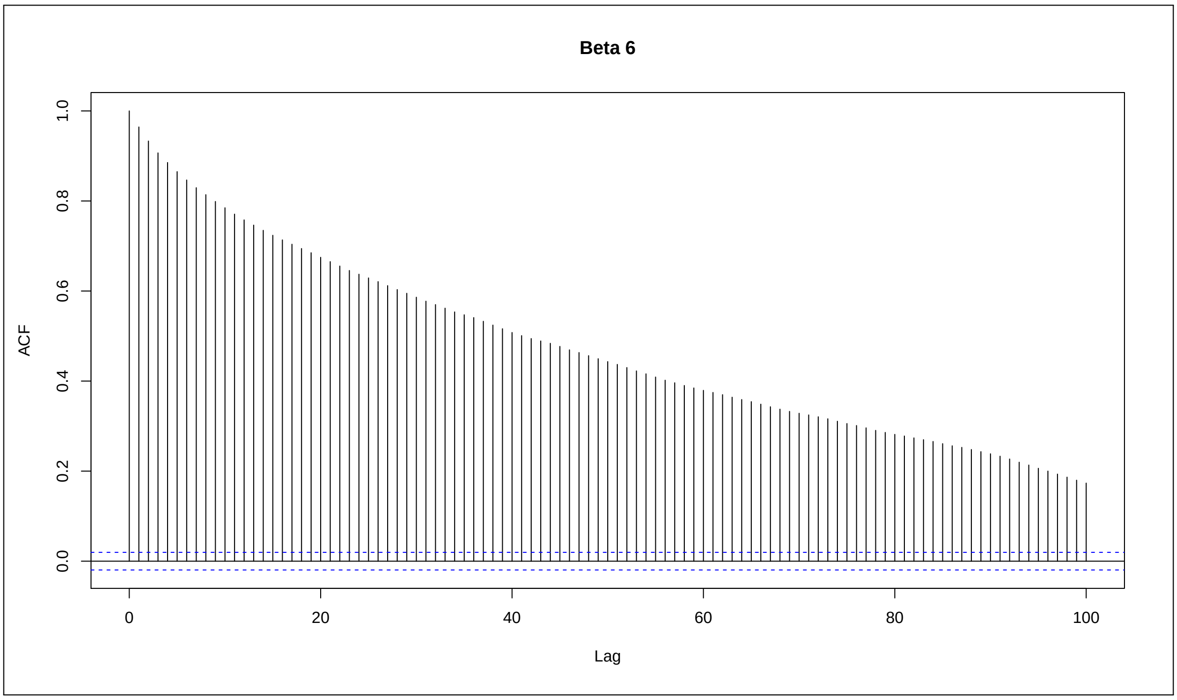

Beta 4
0.2
0.0
−0.2
−0.4
−0.6
0
2000
4000
6000
8000
10000
Time

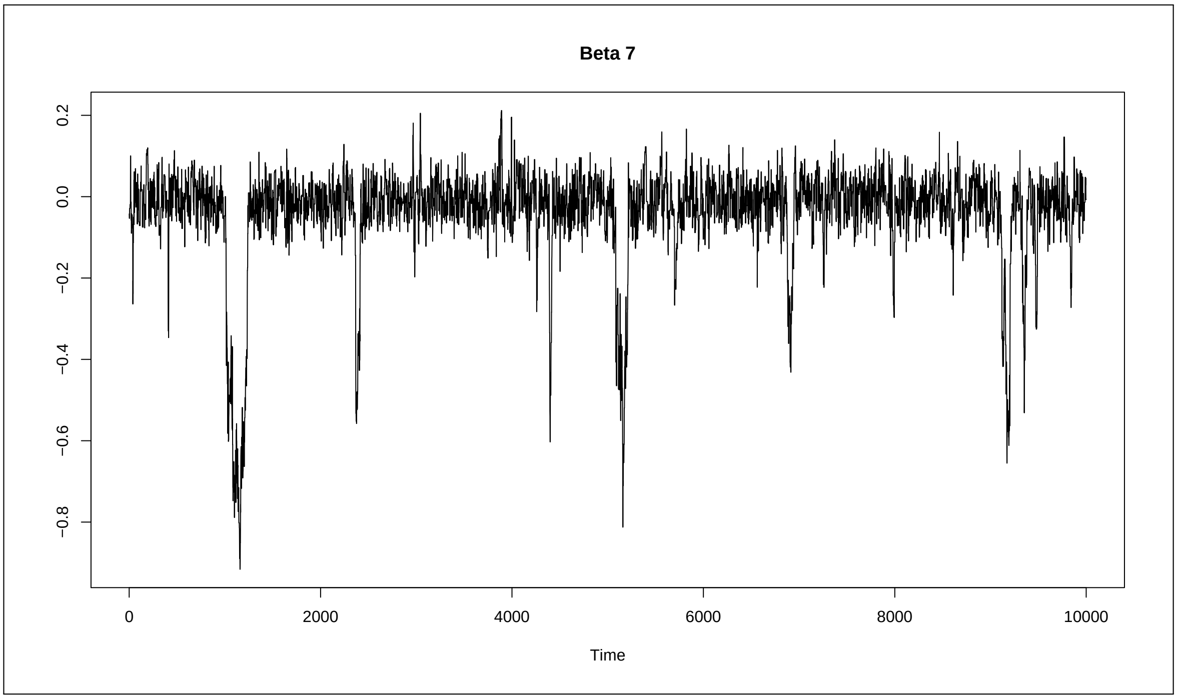

Beta 4
ACF
1.0
0.8
0.6
0.4
0.2
0.0
0
20
40
60
80
100
Lag

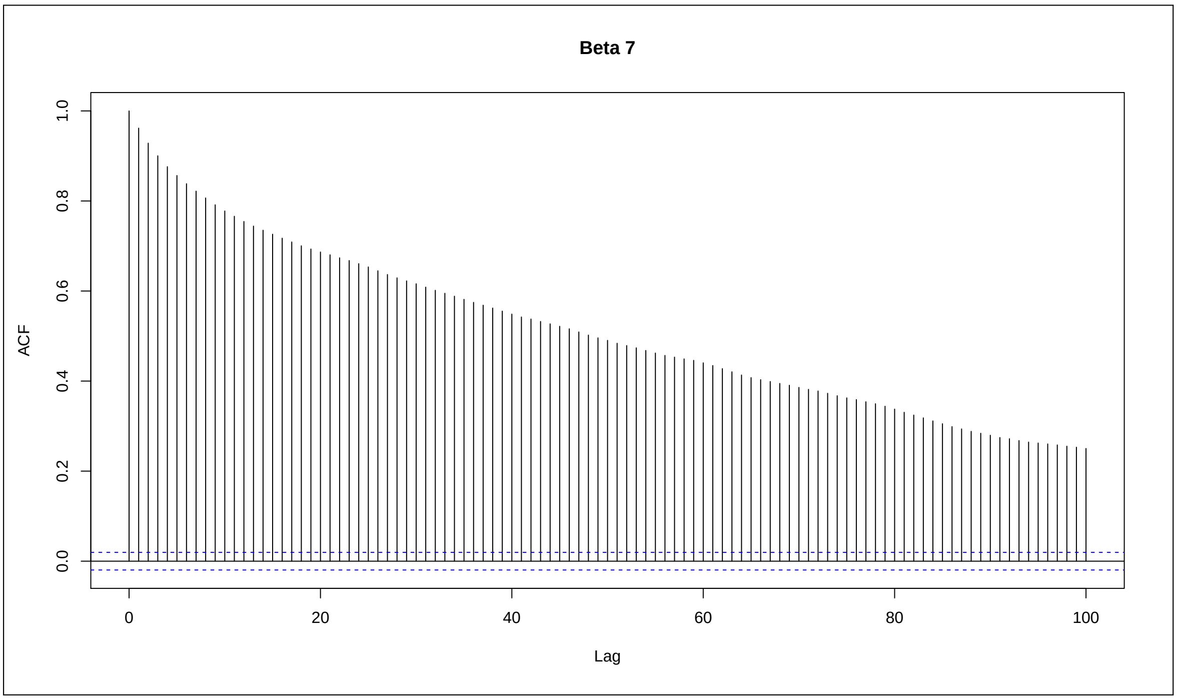

Beta 5
0.8
0.6
0.4
0.2
0.0
−0.2
0
2000
4000
6000
8000
10000
Time

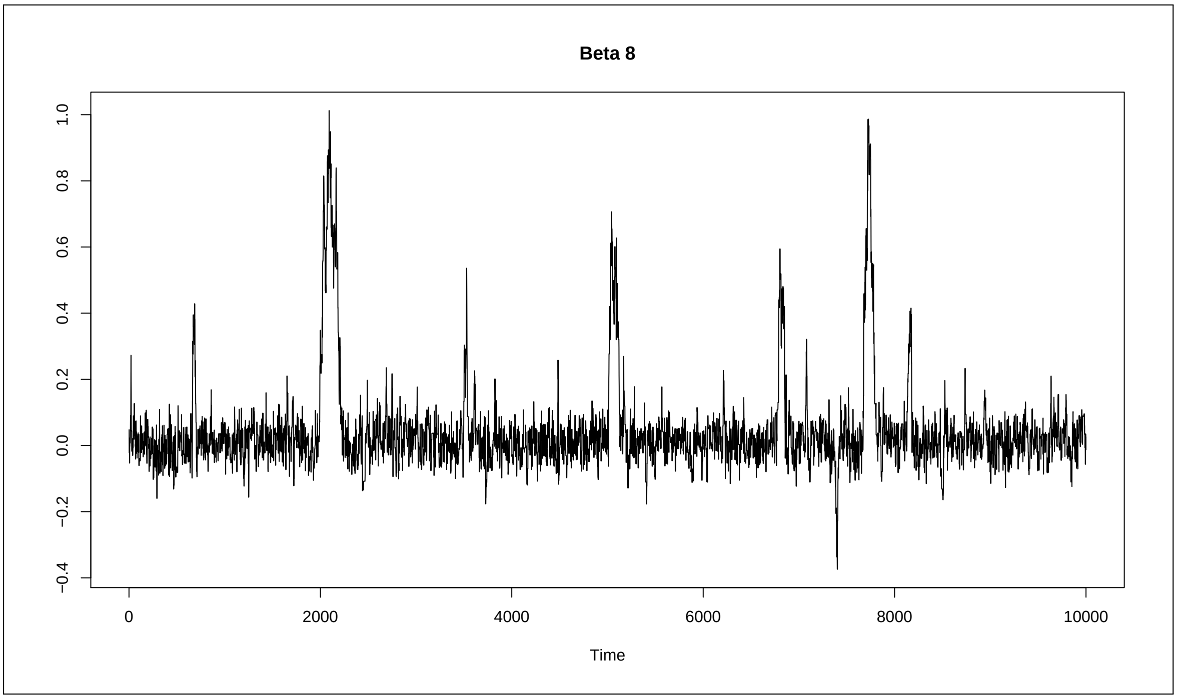

Beta 5
ACF
1.0
0.8
0.6
0.4
0.2
0.0
0
20
40
60
80
100
Lag

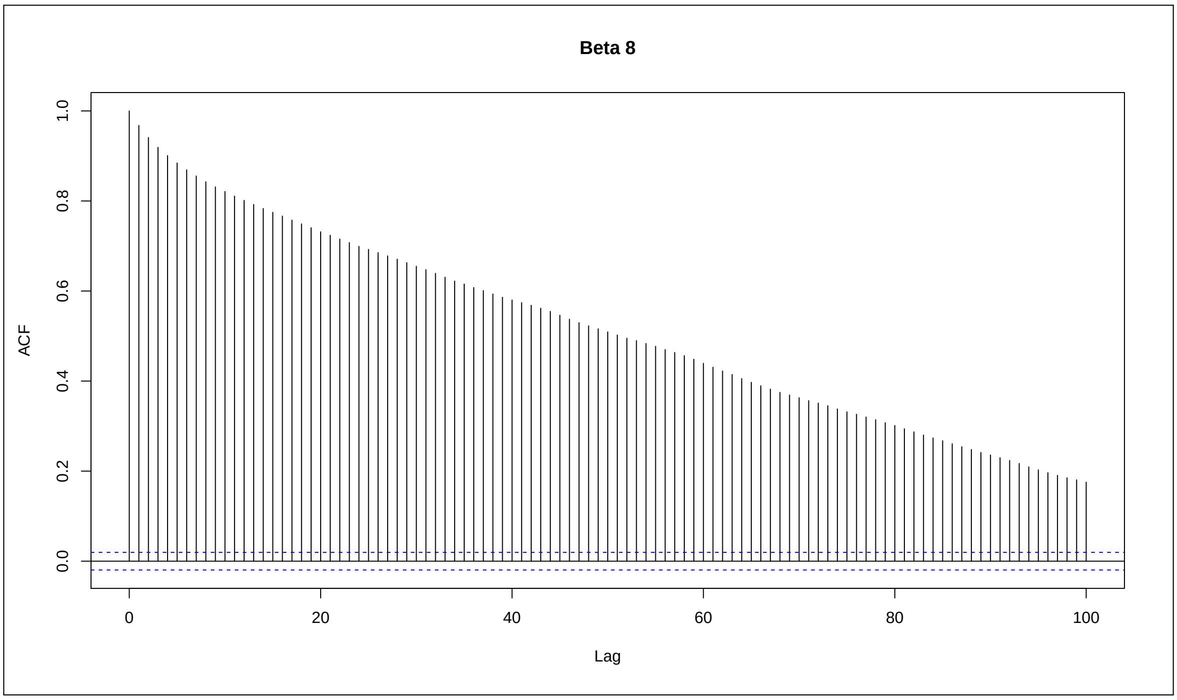

Beta 6
0.5
0.0
−0.5
−1.0
0
2000
4000
6000
8000
10000
Time

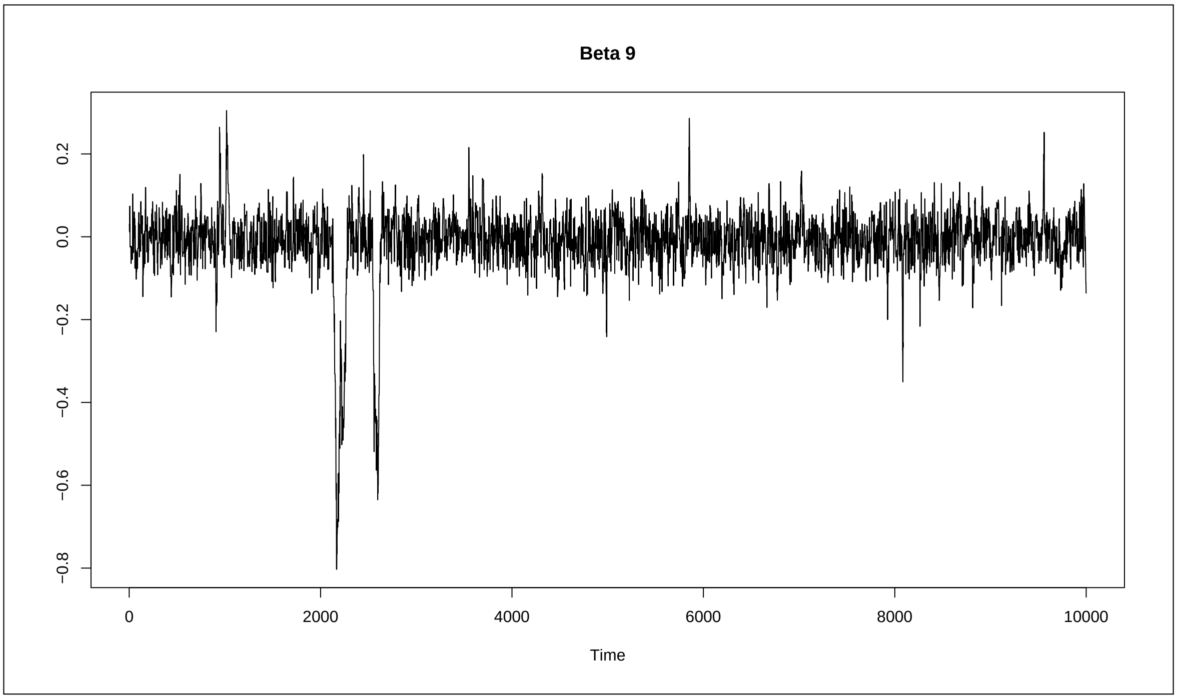

Beta 6
ACF
1.0
0.8
0.6
0.4
0.2
0.0
0
20
40
60
80
100
Lag

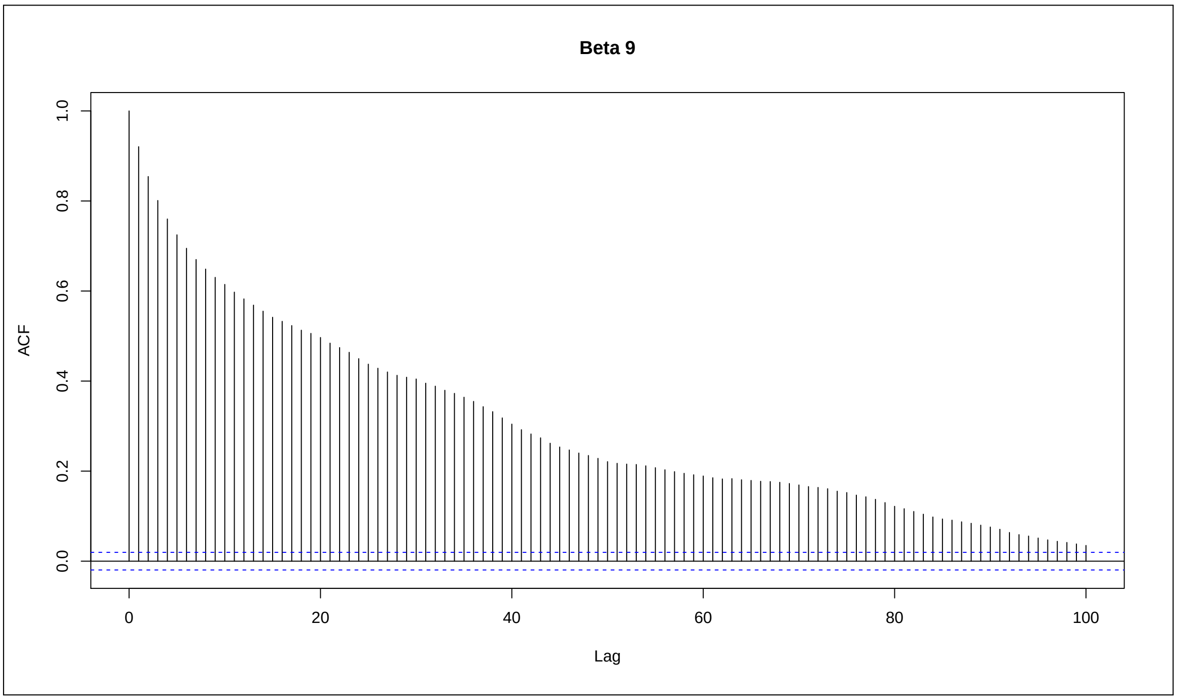

Beta 7
0.2
0.0
−0.2
−0.4
−0.6
−0.8
0
2000
4000
6000
8000
10000
Time

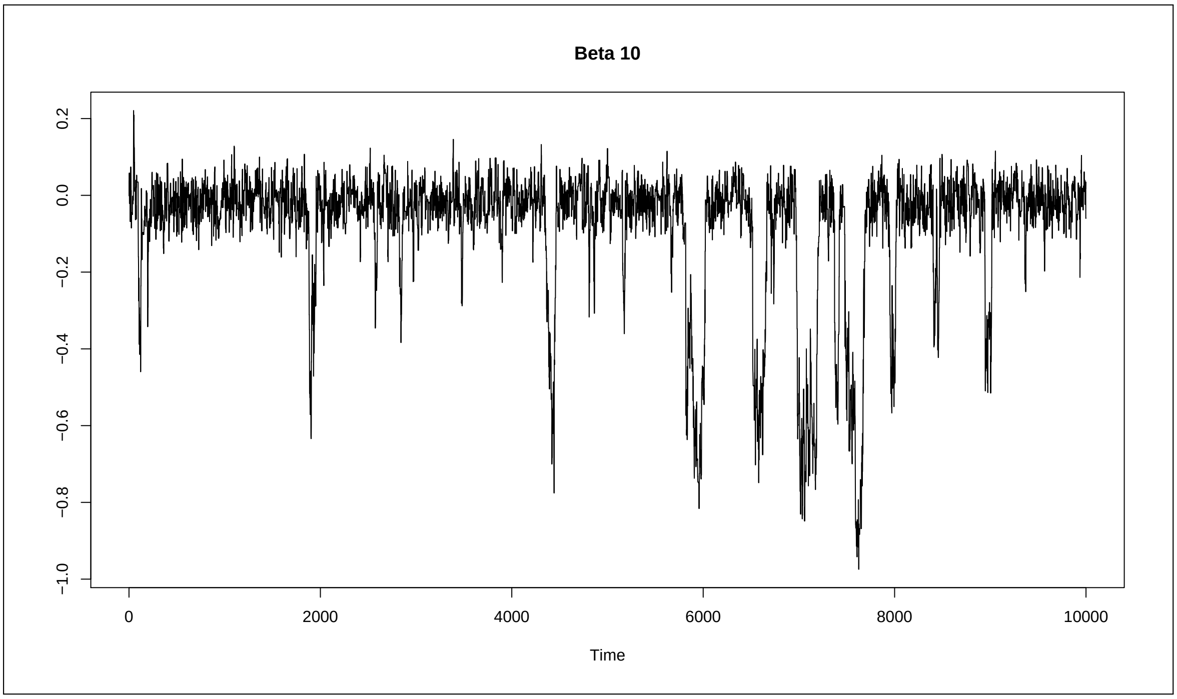

Beta 7
ACF
1.0
0.8
0.6
0.4
0.2
0.0
0
20
40
60
80
100
Lag

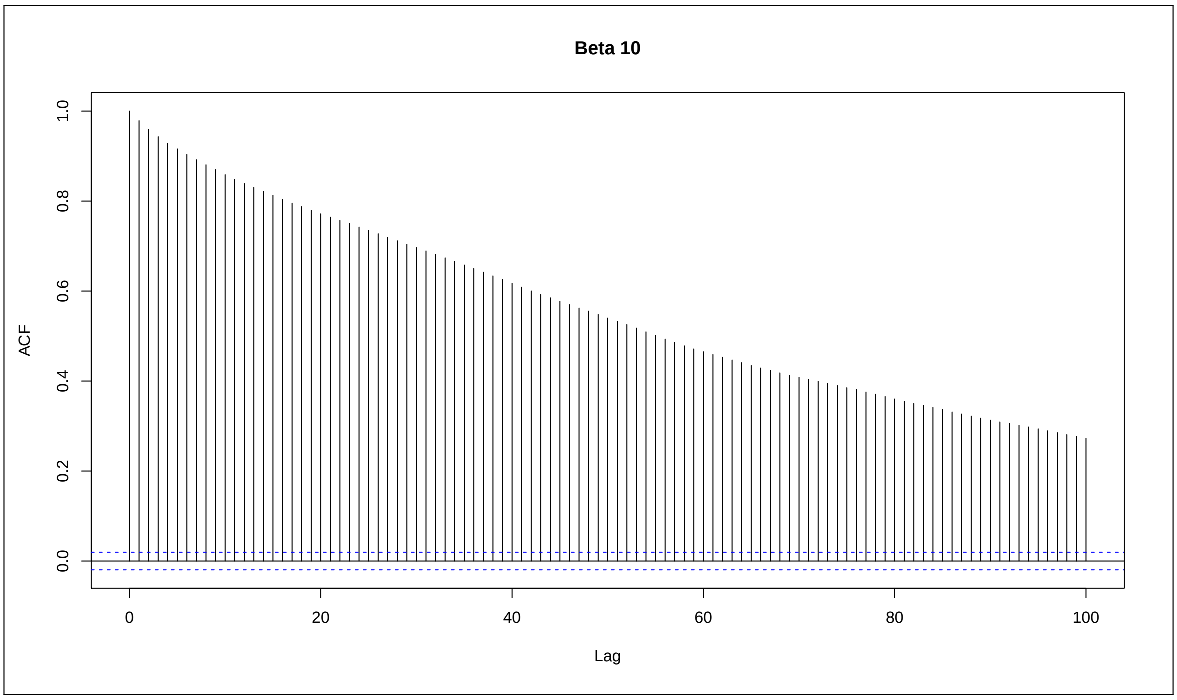

Beta 8
1.0
0.8
0.6
0.4
0.2
0.0
−0.2
−0.4
0
2000
4000
6000
8000
10000
Time

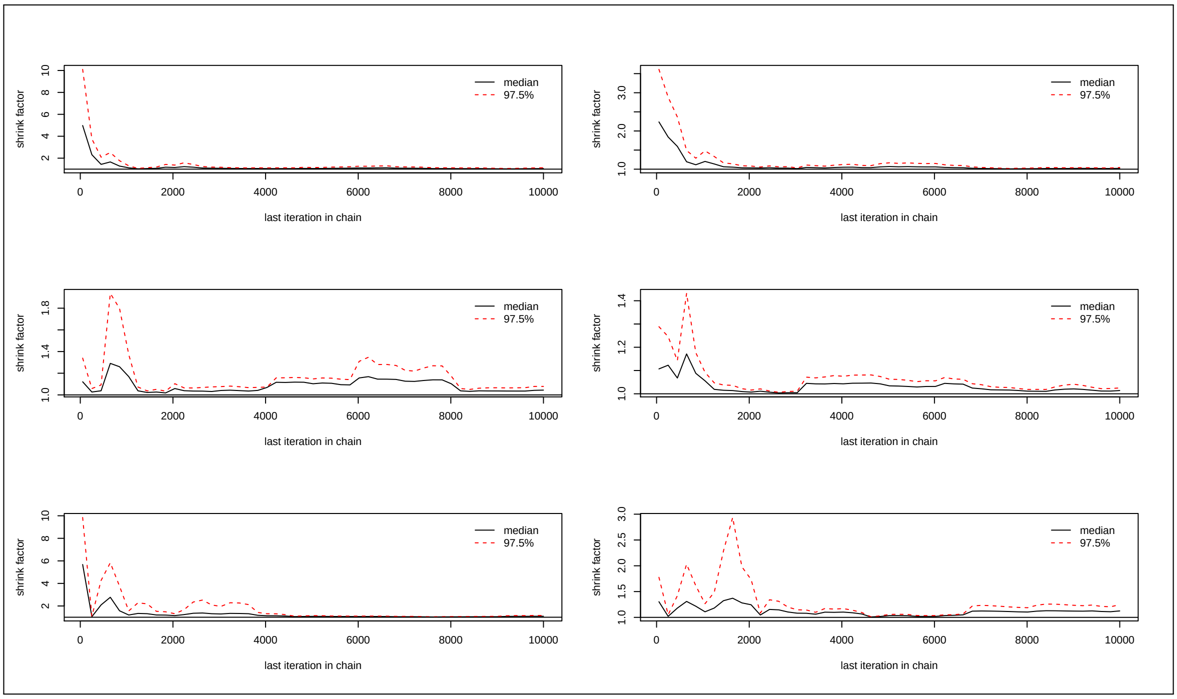

Beta 8
ACF
1.0
0.8
0.6
0.4
0.2
0.0
0
20
40
60
80
100
Lag

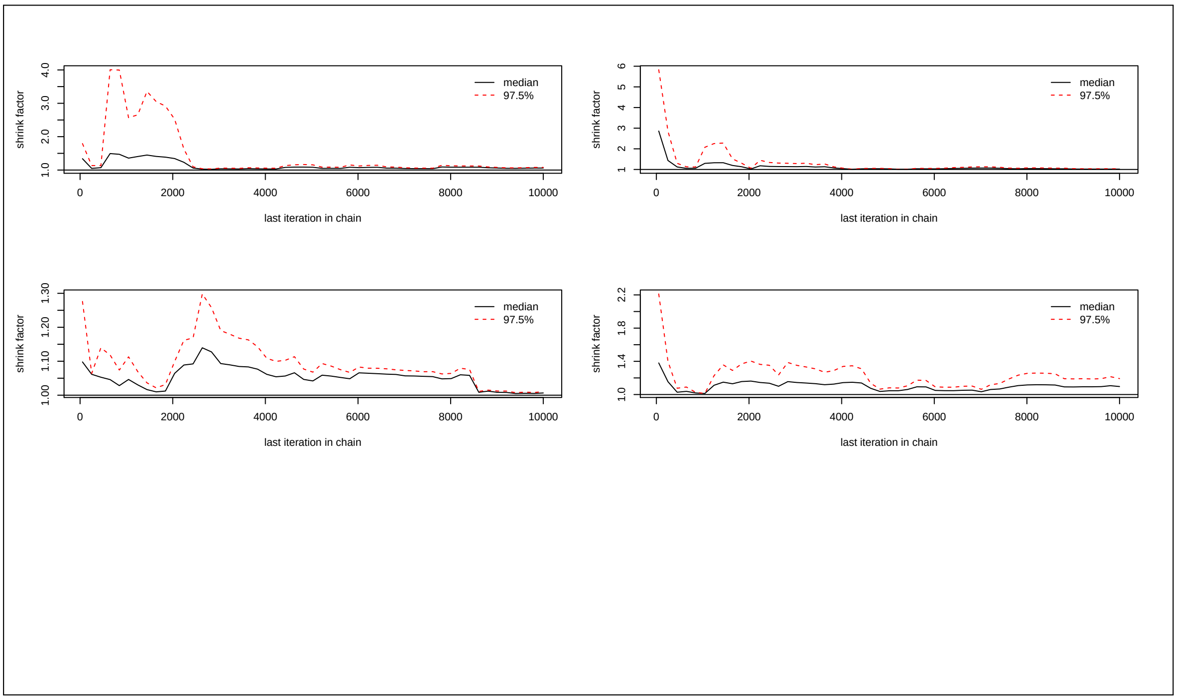

Beta 9
0.2
0.0
−0.2
−0.4
−0.6
−0.8
0
2000
4000
6000
8000
10000
Time


Beta 9
ACF
1.0
0.8
0.6
0.4
0.2
0.0
0
20
40
60
80
100
Lag


Beta 10
0.2
0.0
−0.2
−0.4
−0.6
−0.8
−1.0
0
2000
4000
6000
8000
10000
Time


Beta 10
ACF
0.0
0.2
0.4
0.6
0.8
1.0
0
20
40
60
80
100
Lag

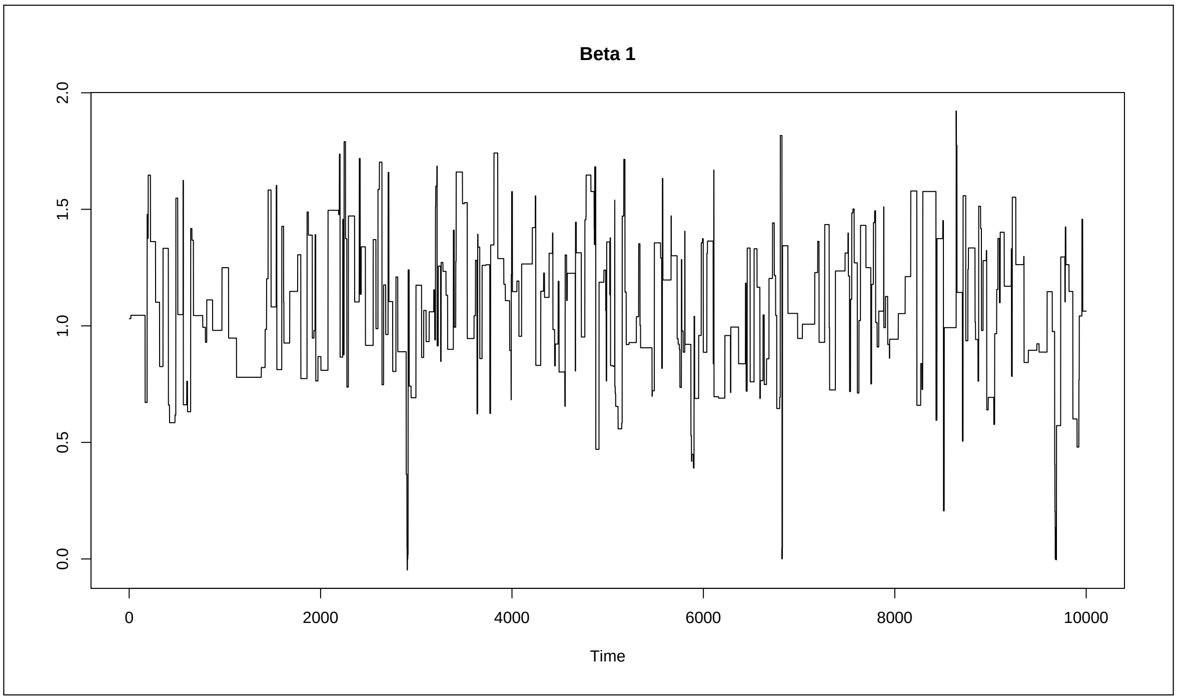

shrink factor
median
97.5%
last iteration in chain
0
2000
4000
6000
8000
10000
2
4
6
8
10
1.0
2.0
3.0
1.0
1.4
1.8
1.0
1.2
1.4
1.0
1.5
2.0
2.5
3.0

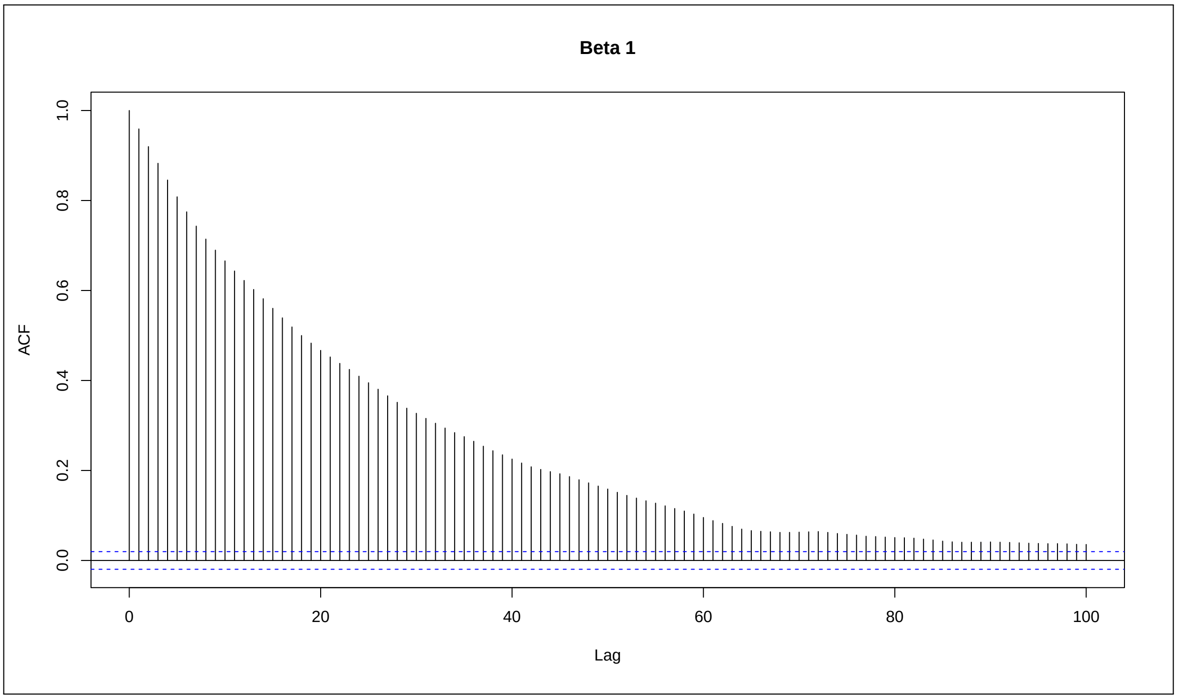


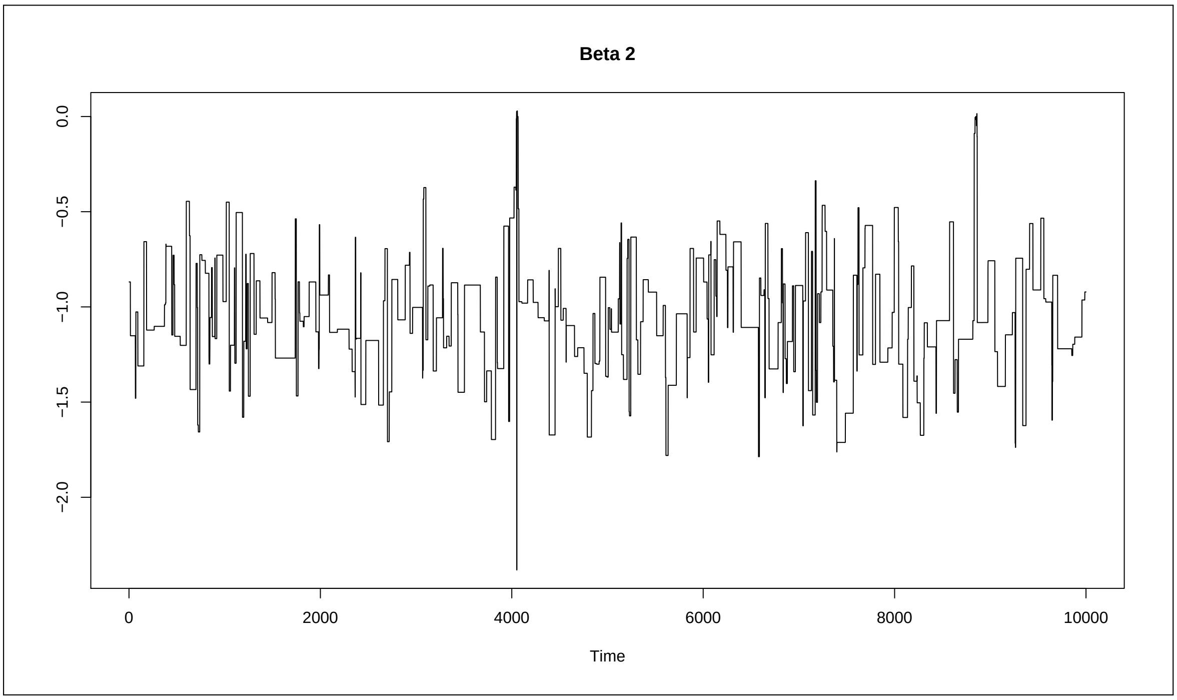
A.3 Single component updating with bi-modal independence sampler

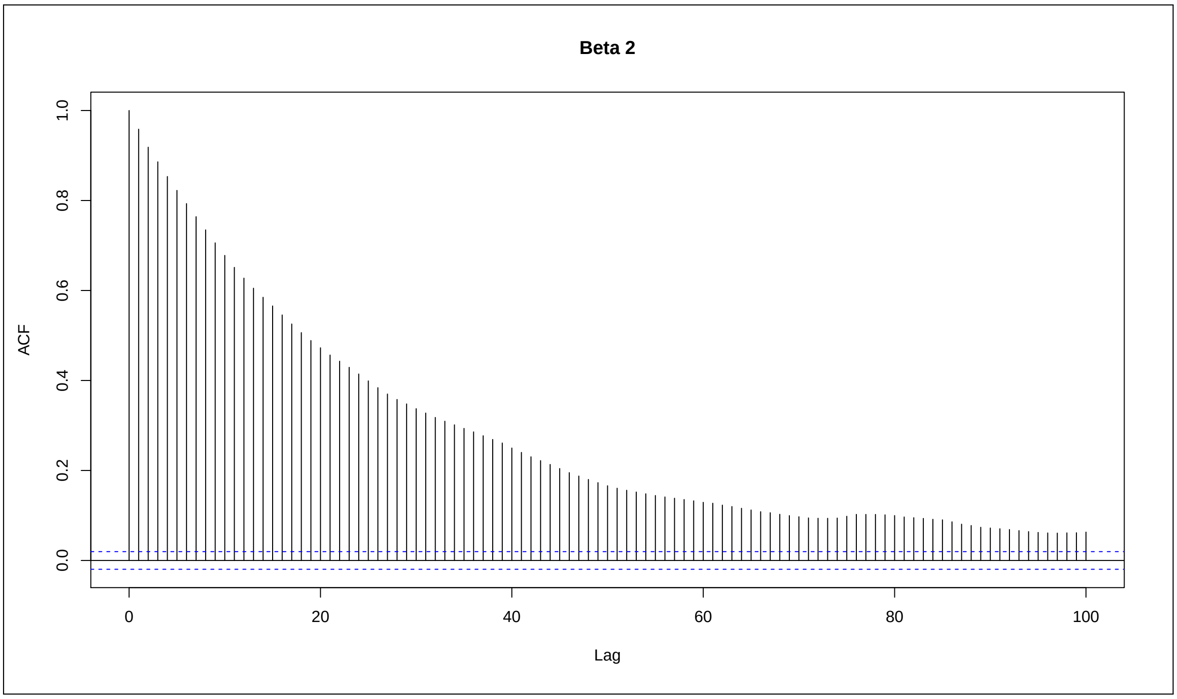
Parameter	G-R Point est.	G-R Upper C.I.	Acceptance rate
β_1	1.0025	1.0037	0.0227
β_2	1.0143	1.0354	0.0221
β_3	1.0016	1.0023	0.4829
β_4	1.0004	1.0006	0.4795
β_5	1.0000	1.0001	0.4702
β_6	1.0028	1.0034	0.4877
β_7	1.0027	1.0042	0.4707
β_8	1.0010	1.0018	0.4707
β_9	1.0008	1.0011	0.4816
β_{10}	1.0010	1.0016	0.4419

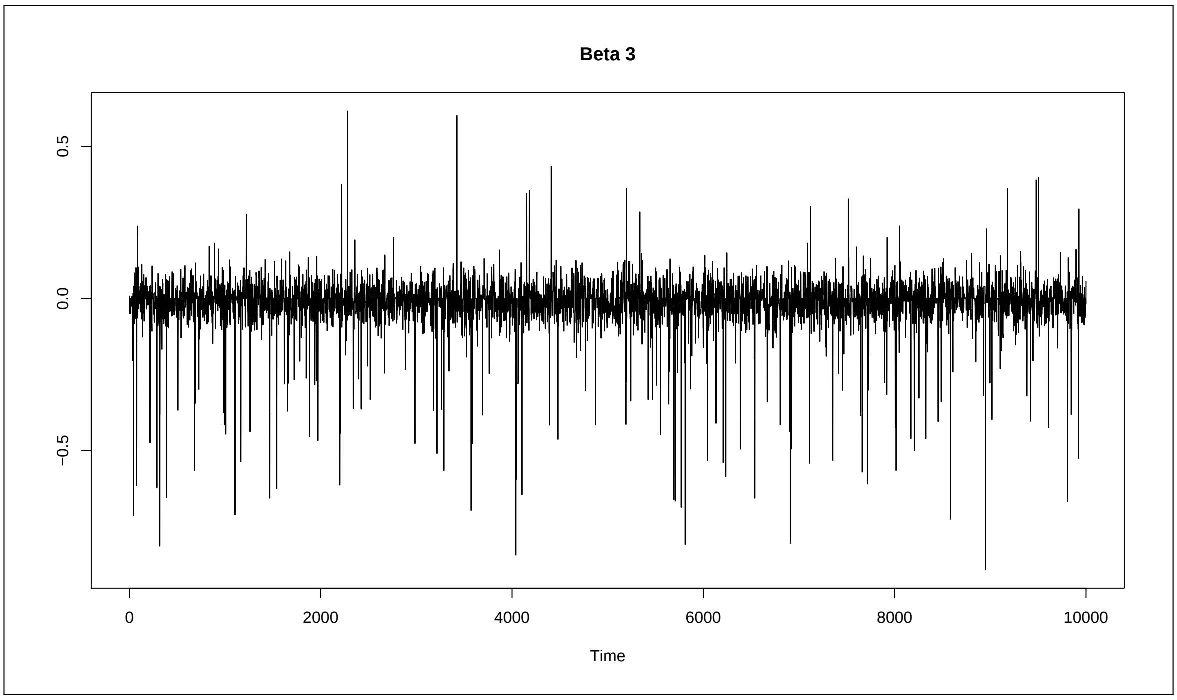
Table A.3: Point estimates of potential scale reduction factor, corresponding upper confidence limits and acceptance rates for the ten logistic slope parameters.

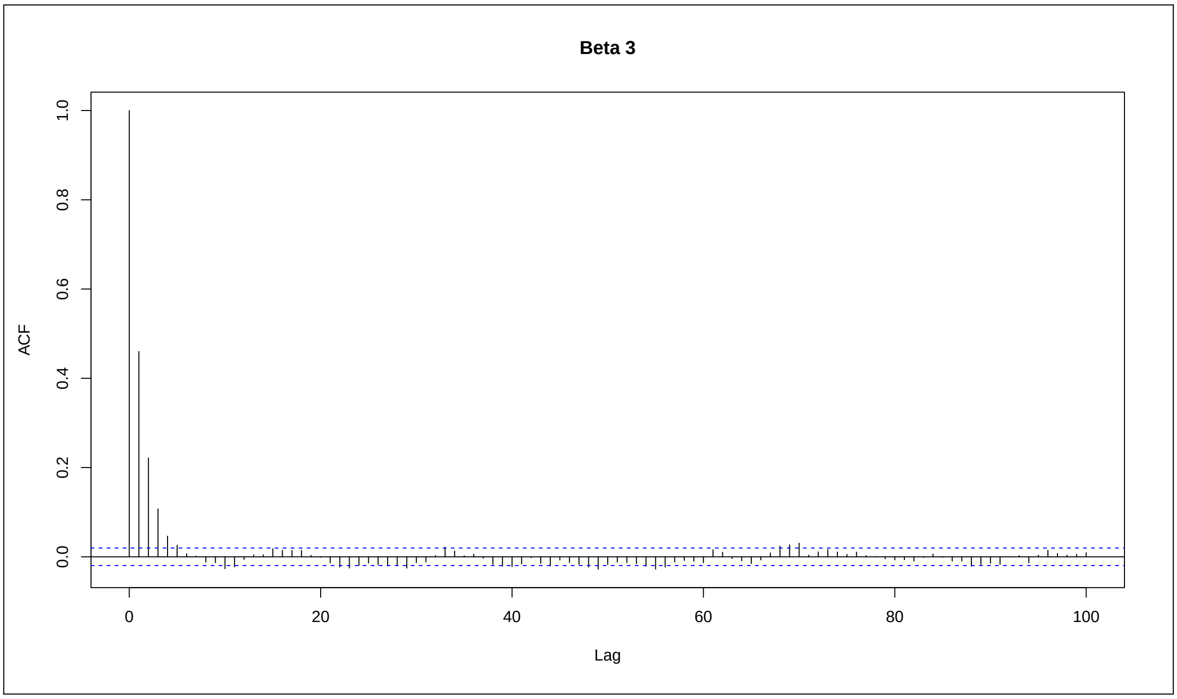

Beta 1
2.0
1.5
1.0
0.5
0.0
0
2000
4000
6000
8000
10000
Time

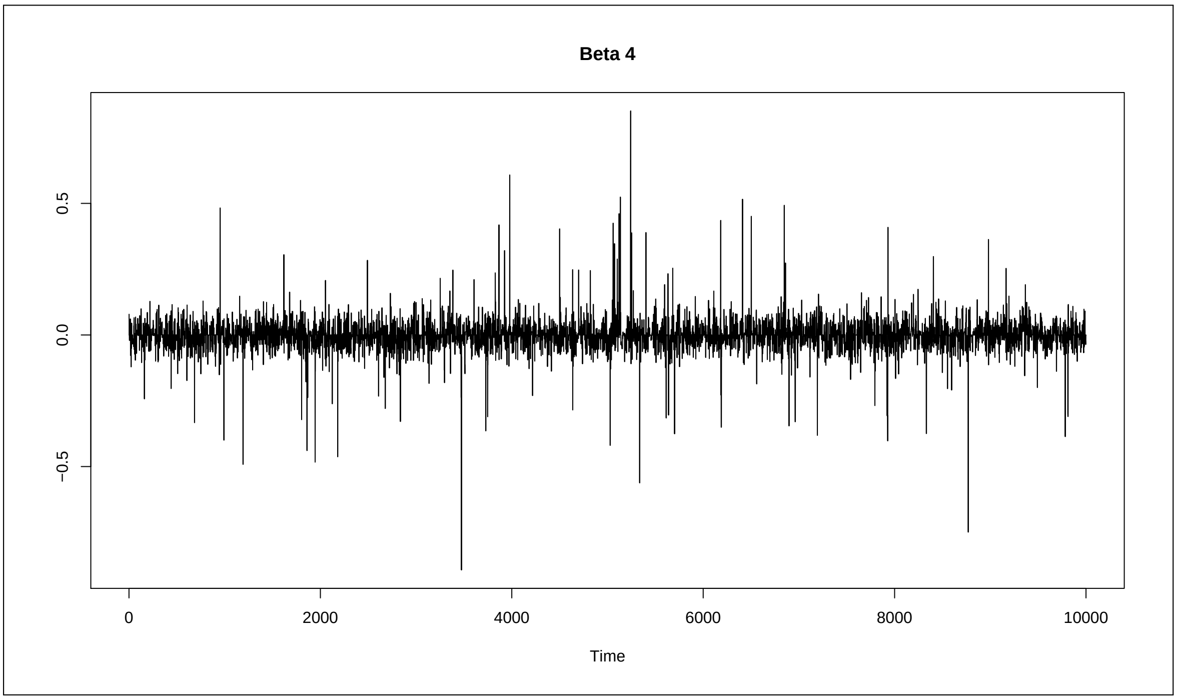

Beta 1
ACF
1.0
0.8
0.6
0.4
0.2
0.0
0
20
40
60
80
100
Lag

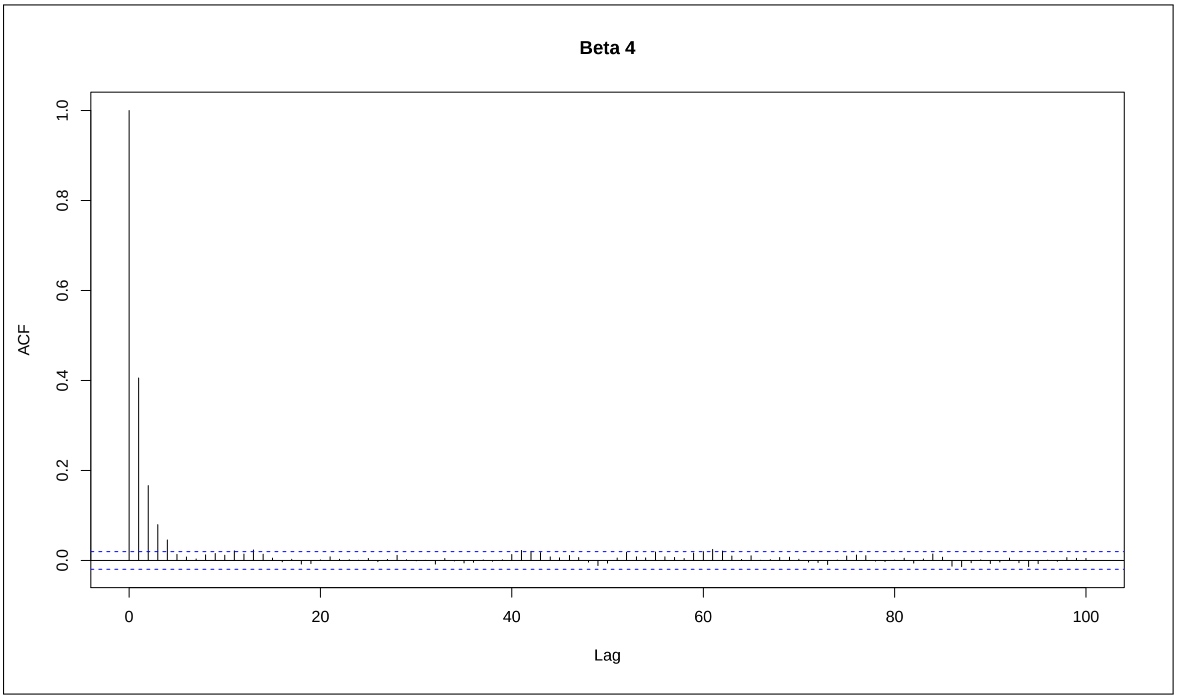

Beta 2
0.0
−0.5
−1.0
−1.5
−2.0
0
2000
4000
6000
8000
10000
Time

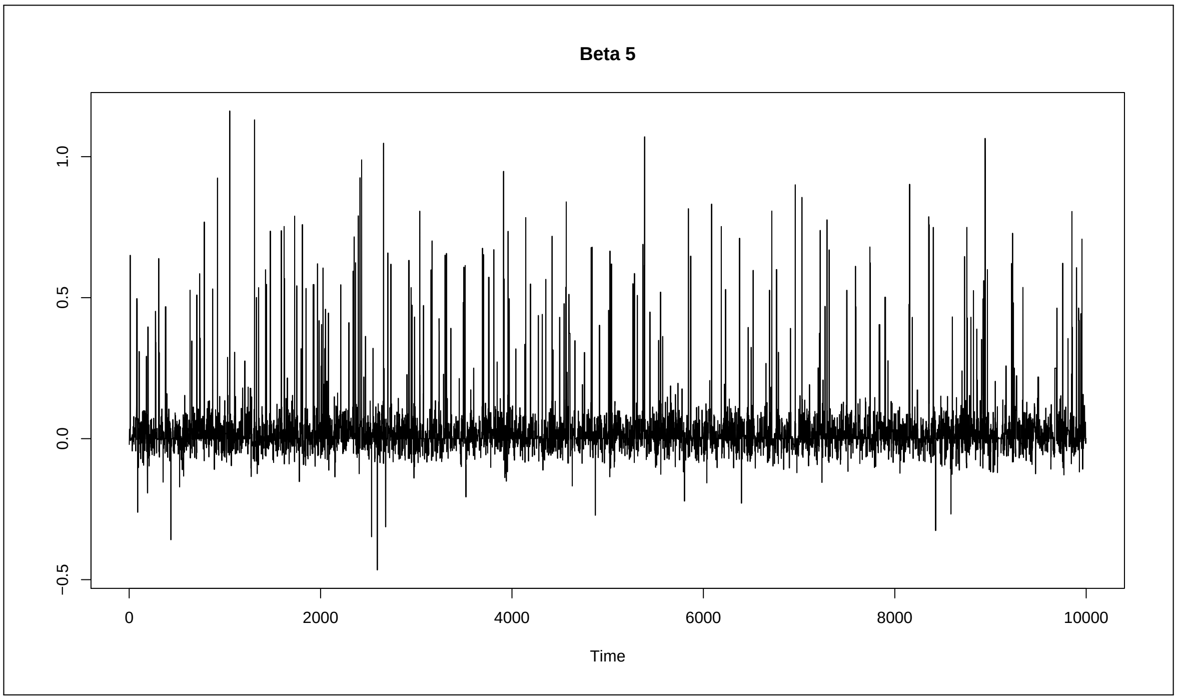

Beta 2
ACF
0.0
0.2
0.4
0.6
0.8
1.0
0
20
40
60
80
100
Lag

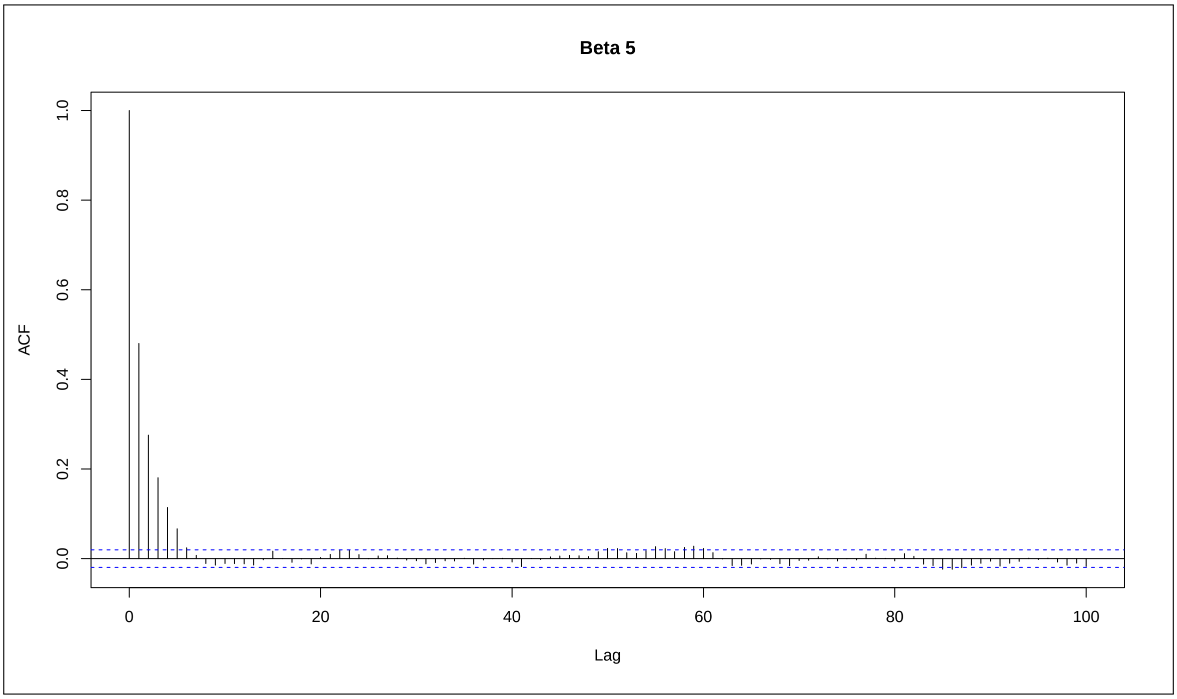

Beta 3
0.5
0.0
−0.5
0
2000
4000
6000
8000
10000
Time

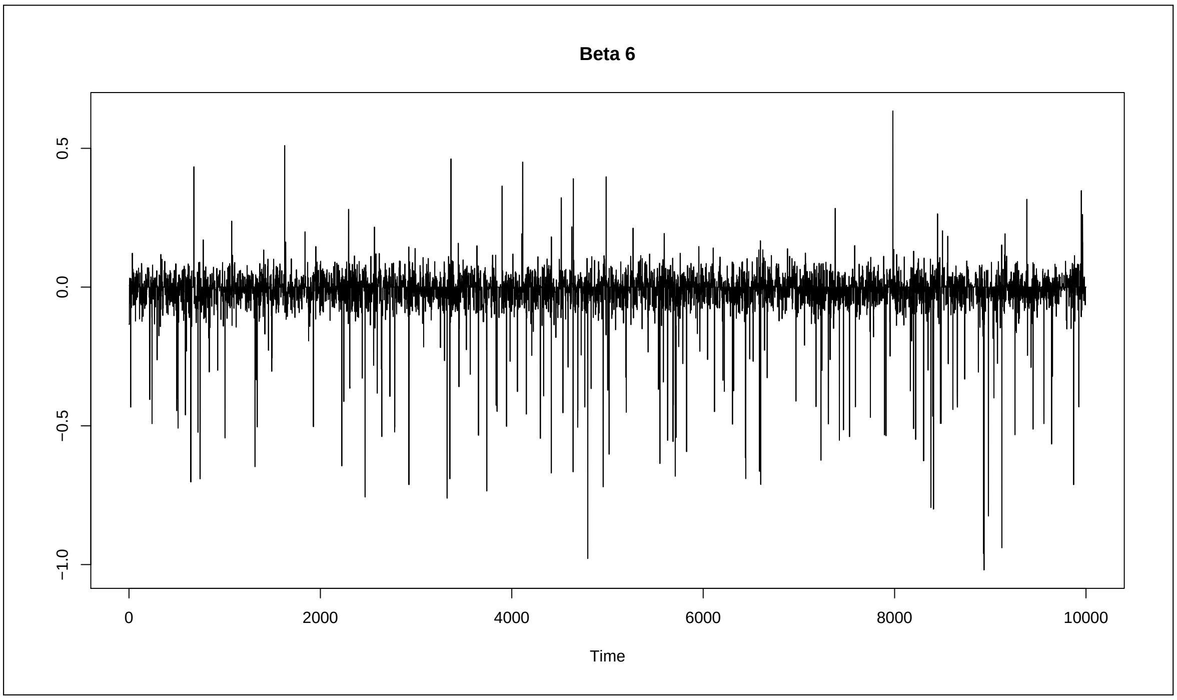

Beta 3
ACF
1.0
0.8
0.6
0.4
0.2
0.0
0
20
40
60
80
100
Lag

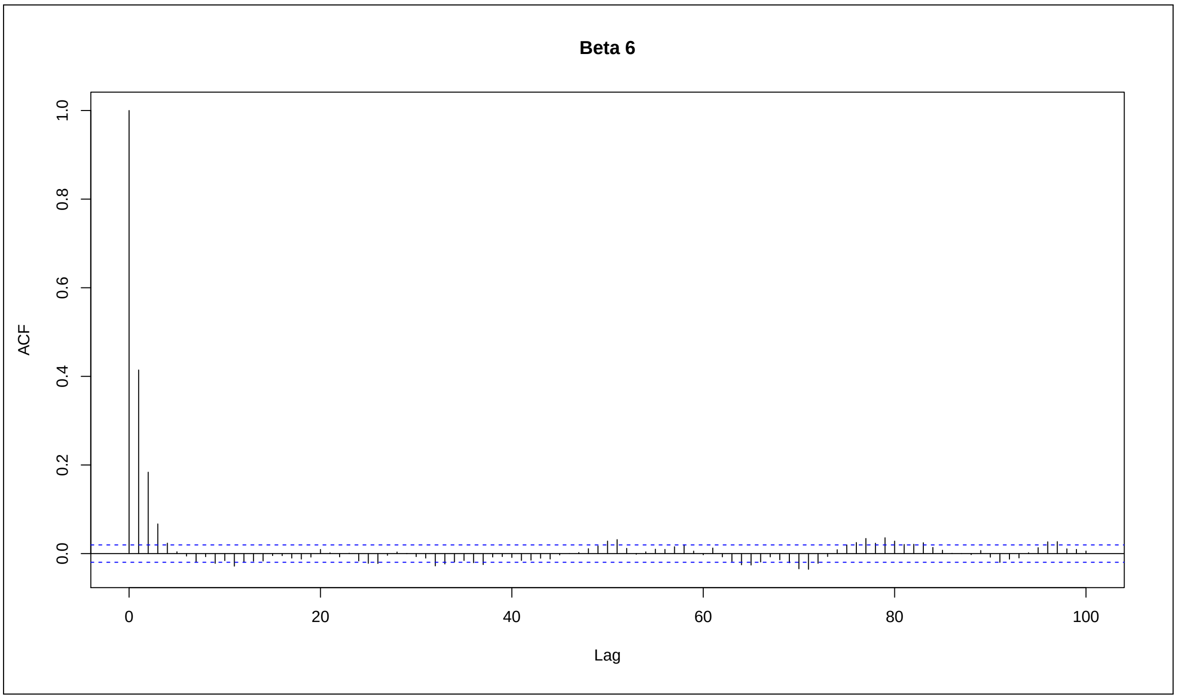

Beta 4
0.5
0.0
−0.5
0
2000
4000
6000
8000
10000
Time

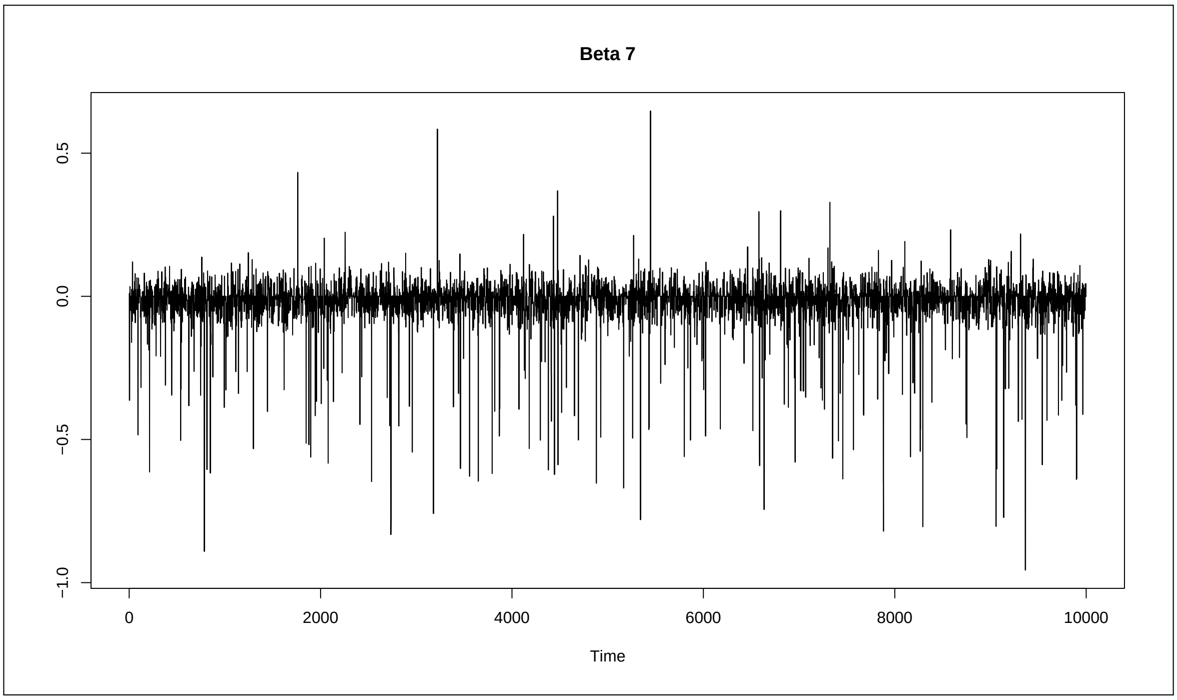

Beta 4
ACF
1.0
0.8
0.6
0.4
0.2
0.0
0
20
40
60
80
100
Lag

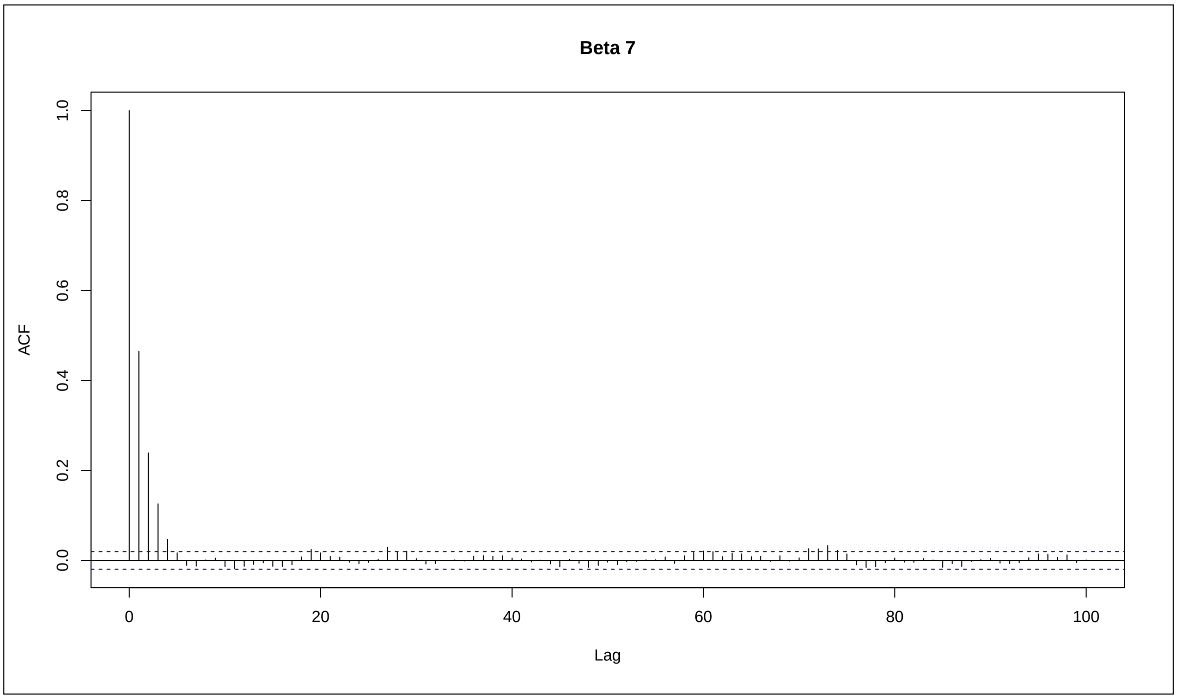

Beta 5
1.0
0.5
0.0
−0.5
0
2000
4000
6000
8000
10000
Time

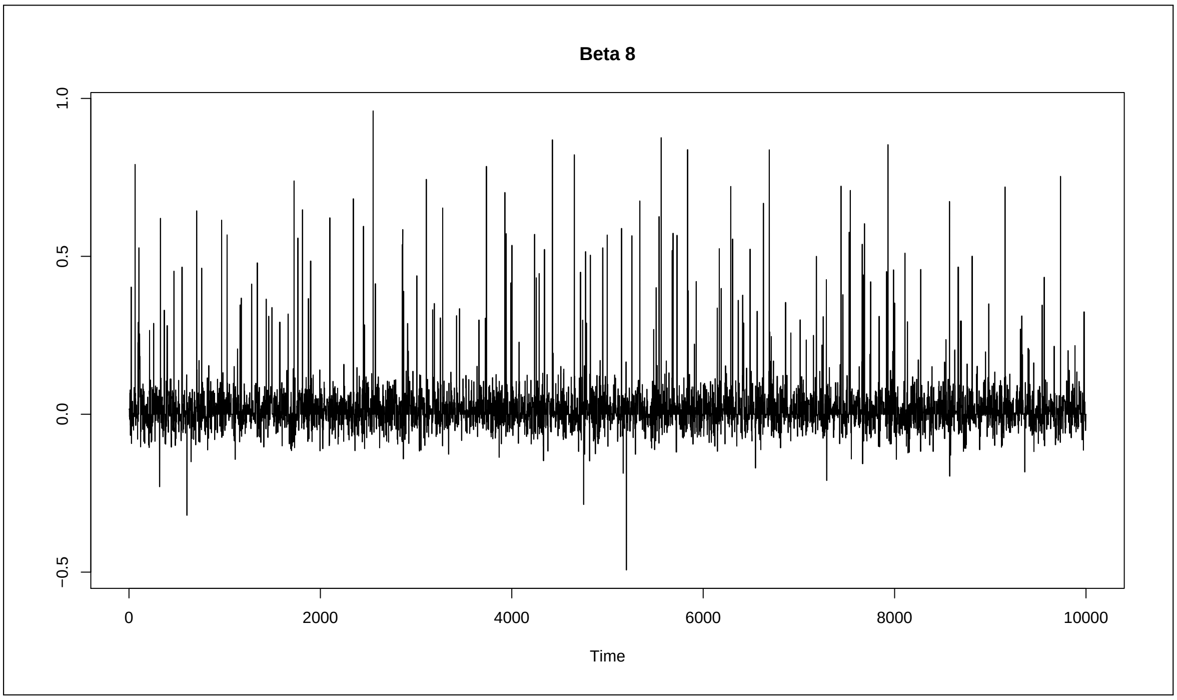

Beta 5
ACF
1.0
0.8
0.6
0.4
0.2
0.0
0
20
40
60
80
100
Lag

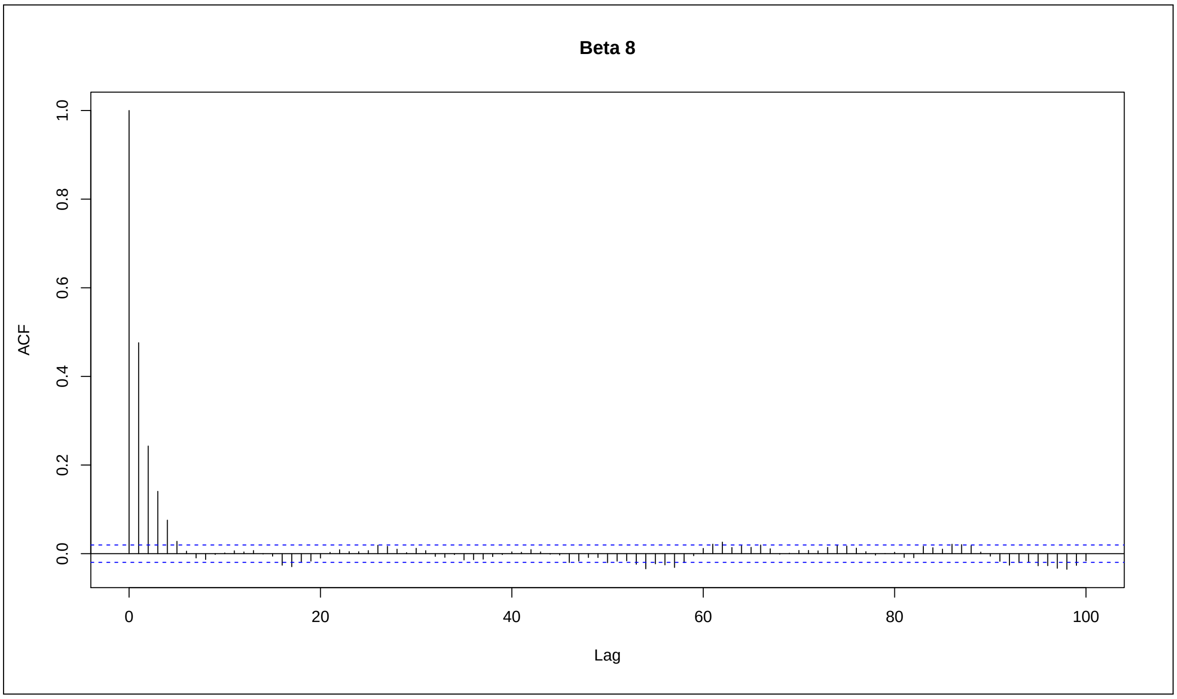

Beta 6
0.5
0.0
−0.5
−1.0
0
2000
4000
6000
8000
10000
Time

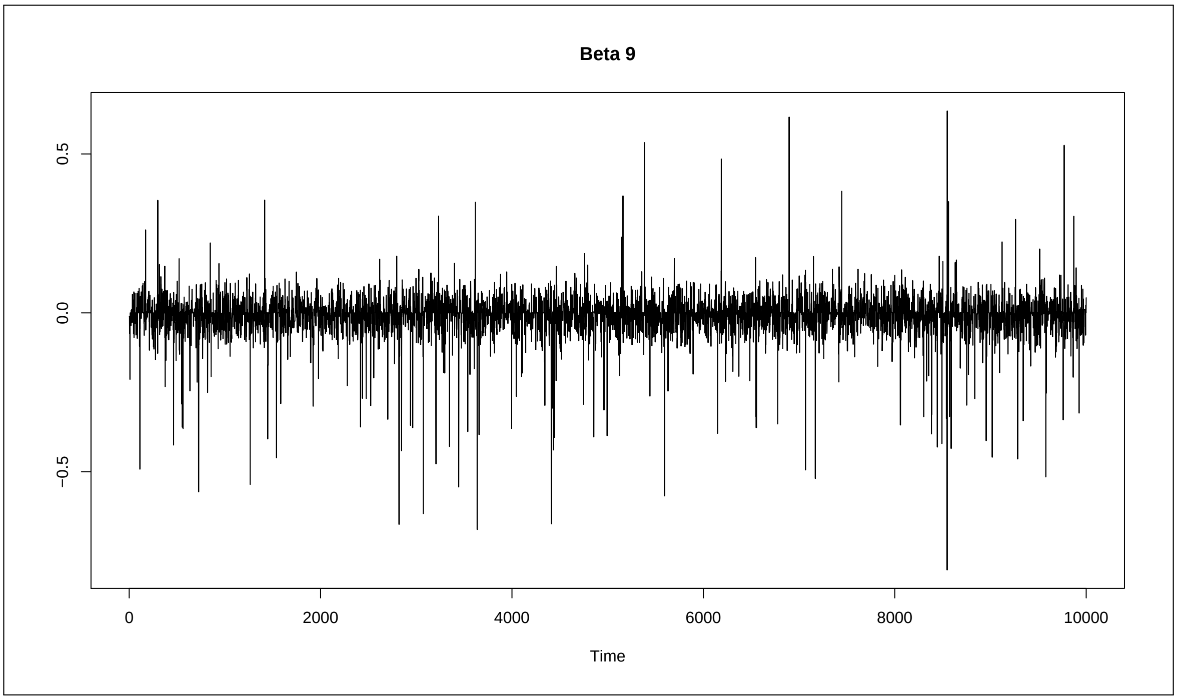

Beta 6
ACF
1.0
0.8
0.6
0.4
0.2
0.0
0
20
40
60
80
100
Lag

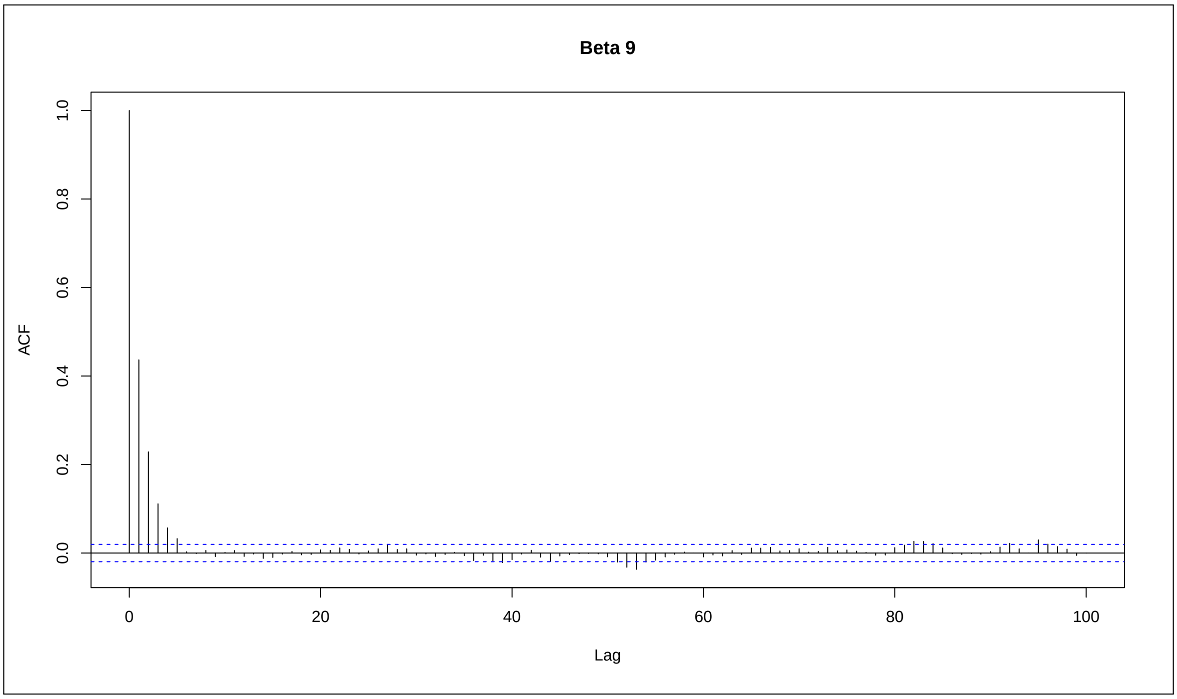

Beta 7
0.5
0.0
−0.5
−1.0
0
2000
4000
6000
8000
10000
Time

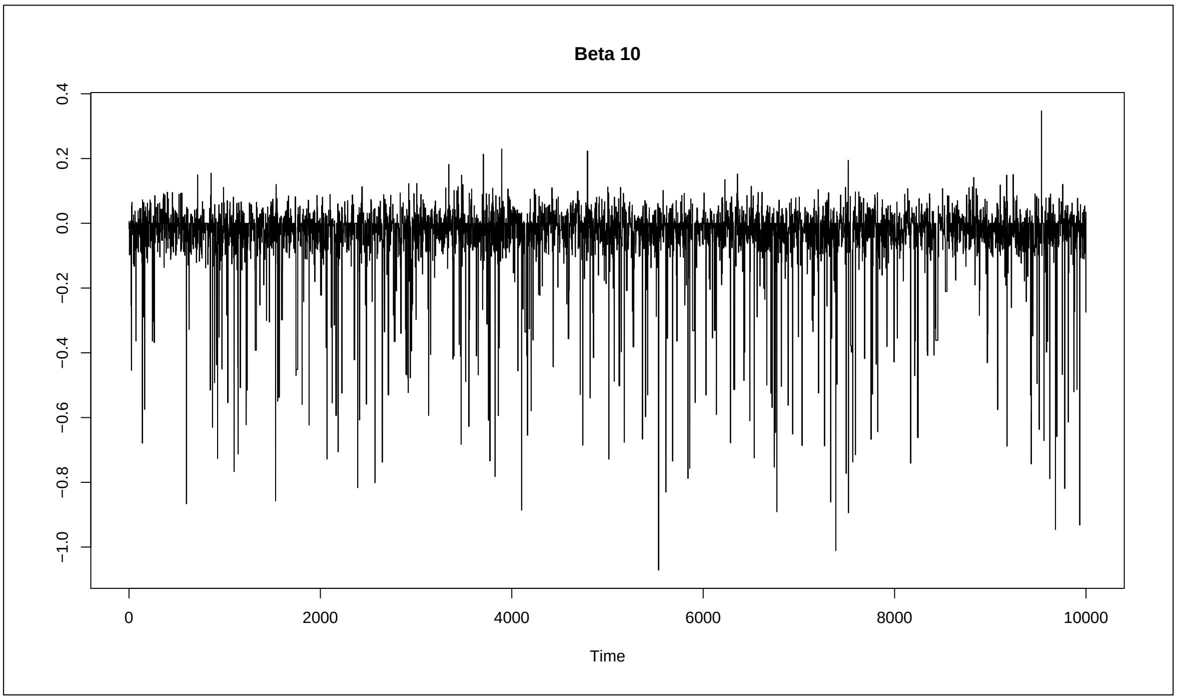

Beta 7
ACF
1.0
0.8
0.6
0.4
0.2
0.0
0
20
40
60
80
100
Lag

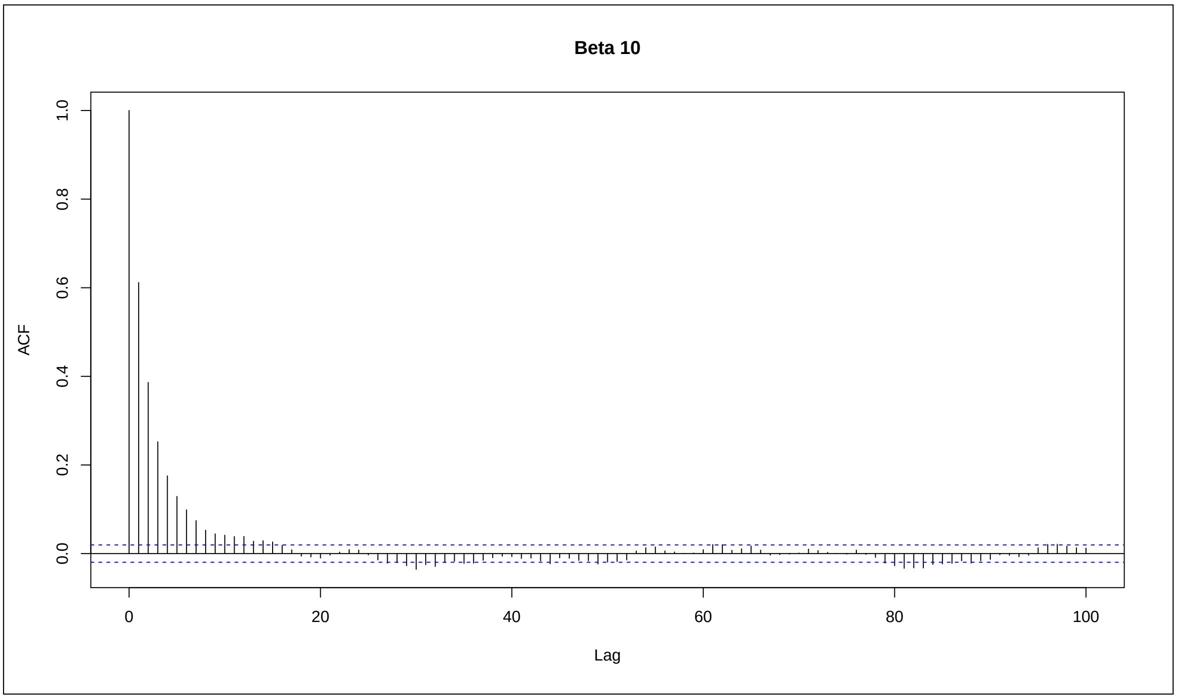

Beta 8
1.0
0.5
0.0
−0.5
0
2000
4000
6000
8000
10000
Time

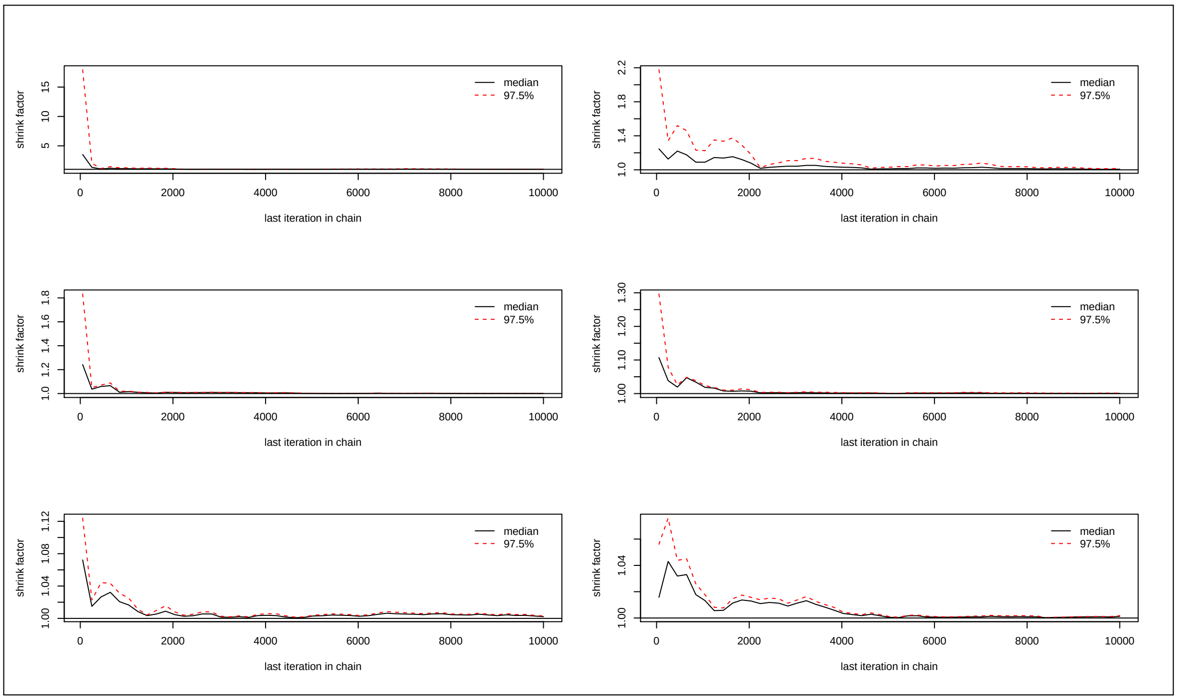

Beta 8
ACF
1.0
0.8
0.6
0.4
0.2
0.0
0
20
40
60
80
100
Lag

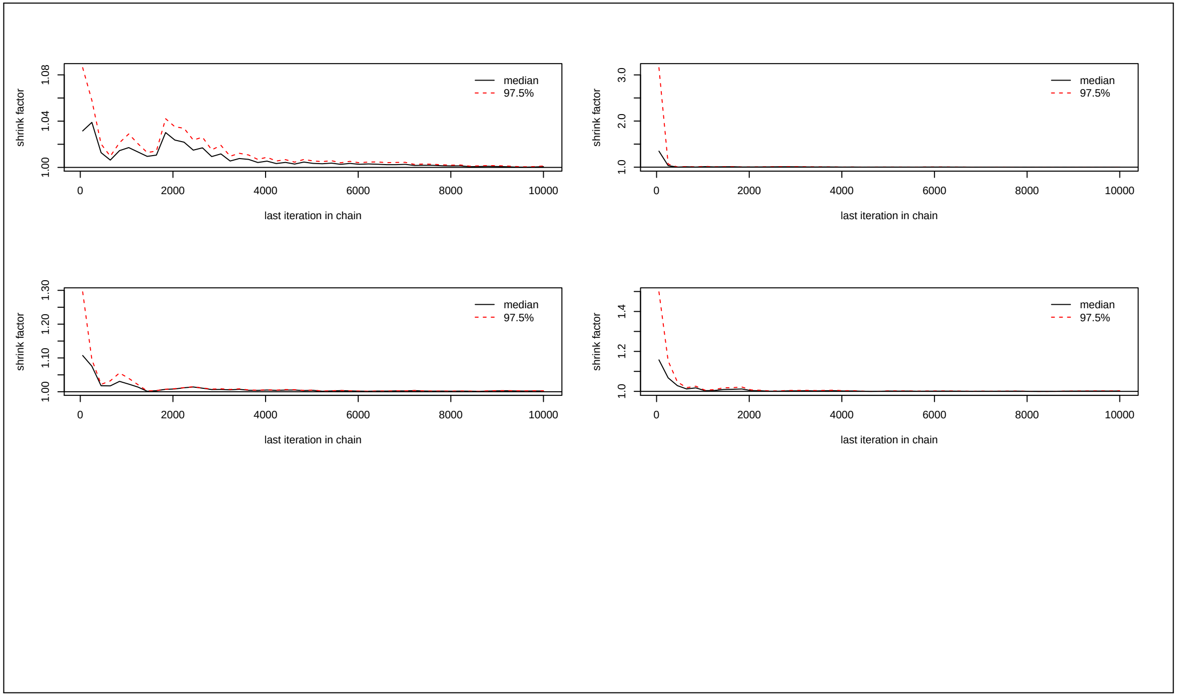

Beta 9
0.5
0.0
−0.5
0
2000
4000
6000
8000
10000
Time


Beta 9
ACF
1.0
0.8
0.6
0.4
0.2
0.0
0
20
40
60
80
100
Lag


Beta 10
0.4
0.2
0.0
−0.2
−0.4
−0.6
−0.8
−1.0
0
2000
4000
6000
8000
10000
Time


Beta 10
ACF
1.0
0.8
0.6
0.4
0.2
0.0
0
20
40
60
80
100
Lag

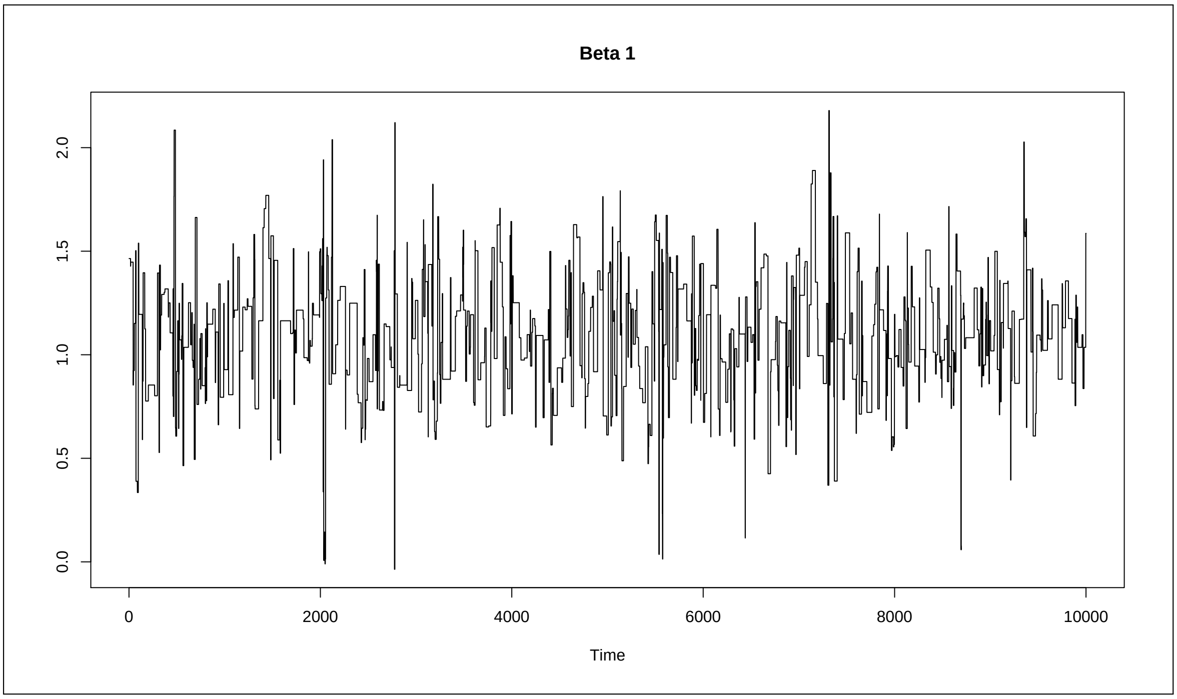


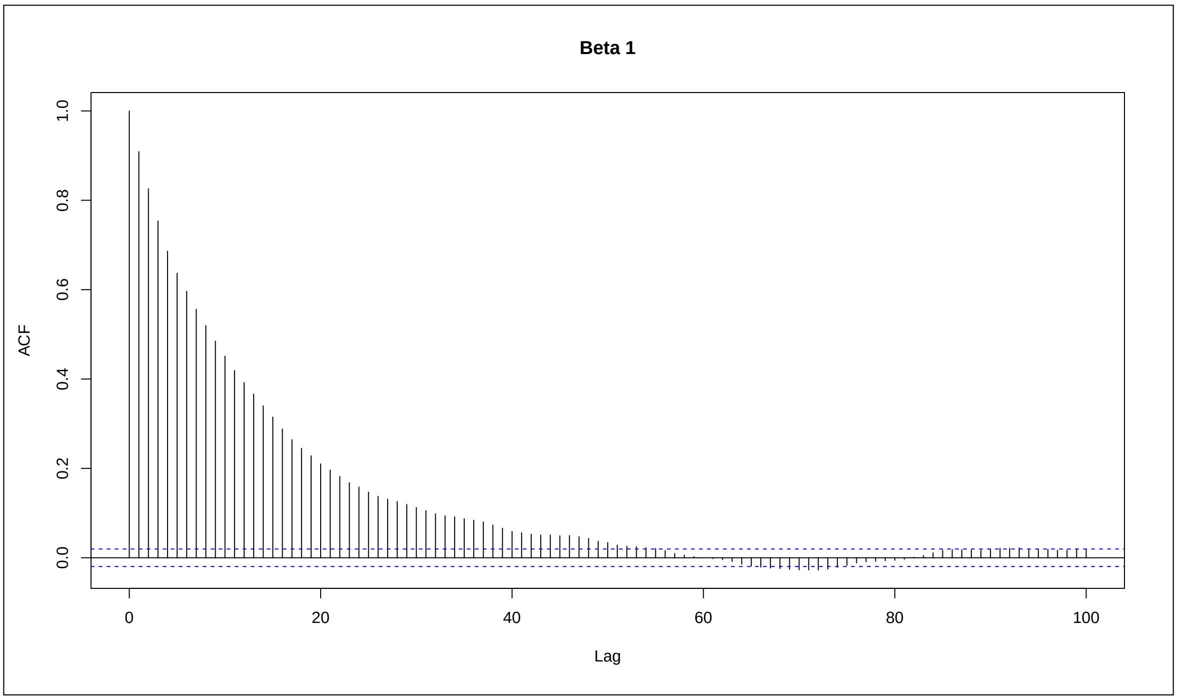


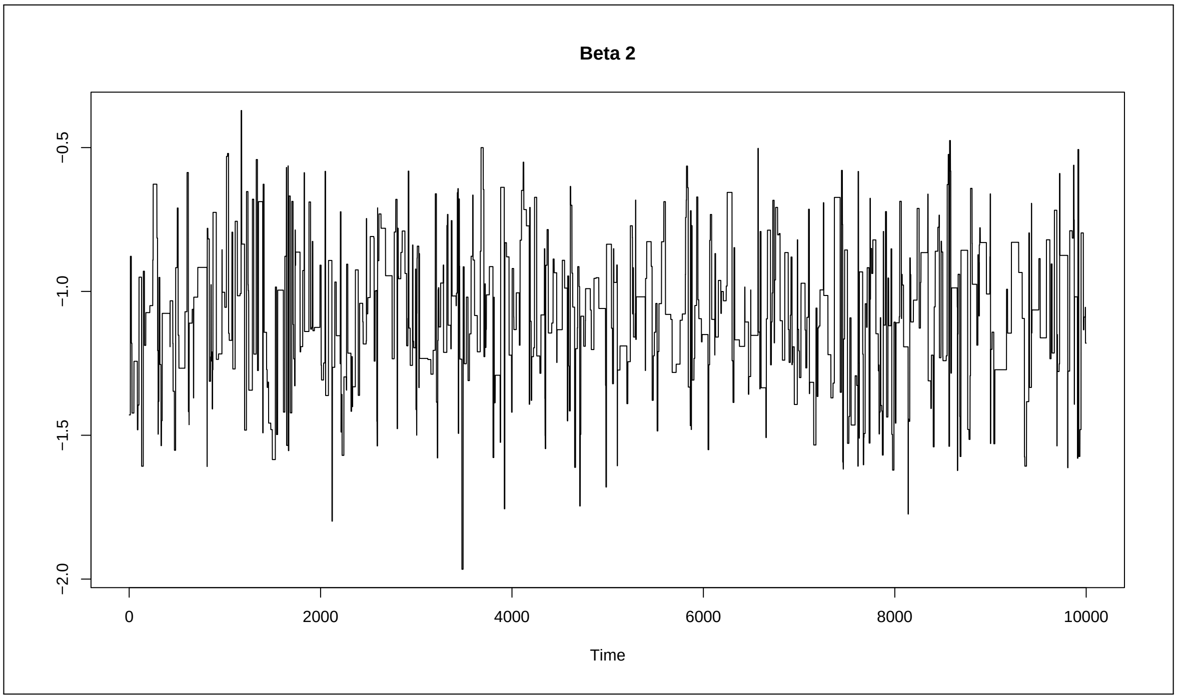
A.4 Single component updating with tri-modal independence sampler

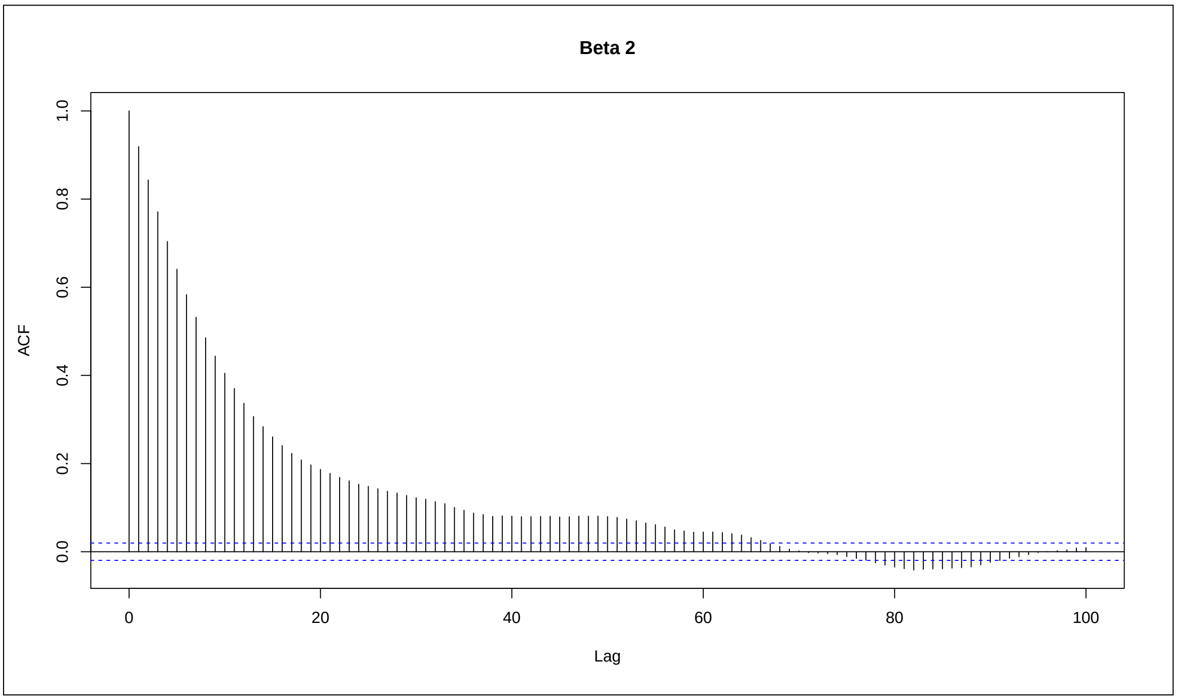
Parameter	G-R Point est.	G-R Upper C.I.	Acceptance rate
β_1	1.0086	1.0212	0.0451
β_2	1.0057	1.0137	0.0422
β_3	1.0006	1.0009	0.2214
β_4	1.0017	1.0018	0.2279
β_5	1.0060	1.0088	0.2184
β_6	1.0013	1.0015	0.2253
β_7	1.0016	1.0024	0.2143
β_8	1.0035	1.0048	0.2145
β_9	1.0040	1.0046	0.2249
β_{10}	1.0008	1.0017	0.2090

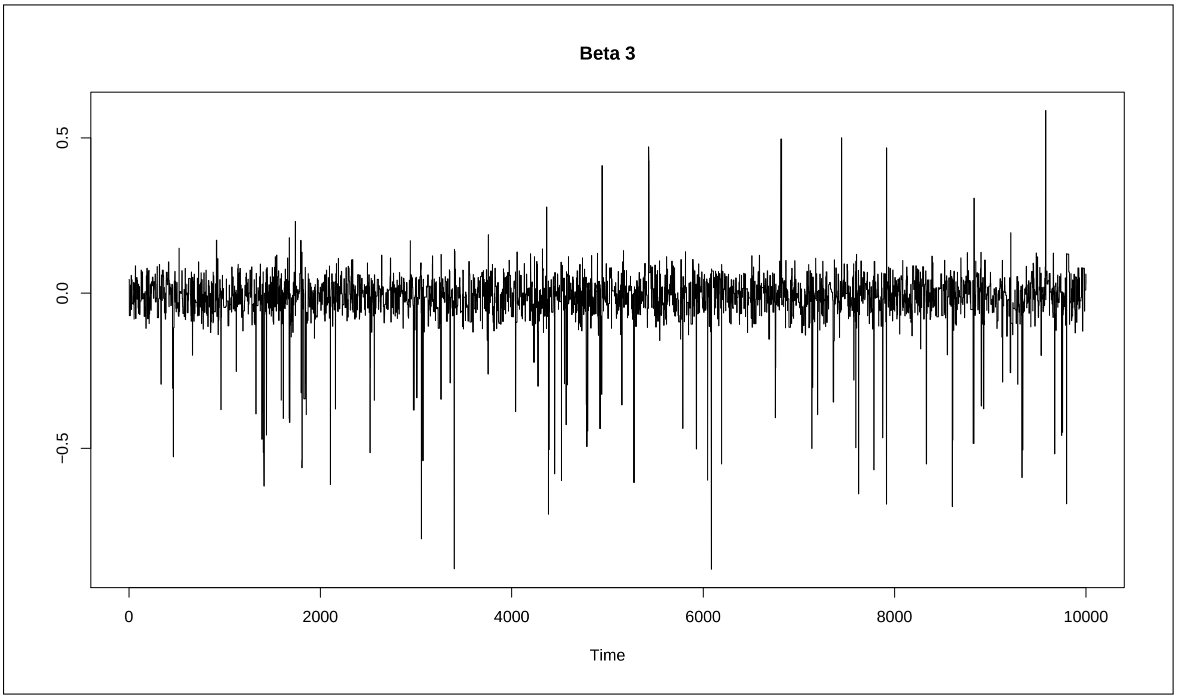
Table A.4: Point estimates of potential scale reduction factor, corresponding upper confidence limits and acceptance rates for the ten logistic slope parameters.

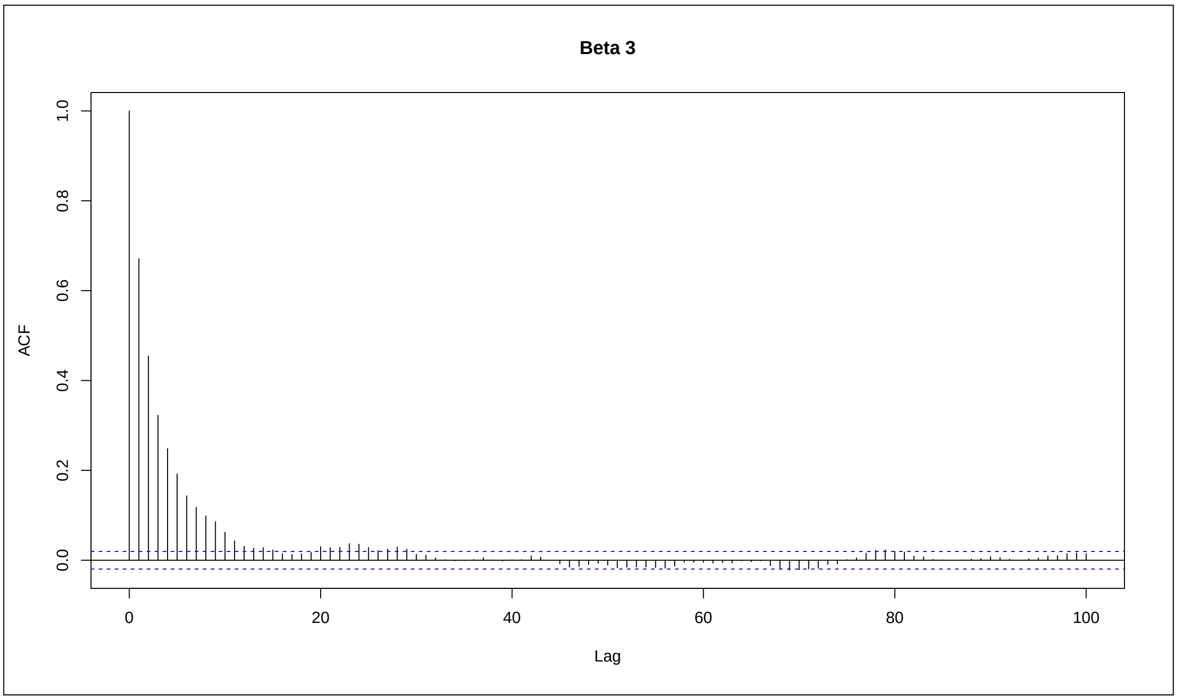


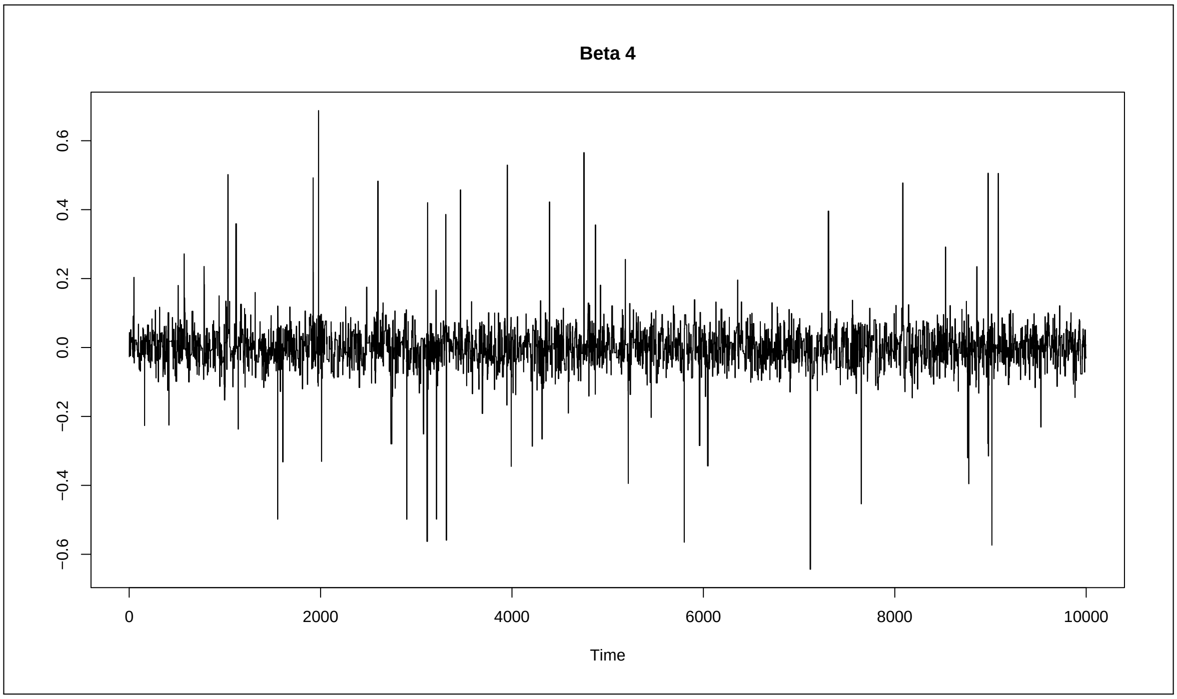

Beta 1
ACF
0.0
0.2
0.4
0.6
0.8
1.0
0
20
40
60
80
100
Lag

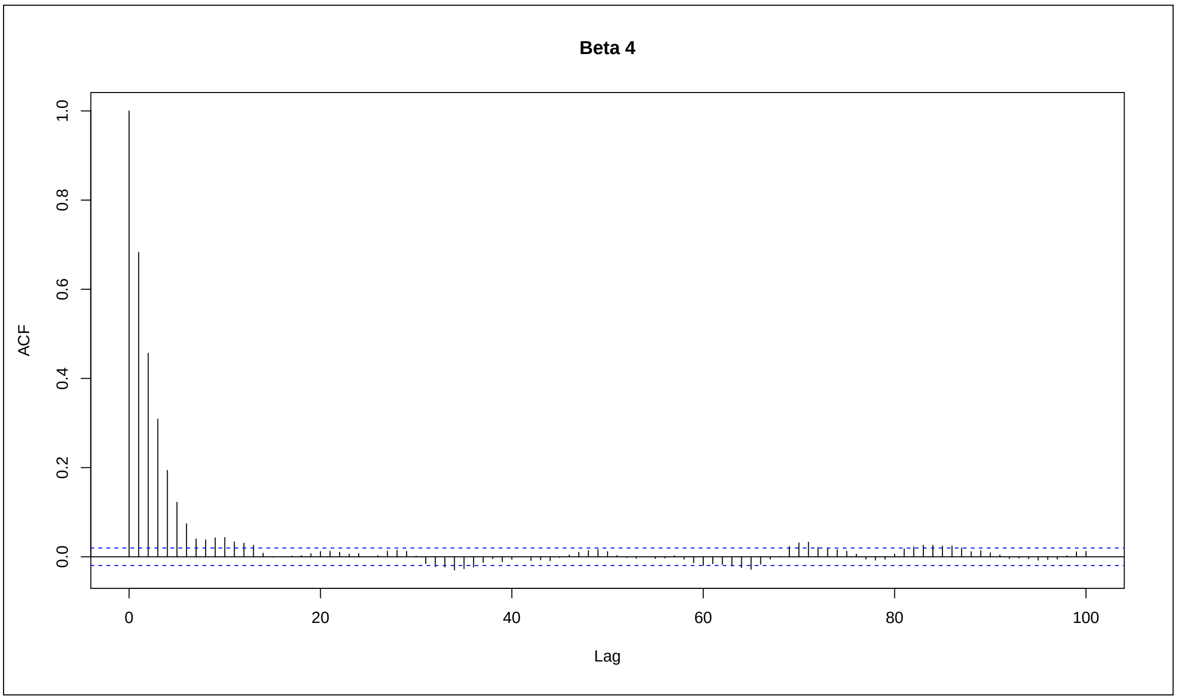

Beta 2
−2.0
−1.5
−1.0
−0.5
0
2000
4000
6000
8000
10000
Time

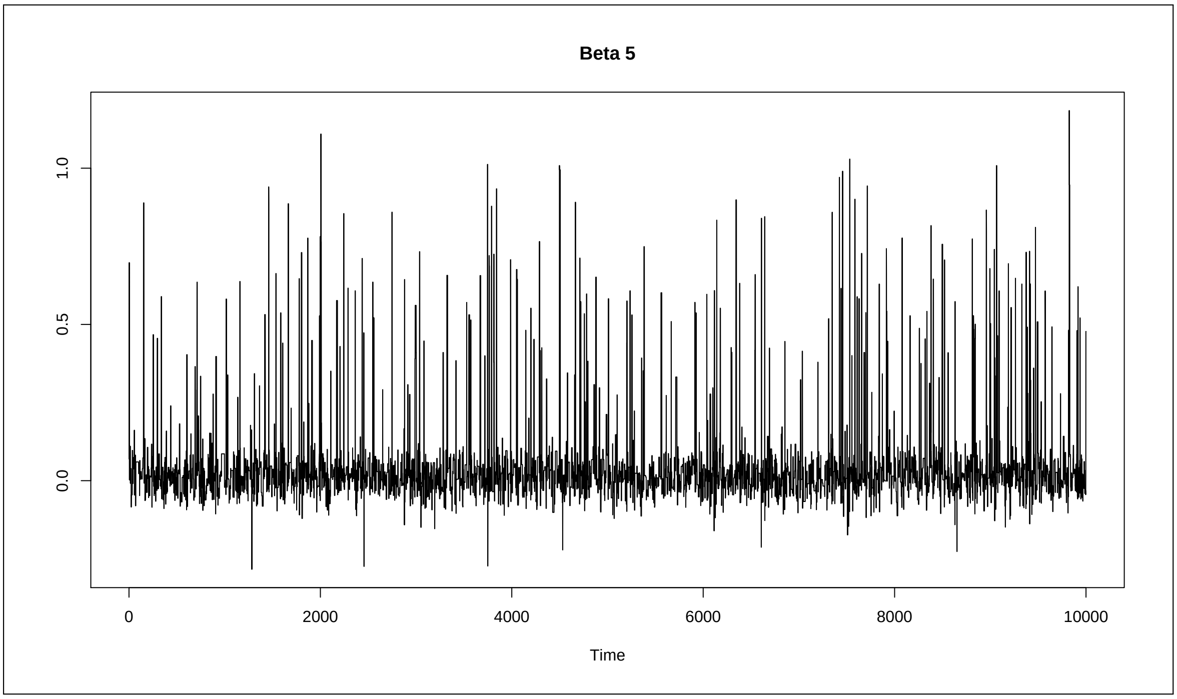

Beta 2
ACF
0.0
0.2
0.4
0.6
0.8
1.0
0
20
40
60
80
100
Lag

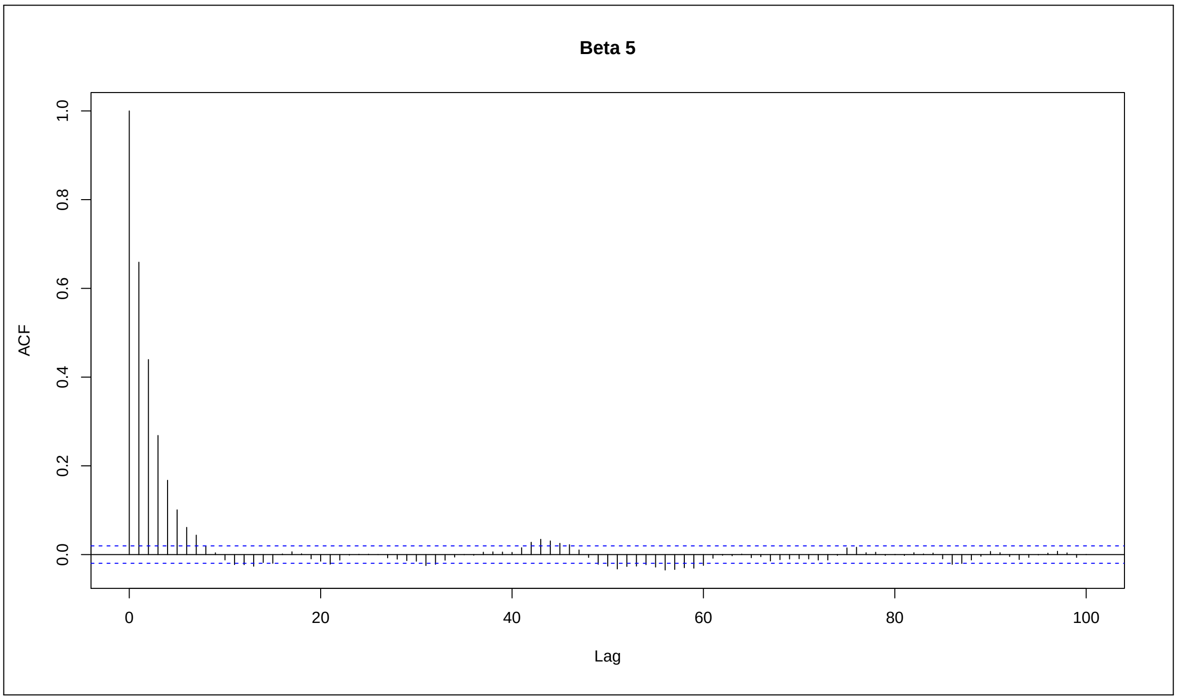

Beta 3
−0.5
0.0
0.5
0
2000
4000
6000
8000
10000
Time

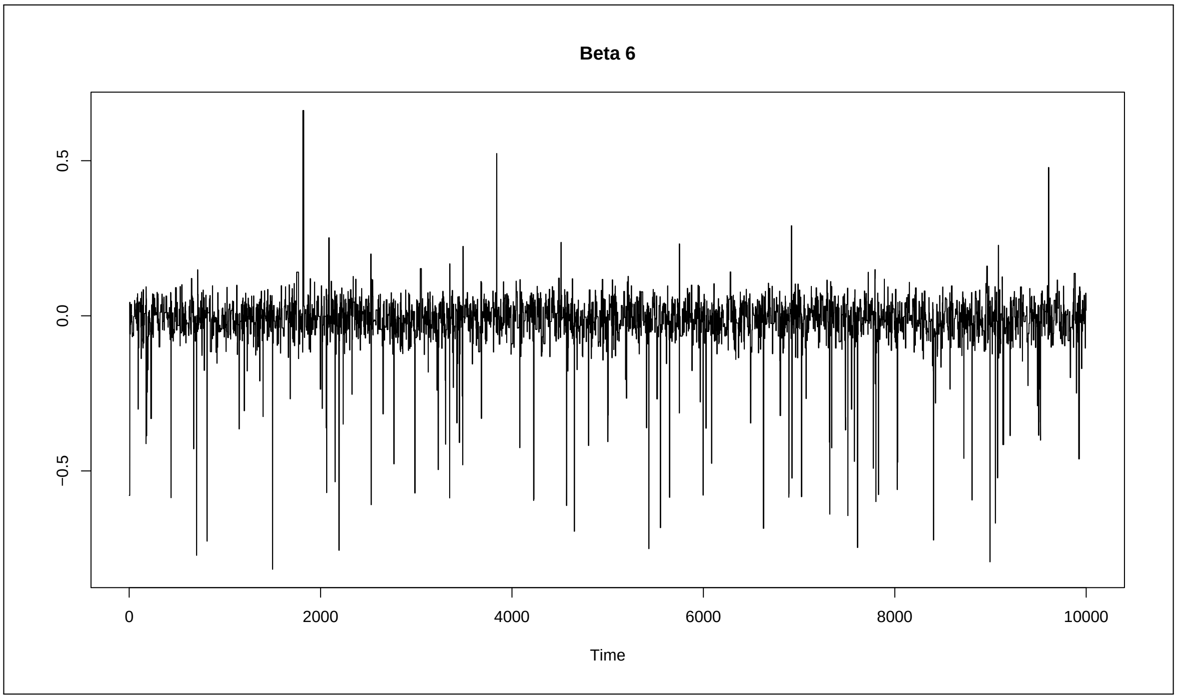

Beta 3
ACF
1.0
0.8
0.6
0.4
0.2
0.0
0
20
40
60
80
100
Lag

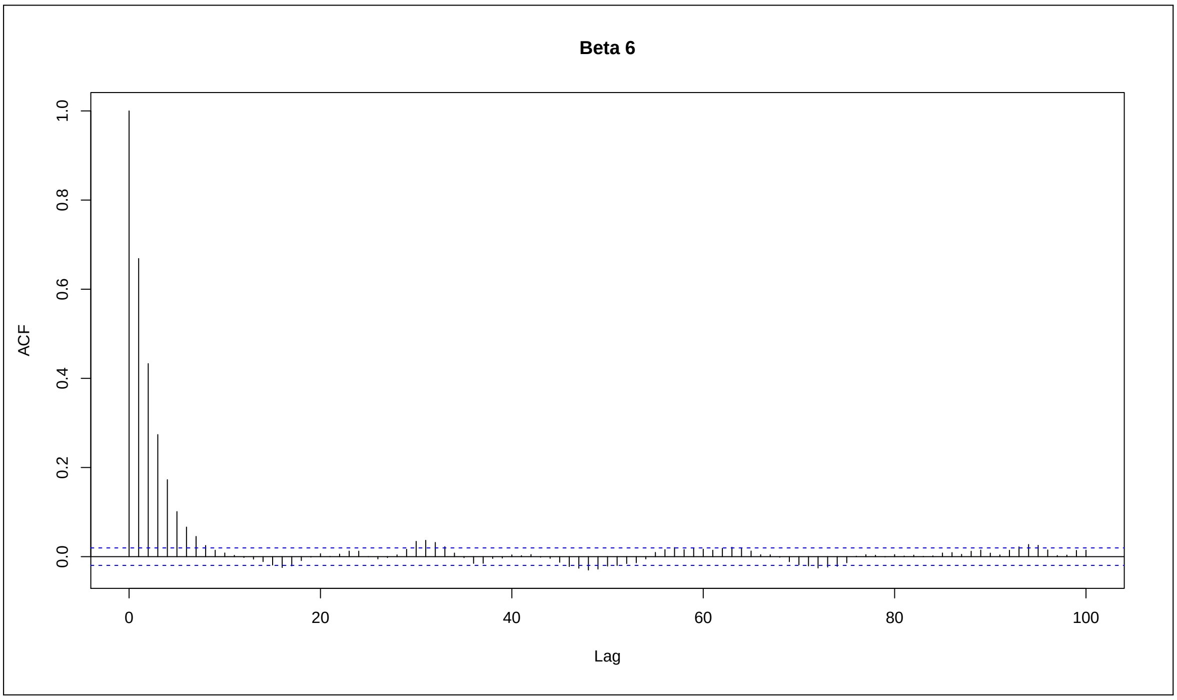

Beta 4
0.6
0.4
0.2
0.0
−0.2
−0.4
−0.6
0
2000
4000
6000
8000
10000
Time

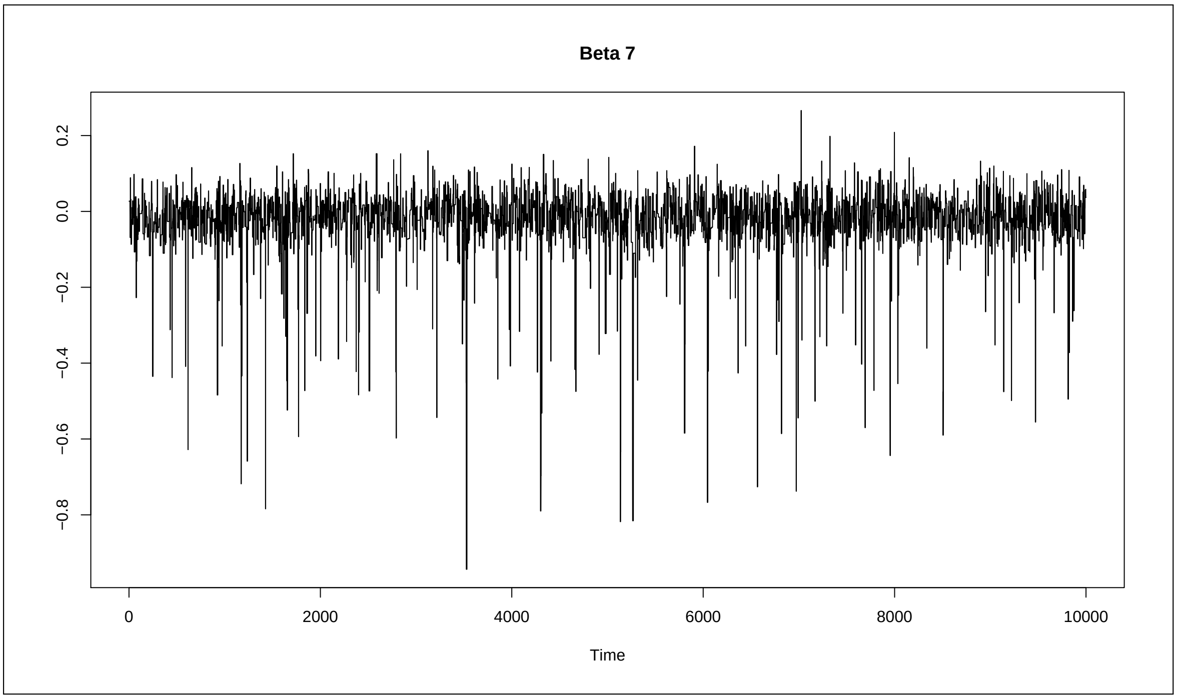

Beta 4
ACF
1.0
0.8
0.6
0.4
0.2
0.0
0
20
40
60
80
100
Lag

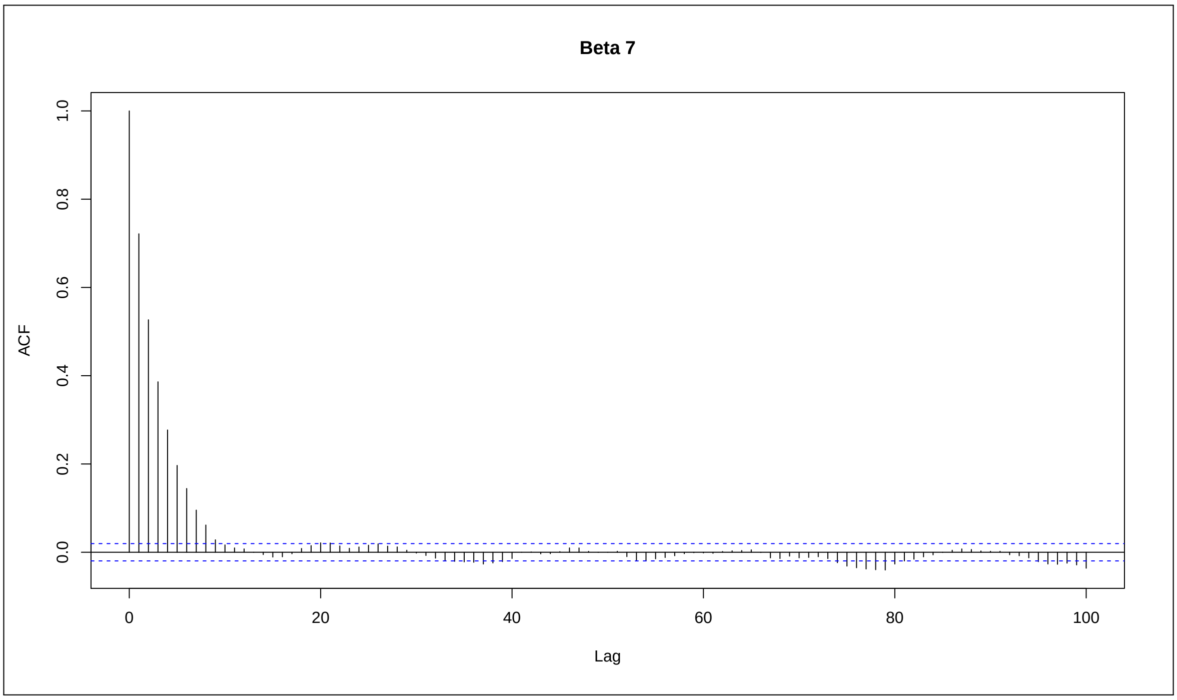

Beta 5
1.0
0.5
0.0
0
2000
4000
6000
8000
10000
Time

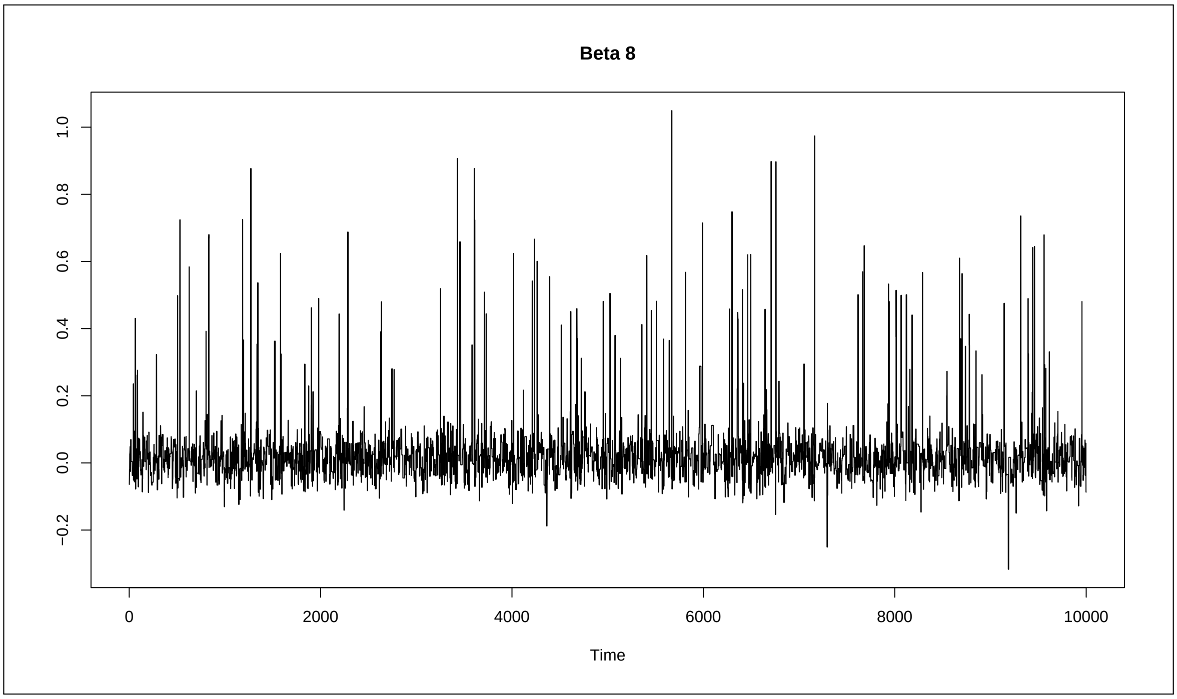

Beta 5
ACF
1.0
0.8
0.6
0.4
0.2
0.0
0
20
40
60
80
100
Lag

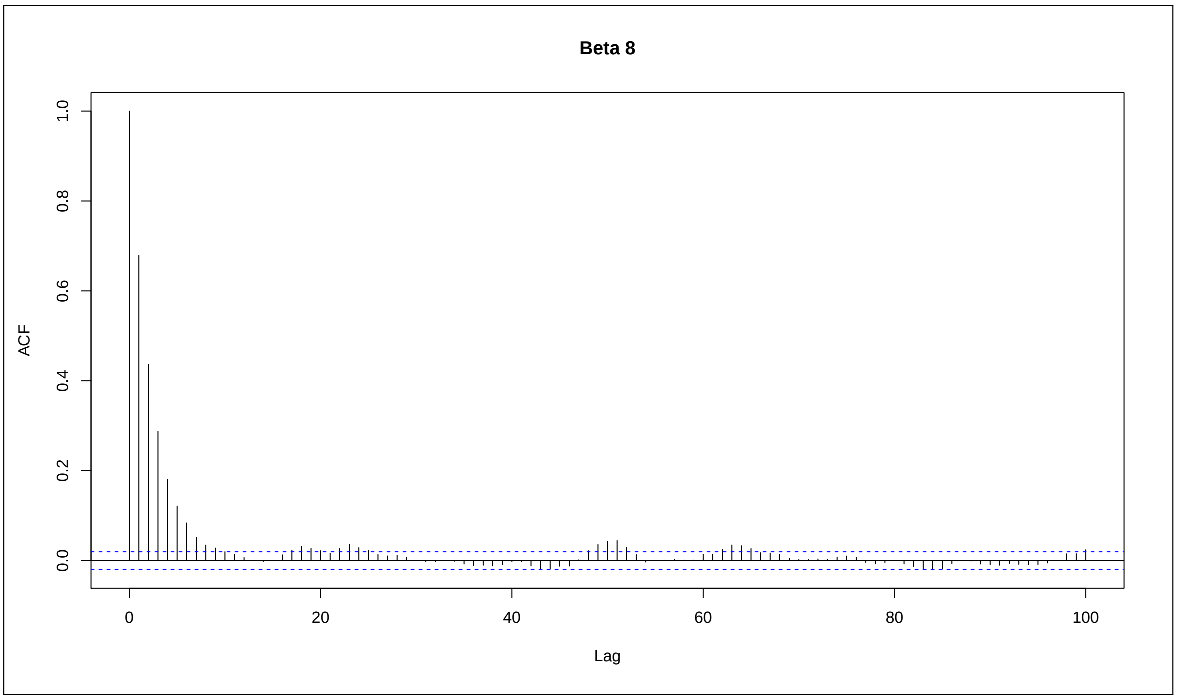

Beta 6
0.5
0.0
−0.5
0
2000
4000
6000
8000
10000
Time

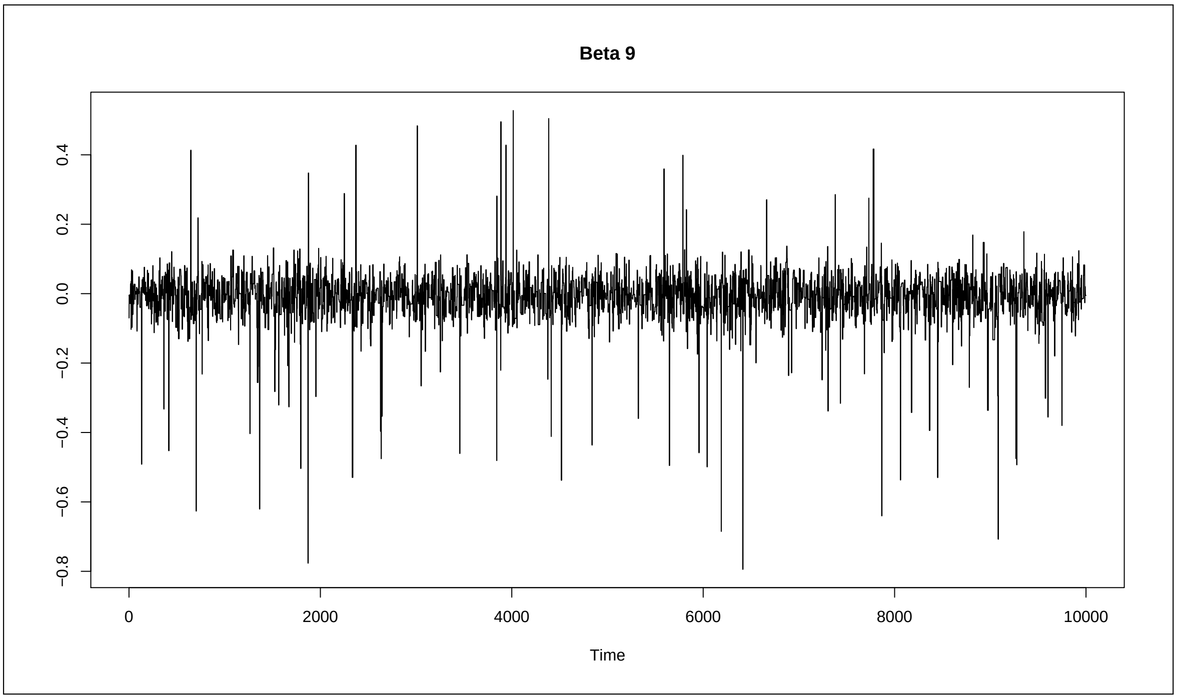

Beta 6
ACF
1.0
0.8
0.6
0.4
0.2
0.0
0
20
40
60
80
100
Lag

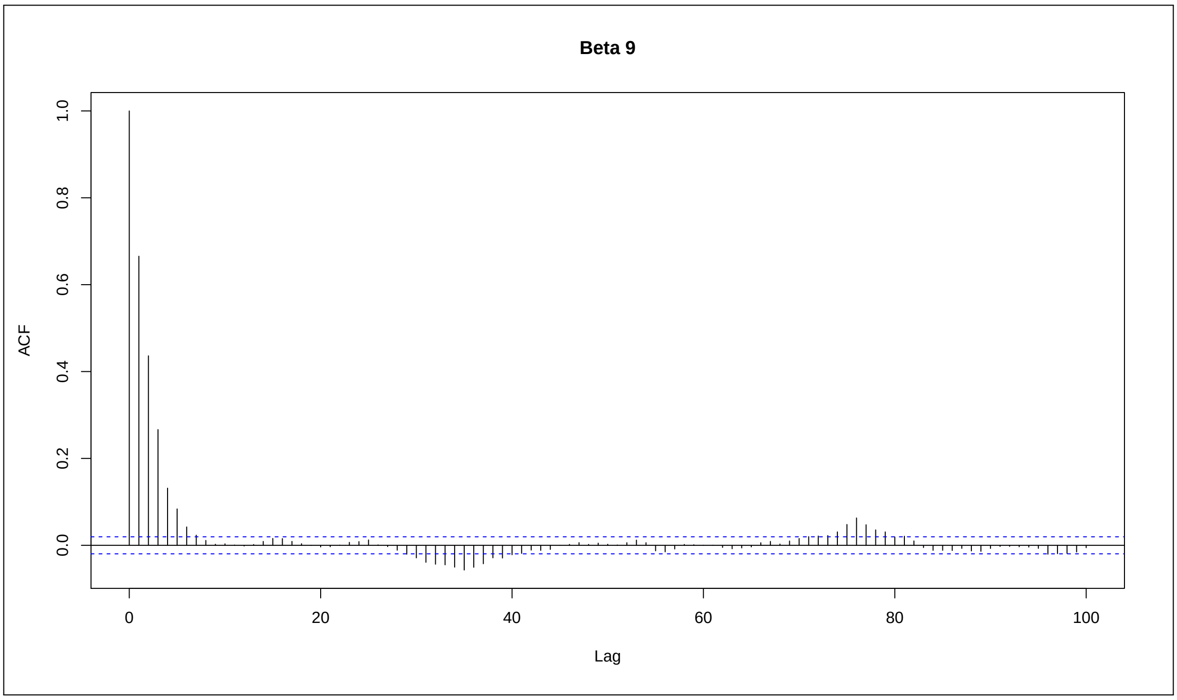

Beta 7
0.2
0.0
−0.2
−0.4
−0.6
−0.8
0
2000
4000
6000
8000
10000
Time

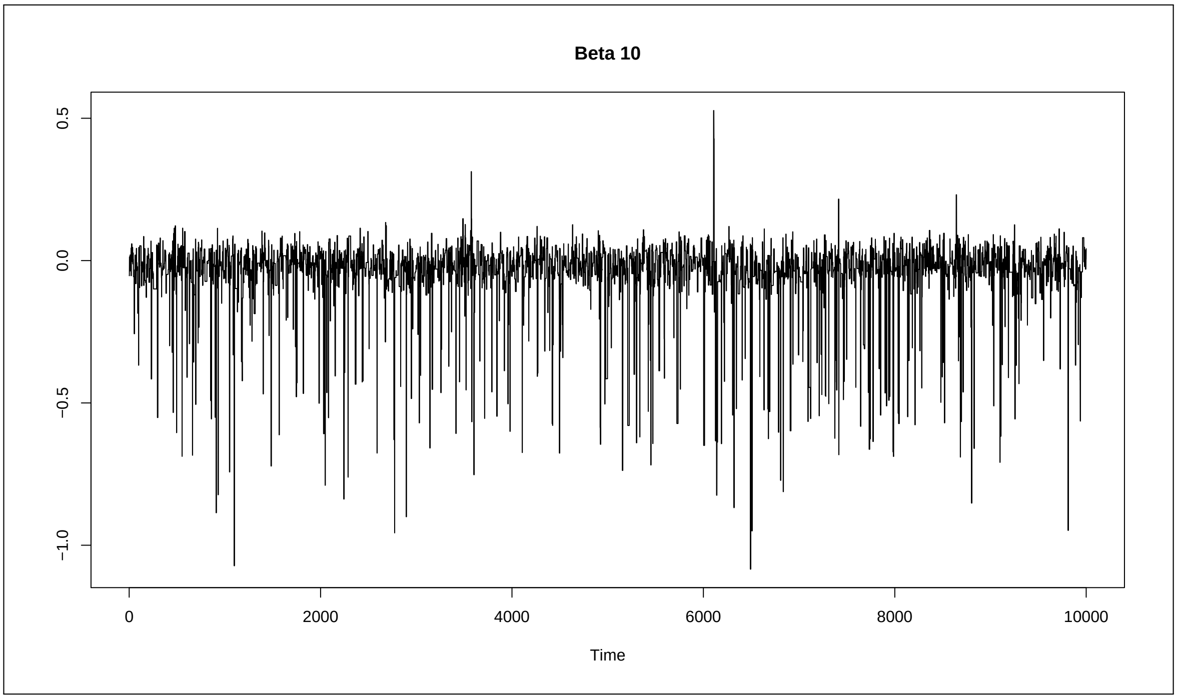

Beta 7
ACF
1.0
0.8
0.6
0.4
0.2
0.0
0
20
40
60
80
100
Lag

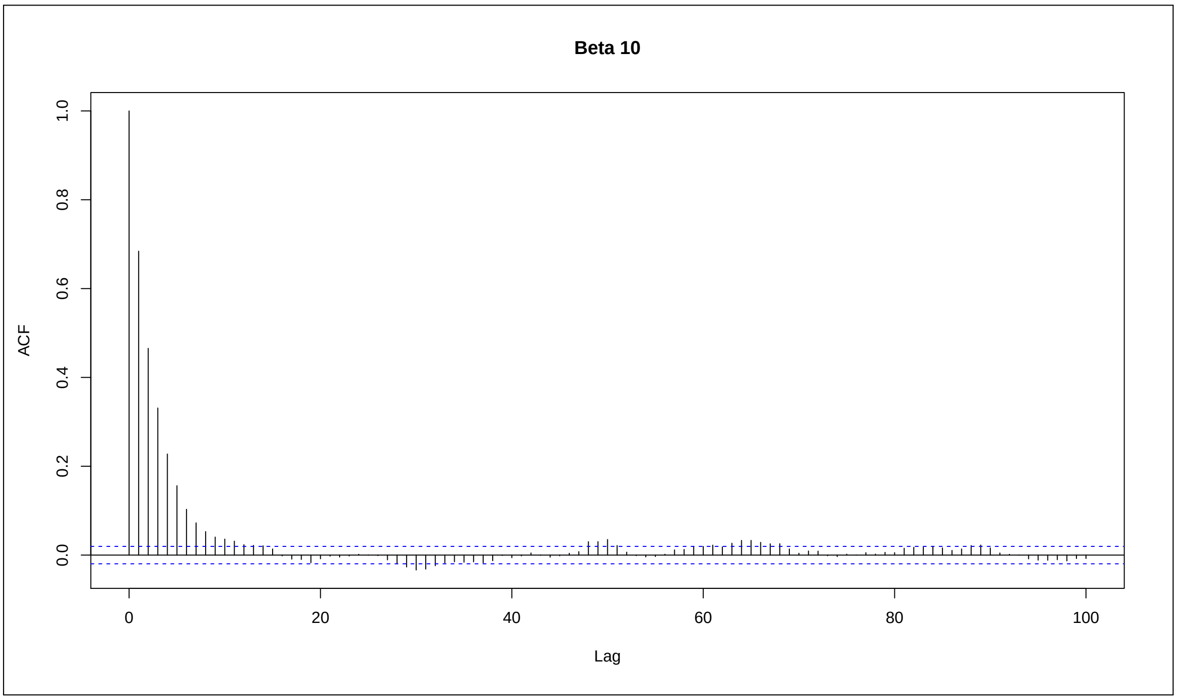

Beta 8
1.0
0.8
0.6
0.4
0.2
0.0
−0.2
0
2000
4000
6000
8000
10000
Time

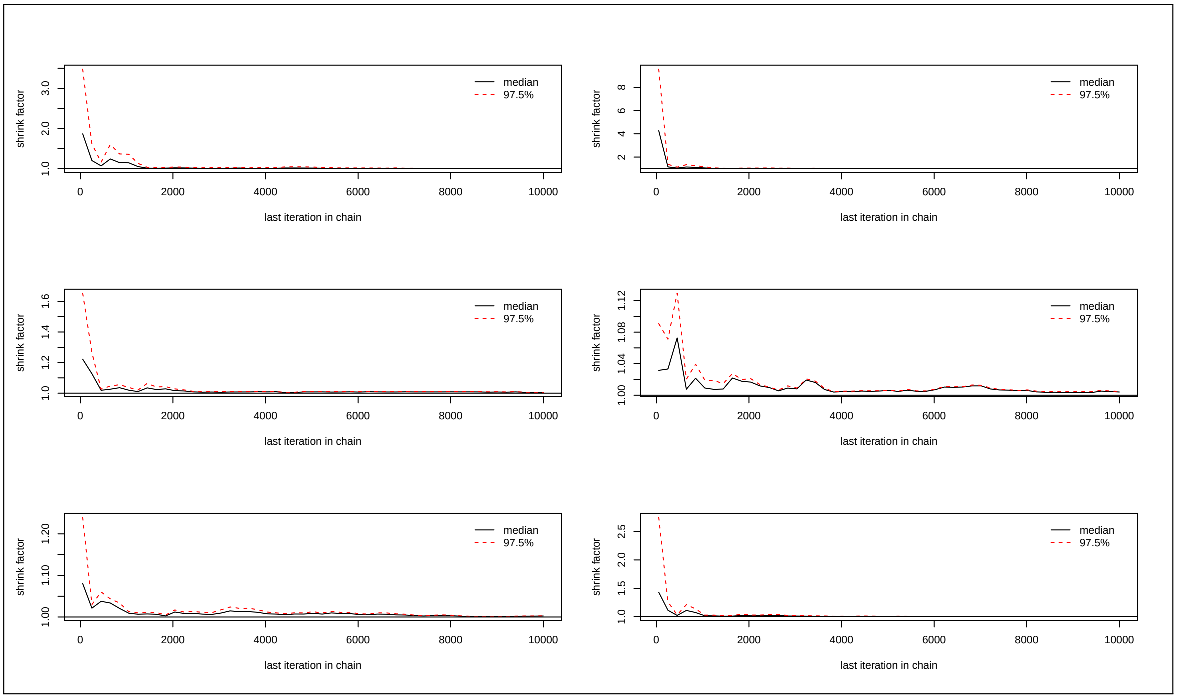

Beta 8
ACF
1.0
0.8
0.6
0.4
0.2
0.0
0
20
40
60
80
100
Lag

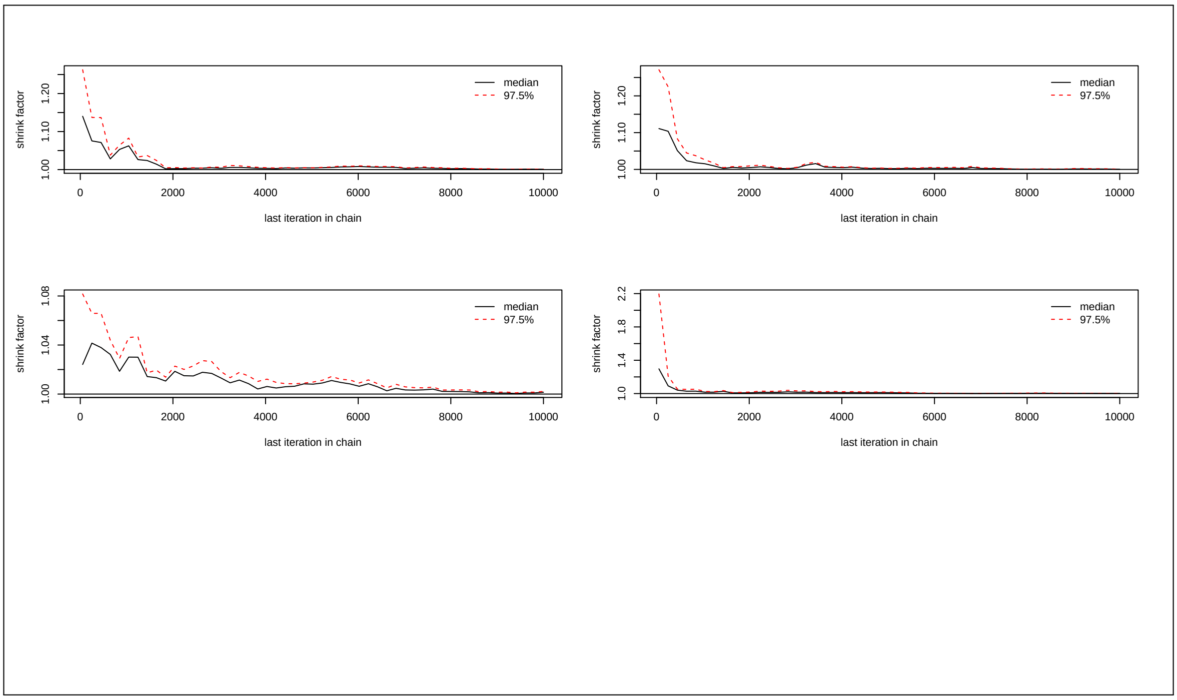

Beta 9
0.4
0.2
0.0
−0.2
−0.4
−0.6
−0.8
0
2000
4000
6000
8000
10000
Time


Beta 9
ACF
Lag
0.0
0.2
0.4
0.6
0.8
1.0
0
20
40
60
80
100


Beta 10
Time
−1.0
−0.5
0.0
0.5
0
2000
4000
6000
8000
10000


Beta 10
ACF
0.0
0.2
0.4
0.6
0.8
1.0
0
20
40
60
80
100
Lag

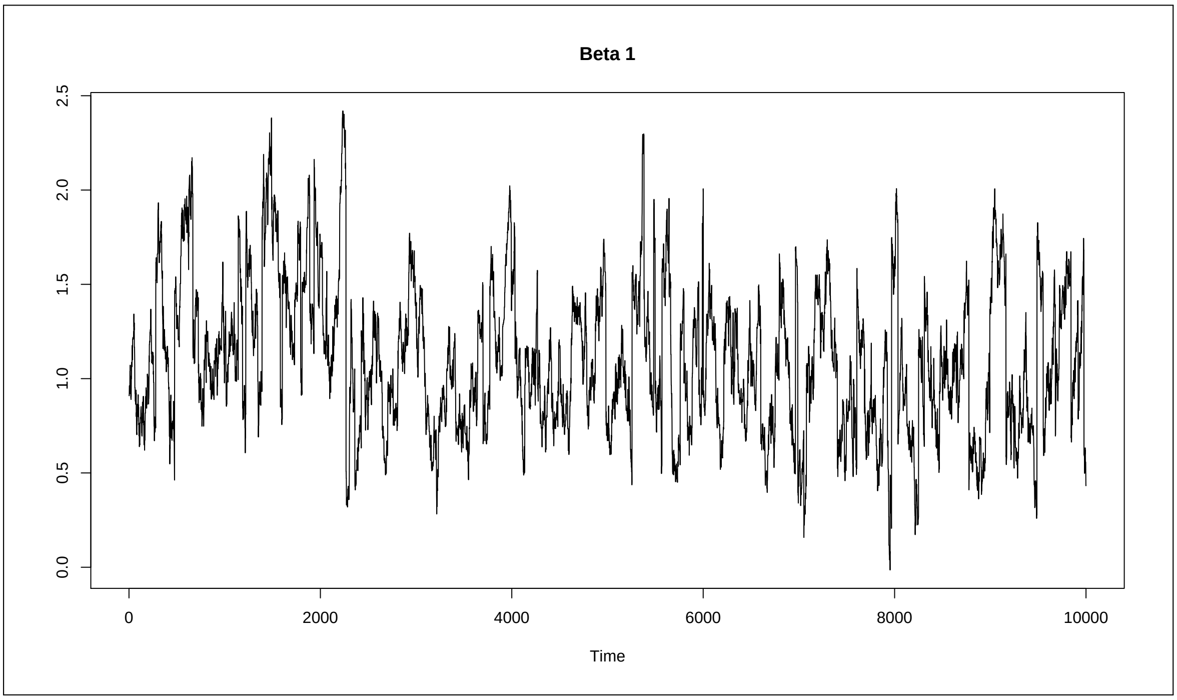

shrink factor
1.0
2.0
3.0
median
97.5%
0
2000
4000
6000
8000
10000
last iteration in chain
shrink factor
2
4
6
8
median
97.5%
last iteration in chain
shrink factor
1.0
1.2
1.4
1.6
median
97.5%
last iteration in chain
shrink factor
1.00
1.04
1.08
1.12
median
97.5%
last iteration in chain
shrink factor
1.00
1.10
1.20
median
97.5%
last iteration in chain
shrink factor
1.0
1.5
2.0
2.5
median
97.5%
last iteration in chain

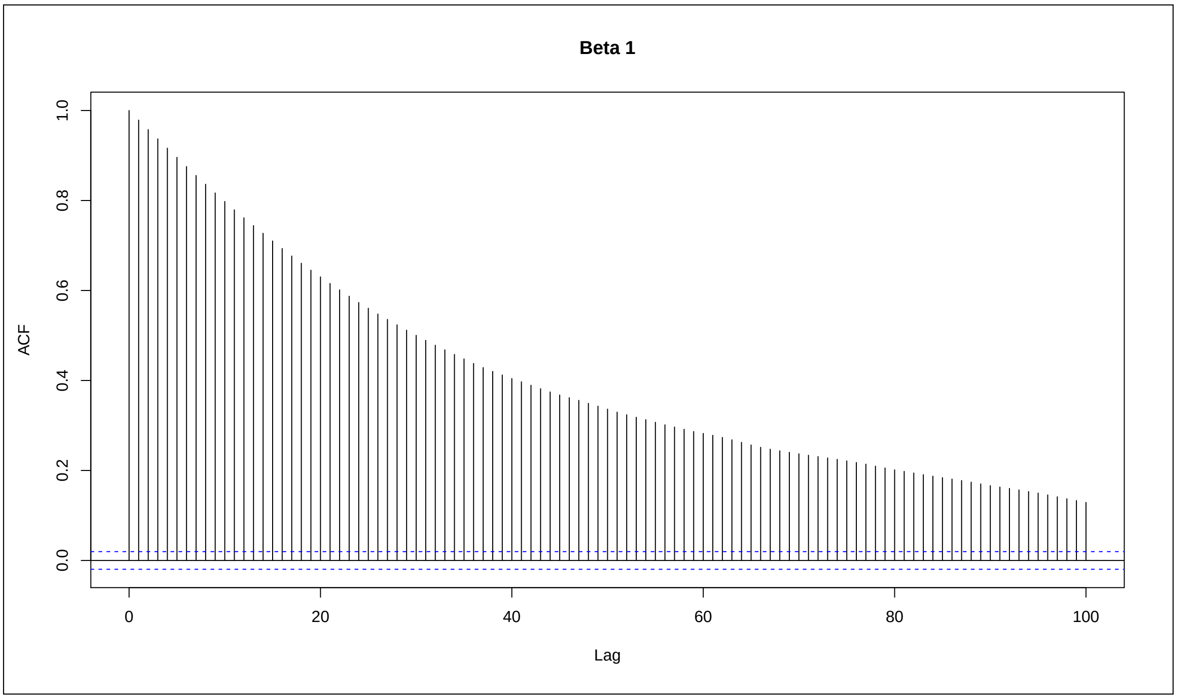


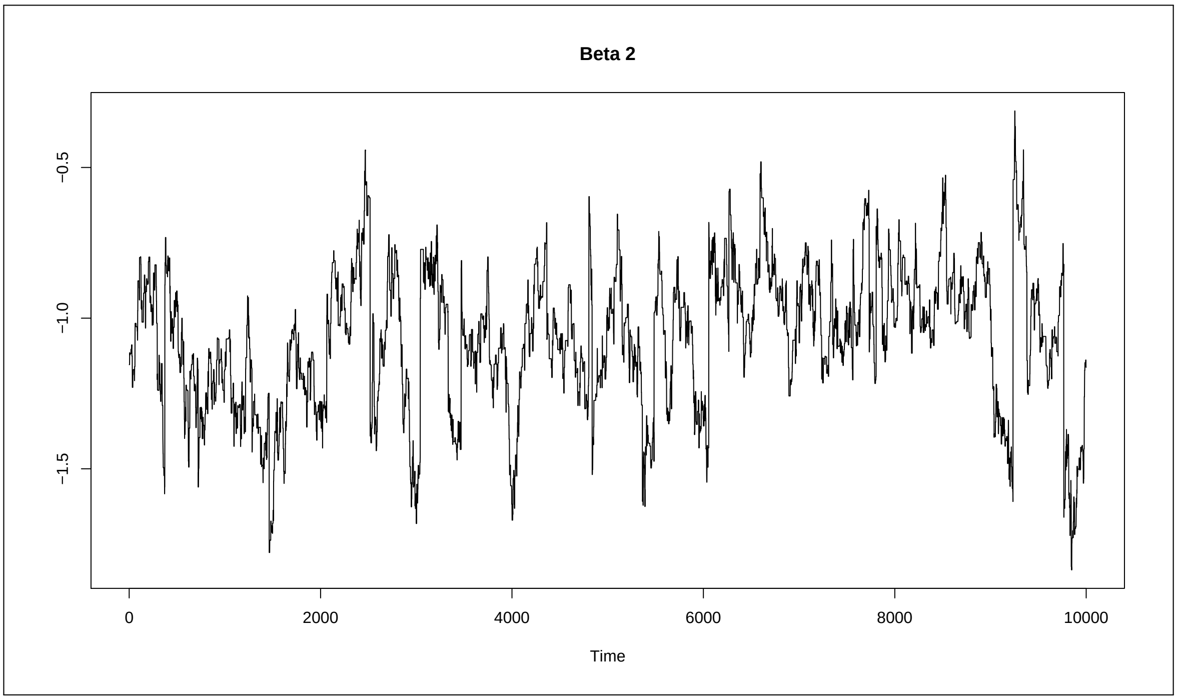
A.5 Single component updating with tri-modal random walk sampler

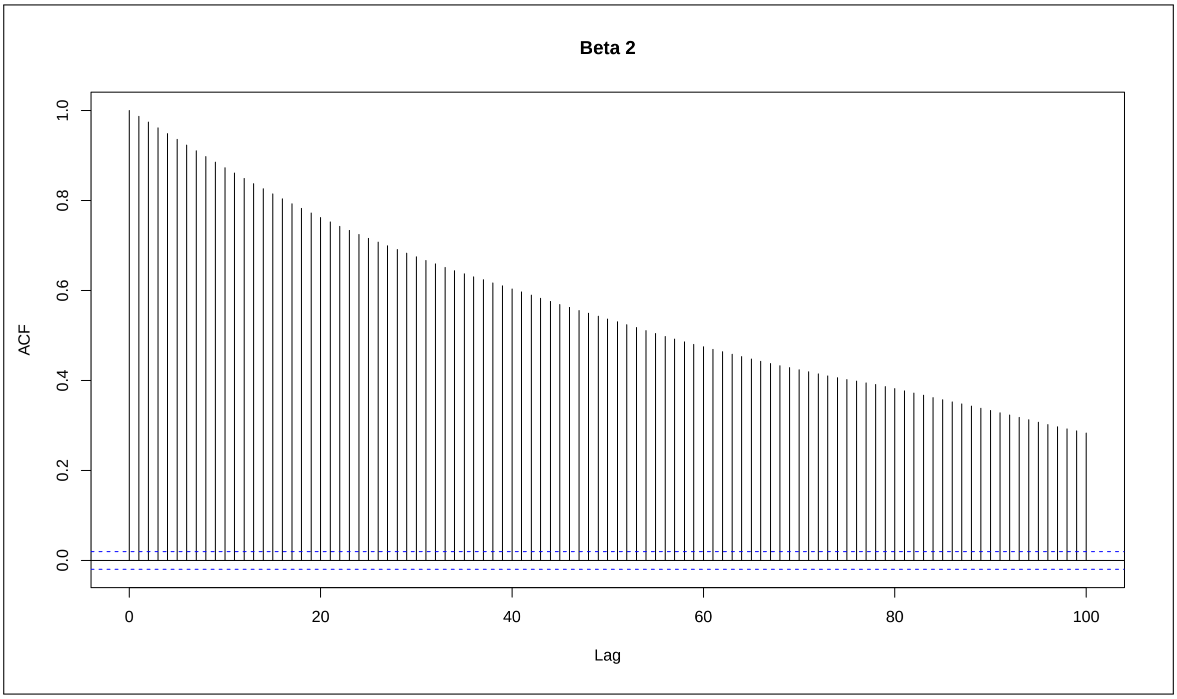
Parameter	G-R Point est.	G-R Upper C.I.	Acceptance rate
β_1	1.0168	1.0433	0.5805
β_2	1.0397	1.0915	0.2839
β_3	1.0138	1.0253	0.4103
β_4	1.0483	1.0623	0.4131
β_5	1.0919	1.2133	0.4296
β_6	1.1573	1.2884	0.4088
β_7	1.0662	1.1006	0.4119
β_8	1.0986	1.1971	0.4231
β_9	1.0711	1.1205	0.4139
β_{10}	1.0214	1.0340	0.4066

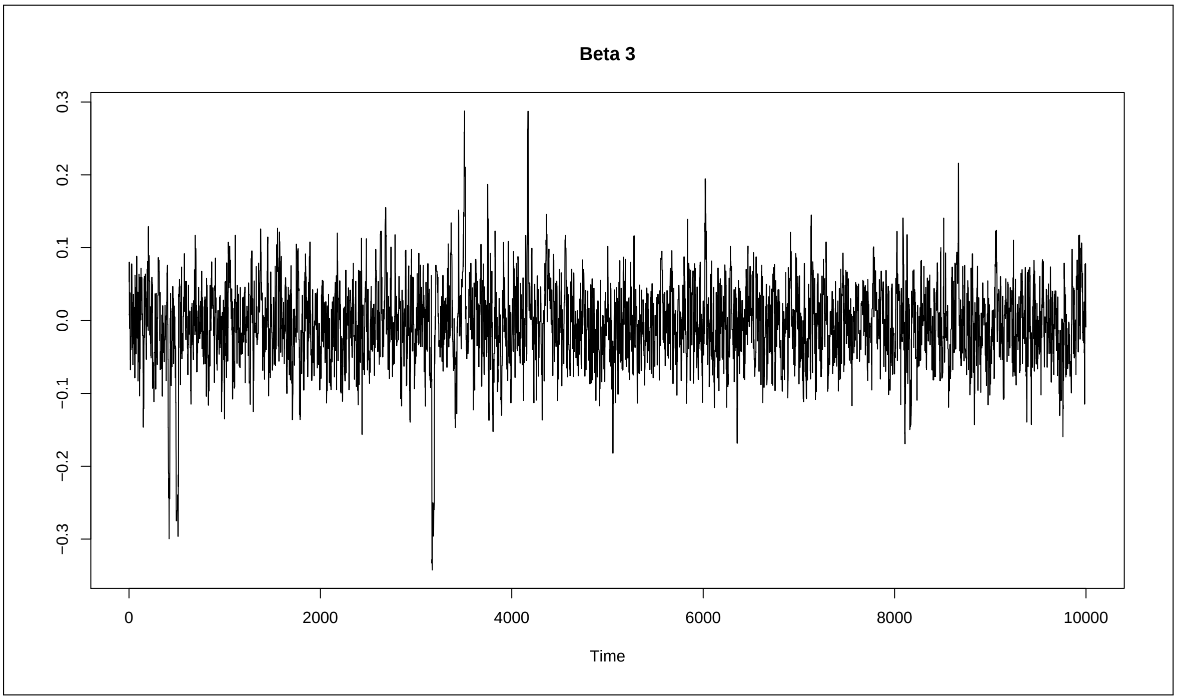
Table A.5: Point estimates of potential scale reduction factor, corresponding upper confidence limits and acceptance rates for the ten logistic slope parameters.

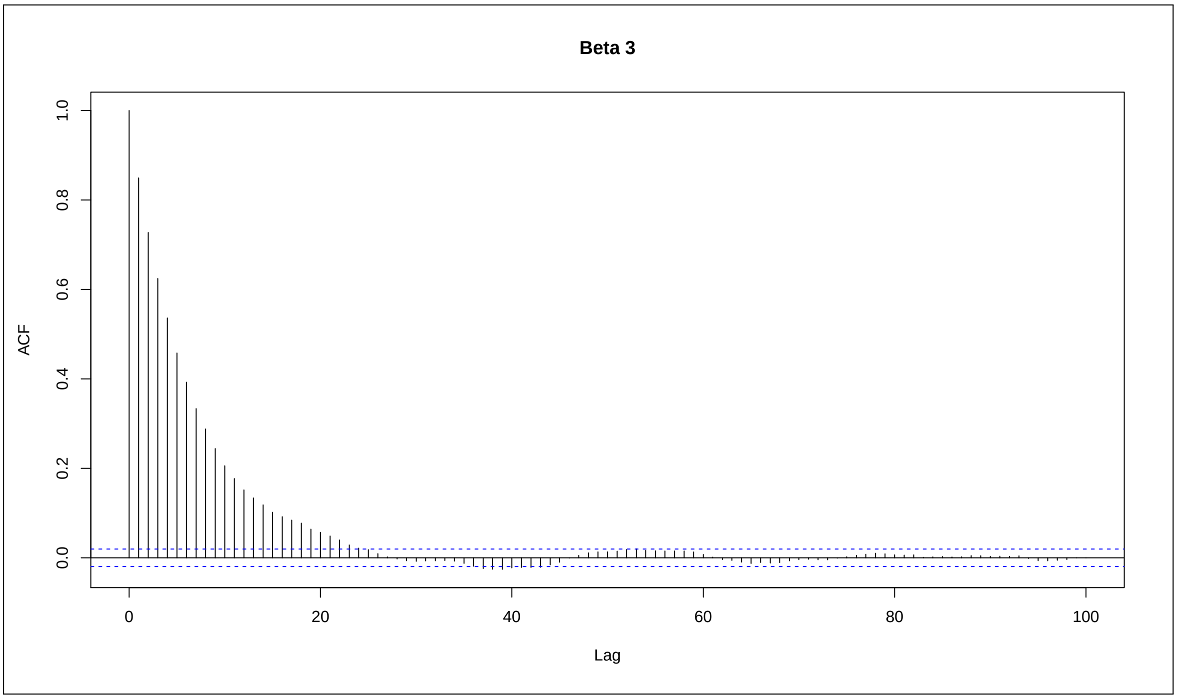

Beta 1
2.5
2.0
1.5
1.0
0.5
0.0
0
2000
4000
6000
8000
10000
Time

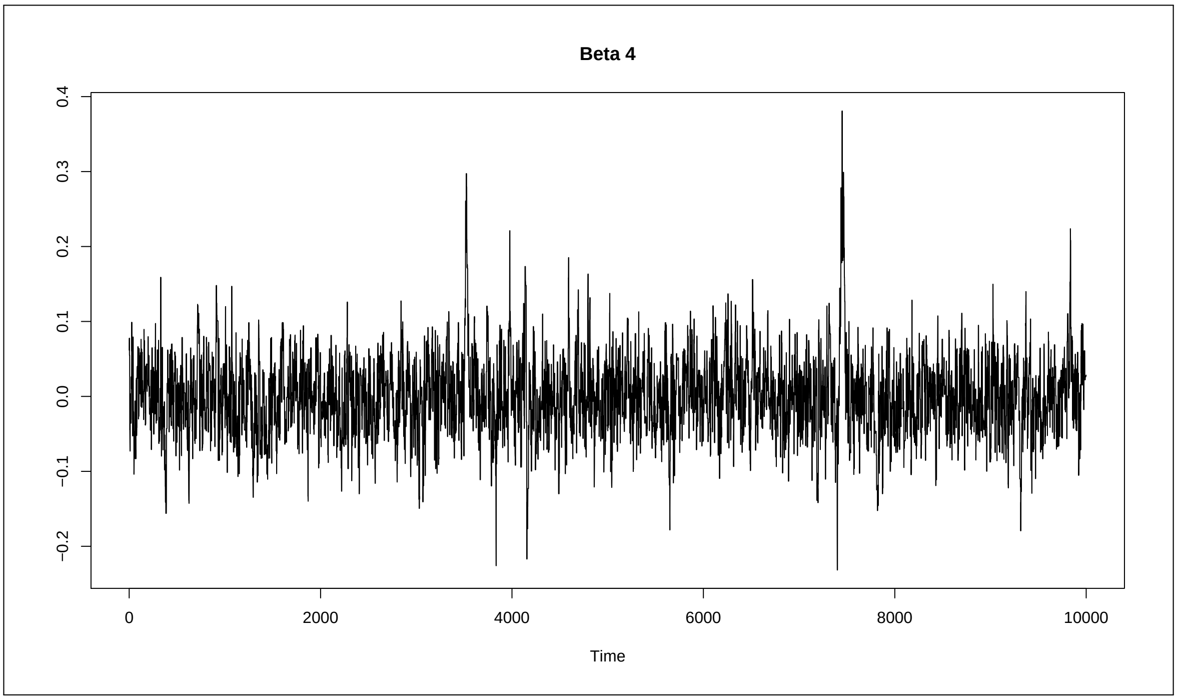

Beta 1
ACF
1.0
0.8
0.6
0.4
0.2
0.0
0
20
40
60
80
100
Lag

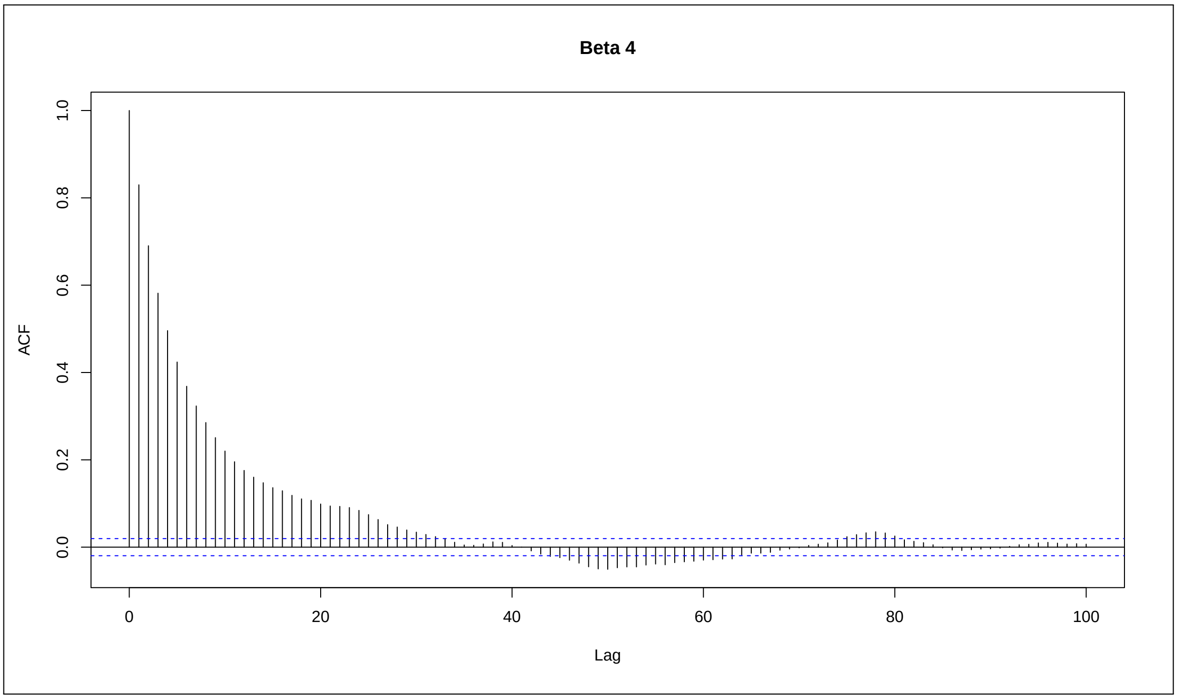

Beta 2
−0.5
−1.0
−1.5
0
2000
4000
6000
8000
10000
Time

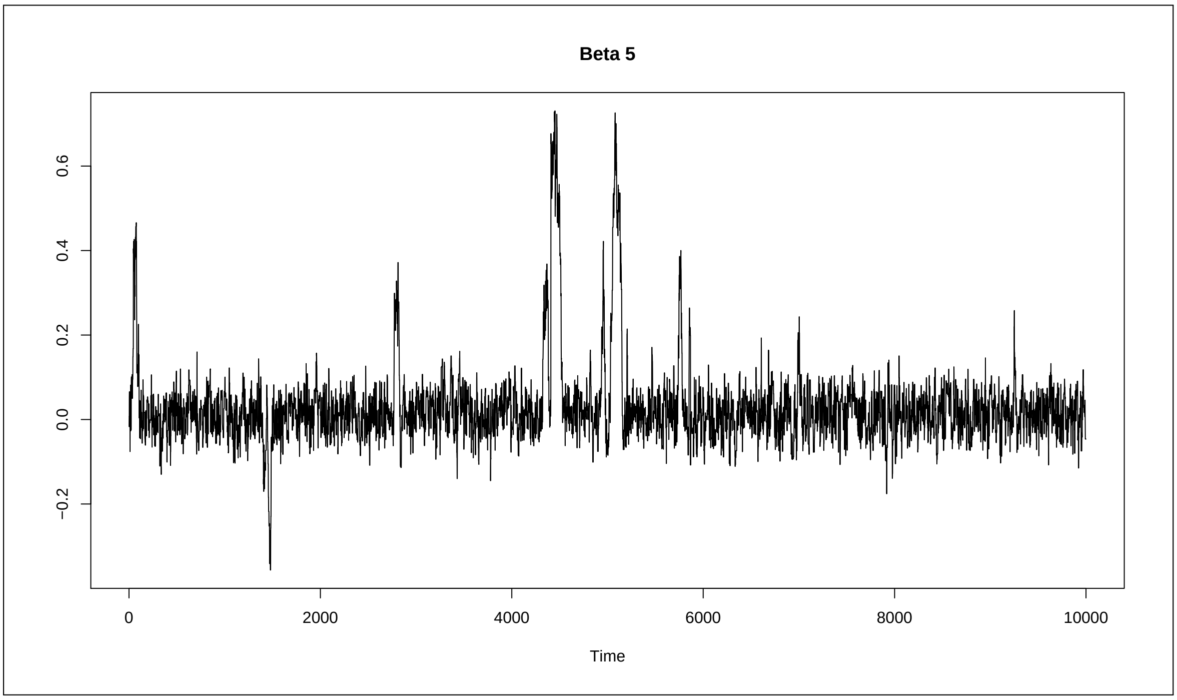

Beta 2
ACF
1.0
0.8
0.6
0.4
0.2
0.0
0
20
40
60
80
100
Lag

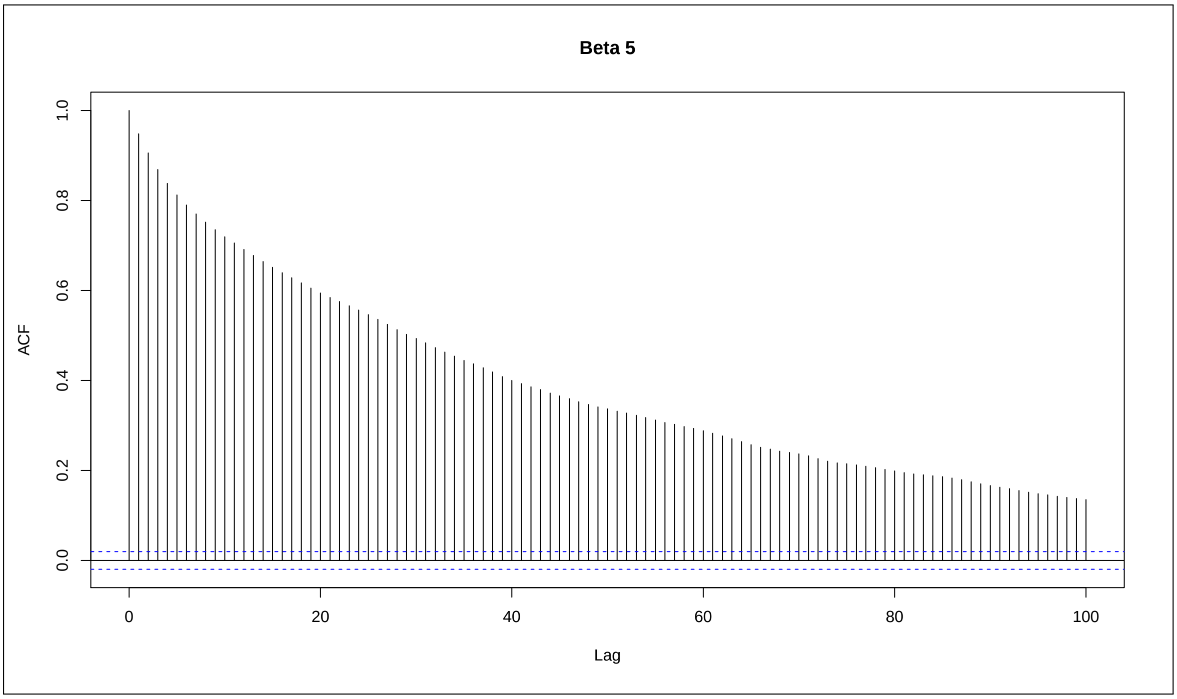

Beta 3
0.3
0.2
0.1
0.0
−0.1
−0.2
−0.3
0
2000
4000
6000
8000
10000
Time

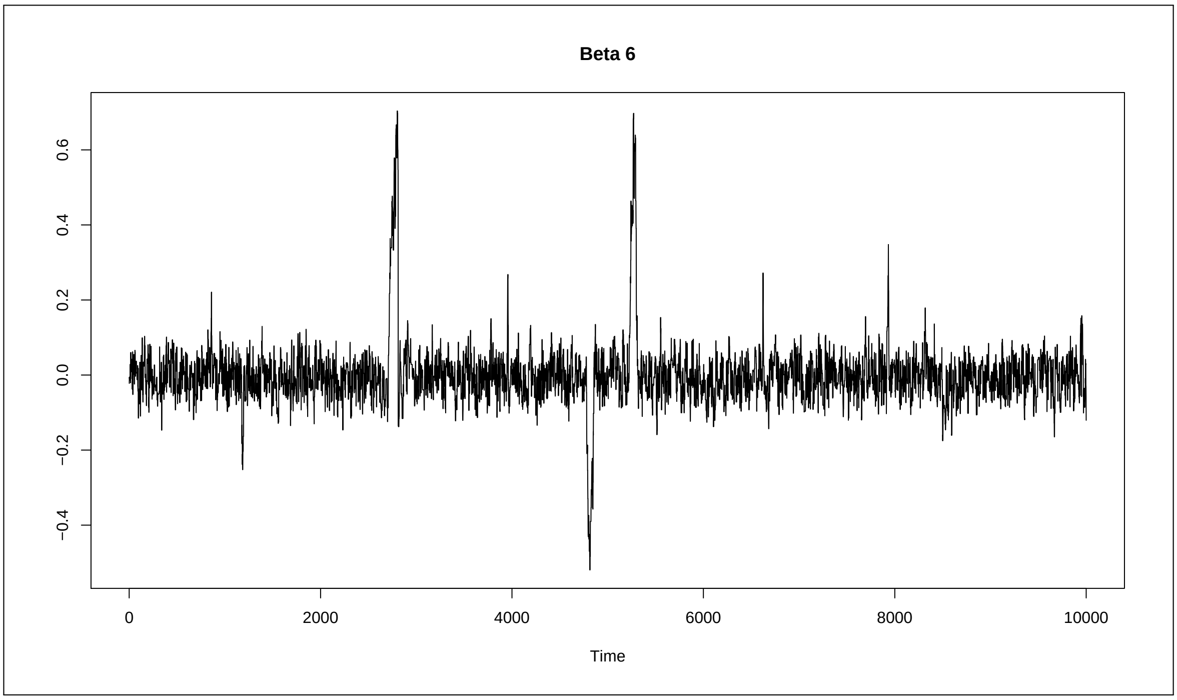

Beta 3
ACF
1.0
0.8
0.6
0.4
0.2
0.0
0
20
40
60
80
100
Lag

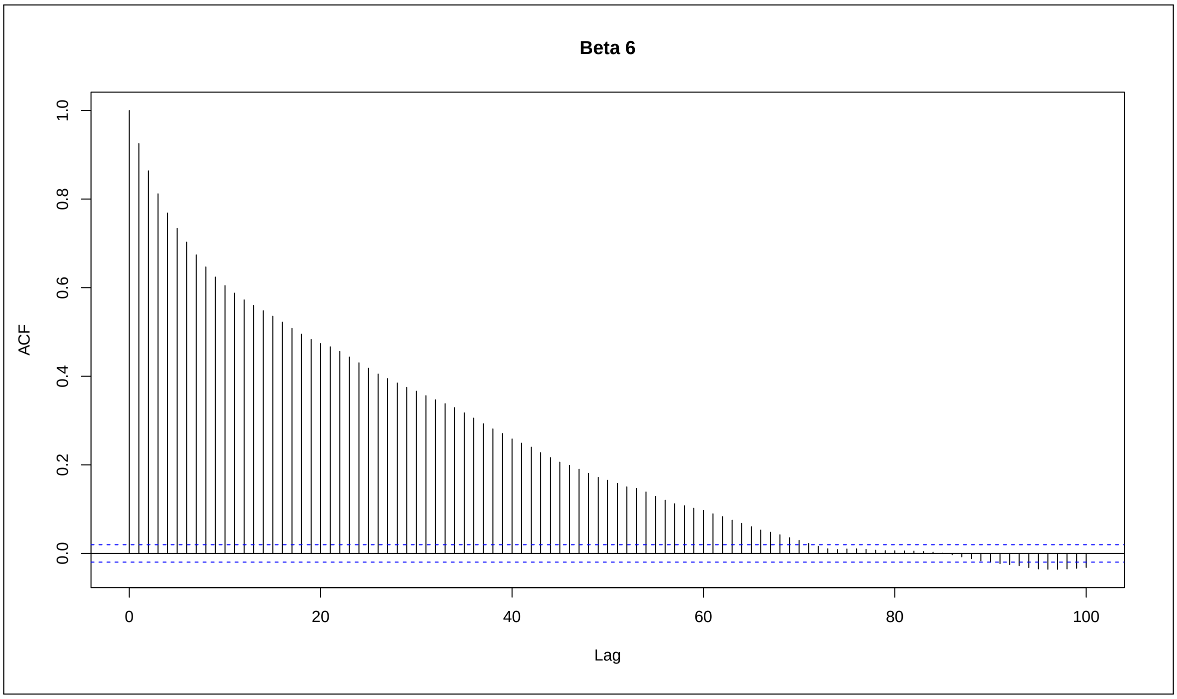

Beta 4
0.4
0.3
0.2
0.1
0.0
−0.1
−0.2
0
2000
4000
6000
8000
10000
Time

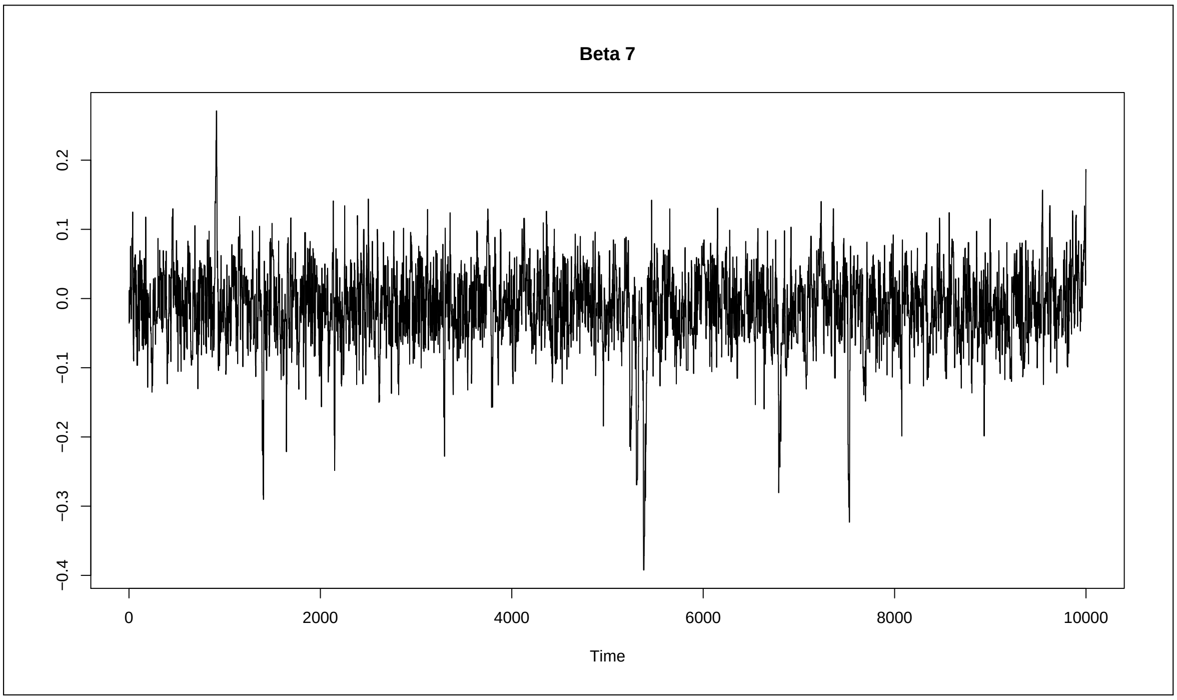

Beta 4
ACF
1.0
0.8
0.6
0.4
0.2
0.0
0
20
40
60
80
100
Lag

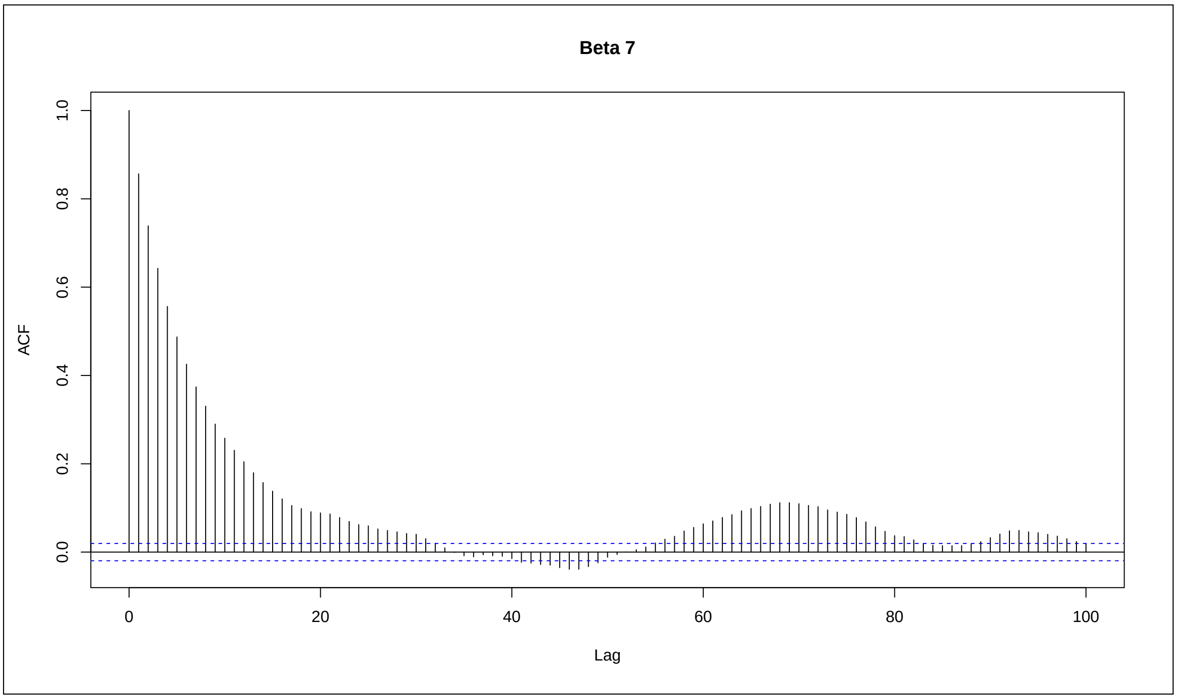

Beta 5
0.6
0.4
0.2
0.0
−0.2
0
2000
4000
6000
8000
10000
Time

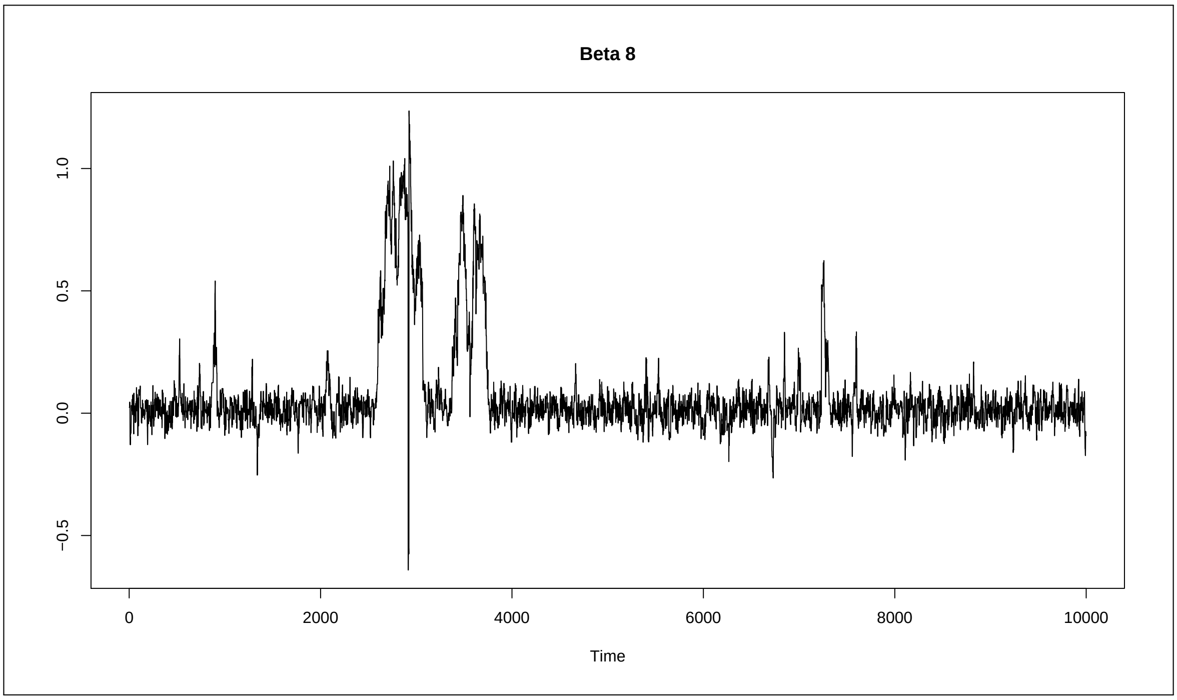

Beta 5
ACF
1.0
0.8
0.6
0.4
0.2
0.0
0
20
40
60
80
100
Lag

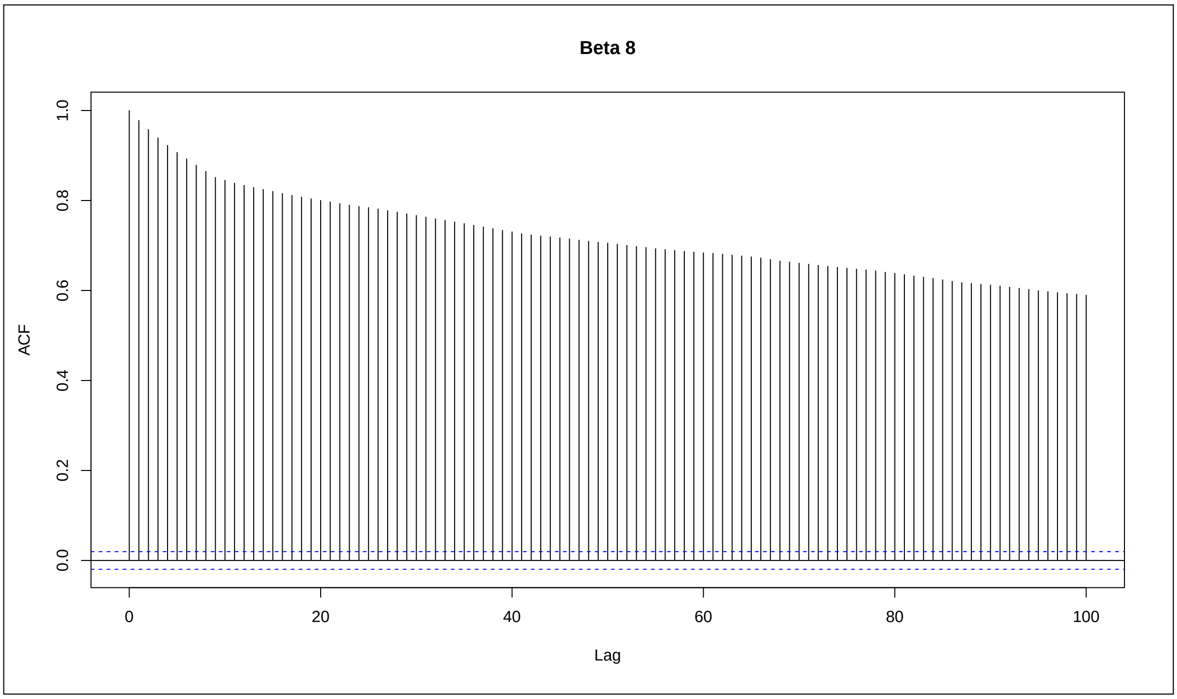

Beta 6
0.6
0.4
0.2
0.0
−0.2
−0.4
0
2000
4000
6000
8000
10000
Time

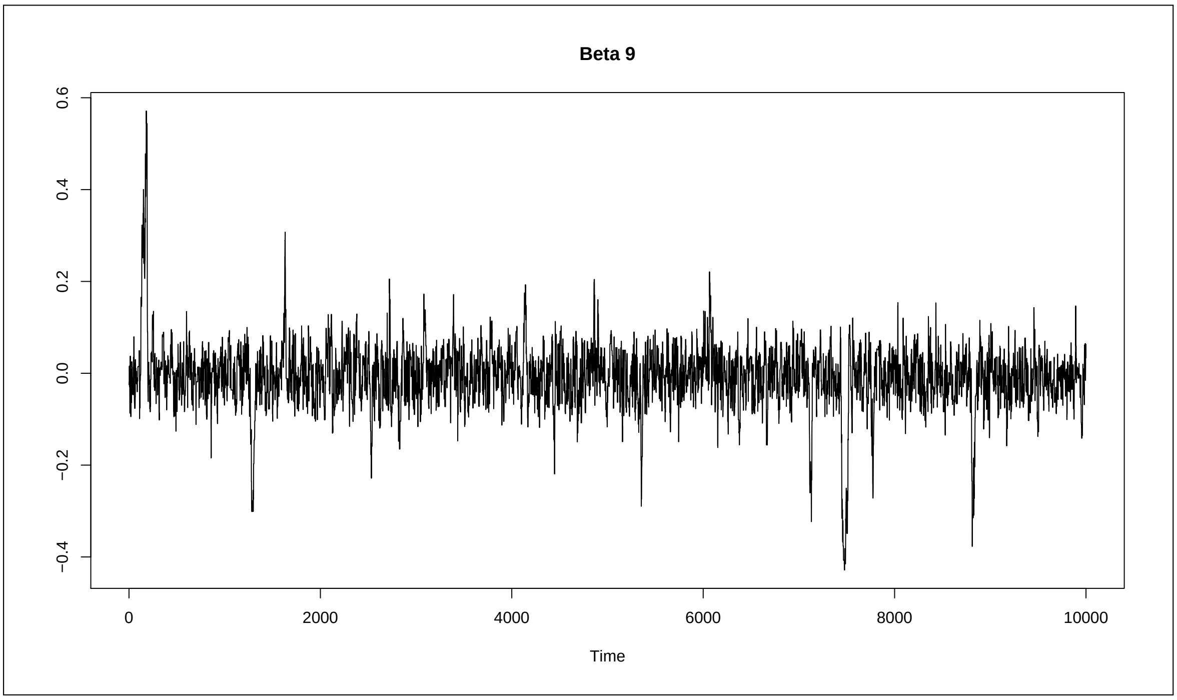

Beta 6
ACF
1.0
0.8
0.6
0.4
0.2
0.0
0
20
40
60
80
100
Lag

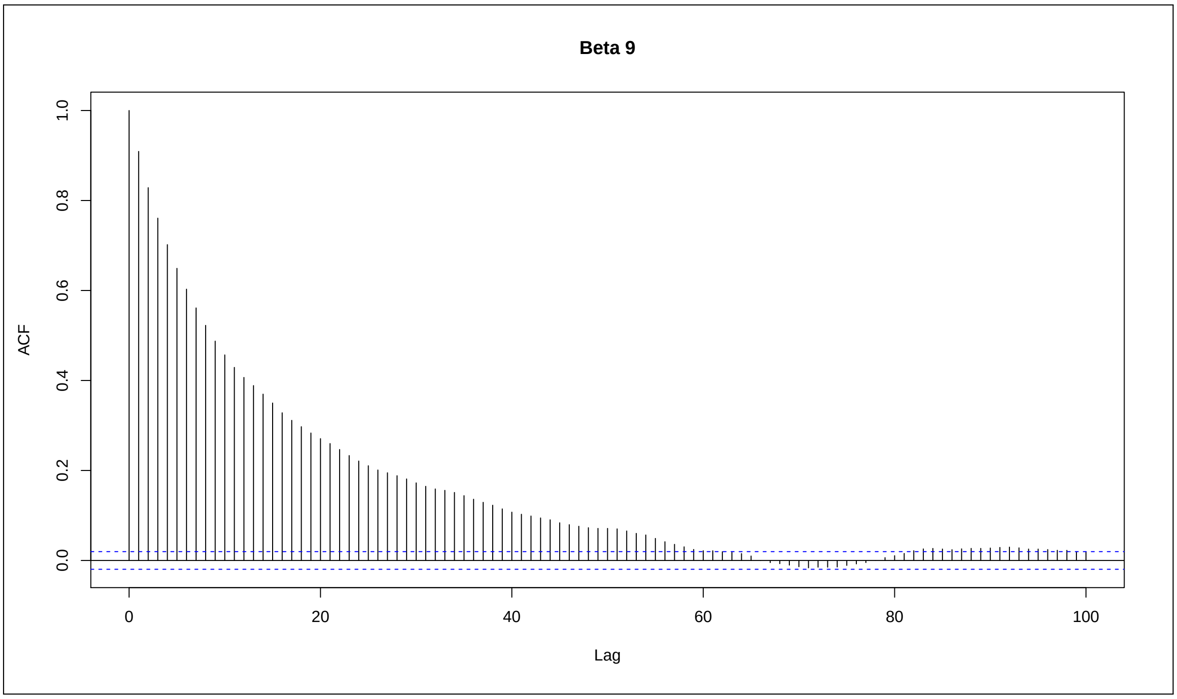

Beta 7
0.2
0.1
0.0
−0.1
−0.2
−0.3
−0.4
0
2000
4000
6000
8000
10000
Time

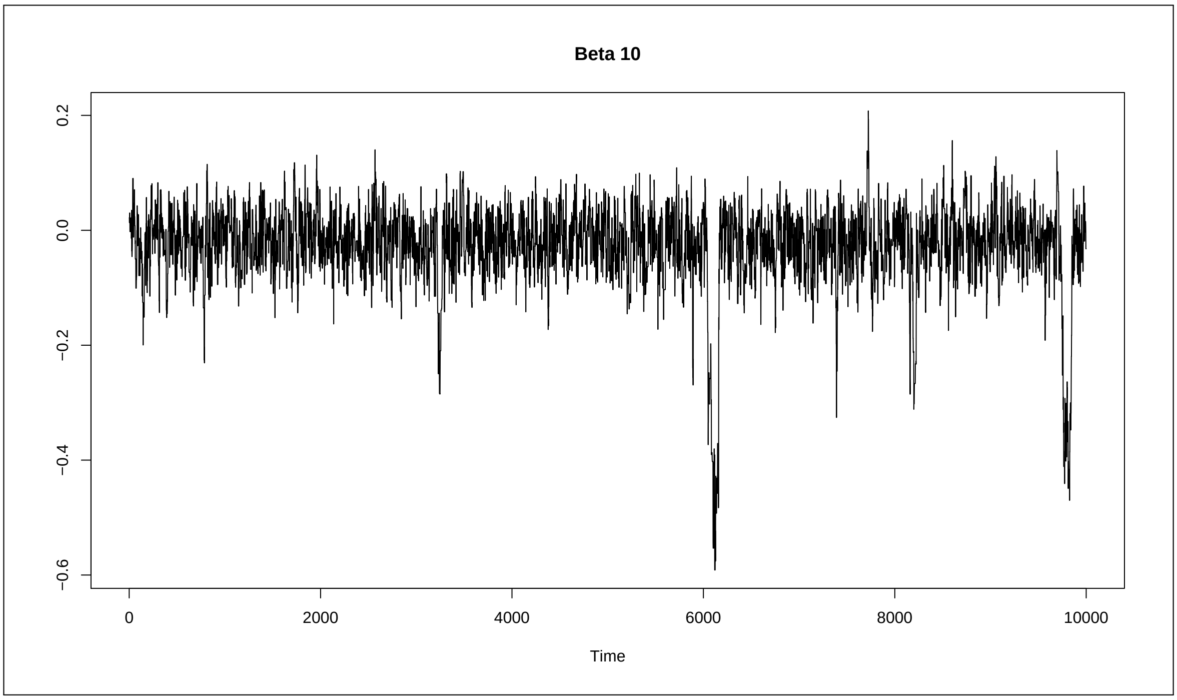

Beta 7
ACF
1.0
0.8
0.6
0.4
0.2
0.0
0
20
40
60
80
100
Lag

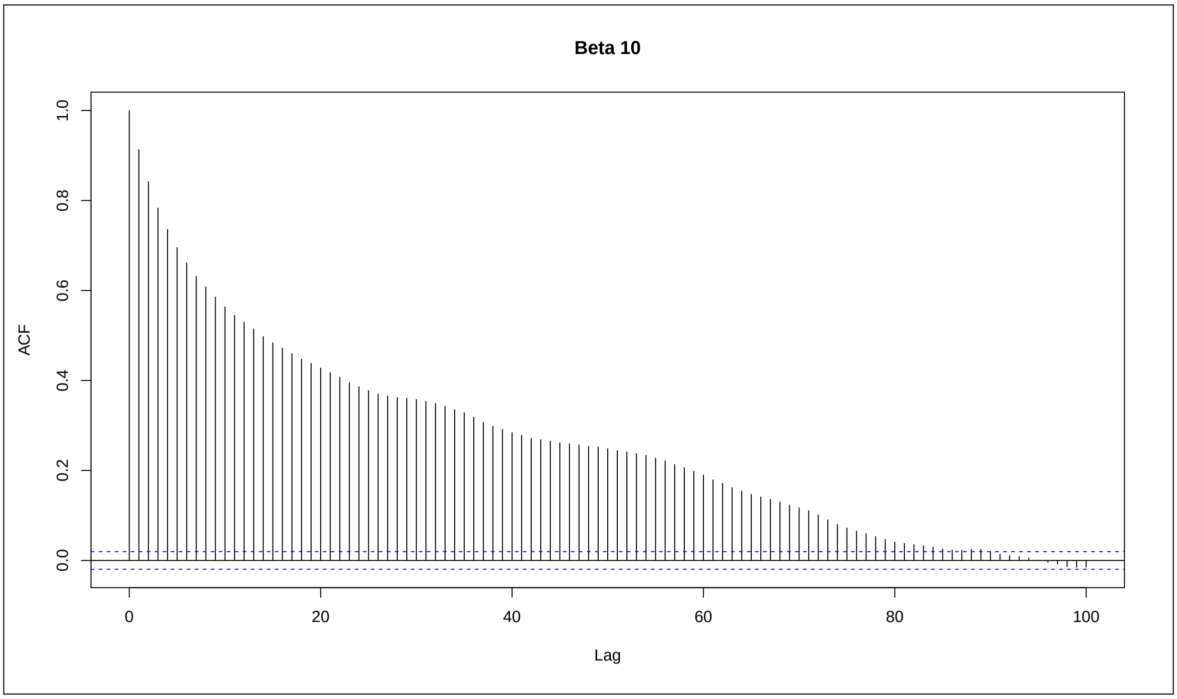

Beta 8
1.0
0.5
0.0
−0.5
0
2000
4000
6000
8000
10000
Time

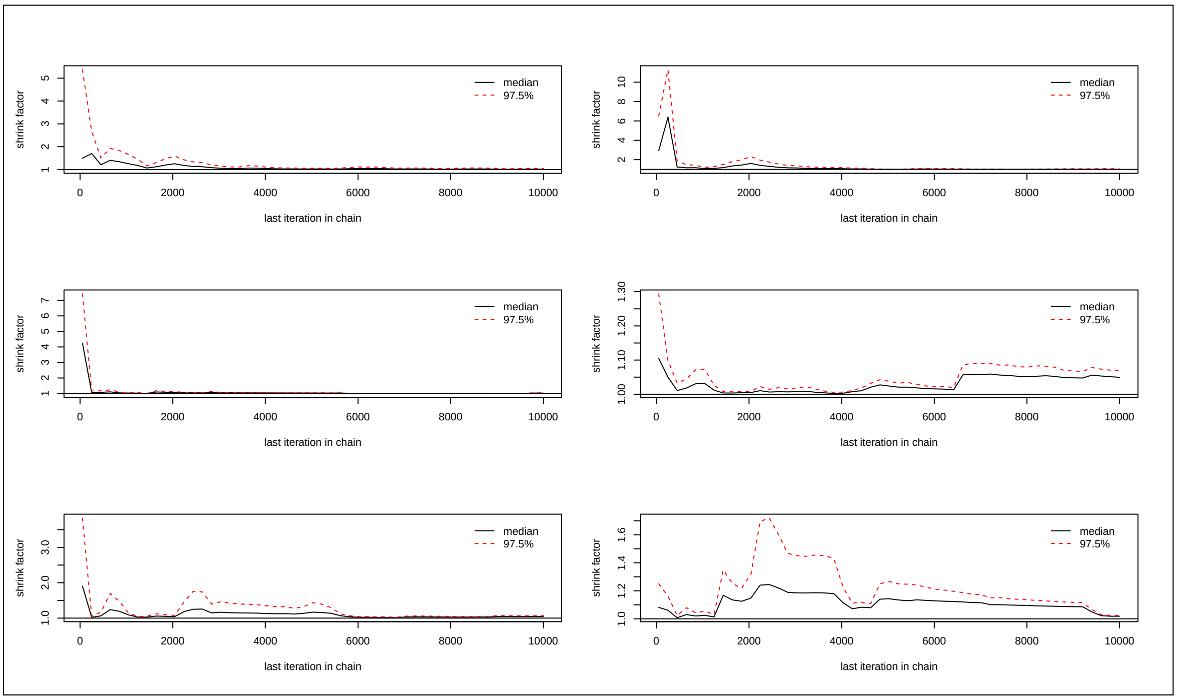

Beta 8
ACF
1.0
0.8
0.6
0.4
0.2
0.0
0
20
40
60
80
100
Lag

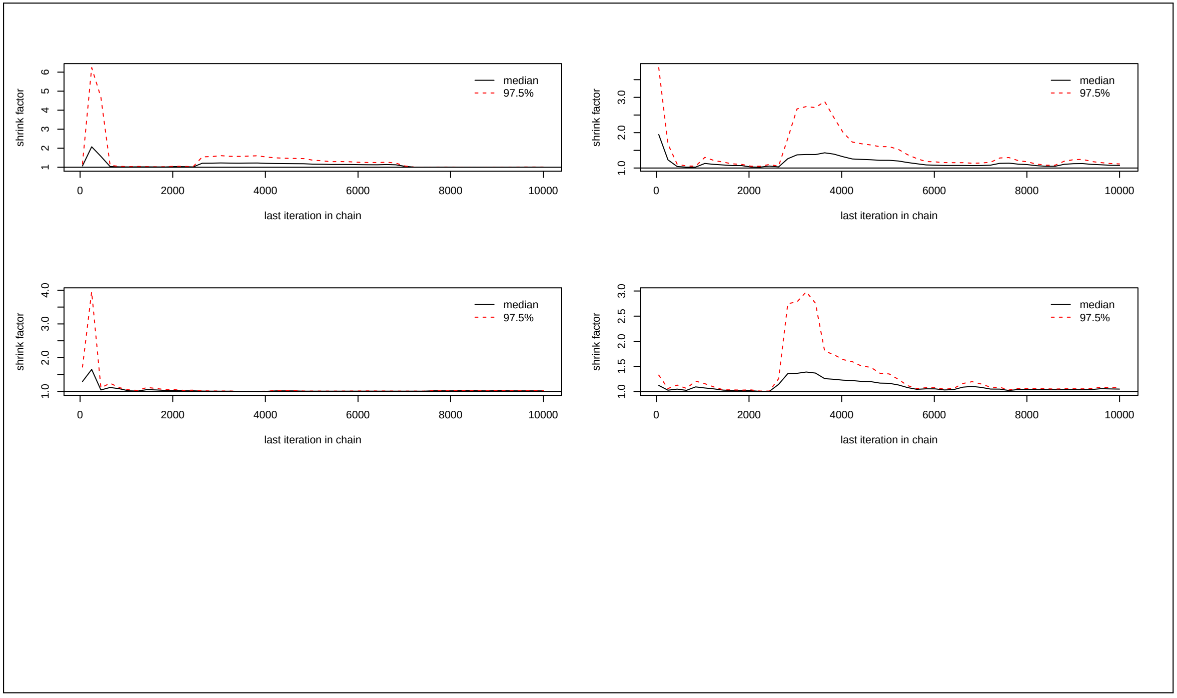

Beta 9
0.6
0.4
0.2
0.0
−0.2
−0.4
0
2000
4000
6000
8000
10000
Time


Beta 9
ACF
1.0
0.8
0.6
0.4
0.2
0.0
0
20
40
60
80
100
Lag


Beta 10
0.2
0.0
−0.2
−0.4
−0.6
0
2000
4000
6000
8000
10000
Time


Beta 10
ACF
1.0
0.8
0.6
0.4
0.2
0.0
0
20
40
60
80
100
Lag

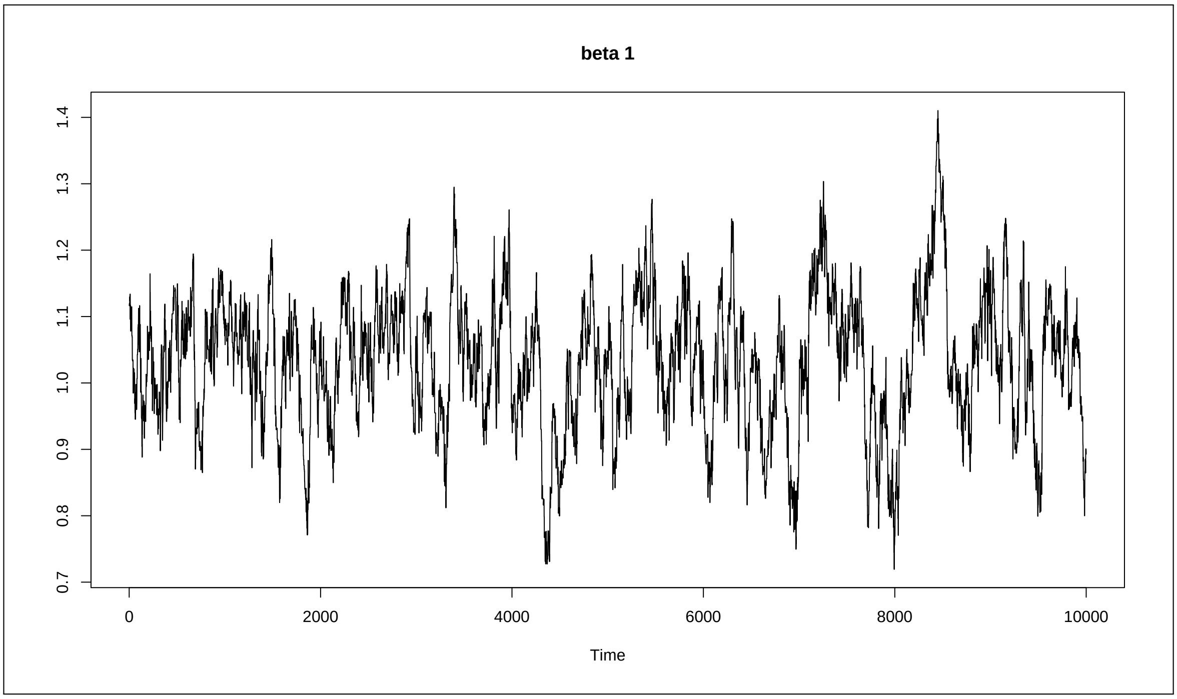


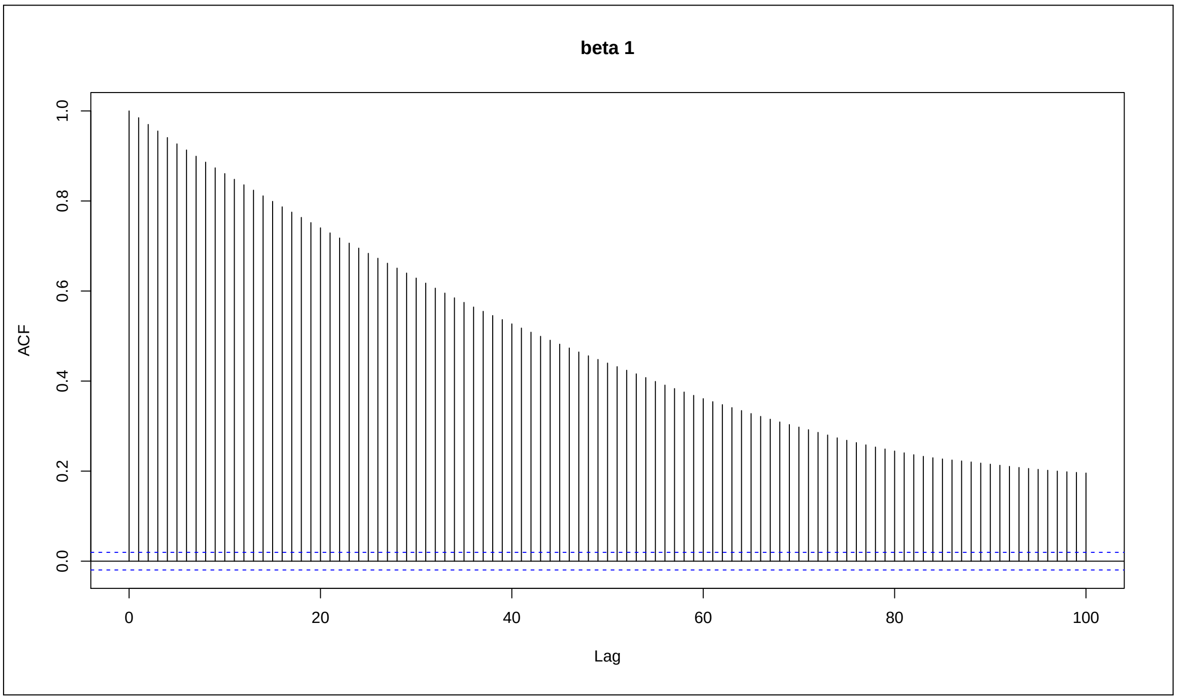


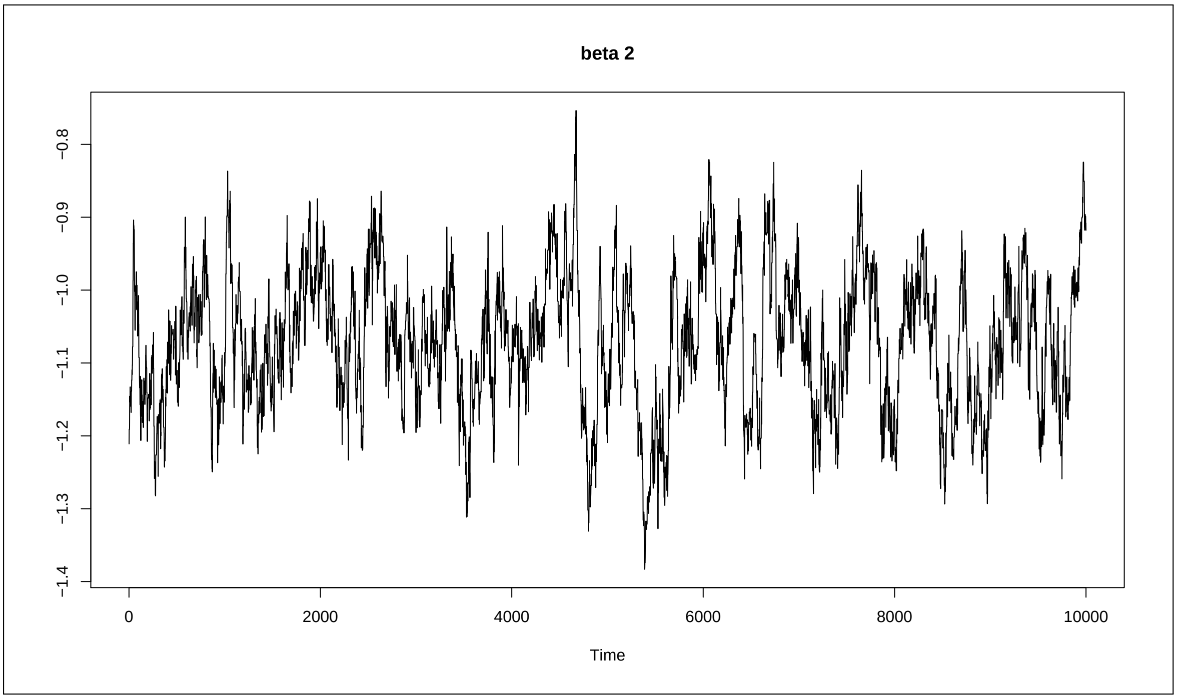
A.6 Block updating with random walk sampler

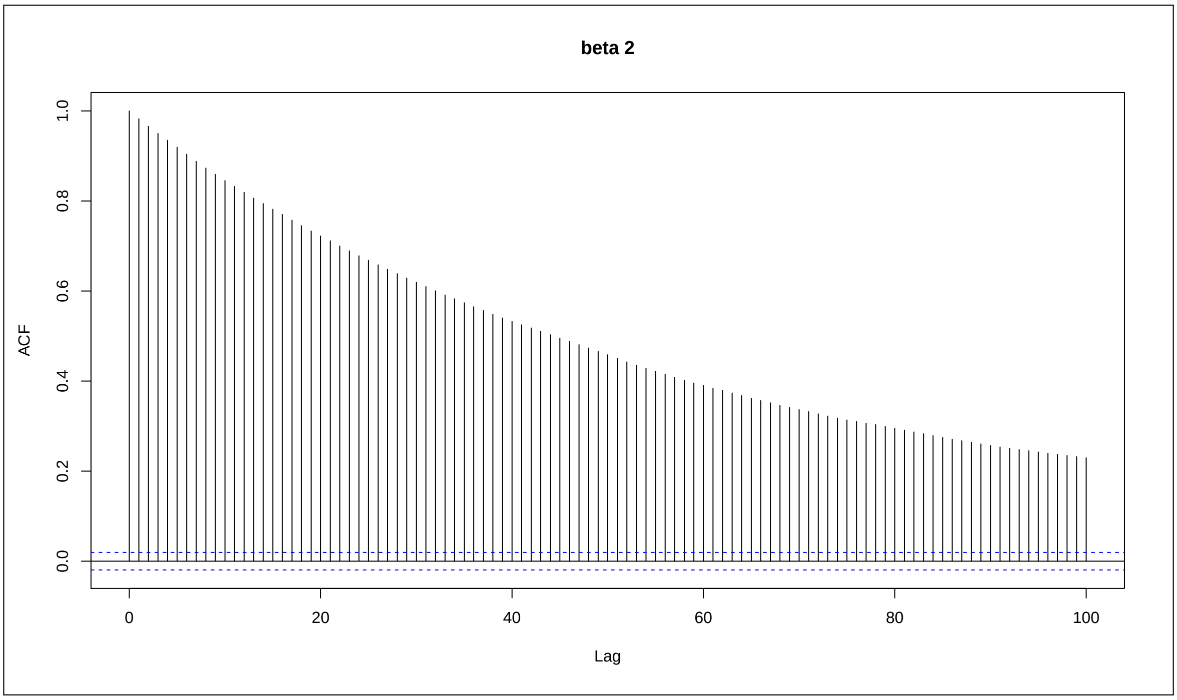
Parameter	G-R Point est.	G-R Upper C.I.
β_1	1.0062	1.0161
β_2	1.0038	1.0104
β_3	1.0054	1.0100
β_4	1.0029	1.0077
β_5	1.0051	1.0120
β_6	1.0085	1.0198
β_7	1.0155	1.0369
β_8	1.0343	1.0862
β_9	1.0016	1.0024
β_{10}	1.0029	1.0069

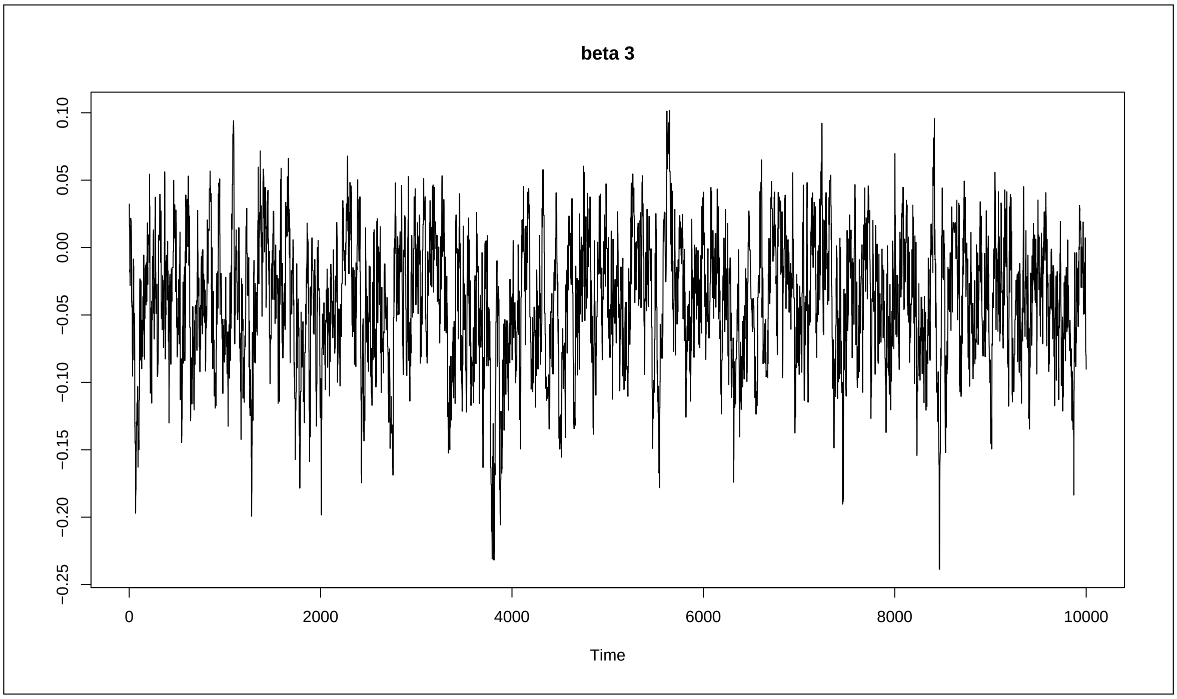
Table A.6: Point estimates of potential scale reduction factor, corresponding upper confidence limits and acceptance rates for the ten logistic slope parameters.

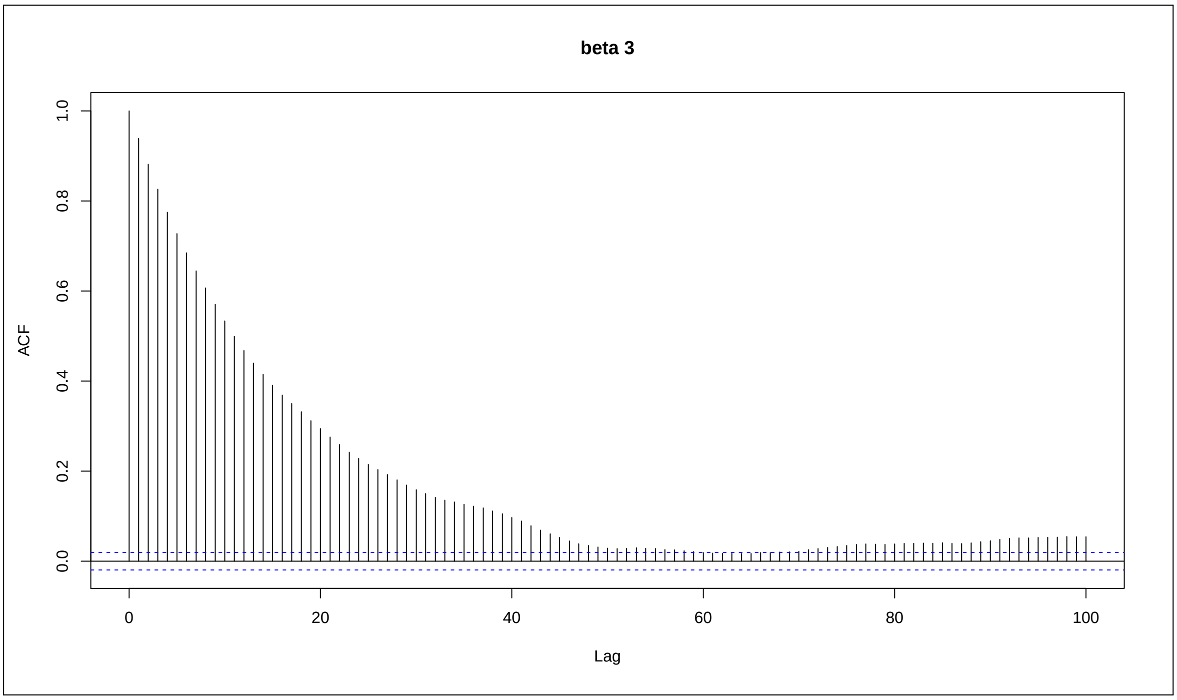


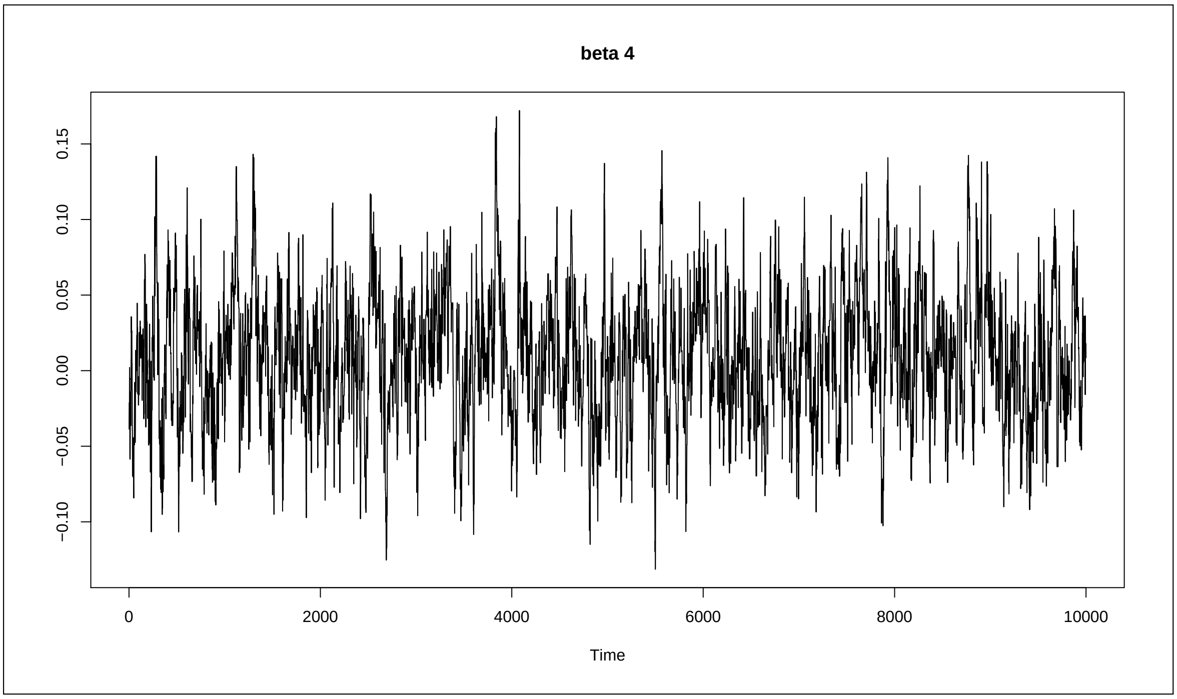

beta 1
ACF
1.0
0.8
0.6
0.4
0.2
0.0
0
20
40
60
80
100
Lag

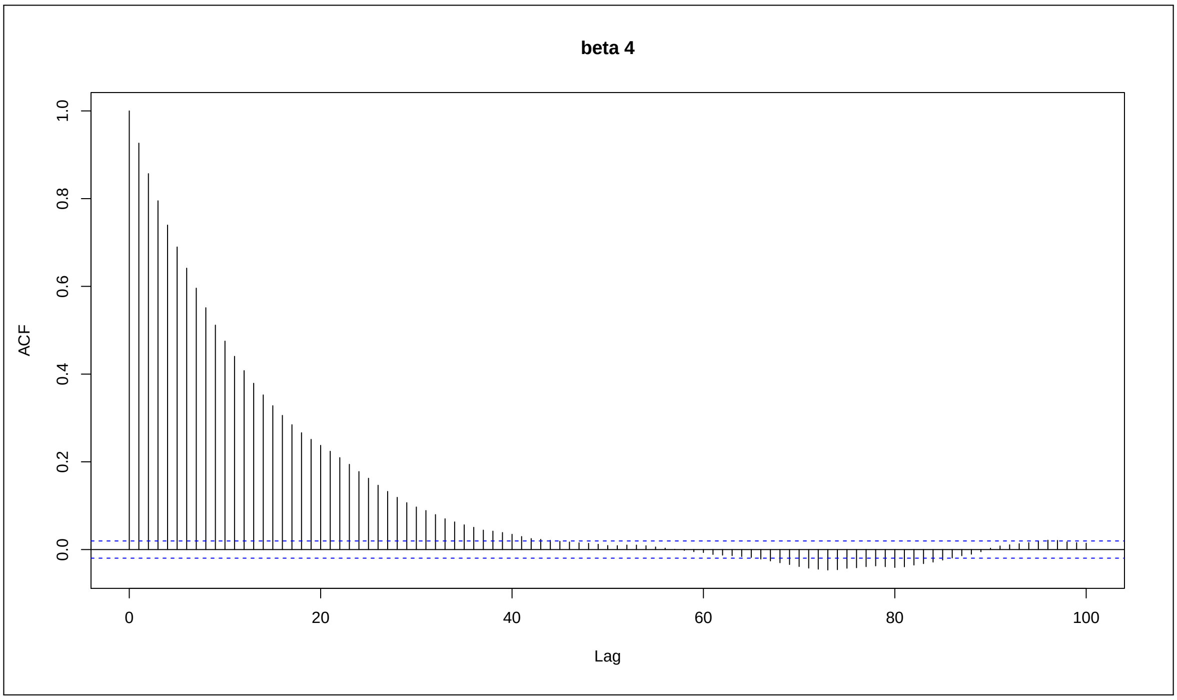

beta 2
−0.8
−0.9
−1.0
−1.1
−1.2
−1.3
−1.4
0
2000
4000
6000
8000
10000
Time

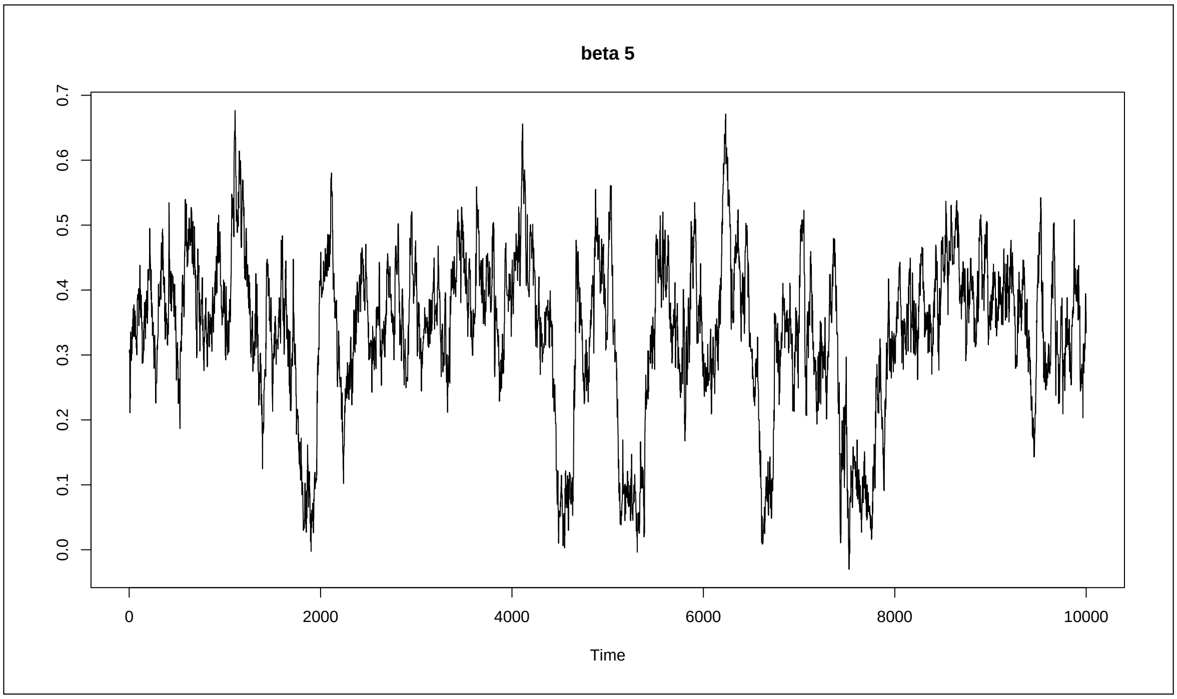

beta 2
ACF
0.0
0.2
0.4
0.6
0.8
1.0
0
20
40
60
80
100
Lag

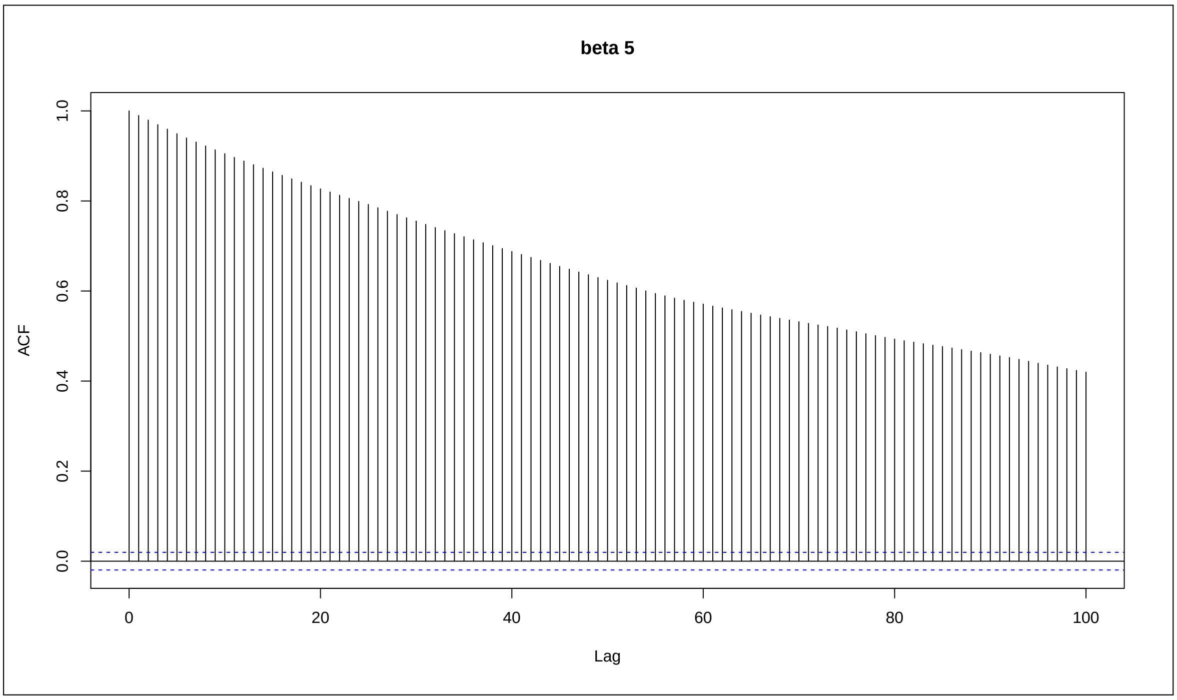

beta 3
−0.25
−0.20
−0.15
−0.10
−0.05
0.00
0.05
0.10
0
2000
4000
6000
8000
10000
Time

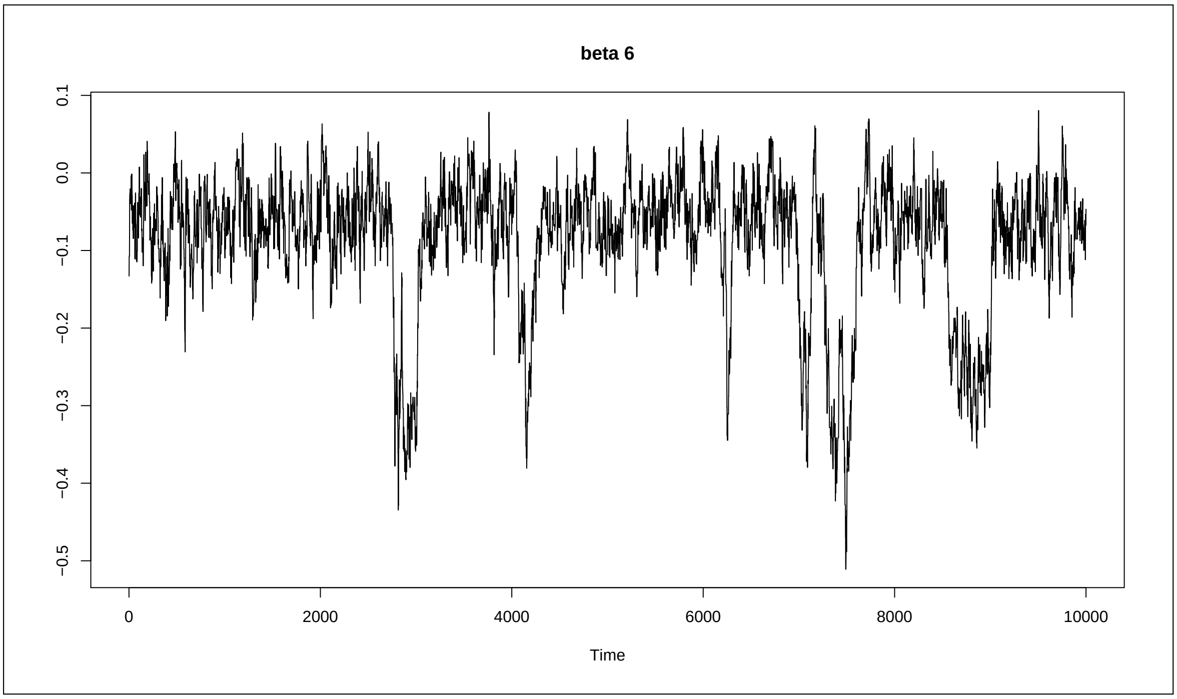

beta 3
ACF
1.0
0.8
0.6
0.4
0.2
0.0
0
20
40
60
80
100
Lag

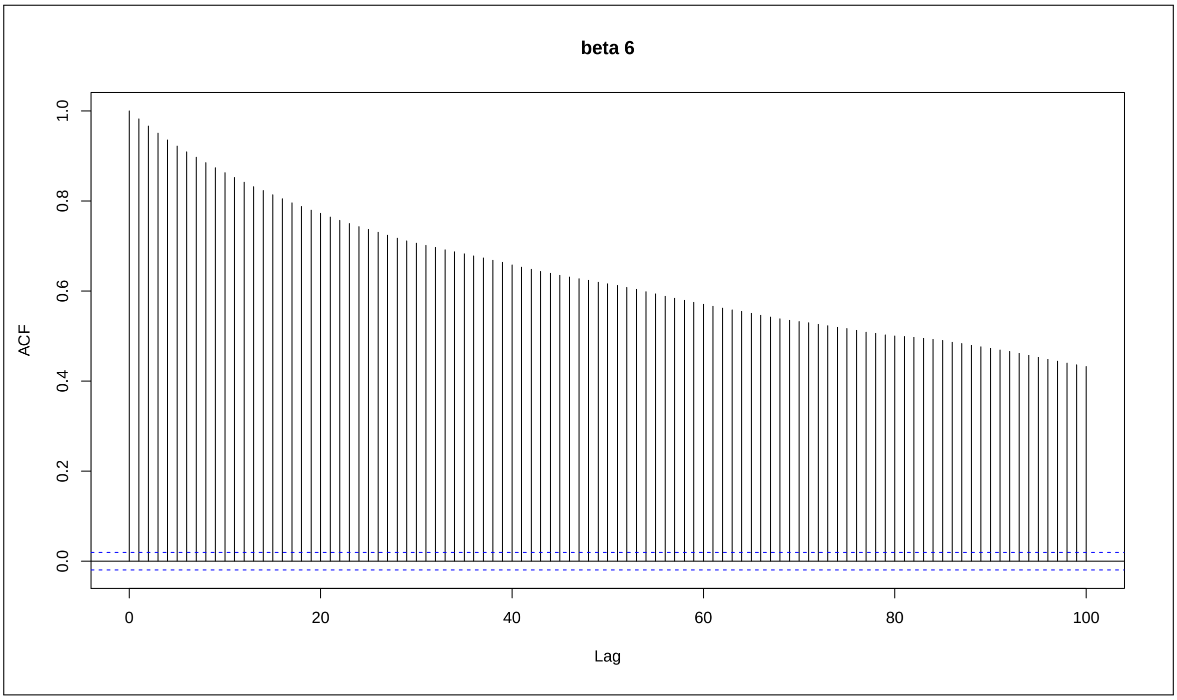

beta 4
0.15
0.10
0.05
0.00
−0.05
−0.10
0
2000
4000
6000
8000
10000
Time

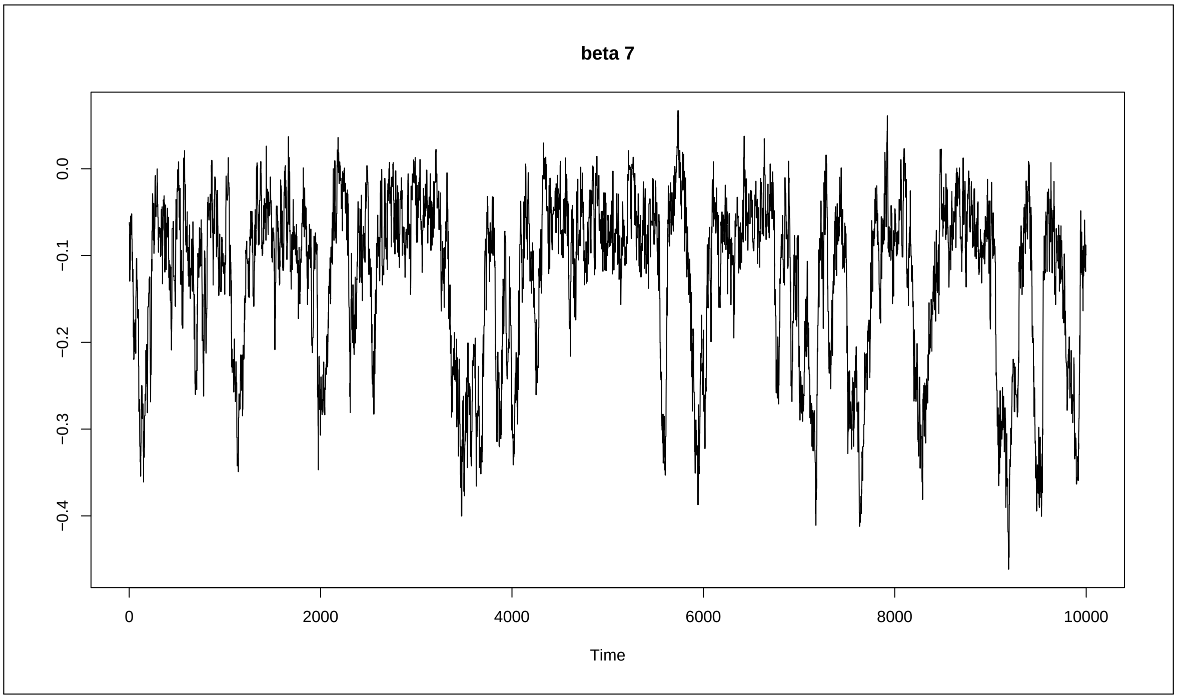

beta 4
ACF
1.0
0.8
0.6
0.4
0.2
0.0
0
20
40
60
80
100
Lag

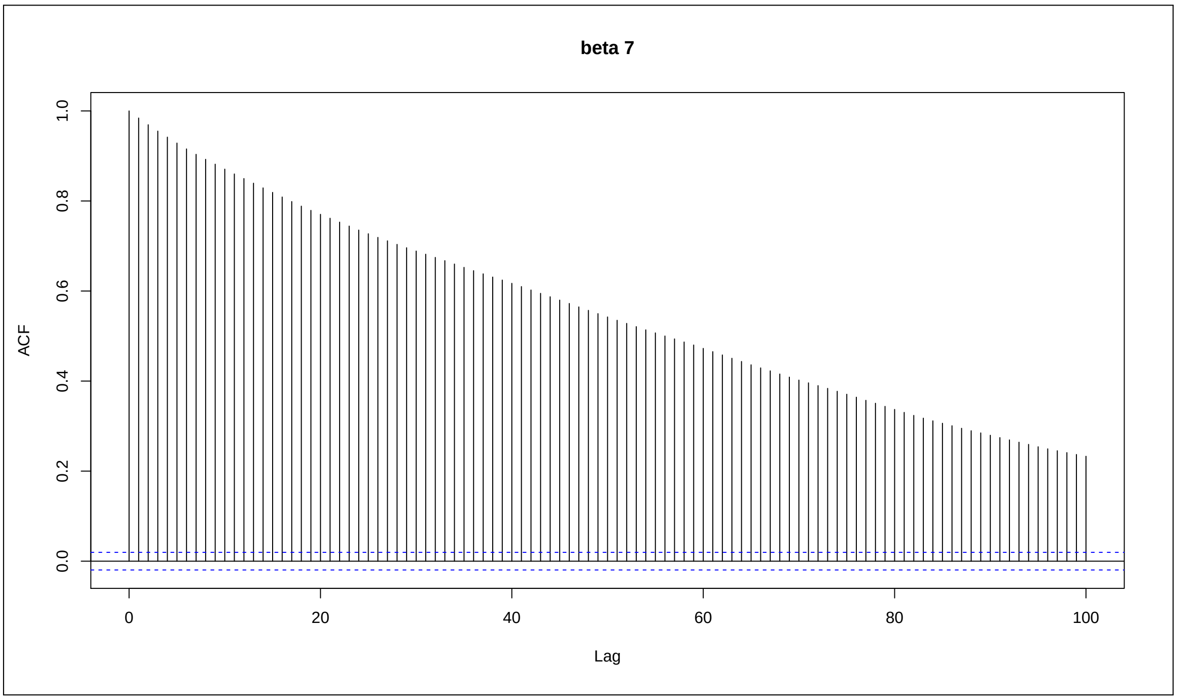

beta 5
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0.0
0
2000
4000
6000
8000
10000
Time

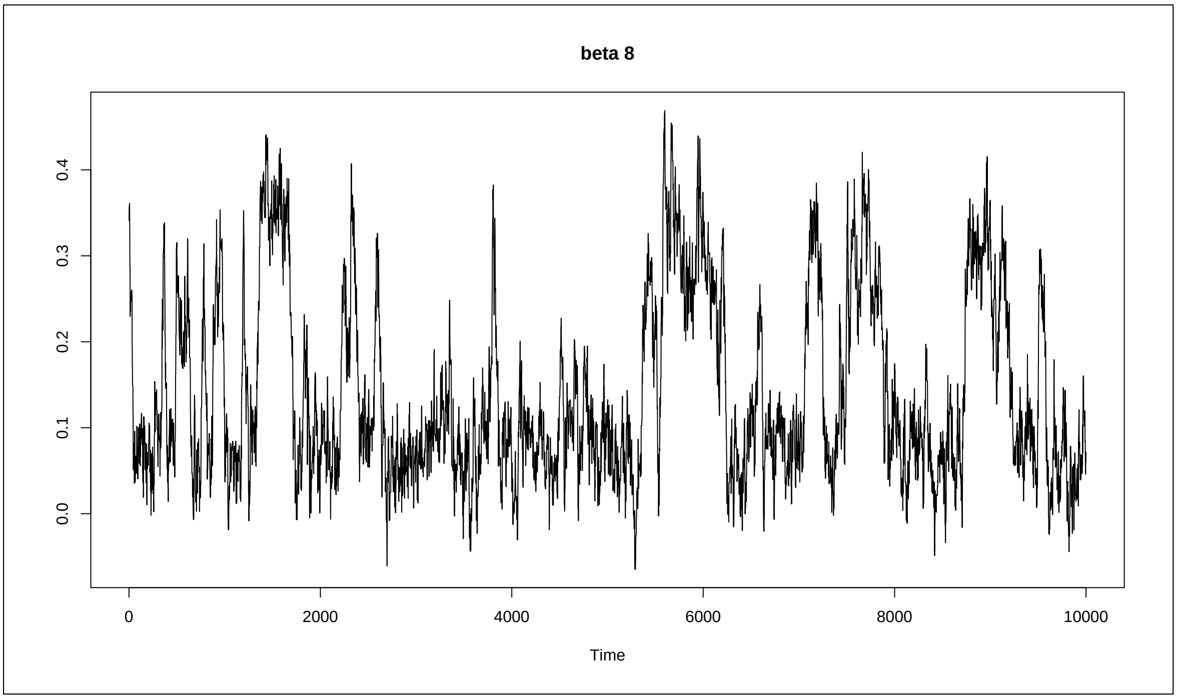

beta 5
ACF
1.0
0.8
0.6
0.4
0.2
0.0
0
20
40
60
80
100
Lag

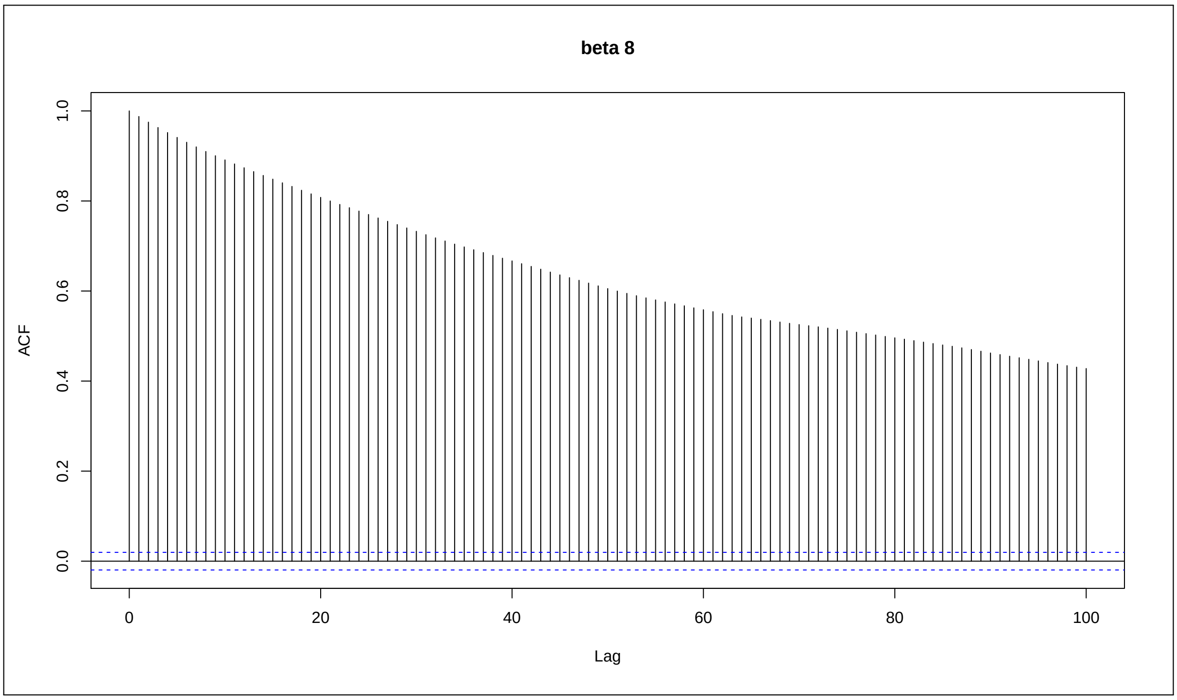

beta 6
0.1
0.0
−0.1
−0.2
−0.3
−0.4
−0.5
0
2000
4000
6000
8000
10000
Time

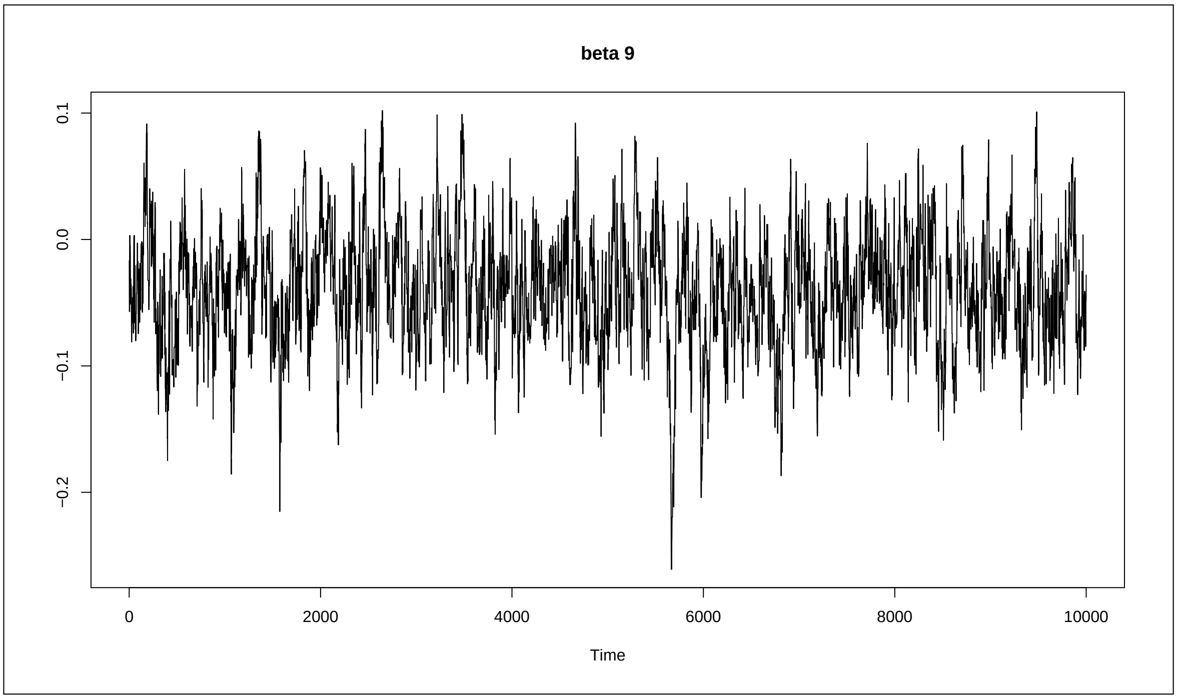

beta 6
ACF
1.0
0.8
0.6
0.4
0.2
0.0
0
20
40
60
80
100
Lag

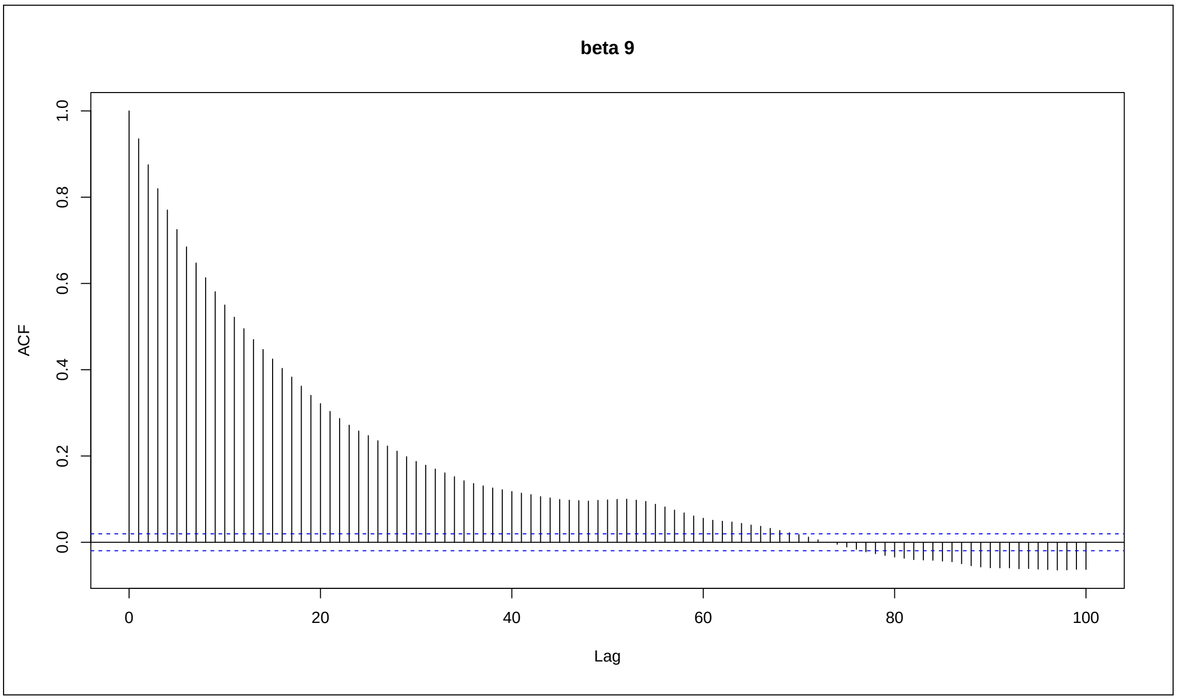

beta 7
0.0
−0.1
−0.2
−0.3
−0.4
0
2000
4000
6000
8000
10000
Time

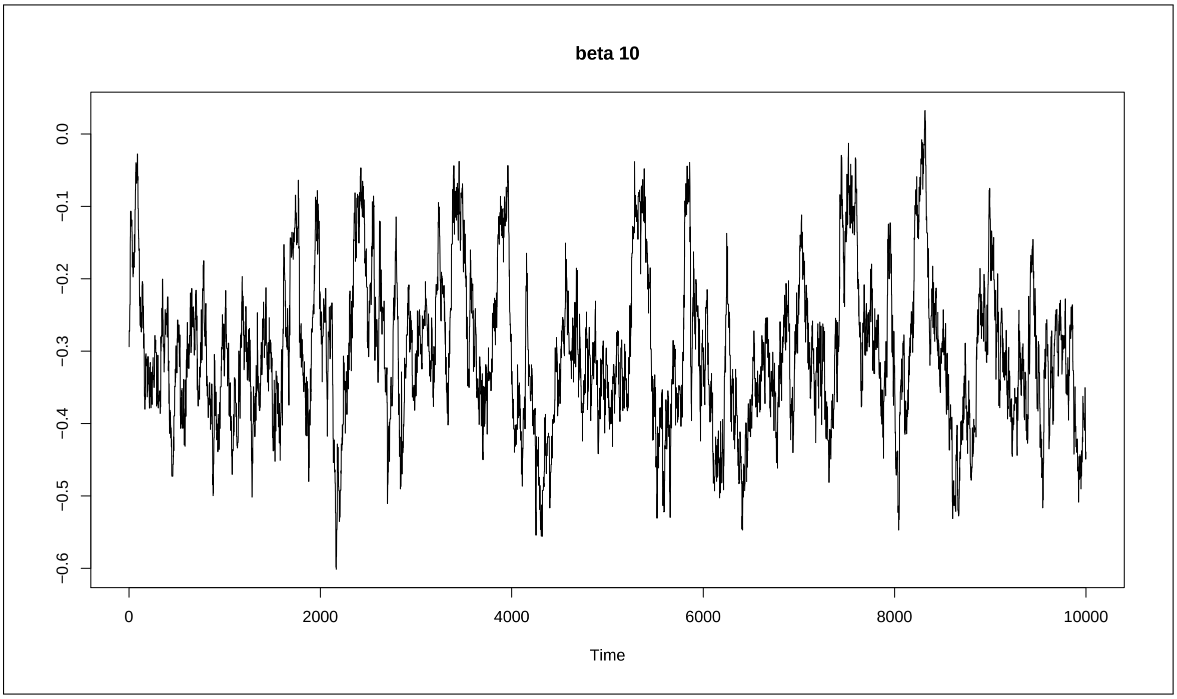

beta 7
ACF
1.0
0.8
0.6
0.4
0.2
0.0
0
20
40
60
80
100
Lag

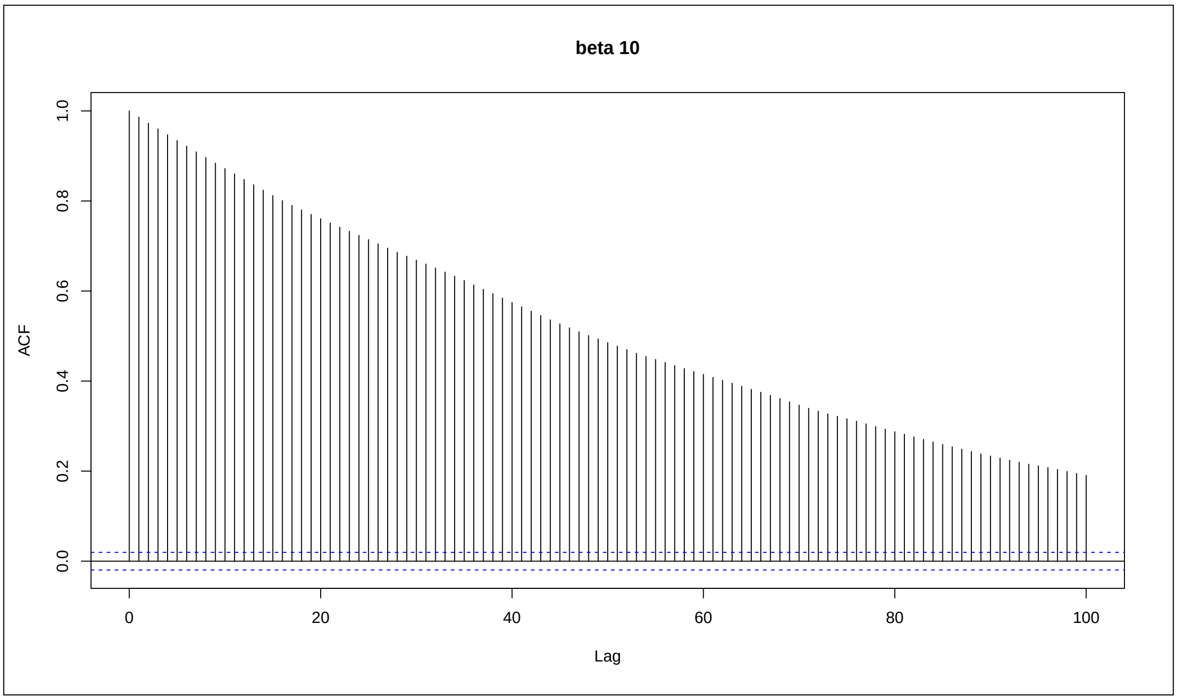

beta 8
0.4
0.3
0.2
0.1
0.0
0
2000
4000
6000
8000
10000
Time

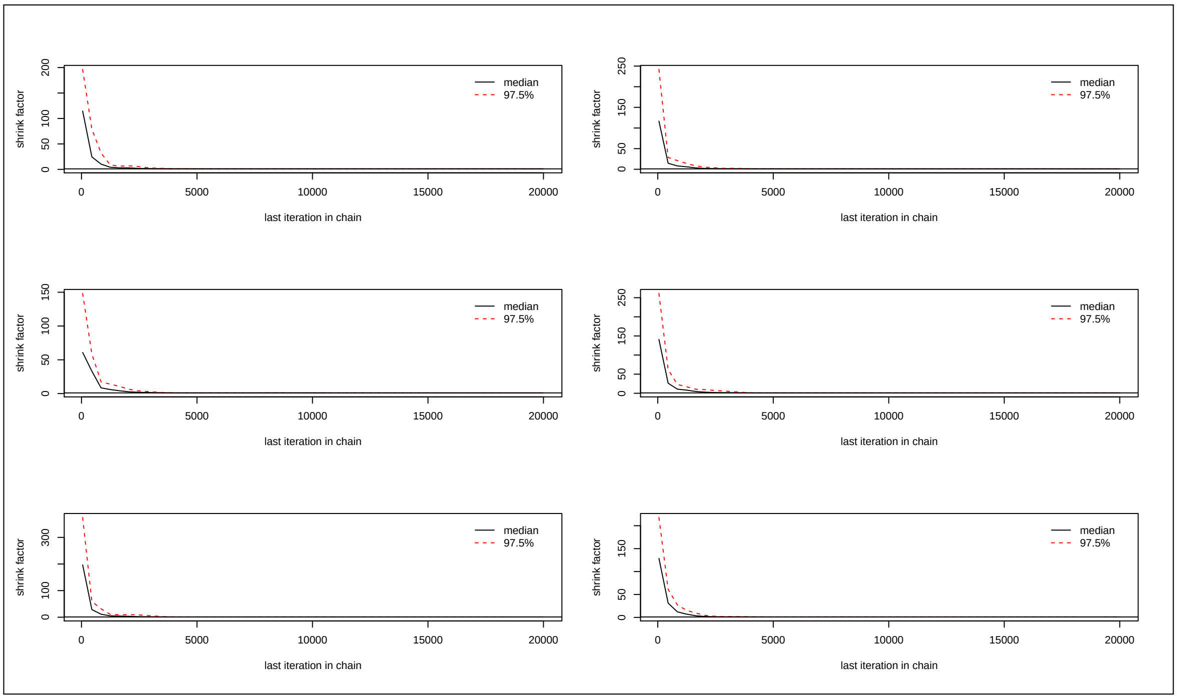

beta 8
ACF
1.0
0.8
0.6
0.4
0.2
0.0
0
20
40
60
80
100
Lag

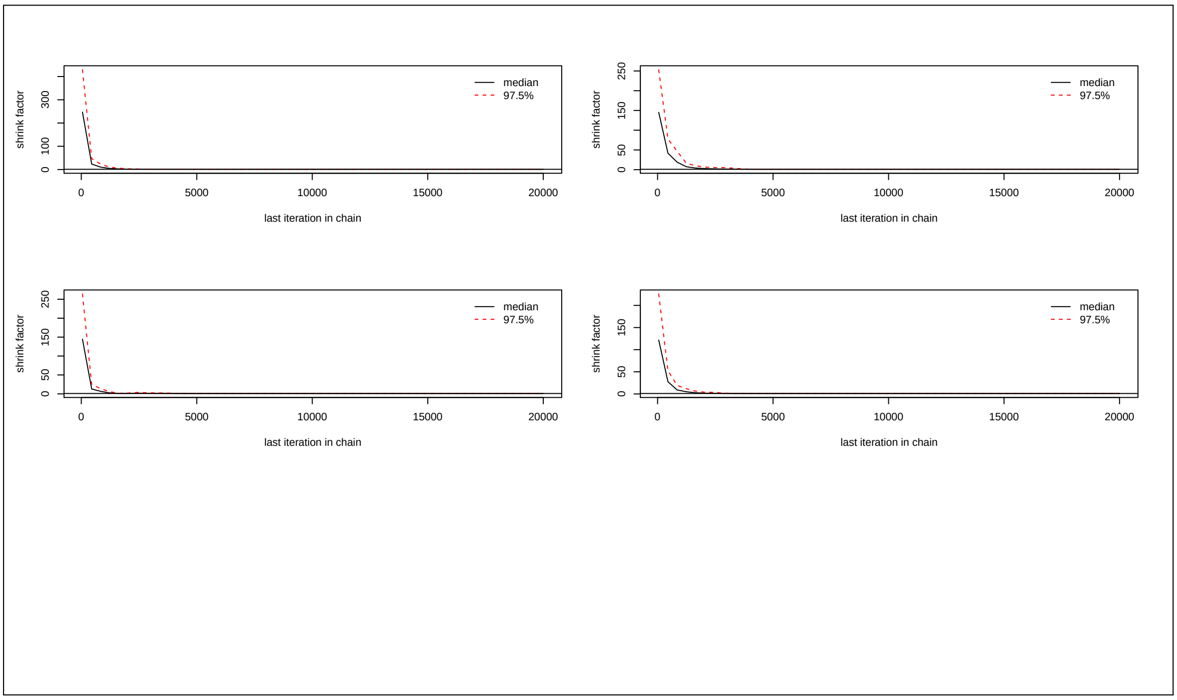

beta 9
0.1
0.0
−0.1
−0.2
0
2000
4000
6000
8000
10000
Time


beta 9
ACF
1.0
0.8
0.6
0.4
0.2
0.0
0
20
40
60
80
100
Lag


beta 10
0.0
−0.1
−0.2
−0.3
−0.4
−0.5
−0.6
0
2000
4000
6000
8000
10000
Time


beta 10
ACF
0.0
0.2
0.4
0.6
0.8
1.0
0
20
40
60
80
100
Lag

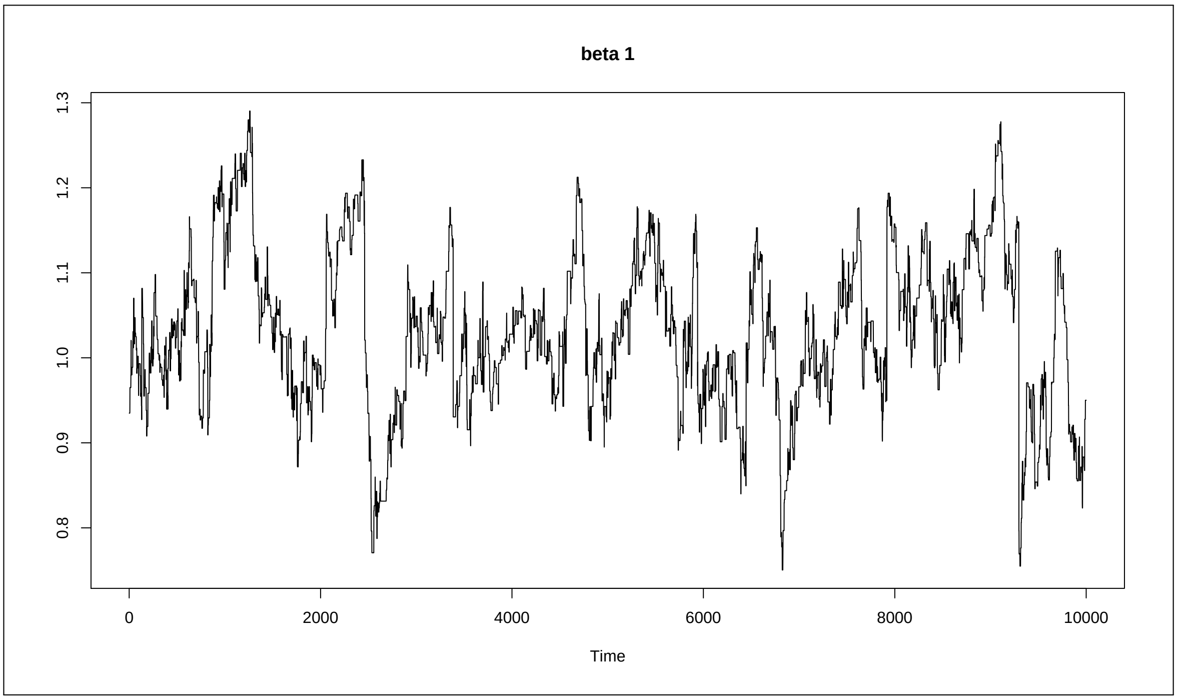

shrink factor
last iteration in chain
median
97.5%
0
5000
10000
15000
20000

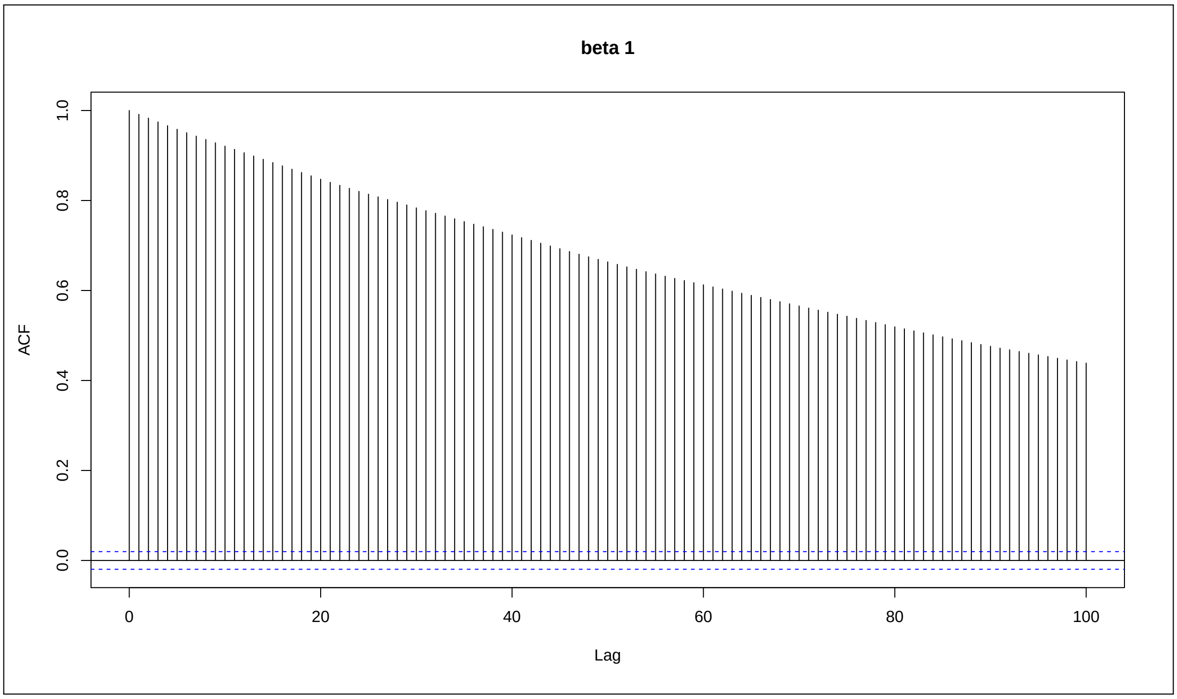


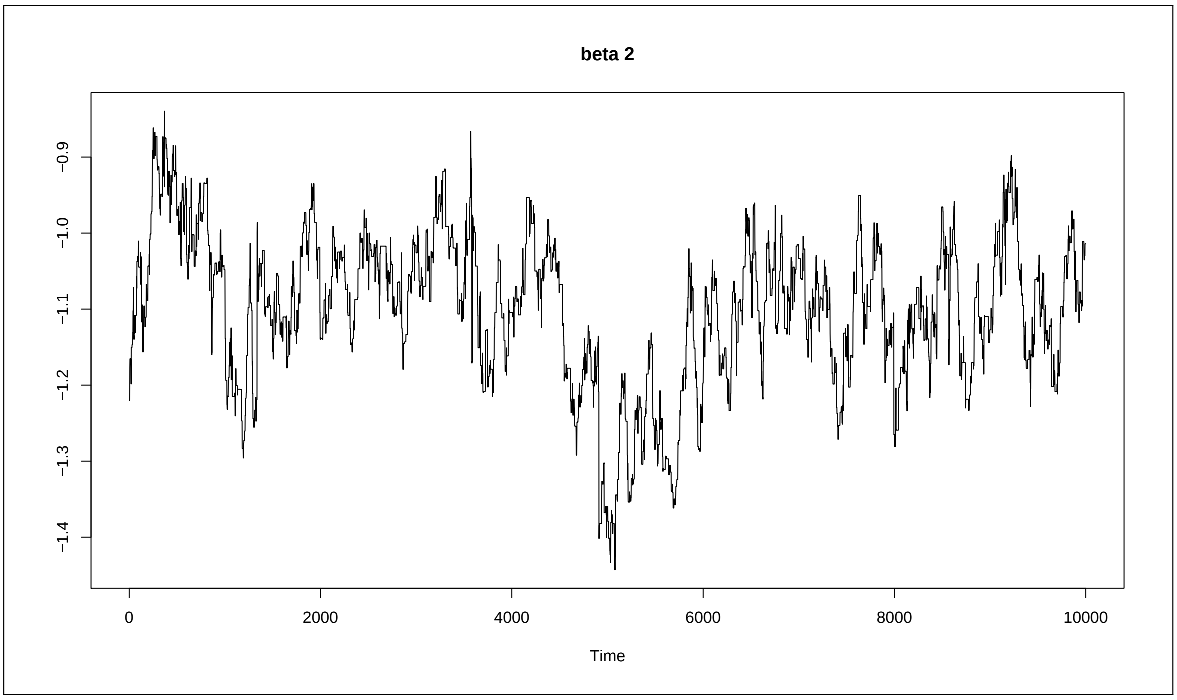
A.7 Block updating with tri-modal random walk sampler

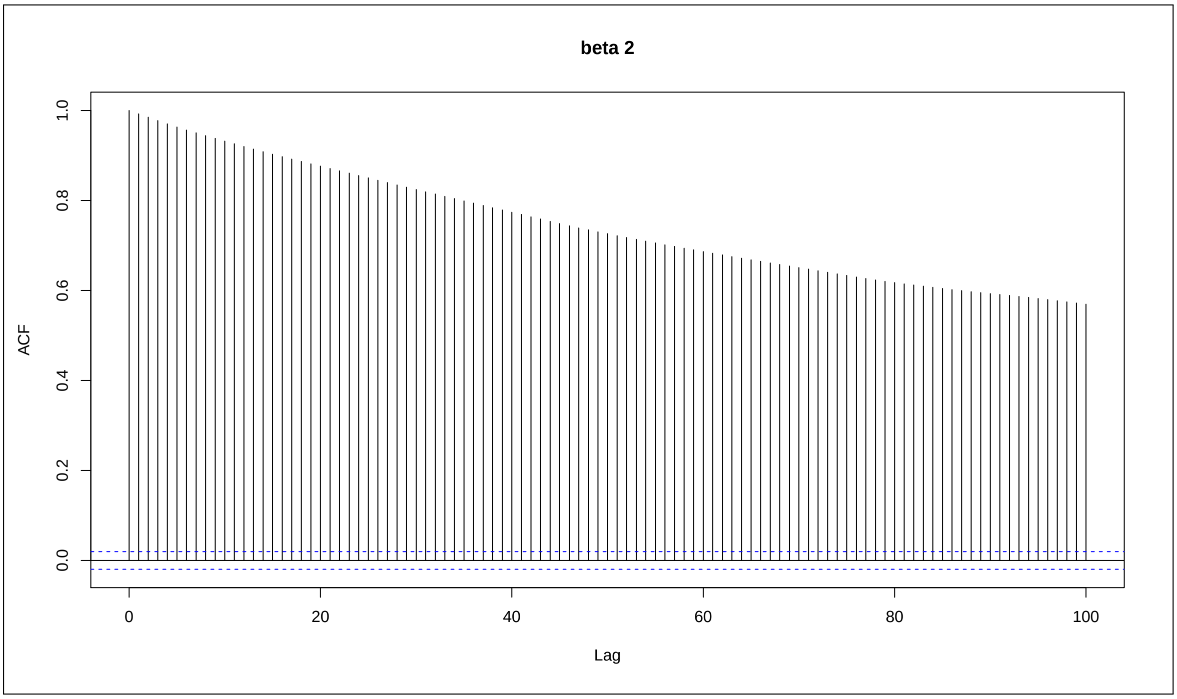
Parameter	G-R Point est.	G-R Upper C.I.
β_1	1.0265	1.0676
β_2	1.0175	1.0444
β_3	1.0245	1.0408
β_4	1.0022	1.0053
β_5	1.0219	1.0552
β_6	1.0332	1.0746
β_7	1.0292	1.0667
β_8	1.0379	1.0922
β_9	1.0204	1.0476
β_{10}	1.0162	1.0380

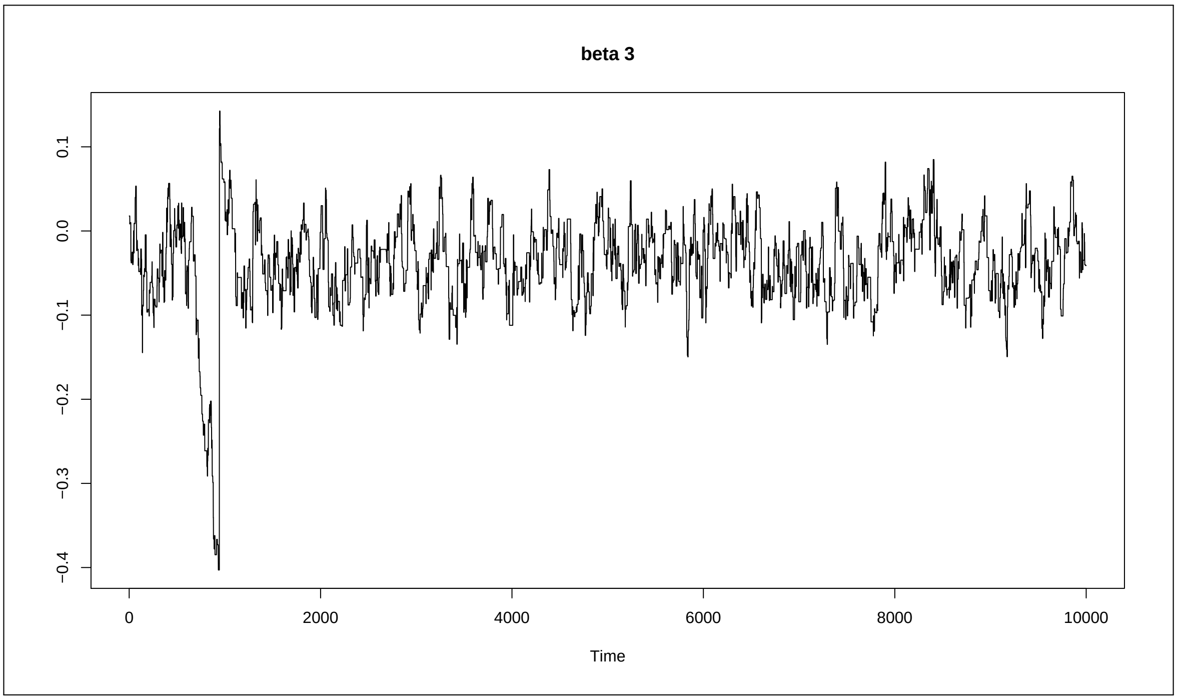
Table A.7: Point estimates of potential scale reduction factor, corresponding upper confidence limits and acceptance rates for the ten logistic slope parameters.

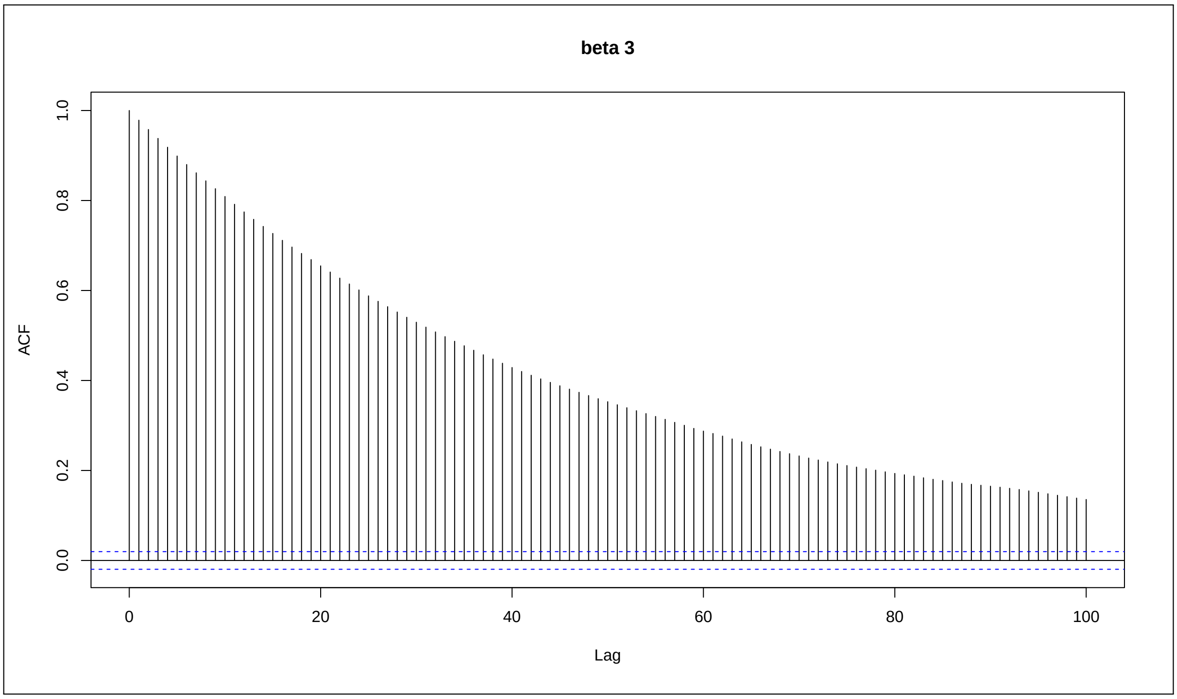

beta 1
1.3
1.2
1.1
1.0
0.9
0.8
0
2000
4000
6000
8000
10000
Time

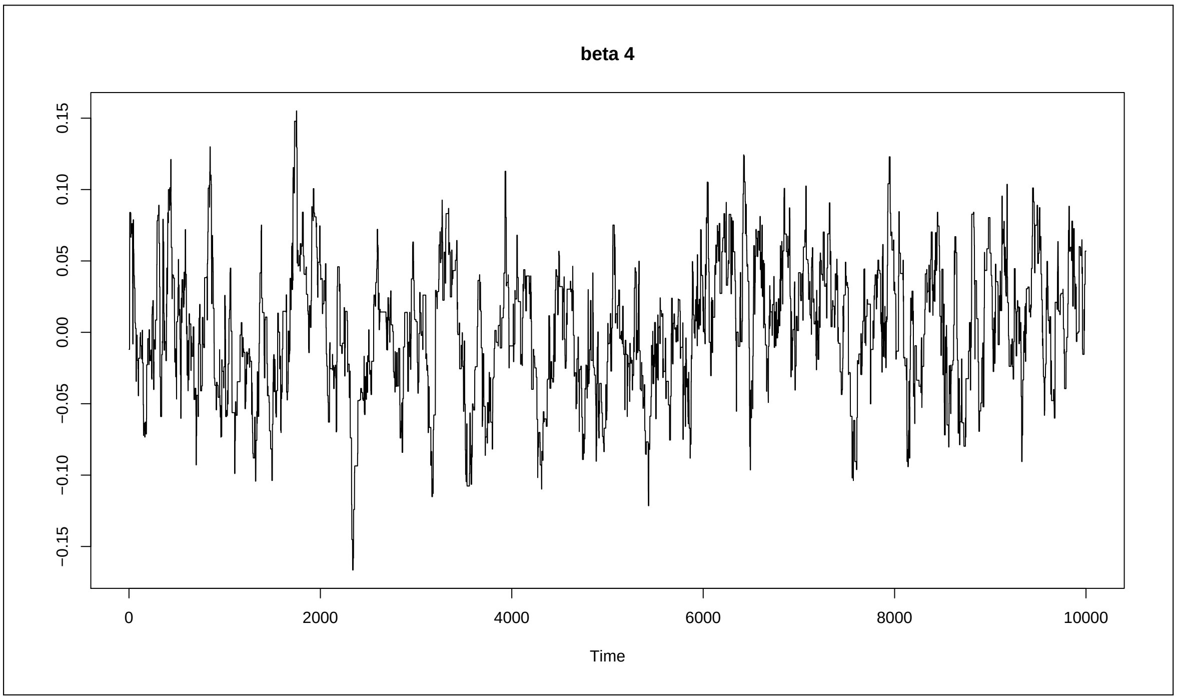

beta 1
ACF
1.0
0.8
0.6
0.4
0.2
0.0
0
20
40
60
80
100
Lag

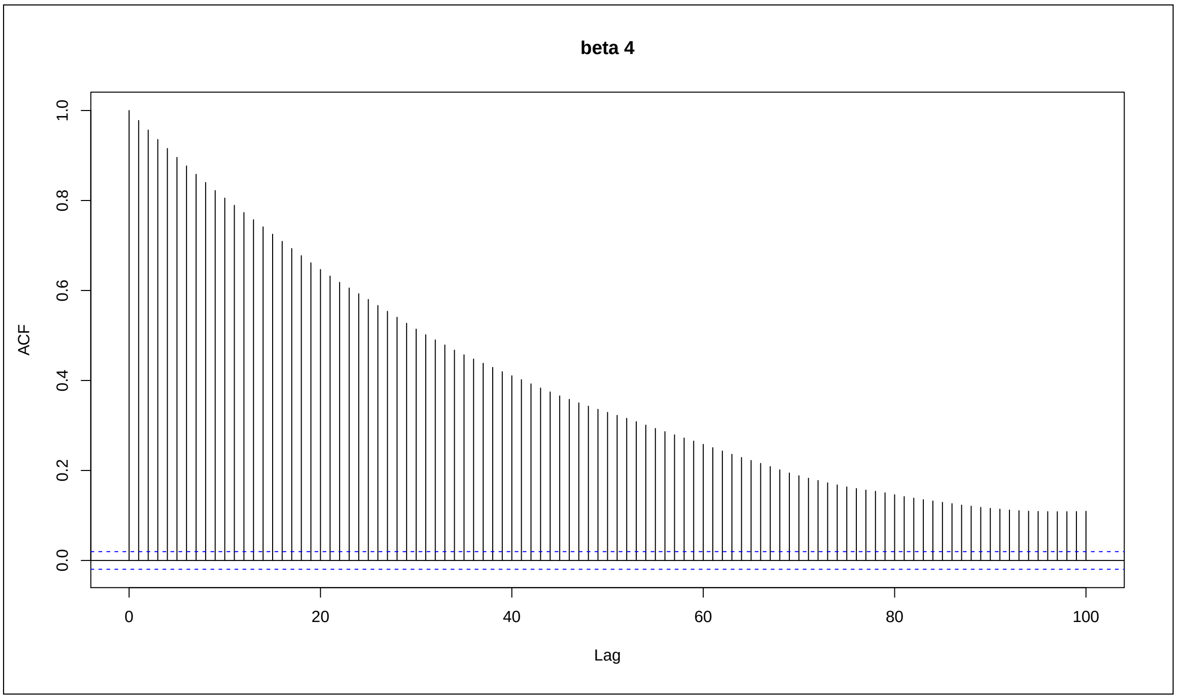

beta 2
−0.9
−1.0
−1.1
−1.2
−1.3
−1.4
0
2000
4000
6000
8000
10000
Time

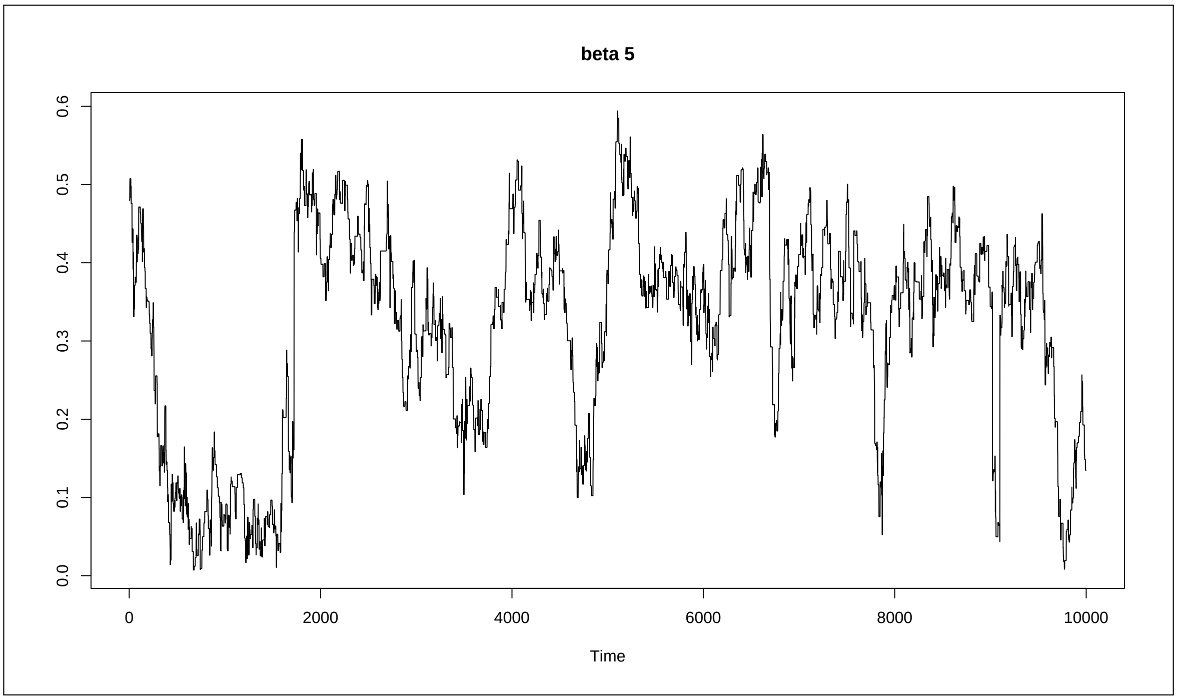

beta 2
ACF
1.0
0.8
0.6
0.4
0.2
0.0
0
20
40
60
80
100
Lag

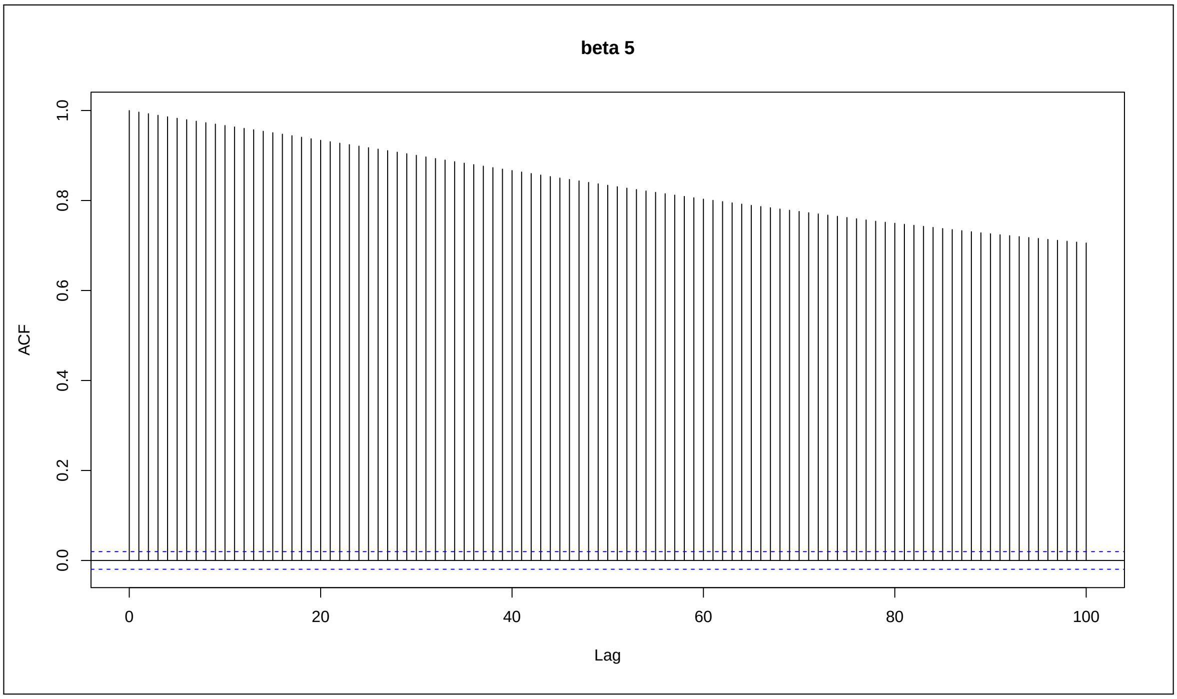

beta 3
0.1
0.0
−0.1
−0.2
−0.3
−0.4
0
2000
4000
6000
8000
10000
Time

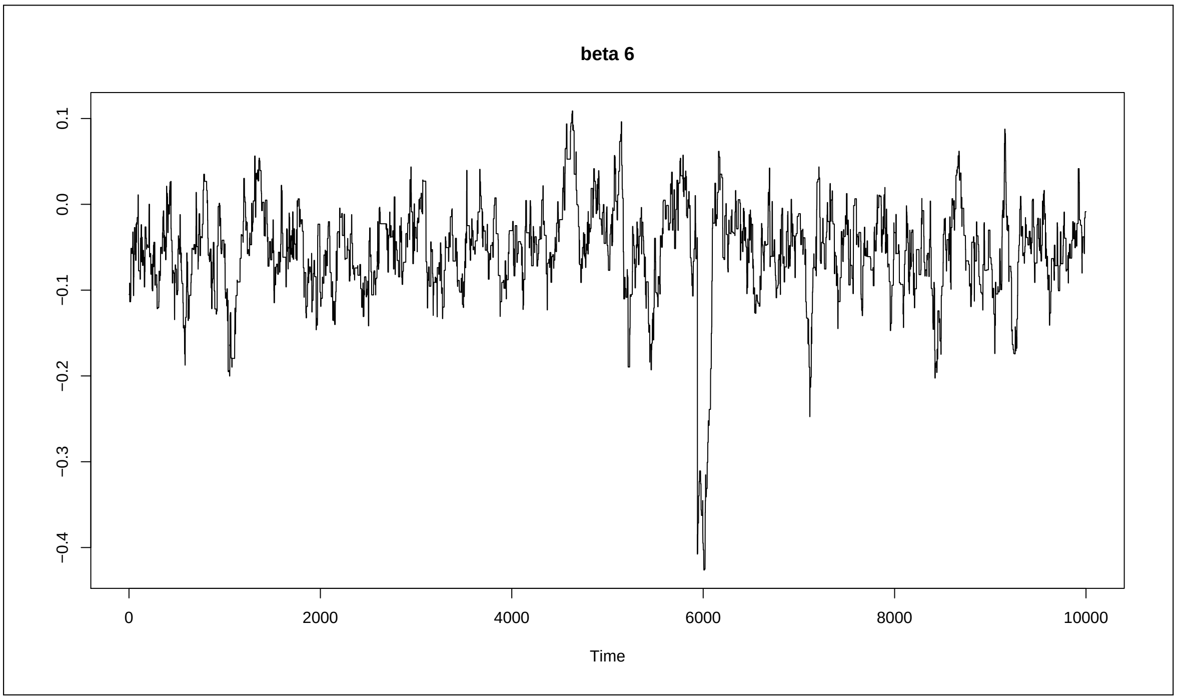

beta 3
ACF
1.0
0.8
0.6
0.4
0.2
0.0
0
20
40
60
80
100
Lag

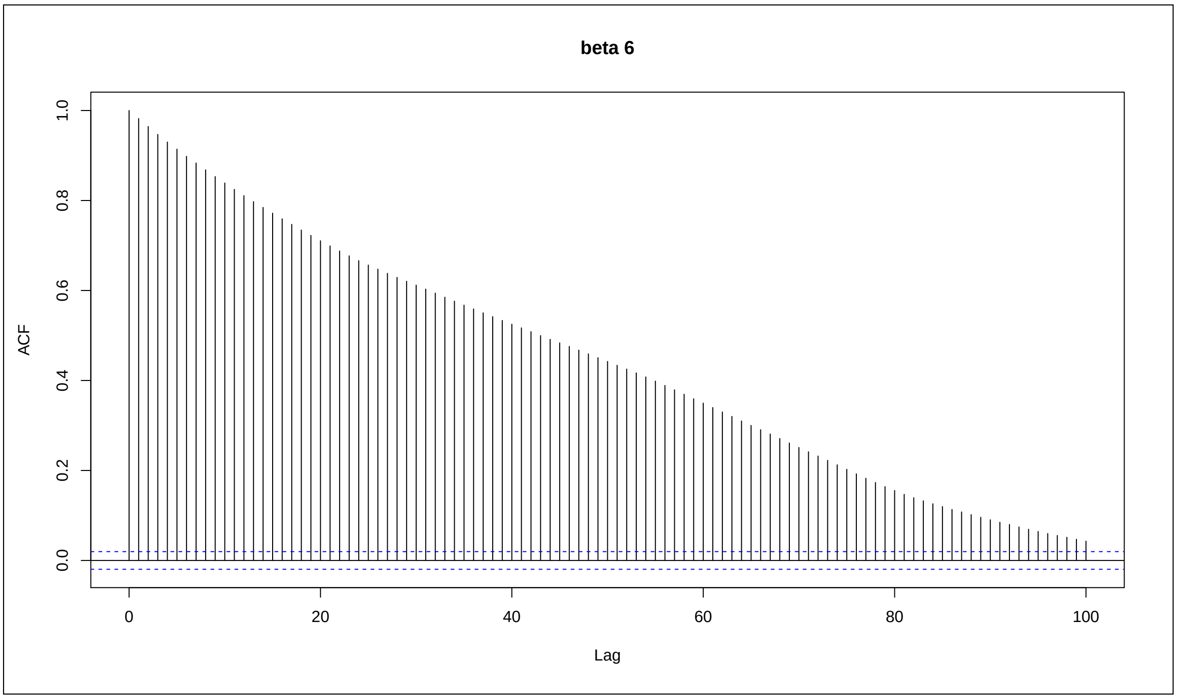

beta 4
0.15
0.10
0.05
0.00
−0.05
−0.10
−0.15
0
2000
4000
6000
8000
10000
Time

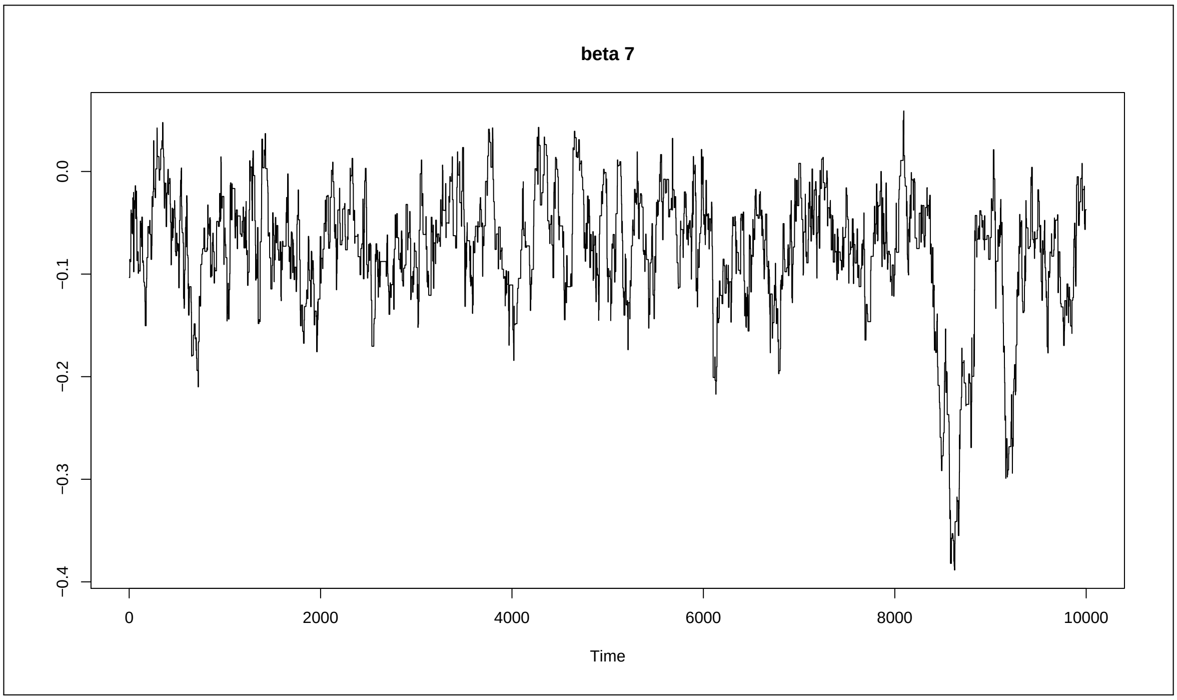

beta 4
ACF
1.0
0.8
0.6
0.4
0.2
0.0
0
20
40
60
80
100
Lag

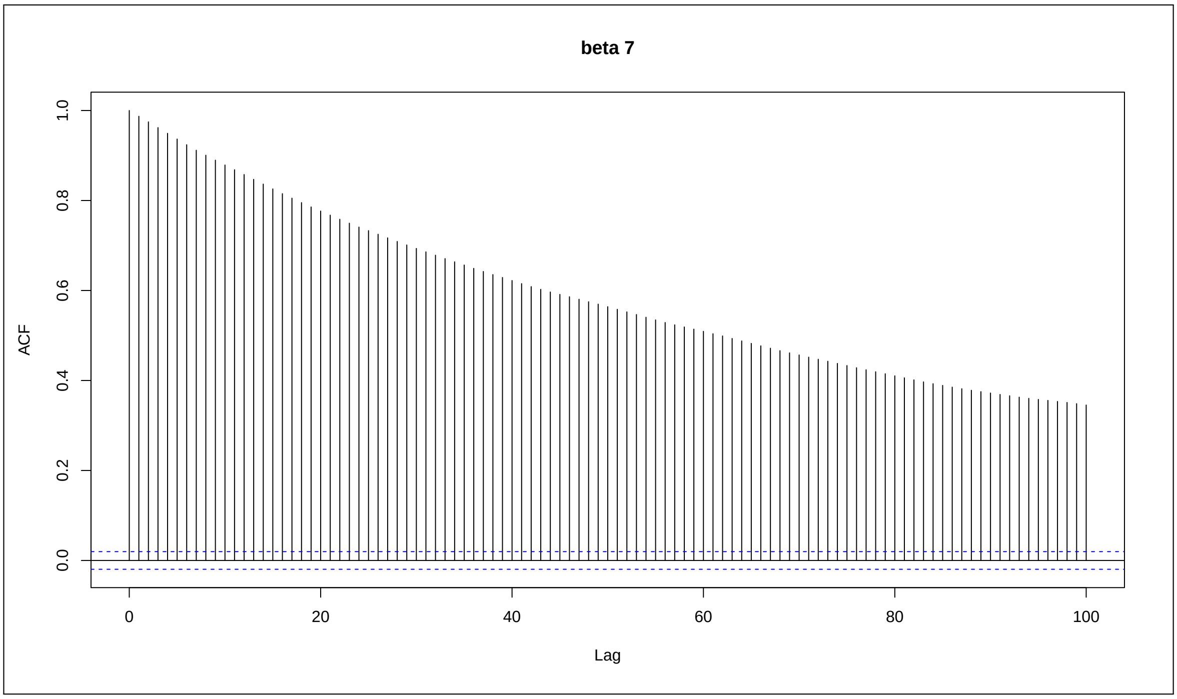

beta 5
0.0
0.1
0.2
0.3
0.4
0.5
0.6
0
2000
4000
6000
8000
10000
Time

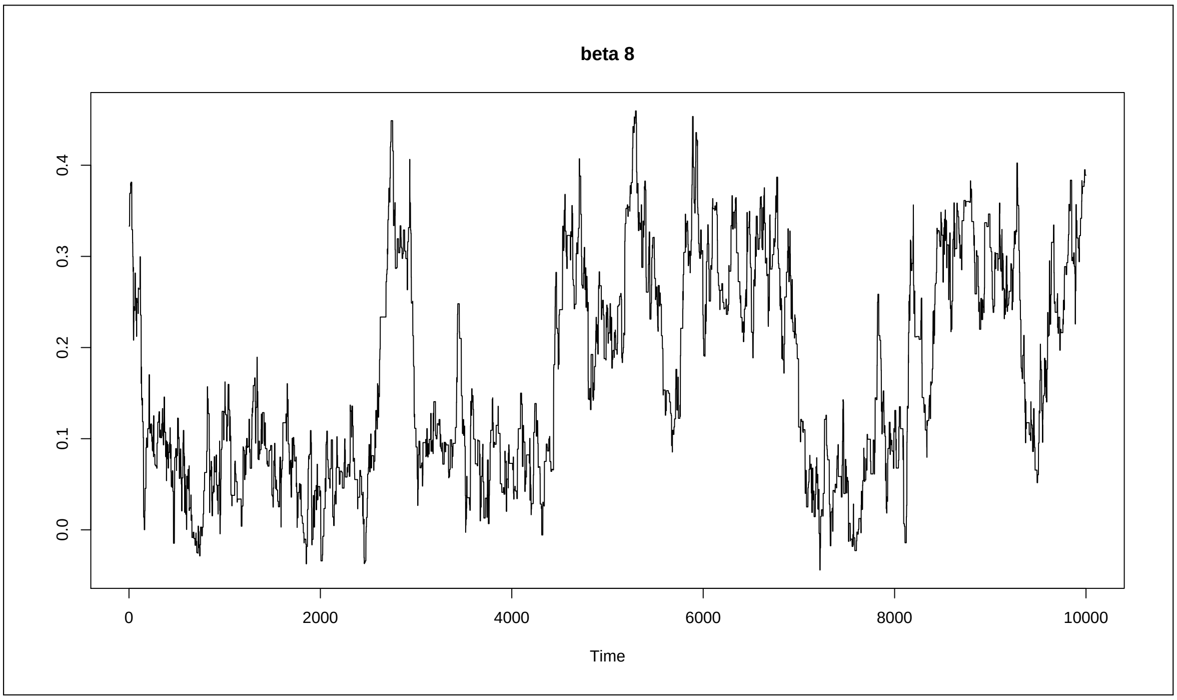

beta 5
ACF
0.0
0.2
0.4
0.6
0.8
1.0
0
20
40
60
80
100
Lag

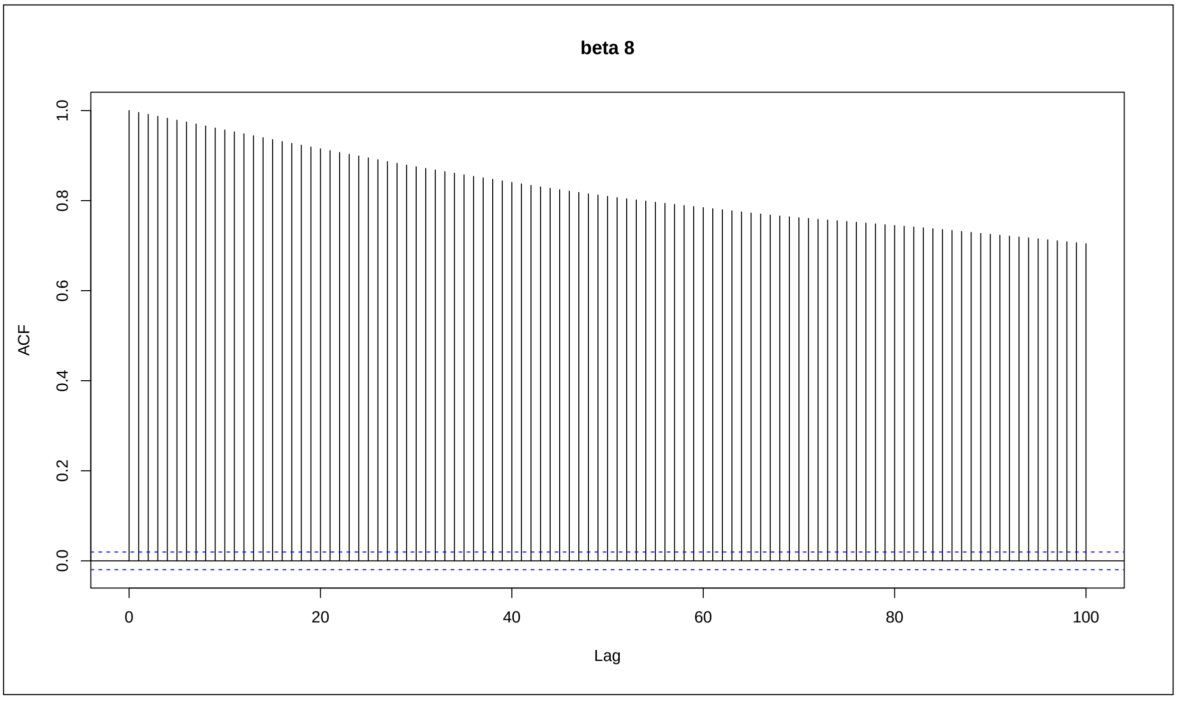

beta 6
0.1
0.0
−0.1
−0.2
−0.3
−0.4
0
2000
4000
6000
8000
10000
Time

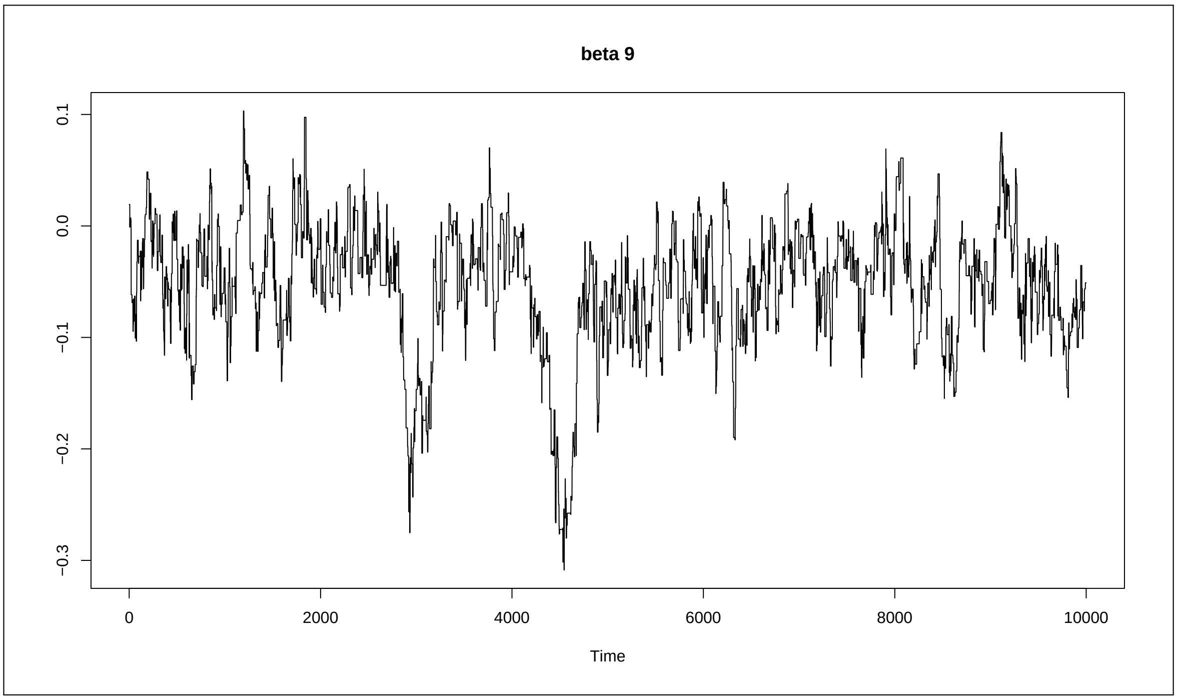

beta 6
ACF
1.0
0.8
0.6
0.4
0.2
0.0
0
20
40
60
80
100
Lag

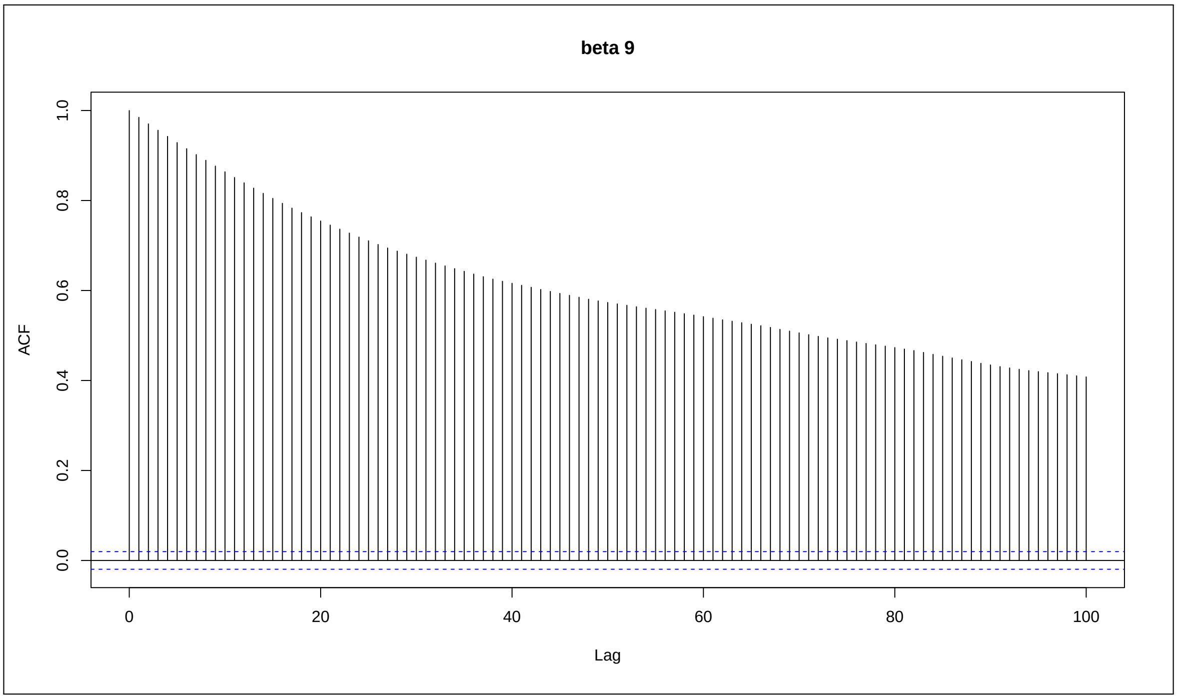

beta 7
0.0
−0.1
−0.2
−0.3
−0.4
0
2000
4000
6000
8000
10000
Time

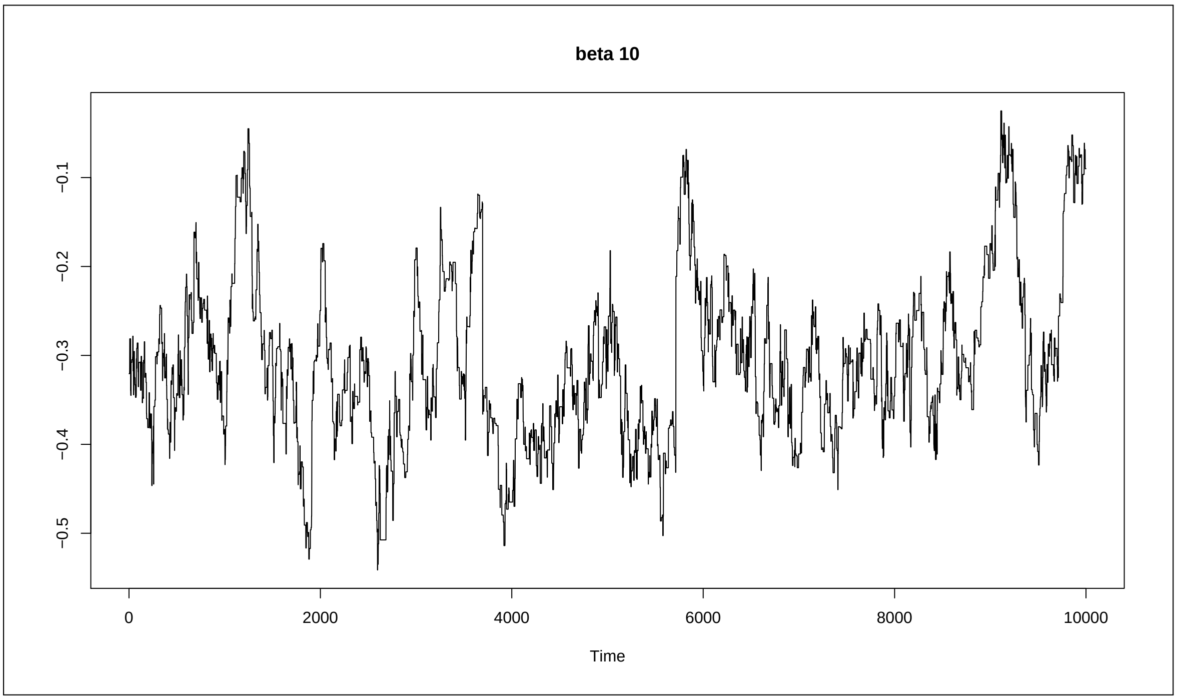

beta 7
1.0
0.8
0.6
0.4
0.2
0.0
ACF
0
20
40
60
80
100
Lag

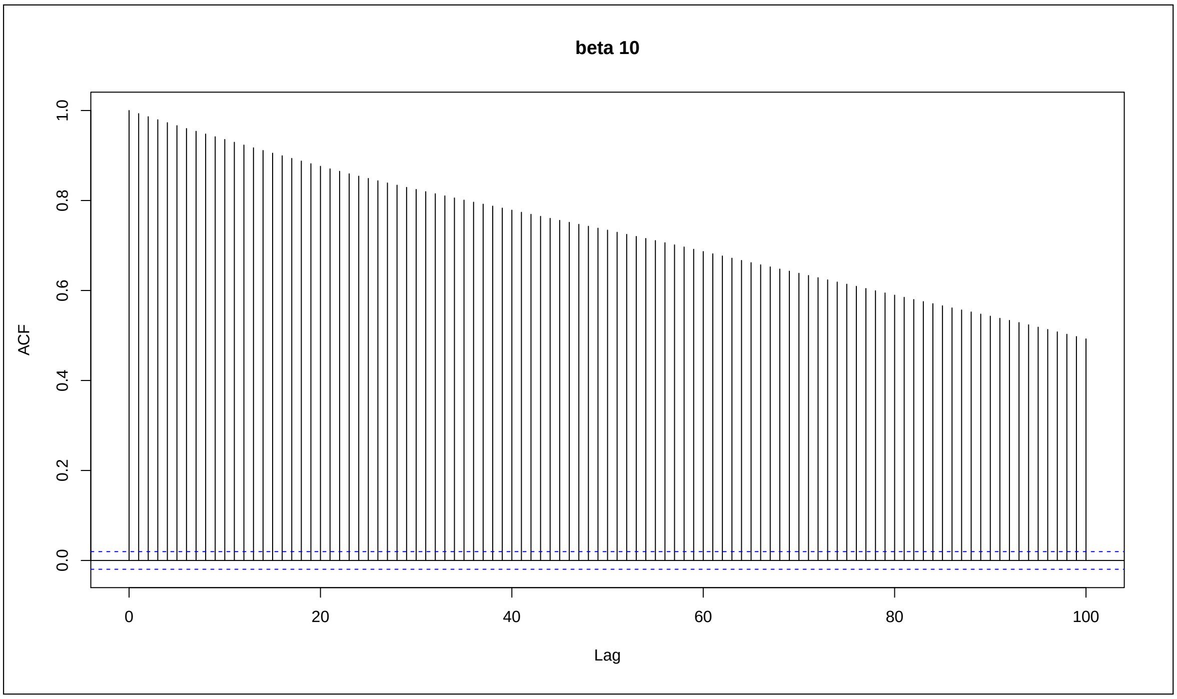

beta 8
0.0
0.1
0.2
0.3
0.4
0
2000
4000
6000
8000
10000
Time

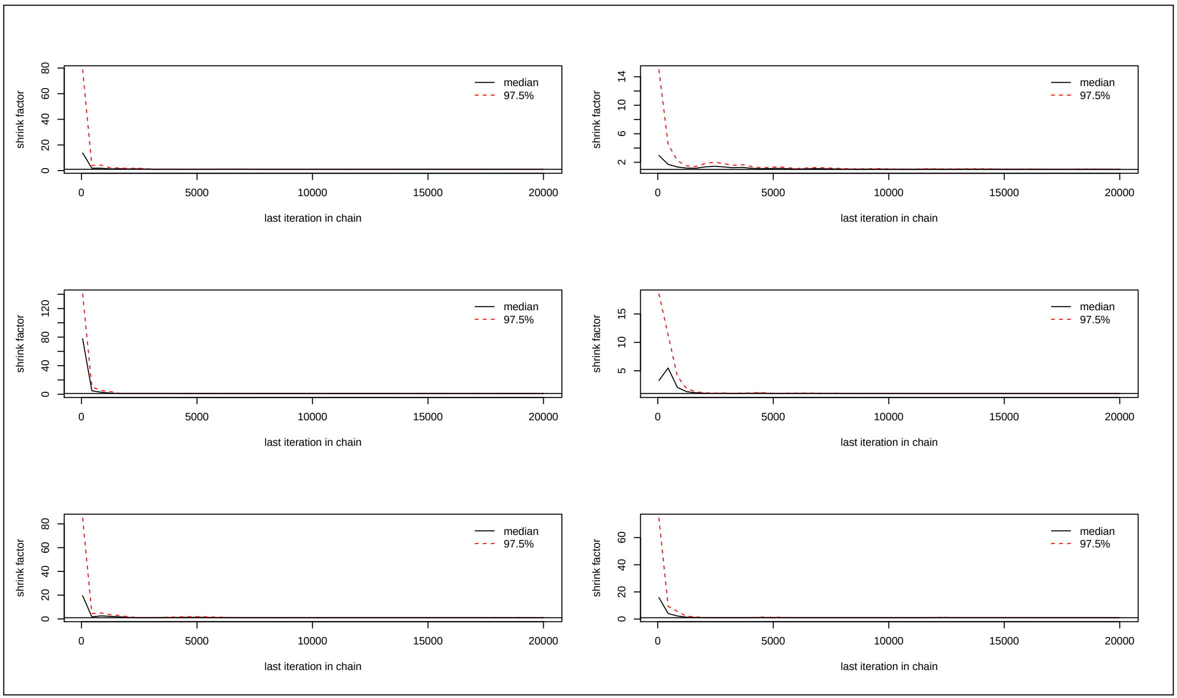

beta 8
ACF
0.0
0.2
0.4
0.6
0.8
1.0
0
20
40
60
80
100
Lag

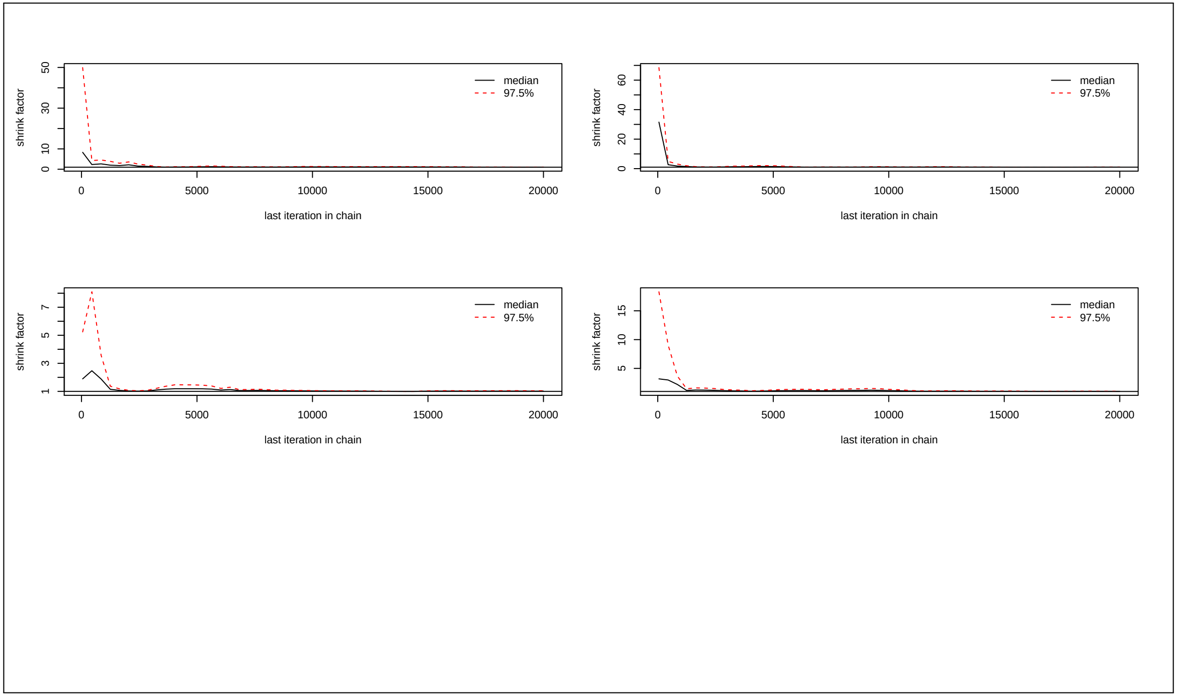

beta 9
0.1
0.0
−0.1
−0.2
−0.3
0
2000
4000
6000
8000
10000
Time


beta 9
ACF
1.0
0.8
0.6
0.4
0.2
0.0
0
20
40
60
80
100
Lag


beta 10
−0.1
−0.2
−0.3
−0.4
−0.5
0
2000
4000
6000
8000
10000
Time


beta 10
ACF
1.0
0.8
0.6
0.4
0.2
0.0
0
20
40
60
80
100
Lag


shrink factor
last iteration in chain
median
97.5%
0
5000
10000
15000
20000
0
20
40
60
80
2
6
10
14
0
40
80
120
5
10
15
0
20
40
60
80
0
20
40
60


shrink factor
last iteration in chain
median
97.5%
0
5000
10000
15000
20000
0
10
30
50
0
20
40
60
1
3
5
7
5
10
15

Glossary of terms

Epigenetics the study of heritable changes in gene expression or phenotype due to processes other than changes in the underlying genetic sequence.

Gene expression The level of protein production from genes.

Gene set A pre-defined group of genes.

Microarray is a collection of microscopic DNA spots attached to a solid surface.

Normalization The process by which raw gene expression data is standardized to minimize experimental error and to indroduce a common scale.

Nucleus A membrane-enclosed organelle found in eukaryotic cells containing most of the cell's genetic material.

Organelle A specialized subunit within a cell that has a specific function.

Phenotype The set of observable characteristics of an individual resulting from the interaction of its genotype with the environment.

Transcription The process of creating a complementary RNA copy of a sequence of DNA.

Translation The process by which messenger RNA (mRNA) is decoded by the ribosome to produce a specific amino acid chain, or polypeptide, that will later fold into an active protein.

List of references

Al-Awadhi, F., M. Hurn, and C. Jennison (2004). Improving the acceptance rate of reversible jump mcmc proposals. *Statistics amp; Probability Letters 69*(2), 189 – 198.

Ashburner, M., C. A. Ball, J. A. Blake, D. Botstein, H. Butler, J. M. Cherry, A. P. Davis, K. Dolinski, S. S. Dwight, J. T. Eppig, M. A. Harris, D. P. Hill, I. L. Tarver, A. Kasarskis, S. Lewis, J. C. Matese, J. E. Richardson, M. Ringwald, G. M. Rubin, and G. Sherlock (2000, May). Gene ontology: tool for the unification of biology. The Gene Ontology Consortium. *Nat Genet 25*(1), 25–29.

Bauer, S., J. Gagneur, and P. N. Robinson (2010). Going bayesian: model-based gene set analysis of genome-scale data. *Nucleic Acids Research 38*(11), 3523–3532.

Benjamini, Y., R. Heller, and D. Yekutieli (2009). Selective inference in complex research. *Phil. Trans. R. Soc. A 367*, 1–17.

Benjamini, Y. and Y. Hochberg (1995). Controlling the false discovery rate: A practical and powerful approach to multiple testing. *Journal of the Royal Statistical Society 57*(1), 289–300.

Brooks, S. P., P. Giudici, and G. O. Roberts (2003). Efficient construction of reversible jump markov chain monte carlo proposal distributions. *Journal of the Royal Statistical Society. Series B (Statistical Methodology) 65*(1), 3–55.

Damian, D. and M. Gorfine (2004, July). Statistical concerns about the GSEA procedure. *Nat Genet 36*(7).

Efron, B. (2005). Local false discovery rates (url:www-stat.stanford.edu/ ckirby/brad/papers/2005localfdr.pdf).

Efron, B. and R. Tibshirani (2002). Empirical bayes methods and false discovery rates for microarrays. *Genetic Epidemiology 23*, 70–86.

Efron, B. and R. Tibshirani (2006). On testing the significance of sets of genes. *The Annals of Applied Statistics 1*(1), 107–129.

Efron, B., R. Tibshirani, J. D. Storey, and V. Tusher (2001). Empirical Bayes Analysis of a Microarray Experiment. *Journal of the American Statistical Association 96*(456), 1151–1160.

Foulkes, A. S. (2009). *Applied Statistical Genetics with R: For Population-Based Association Studies*. Springer. ISBN: 978-0-387-89553-6.

Gelman, A. and D. B. Rubin (1992). Inference from Iterative Simulation Using Multiple Sequences. *Statistical Science 7*(4), 457–472.

Gilks, W., S. Richardson, and D. Spiegelhalter (1996). *Markov chain Monte Carlo in practice*. Chapman & Hall/CRC.

Goeman, J. J. and P. Buhlmann (2007). Analyzing gene expression data in terms of gene sets: methodological issues. *Bioinformatics 23*(8), 980–987.

Green, P. J. (1995). Reversible jump markov chain monte carlo computation and bayesian model determination. *Biometrika 82*, 711–732.

Harris, S. L. and A. J. Levine (2005). The p53 pathway: positive and negative feedback loops. *Oncogene 24*(17), 2899–908.

Hung, J.-H., T.-H. Yang, Z. Hu, Z. Weng, and C. DeLisi (2012, May). Gene set enrichment analysis: performance evaluation and usage guidelines. *Briefings in Bioinformatics 13*(3), 281–291.

Irizarry, R. A., W. Chi, Z. Yun, and T. P. Speed (2009). Gene set enrichment analysis made simple. *Statistical Methods in Medical Research 18*(6), 565–575.

Isci, S., C. Ozturk, J. Jones, and H. H. Otu (2011). Pathway analysis of high-throughput biological data within a bayesian network framework. *Bioinformatics 27*(12), 1667–1674.

Jiang, Z. and R. Gentleman (2007). Extensions to gene set enrichment. *Bioinformatics 23*(3), 306–313.

Khatri, P. and S. Draghici (2005). Ontological analysis of gene expression data: current tools, limitations, and open problems. *Bioinformatics 21*(18), 3587–3595.

Mootha, V. K., C. M. Lindgren, K.-F. F. Eriksson, A. Subramanian, S. Sihag, J. Lehar, P. Puigserver, E. Carlsson, M. Ridderstråle, E. Laurila, N. Houstis, M. J. Daly, N. Patterson, J. P. Mesirov, T. R. Golub, P. Tamayo, B. Spiegelman, E. S. Lander, J. N. Hirschhorn, D. Altshuler, and L. C. Groop (2003, July). PGC-1alpha-responsive genes involved in oxidative phosphorylation are coordinately downregulated in human diabetes. *Nature genetics 34*(3), 267–273.

Nam, D. and S.-Y. Kim (2008). Gene-set approach for expression pattern analysis. *Briefings in Bioinformatics 9*(3), 189–197.

Ogata, H., S. Goto, K. Sato, W. Fujibuchi, H. Bono, and M. Kanehisa (1999, January). KEGG: Kyoto Encyclopedia of Genes and Genomes. *Nucleic acids research 27*(1), 29–34.

Pan, K.-H., C.-J. Lih, and S. N. Cohen (2005). Effects of threshold choice on biological conclusions reached during analysis of gene expression by dna microarrays. *Proceedings of the National Academy of Sciences of the United States of America 102*(25), 8961–8965.

Parmigiani, G., E. Garrett, R. Irizarry, and S. Zeger (2003). *The Analysis of Gene Expression Data*. New York: Springer.

Puzio-Kuter, A. M. (2011). The role of p53 in metabolic regulation. *Genes Cancer 2*(4), 385–391.

R Development Core Team (2011). *R: A Language and Environment for Statistical Computing*. Vienna, Austria: R Foundation for Statistical Computing. ISBN 3-900051-07-0.

Ross, D. T., U. Scherf, M. B. Eisen, C. M. Perou, C. Rees, P. Spellman, V. Iyer, S. S. Jeffrey, M. Van De Rijn, M. Waltham, and et al. (2000). Systematic variation in gene expression patterns in human cancer cell lines. *Nature Genetics 24*(3), 227–235.

Shahbaba, B., R. Tibshirani, C. M. Shachaf, and S. K. Plevritis (2011). Bayesian gene set analysis for identifying significant biological pathways. *Journal of the Royal Statistical Society: Series C (Applied Statistics) 60*(4), 541–557.

Song, S. and M. Black (2008). Microarray-based gene set analysis: a comparison of current methods. *BMC Bioinformatics 9*(1), 502.

Stingo, F. C., Y. A. Chen, M. G. Tadesse, and M. Vannucci (2011). Incorporating biological information into linear models: A bayesian approach to the selection of pathways and genes. *Annals of Applied Statistics 5*, 1978–2002.

Storey, J. (2002). A direct approach to false discovery rates. *J.R.Statist.Soc 64*, 479–498.

Storey, J. (2003). The positive flase discovery rate: A bayesian interpretation and the q-value. *The Annals of Statistics 31*(6), 2013–2035.

Strimmer, K. (2008). fdrtool: a versatile r package for estimating local and tail area-based false discovery rates. *Bioinformatics 24*(12), 1461–1462.

Subramanian, A., P. Tamayoa, V. Mootha, S. Mukherjeed, B. Eberta, M. Gillettea, A. Paulovichg, S. Pomeroyh, T. Goluba, E. Landera, and J. Mesirova (2005). Gene set enrichment analysis: A knowlege-based approach for interpreting genome-wide expression profiles. *Proc. Natl. Acad. Sci. 102*(42), 15545–15550.

Vogelstein, B., D. Lane, and A. J. Levine (2000). Surfing the p53 network. *Nature 408*(6810), 307–310.

Waagepetersen, R. and D. Sorensen (2006). A tutorial on reversible jump mcmc with a view toward applications in qtl-mapping. In *ON QTL MAPPING. INTERNATIONAL STATISTICAL REVIEW*, pp. 49–62.

Zou, M. and S. D. Conzen (2005). A new dynamic bayesian network (dbn) approach for identifying gene regulatory networks from time course microarray data. *Bioinformatics 21*(1), 71–79.