Stockholms
universitet

Estimating convergence of Markov chain Monte Carlo simulations

Kristoffer Sahlin

Masteruppsats i matematisk statistik
Master Thesis in Mathematical Statistics

Masteruppsats 2011:2
Matematisk statistik
Februari 2011

www.math.su.se

Matematisk statistik
Matematiska institutionen
Stockholms universitet
106 91 Stockholm

Mathematical Statistics
Stockholm University
Master Thesis **2011:2**
http://www.math.su.se

Estimating convergence of Markov chain Monte Carlo simulations

Kristoffer Sahlin*

December 2011

Abstract

An important research topic within Markov chain Monte Carlo (MCMC) methods is the estimation of convergence of a simulation. The simulation is divided in to two parts, pre- and post-convergence, where the pre-convergence part known as burn-in is discarded and the post-convergence part is used for inference. Recently, MCMC methods have become a popular way of analyzing phylogenetic models. As more and larger phylogenetic data sets are analyzed, there is a need for automated procedures estimating both convergence and sufficient run length of a simulation. Since MCMC methods are used in a variety of different research fields there are several different methods for evaluating the output of a run. In this thesis, we construct a diagnostic for estimating the burn-in of the chain. We then evaluate this diagnostic together with well known convergence diagnostics used in other fields on simulations performed on three different phylogenetic data sets. We also propose an algorithmic procedure for verifying convergence and sufficient run length of an MCMC-simulation.

[*] Postal address: Mathematical Statistics, Stockholm University, SE-106 91, Sweden. E-mail:ksahlin@kth.se . Supervisor: Sebastian Höhna.

Acknowledgements

I would like to thank my supervisor Sebastian Höhna for a thorough support during the writing of this thesis, for good advice on improving the thesis but also through useful general guidelines about preforming research.

Contents

1 Introduction

1.1 Motivation

In Bayesian inference, one is often interested in finding the posterior probability distribution of some parameter or hypothesis. This posterior distribution can sometimes be very complicated to compute. An alternative to an analytical solution of the posterior probability distribution is to use Markov chain Monte Carlo (MCMC) methods to simulate from the posterior distribution. In order for the sample distribution obtained by an MCMC simulation to make good representation of the true distribution we must rely on the assumption that we can obtain samples from the MCMC simulation with relative frequencies that agrees with sample frequencies obtained from the true distribution.

Due to lack of prior knowledge, the MCMC simulation is often started at a random point in parameter space. This starting point is often far from high density regions of the posterior distribution. Thus, in the early stages of the simulation we will often obtain sample values that are unlikely to occur in samples from the true distribution. This will in turn affect parameter estimates if the chain is not run sufficiently long. When obtaining an estimation for the posterior distribution from an MCMC simulation, the problem is not that the early samples are invalid samples from this distribution, but rather that it is not likely to obtain these values as samples from the true posterior distribution unless our chain is run for a very long time.

One could argue that we do not want to start way out in the tail of some distribution since we want our samples of the chain to be representative of the distribution we are sampling from. What we mean with representative is that highly unlikely sampled values, when having a chain that is fairly short, do not give a good representation of the posterior distribution. However, to determine when this part of the chain is over and we are obtaining samples that are good representations of our posterior distribution is not straight forward. This initial part of the chain is referred to as the "burn-in" period of the chain and the remaining part is called the stationary part or the part where the chain has "converged in distribution".

Since one is often interested in the mean or variance of the posterior distribution of some parameter one tries to get an informative sample that estimates this quantity with high accuracy. The chain is said to have converged with respect to some specified accuracy if this quantity can be measured correct and as accurate as we have specified. This type of convergence (in contrast to convergence in distribution) are often referred to as "convergence of an ergodic estimate" and one often estimates the sufficient run length of the chain in order to obtain a particular accuracy of the ergodic estimate. The burn-in period of the chain will slow down the convergence of an ergodic estimate since it skews the estimates of this quantity. However, we make this burn-in less influential in our sample by having sufficiently many samples from the converged part of the chain.

Thus, there is a relationship between how large part of the chain that is in the burn-in phase and how long it will take for the chain to reach convergence of an ergodic estimate (e.g. the mean value of the parameter sampled). Of course the characteristics of the burn-in (e.g. the magnitude for which the burn-in values differ from the sampled values of the stationary part of the chain) influence this rate of convergence speed but there is nonetheless an underlying relationship between the burn-in period and the rate of convergence of an ergodic estimate.

1.2 Objective

This relationship motivates the use of convergence diagnostics that tries to estimate the number of iterations the chain has spent in this burn-in phase in order to discard these samples from the output. If this estimation is accurate we are able to "speed up" the convergence of the ergodic estimate simply by having reduced the number of "non-representative" observations in our sample. Large data sets often need long simulations due to large parameter spaces that needs to be simulated sufficiently (e.g. the tree topology parameter for which the number of trees grows exponentially with the number of taxa) or due to large autocorrelations between successive samples (decreasing the information obtained from a sample set). Then, finding and discarding burn-in can save a lot of (computational) time since we do not need to run as long simulations as needed for chains containing this initial burn-in period.

An accurate estimate of the burn-in is what we strive for since estimating too short burn-in will leave behind some samples that is in “burn-in phase”, which as we said, slows down convergence. But on the other hand, overestimating the burn-in makes us throw away samples that we could have used to get more accurate estimate for our distribution, thus in a way also slows down convergence. In this thesis, we will therefore focus on finding the optimal burn-in to discard.

1.3 Project outline

In the background section we will begin with an introduction to phylogenetics, Bayesian statistics, MCMC algorithms and the topic of convergence of MCMC output. In the methods section we discuss different diagnostics that can be used for assessing convergence of these simulations. We will then proceed with the analysis section that is split into two parts. In the first part, we evaluate how different convergence diagnostics perform in assessing convergence and estimating burn-in for parameters in phylogenetic models. In the second part, we evaluate if the run length is sufficient for these chains. The analysis is performed on three different phylogenetic data sets of different sizes that contains DNA from related species. In the remaining section we discuss problematic issues that we encountered and also propose an alternative analysis method for the run length test. We finally give an algorithm for verifying that a simulation has be run sufficiently long and that an appropriate burn-in is removed.

2 Background

2.1 Phylogenetics

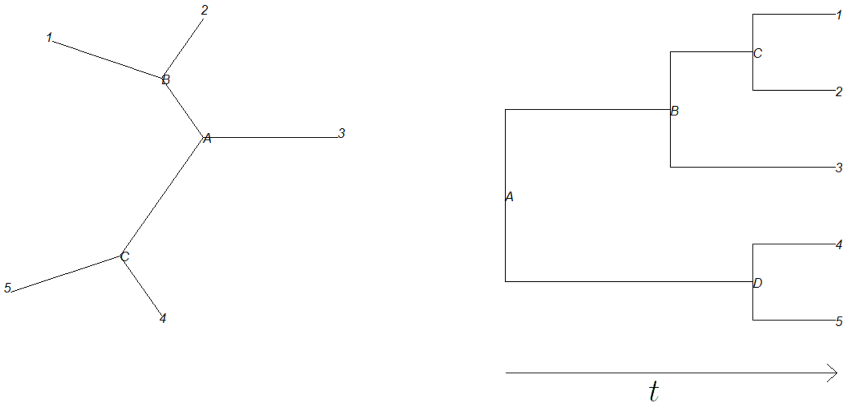
Phylogenetic systematics (or phylogenetics) is the field within biology that deals with identifying and understanding the evolutionary relationships among different kinds of life forms on earth. In evolutionary theory it is believed that similarity among individuals or species is due to common descent, i.e. inheritance from a common ancestor. Thus, the relationships obtained by phylogenetic research often describe the historical relationships among organisms or their parts, such as their genes. The data sets examined in this area are either morphological (warm blooded, unicellular etc.) or molecular data (DNA, RNA, or other genetic information).

When performing research in this area the aim is often to come up with a well supported hypothesis of the evolutionary history between the organisms investigated. This evolutionary relationship is often represented as a binary tree. A binary tree is a graph (vertices that are joined by edges) that is connected, contains no cycles and satisfies the condition that all vertices connect no more than three edges. A leaf is a vertex that has only one edge connected to it. The edges that are not incident with a leaf is said to be internal. The number of edges that are incidental with a vertex is called the degree of a vertex. If the binary tree has only one internal vertex of degree two, while the others are of degree three, then the tree is said to be rooted. Within phylogenetics, one often refer to this binary rooted three as an evolutionary tree. Figure 1(b) shows an example of an evolutionary tree with labeled internal nodes and leafs.

In phylogenetics research, although one is often primarily interested in the the structure of the evolutionary tree, the models one works with contain many other parameters that might be of interest. Examples of other parameters are branch- or total tree lengths (represents the time of evolution), substitutions (changes in the DNA) over time, the rates for each substitution (e.g. from A to C) or speciation times. Figure 1(b) shows a tree with different branch lengths. If the numbers 1-5 are different species and evolutionary time are measured horizontal with present time at species 1 and 2, this picture could for example illustrate that species 3, 4 and 5 has gone extinct.

When trying to find the best hypothesis (evolutionary tree) for a data set of a number of taxa we must turn to some kind of criteria to find this tree. There are a number of different such criteria proposed such as maximum parsimony, minimum evolution and maximum likelihood [8] [1][7][10][9].

This thesis, however, considers a Bayesian inference approach with the aid of Markov chain Monte Carlo algorithms. It is a more recently discovered approach in the area of phylogenetics which incorporates the



(a) A binary unrooted tree consisting of seven edges (two internal) and eight vertices comprising of five leaves (vertices of degree one) labeled 1-5 and three internal vertices (all of degree three) labeled A-C.

(b) A binary rooted tree consisting of eight edges (three internal) and nine vertices comprising of five leaves (vertices of degree one) labeled 1-5 and four internal vertices (the root of degree two and the others of degree three) labeled A-D. This tree also shows different lengths of the branches.

Figure 1: Example of two labeled binary trees.

likelihood function as well as prior information [25],[4].

2.2 Bayesian Inference

Bayesian inference is a part of statistics where we use prior knowledge (that might be arising from previous observations or studies) together with new data to update the probability that a hypothesis is true. This genre of statistics has got its name from the use of Bayes theorem which states

$$P(A|B) = \frac{P(B|A)P(A)}{P(B)}.$$

Here, A represents the hypothesis, $P(B|A)$ is the marginal likelihood function, $P(B)$ is the probability of the data, $P(A|B)$ is our updated probability function (posterior probability) and $P(A)$ is the prior probability of our hypothesis that was inferred before our new information B became available. Applied to the area of phylogenetics, if n is the total amount of possible different evolutionary trees, τ_i is a given phylogenetic tree where $i = 1, \ldots, n$, $\boldsymbol{\theta}$ are the remaining parameters such as tree length or substitution rates and $\mathbf{X}$ is the observations, then Bayes formula can be written as

$$f(\tau_i|\mathbf{X}) = \frac{f(\mathbf{X}|\tau_i)f(\tau_i)}{\sum_{j=1}^{n} f(\mathbf{X}|\tau_j)f(\tau_i)} \tag{2.1}$$

here

$$f(\mathbf{X}|\tau_i) = \int_{\boldsymbol{\theta}\in\boldsymbol{\Theta}} f(\mathbf{X}|\tau_i, \theta)f(\boldsymbol{\theta})d\boldsymbol{\theta}.$$

That is, all other parameters are integrated over such that the likelihood function $f(\mathbf{X}|\tau_i)$ only depends on τ_i.

The denominator above in Bayes theorem for the phylogenetic model is often very difficult to compute since this expression involves integration over all parameter values as well as summing over all trees (which is increasing exponentially with the number of taxa [8]). This computational problem is overcome by the Markov chain Monte Carlo algorithms which we will focus on in the following section.

2.3 Markov chain Monte Carlo algorithms

Markov chain Monte Carlo (MCMC) methods are a class of algorithms which are used for simulating samples from a posterior distribution that has the desired true posterior distribution as its stationary distribution. Briefly described in words we obtain through a Markov chain a large number of samples from the distribution. Since the samples are Markov generated two successive samples are usually strongly dependent. One often samples only every m^{th} $(m \geq 2)$ iteration since the large amount of output that is generated if sampling every generation may be unpractical due to memory constraints of the computer. This procedure is called thinning of the chain. Notice that since successive samples are dependent, one does not loose nearly as much information as discarding independent observations. By the Monte Carlo property we then get our desired posterior distribution by observing the relative frequencies of the samples generated from this chain. We will now mention two of the most well-known MCMC algorithms and see in more detail how they work.

2.3.1 The Metropolis-Hastings algorithm

We explain this algorithm by continuing on the same simplified phylogenetic model given by equation (2.1). As we stated above, we are only considering the tree topology τ as our parameter. Thus, if $\mathbf{X}$ denotes our observations, the desired posterior distribution $\pi(\tau_i|\mathbf{X})$ for some topology τ_i is one dimensional and is given by $\pi(\tau_i|\mathbf{X}) = \frac{f(\mathbf{X}|\tau_i)f(\tau_i)}{\sum_{j=1}^{n} f(\mathbf{X}|\tau_j)f(\tau_j)}$ where the other parameters $\boldsymbol{\theta}$ in the model are integrated over. To obtain a Markov chain[1] $\{\tau^1, \tau^2, \ldots, \tau^t, \ldots\}$ with equilibrium distribution $\pi(\tau_i|\mathbf{X})$ the Metropolis-Hastings algorithm [24] works as follows. Let $Q(\tau^*|\tau^t)$ be the proposal density which is a function that depends on the current state τ^t and generates a new proposed sample τ^*. This proposed value τ^* is accepted as the next value τ^{t+1} if a value $\alpha^t \in U(0,1)$ satisfies

$$\alpha^t < \min\left\{\frac{Q(\tau^t|\tau^*)}{Q(\tau^*|\tau^t)}\frac{\pi(\tau^*|\mathbf{X})}{\pi(\tau^t|\mathbf{X})}, 1\right\} \tag{2.2}$$

If the proposal is not accepted, then $(\tau^{t+1}, \boldsymbol{\theta}^{t+1}) = (\tau^t, \boldsymbol{\theta}^t)$. That is, the chain remains in its current state in time $t+1$. The proposal function might generate new values of a parameter that are accepted with very low probability, thus the parameter will stay in the same state for long time periods before moving. Poor implementations or an algorithm that is stuck in the peak of a very highly peaked distribution are examples where this may be the case.

A good property of this algorithm is that it completely ignores the normalization factor which can be very time consuming to calculate (see A.1).

One particular form of MCMC algorithm is called the Metropolis Coupled MCMC algorithm (MC^3) [14] which runs m chains in parallel with different stationary distributions $\pi_j(\cdot), j = 1, 2, \ldots m$. The first chain $\pi_1(\cdot)$, called the cold chain, has the desired stationary distribution and the others, called the heated chains, have stationary distributions on the form

$$\pi_j(\cdot) = \pi(\cdot)^{a_j}$$

where $a_j < 1$. The algorithm proposes a swap of states between two randomly chosen chains after each iteration. Raising the density with a power less than one flattens out the distribution. Thus, we get by the swapping of states that the chain with the correct target distribution are more likely to explore the state space better and traverse between possible peaks of the target distribution.

2.3.2 The Gibbs sampler

In the previous section we looked at an example where the desired posterior distribution was one dimensional (we focused only on the tree topology parameter). The Metropolis-Hastings algorithm will work when the

[1] Here superscript is used to denote iteration step while subscript denotes a particular tree topology.

desired distribution is multidimensional as well. We will here discuss a special case of the Metropolis-Hastings algorithm known as the Gibbs sampler [12],[11], which is used to generate samples from a joint probability distribution. The algorithm can be applied when the joint distribution is not known explicitly but the conditional distribution of each parameter is known.

We here consider a two-dimensional case which can be generalized. Let (X, Y) be a bivariate parameter in some model. Suppose we wish to compute one or both marginal densities, $p(X)$ and $p(Y)$ or the joint distribution $p(X, Y)$. The sampler is based on the idea that it is easier to consider the conditional distributions, $p(X|Y)$ and $p(Y|X)$, than it is to obtain the joint density $p(X, Y)$. The sampler starts with some initial value y^0 for Y and thereafter obtains x^0 by generating a parameter value of X from the conditional distribution $p(X|Y = y^0)$. The sampler then uses x^0 to generate a new value y^1, using the conditional distribution with the value x^0 as condition, $p(Y|X = x^0)$. This procedure is repeated by the two following steps

$$\begin{aligned} x^i &\sim p(X|Y = y^{i-1}) \\ y^i &\sim p(Y|X = x^i). \end{aligned}$$

If this process is repeated m times, it gives a sequence of m points where the points are vectors of two parameters (e.g. (x^j, y^j)).
This process is extended when $l > 2$ parameters are involved. In particular, the value of the kth parameter at the i^{th} iteration is drawn from the distribution $p(\theta^i_{(k)}|\boldsymbol{\theta}^i_{(-k)})$ where

$$\boldsymbol{\theta}^i_{(-k)} = \big(\theta^i_{(1)}, \ldots, \theta^i_{(k-1)}, \theta^{i-1}_{(k+1)}, \ldots, \theta^{i-1}_{(l)}\big)$$

denotes a vector containing all of the parameters but k (note how parameters with index $< k$ have simulated values for iteration i in the algorithm while the parameters with index $> k$ have values from the $(i-1)$th iteration). Thus, during the ith iteration of a sample point, we get the value of the kth parameter $\theta^i_{(k)}$ from

$$\theta^i_{(k)} \sim p\big(\theta_{(k)}|\theta_{(1)} = \theta^i_{(1)}, \ldots, \theta_{(k-1)} = \theta^i_{(k-1)}, \theta_{(k+1)} = \theta^{i-1}_{(k+1)}, \ldots, \theta_{(l)} = \theta^{i-1}_{(l)}\big).$$

The Gibbs sampler is a special case of the Metropolis-Hastings algorithm where the proposed random vector $(\theta^*_{(1)}, \ldots, \theta^*_{(l)})$ is always accepted (see Appendix A.2).

2.4 Estimating convergence

As we mentioned in the introduction, two different types of convergence occurs in literature of these matters, convergence in distribution and convergence of ergodic estimates. Therefore it is important to clarify this matter.

- ***Burn-in**: Let us denote the burn-in as the part of a chain where the current state of the chain is dependent on its starting point. We let the iteration n_0 denote the cut-off point where the chain looses this dependence.*

Here we say that after n_0 iterations are removed, the remaining part of the chain has converged in distribution or has reached the stationary phase. This definition is also consistent with our informal discussion of the burn-in earlier, namely as part of the chain where the samples cannot be considered to give a good representation of the posterior distribution. We will also treat the convergence of an ergodic estimate.

- ***Run length**: For some ergodic estimate ρ, we say that we have convergence of ρ if ρ can be estimated correctly with an accuracy level higher than some specified accuracy level. We say that the chain has a sufficient run length if we have convergence of ρ.*

Noticed that we have not specified any accuracy measure/threshold here since we will investigate different types of measures. In this thesis we will only treat the convergence of the mean estimate. The definition of run length says that we have convergence of the mean estimate or equivalently, sufficient run length of the chain when we have a correct and accurate measure (with respect to some specified accuracy threshold) of the true mean value. Since we must have sufficiently many samples from the converged part of the chain for this, we refer to this estimation as estimating the run length of the chain with respect to some accuracy level for the mean. We denote this suggested run length n and it, of course, satisfies $n > n_0$.

Getting a sufficient sample for a parameter estimate can be a time consuming task for large data sets. The reason that the topic of estimating a correct burn-in has gotten so much attention is that it is a way to reach convergence of an ergodic estimate faster. As we said in the introduction, discarding a too short burn-in period will leave some "unrepresentative" samples in the chain. Since these samples often differ significantly in value from samples of high posterior density areas, they will be the largest contributors to the standard deviation of the parameter estimate. This gives that the convergence w.r.t. some accuracy level (often a confidence interval) takes longer time (i.e. requires more samples from the converged part of the chain) to reach. Estimating a too large n_0 will instead make us to throw away representative samples that could be used for improving accuracy for the parameter estimate. Thus, we see that the true run length is related to the burn-in in the way that discarding a correct burn-in will often make us reach n faster. This relationship motivates the use of burn-in estimation.

We want the burn-in estimator n_0 to be such that after the first n_0 samples are discarded, the remaining samples of the chain give an as accurate (and of course correct) estimate of the parameter as possible. Since we focus on the mean of a parameter in this thesis, we measure accuracy by the width of the confidence interval of the mean or equivalently, the size of the standard error of the mean SEM. Therefore, we say that the best burn-in estimate is the one that gives the smallest SEM of the parameter estimate. Since we are considering correlated samples the SEM must be calculated somewhat different from the case where we have independent samples. Formulas for calculating SEM are found in B.4.

3 Methods

There are several methods for estimating n_0 and n availible in the literature [5], [6]. The more theoretical methods involve determination of bounds between the transition probabilities of MCMC chains and the stationary distribution [17]. There are also more empirical methods based on results of time series and data analysis. Papers such as [5], [23] compare different convergence diagnostics applied to academic examples or practical examples in areas other than phylogenetics. We will here discuss three well known practical methods for estimating burn-in applied to MCMC output. Also, we construct a burn-in estimate diagnostic that is based on evaluation of how much information a sample of a parameter contains. Furthermore we briefly discuss three run length diagnostics. Most of these diagnostics are implemented in the package coda [18] for R [19] which we have used in this thesis.

3.1 Burn-in estimators

3.1.1 Geweke

This diagnostic compares the location of the sampled parameter on two different time intervals of the chain. If the mean values of the parameter in the two time intervals are somewhat close to each other we can assume that the two different parts of the chain have similar locations in the state space, and it is assumed that the two samples come from the same distribution. Usually one compares the last half of the chain, which is assumed to have converged (in order for the test to make sense), against some smaller interval in the beginning of the chain. The Geweke diagnostic uses spectral density estimation for this analysis. The estimation of the spectral density is used for detecting periodical behaviors in data but will not be covered in this thesis.

To describe this diagnostic more precise, we introduce some parameters. Let $\theta(X)$ be some functional and we denote $\theta^t = \theta(X^{(t+n_0)})$. Here n_0 is the start iteration from where we want to test if the chain has converged. If we define $A = \{t; 1 \leq t \leq n_A\}$, $B = \{t; n^* \leq t \leq n\}$ where $1 < n_A < n^* < n$ and $\frac{n_A+n_B}{n} < 1$,

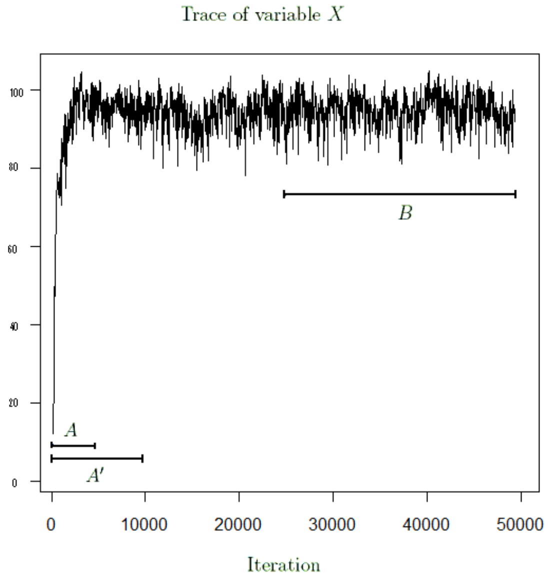


Figure 2: This figure shows a trace plot of a parameter X and a first window of two different sizes A and A' that is tested in the Geweke diagnostic against the second window B. We here see that window A' is chosen too large and the mean of the values in this window will be too close to the mean of the values in window B, thus the Geweke diagnostic does not discover the initial burn-in. However, window A is in suitable size to detect the burn-in.

let

$$\overline{\theta}_A = \frac{1}{n_A}\sum_{t\in A}\theta^t \qquad \overline{\theta}_B = \frac{1}{n-n^*+1}\sum_{t\in B}\theta^t.$$

Here $\overline{\theta}_A$ and $\overline{\theta}_B$ are means for the two different time periods. Furthermore, if the MCMC process and and the functional θ^t imply an existence of a spectral density $\widehat{S}_\theta(0)$ for this time series with no discontinuities at frequency 0, then $\widehat{S}^A_\theta(0)/n_A$ and $\widehat{S}^B_\theta(0)/(n-n^*+1)$ are the asymptotic variances of $\overline{\theta}_A$ and $\overline{\theta}_B$. Hence, the square root of these asymptotic variances estimates the standard error of $\overline{\theta}_A$ and $\overline{\theta}_B$.

This diagnostic suggests that if the chain has converged at time n_0, that is, both of the subsamples are drawn from the stationary distribution of the chain, then the two means $\overline{\theta}_A$ and $\overline{\theta}_B$ should be equal and Geweke's statistic has an asymptotically standard normal distribution. That is

$$Z_n = \frac{(\overline{\theta}_A - \overline{\theta}_B)}{\sqrt{\frac{1}{n_A}\widehat{S}^A_\theta(0) + \frac{1}{n-n^*+1}\widehat{S}^B_\theta(0)}} \longrightarrow N(0,1) \qquad n\to\infty.$$

This result gives that we can test the null hypothesis of equal location in the state space for θ. The null hypothesis is rejected if $|Z_n|$ is large and this indicates that the chain has not yet converged by time n_0. One procedure is then to let n_0 be a greater value and apply the test again. However, one must be careful with repeated hypothesis testing like this since each test carries some uncertainty. More specific we have a certain probability of a type II error in each test, the probability to accept a false null hypothesis. In this case the null hypothesis is that the sample means of A and B are equal. The repetitive hypothesis testing will increase the total probability of this error.

Furthermore, the length of the so called "windows" A and B is open for the user to specify and we will see in our analysis that the diagnostic is sensitive to these specifications. Too wide A will some times "hide" the burn in part within the converged part of the chain and the difference in means may not be large enough for the diagnostic to give an indication of non convergence. On the other hand, a too narrow A will be very sensitive to where the parameter is in state space at that time since it does not contain many samples. This especially occurs if the parameter is slow mixing, i.e. takes long time to travel in state space. Figure 2 shows a trace plot of a parameter X with two different sizes of the first window A and A' (which may affect the outcome of the test) and a second window B.

We will for the Geweke diagnostic use the pre-determined settings in coda [18]. That is, window sizes of $A = \frac{n}{10}$ and $B = \frac{n}{2}$ for window A and B. We choose these sizes because they are suggested by Geweke 1992 [16]. First, the statistic is applied to the whole chain, if the Z-statistic is outside the 95% confidence interval, we continue to apply the diagnostic after discarding $10\%, 20\%, 30\%$ and 40%. If the Z-statistic is still outside 95% confidence interval for the last test, the chain is reported as “failed to converge”.

3.1.2 Heidelberger-Welch

The Heidelberger-Welch diagnostic [15] is based on the assumption that we have got a weakly stationary process (see B.1) when the chain has reached convergence. A weakly stationary process has the properties that, if X^j is defined as the j^{th} iteration in the sequence, the mean function $E[X^j]$ is constant in time and $Cov(\theta^j, \theta^{j+s})$ does not depend on j but only on the size of s. This is a sensible assumption since our sequence is generated by a Markov chain and therefore satisfies full stationarity.

Let $S(0)$ denote the spectral density of the output sequence evaluated at frequency 0 obtained from the second half of the sequence to avoid estimation error that comes from an possible initial transient phase of the chain. Let also

$$Y_0 = 0, \quad Y_n = \sum_{j=1}^{n} X^j, \quad \overline{X} = \frac{1}{n}\sum_{j=1}^{n} X^j$$

and define

$$B_n(t) = \frac{Y_{\lfloor nt \rfloor} - \lfloor nt \rfloor \overline{X}}{(n\widehat{S}(0))^{1/2}} \qquad 0 \le t \le 1,$$

where $\lfloor a \rfloor$ denotes the greatest integer less or equal to a.
Then, under the null hypothesis of stationarity for large n, $B_n = \{B_n(t), 0 \le t \le 1\}$ will be distributed approximately as a Brownian bridge and the Cramer-von Mises statistic $T = \int_0^1 B_n(t)^2 dt$ can be used to test this. $T >$ tabulated value rejects that B_n shows similar shape to a brownian bridge thus rejecting the hypothesis of stationarity of the chain.

To estimate the length of the burn-in, an iterative procedure is suggested by Heidelberger-Welch [15] that is based on repetitive hypothesis testing of the constructed Cramer-Von Mises statistic for different parts of the chain.

Once again it should be noted as for the Geweke statistic, that this test-procedure uses repetitive hypothesis testing. The test has also very little power to detect a transient phase of the chain when the whole output sequence is within this transient phase.

To get an intuitive feeling of how the $B_n(t)$ is constructed we see an example in Figure 3 which is constructed from 1 000 simulated values from a random variable $X \sim N(1, 4^2)$. Observe that these simulated values are independent of each other, therefore we do not have any autocorrelation. Due to this independence, the trace of X shows nice behavior considering the mixing of the chain.

We use coda’s [18] procedure with the Heidelberger-Welch diagnostic. That is, the whole output sequence is tested with Cramer-Von Mises statistic to see if the output can be assumed to be distributed as a Brownian Bridge. If the null hypothesis of stationarity is rejected, the test is repeated with $10\%, 20\%, \ldots$ of the chain discarded. This procedure continues until either stationarity is reached for some subsequence or if 50% of the sequence has been discarded. The latter outcome results in a ”failure of stationarity” of the chain.

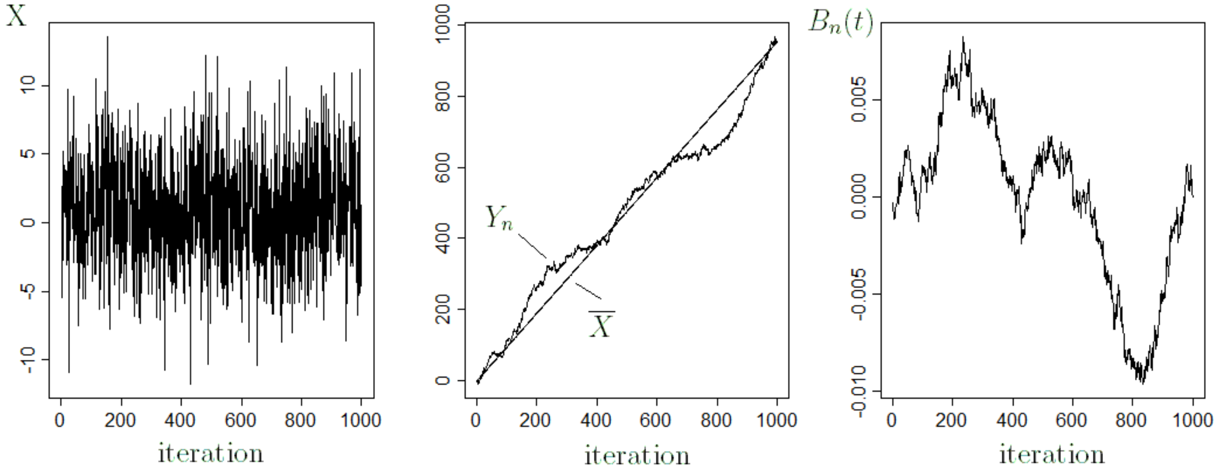


Figure 3: The first plot in this figure shows the trace of a random variable $X \sim N(1, 4^2)$. The second plot shows how the two quantities Y_n and $\overline{X}$ which are used in the $B_n(t)$-transformation develops as the number of iterations increases for this particular simulation. The third plot shows the complete transformation $B_n(t)$ which Heidelberger-Welch suggests should be tested with Cramer-von Mieses statistic. The test basically tries to verify it the trace of $B_n(t)$ behaves like a brownian bridge.

3.1.3 Raftery-Lewis

Suppose that we monitor the parameter $\theta(X)$ and we are interested in estimating the value of u such that

$$P(\theta(x) \leq u) = q,$$

for some quantile value q. Since we are estimating this value u we choose which precision r and probability p we want to have on our estimate $\widehat{u}$. That is, we want that $\widehat{u} \in [u - r, u + r]$ for some probability p. These estimates are useful to obtain since by looking at a few different quantiles q, and the corresponding values of u, we can get a picture of what the distribution of the parameter $\theta(X)$ looks like. Figure 4 shows these quantities in a more easy interpretable way.

Raftery and Lewis [20] propose a method that calculates the total run length and the estimated burn-in of the chain in order to estimate the above probability within the required accuracy r and probability p. The method to calculate this is as follows.

First they introduce the indicator function $Z_t = I(\theta(X^t) \leq u)$. Then they go ahead and determine the value of $s \geq 1$ such that the thinned out sequence Z_{st} approximates a finite state first order Markov chain (see Appendix B.2), where $Z_{st} = Z_{1+(t-1)s}$. With this construction Z_{st} now have a transition matrix of the following form

$$P = \begin{pmatrix} R_{00} & R_{01} \\ R_{10} & R_{11} \end{pmatrix}$$

Where we obtain our approximation of the transition matrix as

$$\widehat{R}_{ij} = \frac{\#\{t : Z_{st} = j, Z_{s(t-1)} = i\}}{\#\{t : Z_{s(t-1)} = i\}} \qquad i, j = 0, 1.$$

The length of the burn in period n_0 can be obtained from

$$|P(Z_{n_0} = i | Z_0 = j) - \pi_i| \leq \epsilon$$

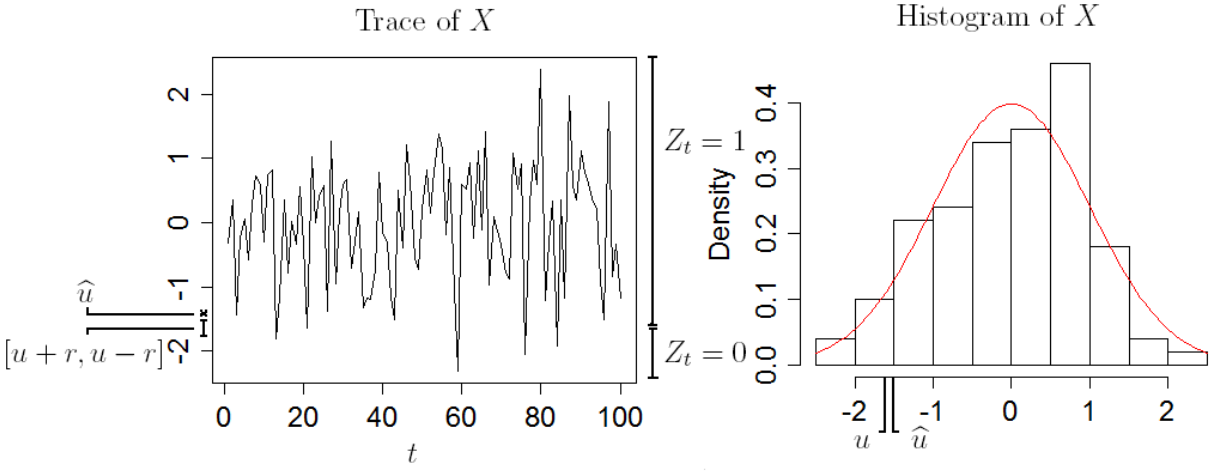


Figure 4: *Note that this is only a graphical interpretation of the values occurring above and does not explain the method of the Raftery-Lewis diagnostic.* In the first plot we se the trace for 100 iterations of the random variable $X \sim N(0,1)$ and in the second plot we see the histogram for the simulated values as well as the normal density curve for X. In this graphical example we have that $\theta(X) = X$, $q = 0.05$ (i.e. the 5% quantile) and $r = 0.1$. This gives $P(x \leq u) = 0.05$ and since $X \sim N(0,1)$ we know that the true u is -1.645. We see that for this simulation our estimate $\widehat{u}$ (which we got to -1.521) is not contained in the interval $[u-r, u+r] = [-1.745, -1.545]$. With more simulations $\widehat{u}$ would probably be contained in this interval. In this example we know the distribution of Z_t since we know u but in the Raftery-Lewis diagnostic it is estimated from the thinned out process Z_{st}.

where π_i is the probability of the stationary distribution to be in state i (equilibrium probability for state i) obtained from the estimated transition matrix P and ϵ is the precision of estimated time to convergence. Using the above formulas with results from two state Markov chains one now obtains values of the estimated burn in n_0 of the chain and the total run length n for for the chain in order to obtain some specified accuracy of $\widehat{u}$ for some quantile q (for formulas, see B.3).

Brooks and Roberts [6] suggests that, when one is interested in a number of different quantiles, it is sometimes most sensible to apply this diagnostic to a number of different q-values and take the largest of the estimated burn-in lengths. Since we are interested in the convergence in distribution here, we apply this diagnostic to the quantiles $\{0.1, 0.2, \ldots, 0.9\}$ and choose the largest burn-in estimate from these.

3.1.4 The Effective Sample Size

After having obtained a set of samples, say $\mathbf{x}$, for some parameter by an MCMC algorithm, one might ask the question: *"How much information do we actually have about this parameter?"*. If there are some correlation between successive samples in $\mathbf{x}$, then we might expect that our sample has not revealed as much information of the posterior distribution of our parameter as we could have gotten if the samples in $\mathbf{x}$ were independent. This "information size" is often estimated when working with MCMC output and are usually called the Effective Sample Size (ESS). The ESS is a quantity that estimates the number of independent samples obtained from $\mathbf{x}$. This quantity can be used as a run length diagnostic, e.g. it is frequently used by biologists to determine if they have obtained a sufficient estimation of some parameter in a phylogenetic model. In practice, they discard some predetermined burn-in value (e.g. 10%, 25% or 50% of the total chain length). Then, they check if the ESS is greater than some threshold value for the remaining part of samples. If so, they accept the simulation length as sufficient.

In this thesis, we have constructed the burn-in estimate ESS_{max} which is based on the method of

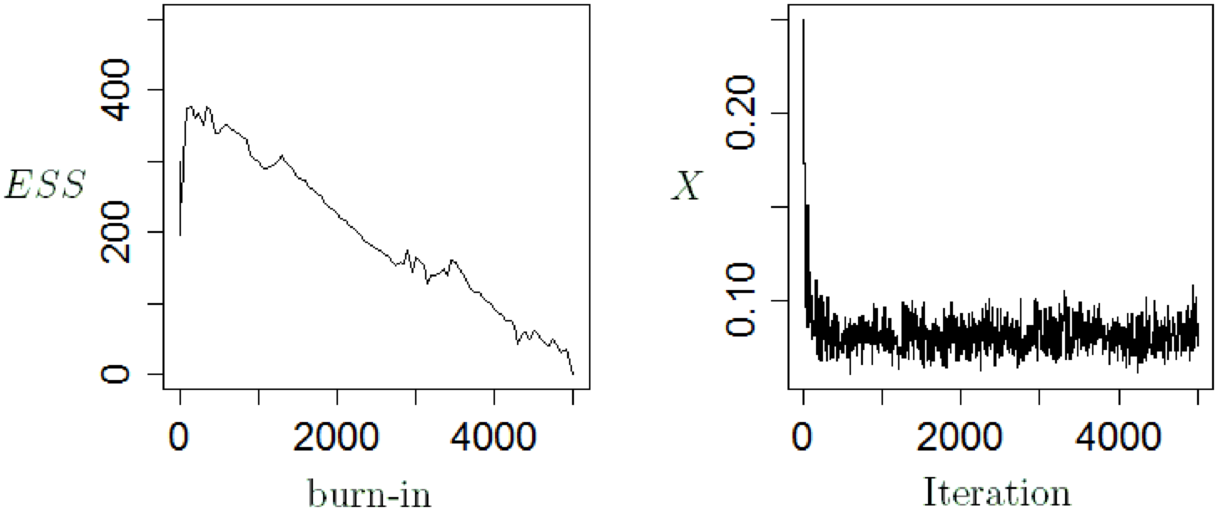


Figure 5: This figure illustrates the relationship between ESS and the trace of the chain for a simulation of a variable X. We see that ESS_{max} is obtained for a burn-in of about 200 iterations. At the same time the chain seems to reach the stationary part. After that, the ESS is decreasing somewhat evenly as longer burn-in parts are discarded. This could be an indication that the stationary part of the chain is reached and that we are throwing away more and more valuable information of our posterior distribution.

estimating the amount of information obtained from a sample. ESS_{max} is defined as the largest effective sample size obtained from a chain when discarding different burn-in lengths. That is, for a chain of n_{tot} samples, the ESS_{max} function is defined as

$$ESS_{max} = \max_{0 \leq n \leq n_{tot}} \big(ESS(n)\big).$$

That is, we can think of the ESS as being a function of burn-in length n_0 as $ESS(n_0)$. Then for different numbers of n_0 as input we get the estimated number of effective samples of the parameter as output. Then for some burn in n'_0, we reach the maximum value of the estimated number of effective samples ESS_{max}. This value of n'_0 is used as the suggested burn-in by ESS_{max}.

Figure 5 shows an example of how the ESS quantity varies with different burn-in sizes discarded from a MCMC simulation. One can see a clear relationship between the ESS trace and the trace of X.

There are several slightly different ways of estimating this ESS quantity, we have got our ESS estimate following the calculations from the MCMC trace analysis program Tracer [2]. Formulas are found in B.4.

For the ESS_{max} diagnostic, we find an approximate ESS_{max} of each chain by calculating the ESS for a number of different burn-in sizes discarded.

3.2 Run length diagnostics

As with burn-in estimators there are several run length diagnostics. We will treat three of them in this paper.

3.2.1 Stability test

We refer to

$$SEM/\widehat{\mu} < c$$

as the stability test of the parameter mean, where SEM is the standard error of the mean. We see that the test allows for larger variance if the mean increases. Thus this test is related to measure the accuracy with a *relative error* (see Appendix B.5). The relative error is an important approximation error since journals in applied science often only accepts analysis if the relative error is smaller than than some boundary value.

3.2.2 Sample size test

We refer to

$$ESS > k$$

as the sample size test. This diagnostic was mentioned in section 3.1.4. It is based on the information considered to be obtained from a sample. The relation of formulas in Appendix B.4 gives that

$$ESS > k \Longleftrightarrow SEM < \sqrt{\frac{\widehat{s}^2_{ML}}{k}}$$

Where $\widehat{s}^2_{ML}$ is the maximum likelihood estimator of the variance of the trace.

3.2.3 Raftery-Lewis

We explained both the Raftary-Lewis burn-in and run length diagnostic in section 3.1.3. However, to see the relation of the run length diagnostic with the other two run length diagnostics recall that with probability $p = 0.95$ we want

$$P(\theta(x) \leq \widehat{\mu}) \in [0.5 - r, 0.5 + r].$$

Or expressed differently

$$|P(\theta(x) \leq \widehat{\mu}) - 0.5| \leq r.$$

Thus, this test is an *absolute error* test(see Appendix B.5).

3.2.4 Implementation

Note that our three run length diagnostics slightly differ in how they are testing appropriate run length. Two of them are based on testing relative error, the stability test that varies with the mean estimate and the sample size test that varies with the ML sample variance estimate. The third (Raftery-Lewis test), is based on an absolute error test. We will implement these diagnostics as follows. After the burn-in estimation is done, we will choose the diagnostic that performs the best on average with respect to the criterium of giving the smallest SEM (see section 2.4). The burn-in values suggested by this diagnostic will then be removed from the chains. We then apply the run length diagnostics on the remaining part of the chains. These three run length diagnostics will be tested with different values of the accuracy measures c, k and r to see which values that maximizes the percentage of correct run length assessments.

Here, we say that a correct run length assessment is an assessment (convergence/non convergence) that agrees with the result of “true mean-test”. The true mean-test is a test for verifying sufficient run length having the true mean of the parameter at hand. We have obtained an accurate estimate μ_{ref} of the true mean through several long "reference chains". We then check if μ_{ref} is inside the confidence interval of the mean estimate for the chain. If not, we conclude that the run length is insufficient (see B.6 for more exhaustive review of the true mean-test).

4 Analysis

As we have mentioned earlier, we first investigate what can be regarded as an appropriate burn-in for our simulated chains. We are aiming for an accurate measure of the mean estimate (narrow confidence interval) of a parameter and will thus consider a good burn-in diagnostic to give a small SEM of the parameter.

We then investigate how the three run length diagnostics makes assessments about sufficient run lengths. Here we use the true mean-test as an indicator that can tell us if convergence of the mean estimate has *not* been reached, simply by indicating that μ_{ref} is outside the confidence interval of the sample mean of the chain. In case μ_{ref} is inside this confidence interval it is up to the run length diagnostic to conclude if the chain has reached its given accuracy level. Thus, the true mean-test is used mainly to see how many assessments of sufficient run length we get in the cases where μ_{ref} does not belong to our parameter estimate's confidence interval.

The general question we want to answer when performing the run length analysis is: *How do we choose different accuracy level constants (i.e. r,k and c) so that we get as similar results to the true mean-test as possible?.* However, when difference occurs between the run length diagnostics and the true mean-test, the more important case is when a diagnostic assesses sufficient run length while we still have a confidence interval of the sample mean that does not contain μ_{ref} (indicating a false assessment of convergence of the mean estimate).

4.1 Data sets and parameters

We will work with three data sets of different sizes provided by MrBayes [22]. The three sets contains a DNA sequence (e.g. a gene) for different numbers of related species. The first data set contains 12 different primates, the second data set contains 30 different lizards of the genus anolis and the third data set contains 54 different flies belonging to the genus drosophila (fruit flies). The phylogenetic models for these data sets can be found in the manual of MrBayes [22].

Table 1 shows information of the simulations used in this thesis. As we mentioned, the reference runs are for obtaining reliable and accurate mean estimates μ_{ref} that are used in the true mean-test. The chains we have performed the analysis on is referred to as test chains. Notice that the sample frequency (number of iterations between two successive samples) of the test chains are chosen smaller than the reference runs to preserve the correlation between the samples in order to study the effects of this.

Data set	primates	anolis	adh
♯ taxa	12	30	54
Length of reference runs	1 000 000	5 000 000	10 000 000
Sample frequencies reference runs	50	250	1000
Total ♯ of reference samples	100 000	100 000	50 000
Length of test runs	100 000	500 000	1 000 000
Sample frequencies test runs	20	100	200
♯ of samples test runs	5001	5001	5001

Table 1: Description of data sets

We have chosen to investigate two parameters that are commonly included in a phylogenetic model. The first parameter is the probability of each site in the alignment (when data is DNA or RNA) to be invariant (does not change throughout the evolution of species considered). This parameter is denoted p_{inv}. The second parameter is the stationary nucleotide frequency parameter for nucleotide G, $\pi(G)$. It represents the frequency of how often you would expect nucleotide G to occur at stationary distribution.

4.2 Burn-in estimation/convergence in distribution

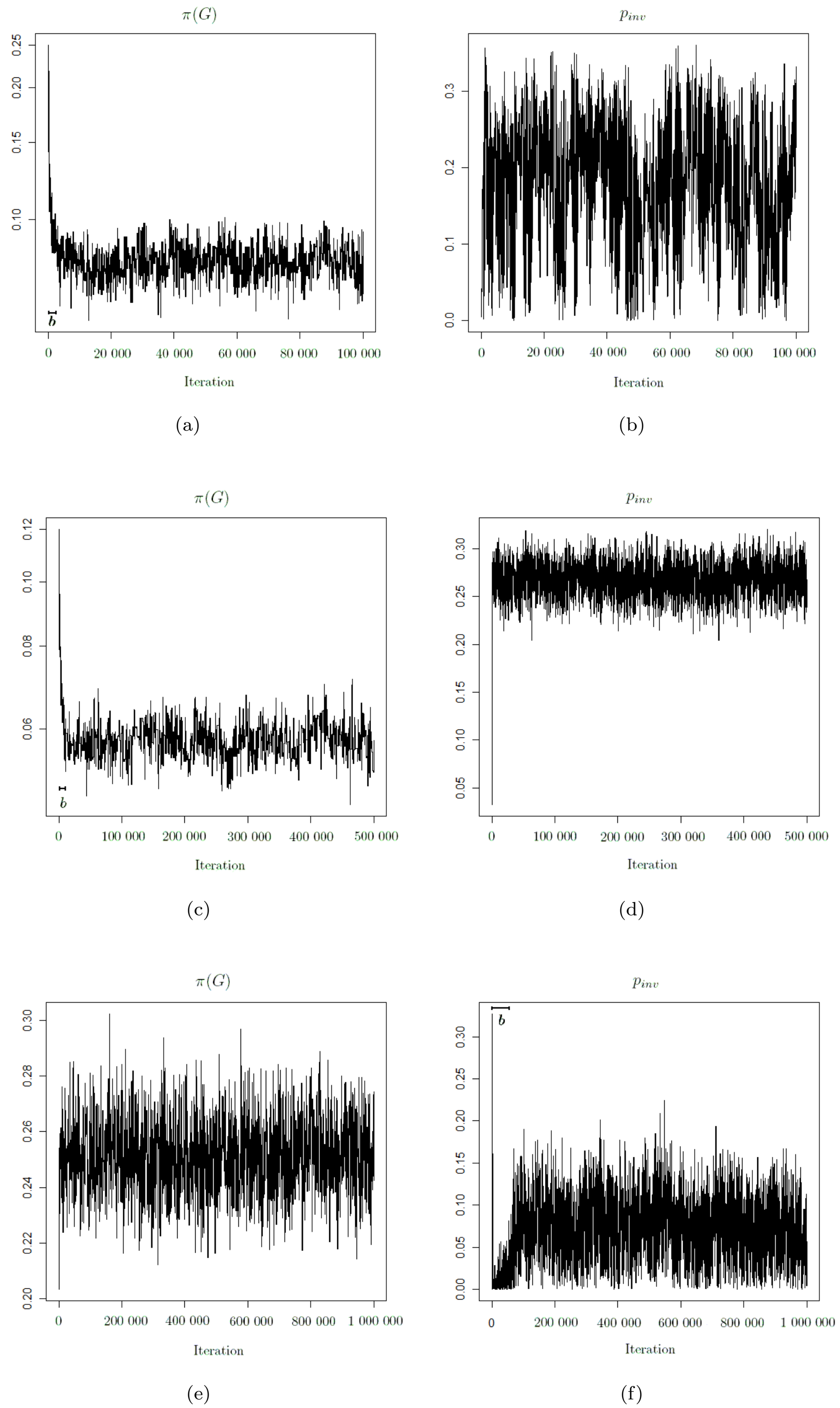
4.2.1 Results

Figure 6 shows examples of how the traces looks like, all 10 traces within a given parameter and data set show similar characteristics. In the beginning of the chains 6(a), 6(c) and 6(f) we see a period constituting a part or whole of the burn-in of the chains that is marked by b. However, one should not be to hasty with drawing conclusions that this is always the optimal burn-in to discard (remember our discussion in section 2.4). 6(b), 6(d) and 6(e) has no obvious (for the eye) burn-in periods and in these cases the help of diagnostics are necessary.

Table 2 shows the burn-in estimations given by the four diagnostics. The estimations with the smallest SEM are showed in bold-faced numbers. We see that in most of the cases, the ESS_{max} is giving the smallest SEM.

When comparing the estimated burn-ins given in the tables with the trace plots we see that the Raftery-Lewis diagnostic has a tendency to underestimate the burn-in. This is seen by comparing Figure 6(a), 6(c) and 6(f) together with the burn-in estimates given by the Raftery-Lewis diagnostic in Table 1(a), 1(c) and 1(f). In most cases ESS_{max} shows a clear relationship between the estimated n_0 and the end of the "obvious" visual part belonging to the burn-in. Examples of this are the estimations of 10 000-40 000 iterations in the traces of $\pi(G)$ in data set anolis and 50 000-100 000 iterations in the traces of p_{inv} in data set adh, as well as the close to zero burn-in estimates of traces 6(d) and 6(e). There are however a few exceptions where the ESS_{max} has found a maximum of the ESS far from the burn-in where we "by eye" conclude that a reasonable burn-in estimate should be. Examples of this are the two 700 000 iteration burn-in estimations in Table 1(f) as well as the 170 000 iteration burn-in estimate in Table 1(d).

Heidelberger-Welch and the Geweke diagnostics are not "pure" burn-in diagnostics but rather implicitly gives a burn-in estimation since n_0 is the point that divides the chain in two parts, the burn-in part and the stationarity part (where we have convergence in distribution). For the Geweke diagnostic, the window size of the first window seems to be too wide to detect the initial transient period of $\pi(G)$ in data set anolis. They also seem to differ in estimating n_0 for p_{inv} in data set primates. This trace is difficult to analyze and one suggestion to the difference in estimations is that the Geweke diagnostic is using too small window size of the first window here (choosing window sizes and analyzing the results of the Geweke diagnostic are more thoroughly discussed in C.1). Otherwise, both the Heidelberger-Welch and the Geweke diagnostic seem to give (indirect) estimates of n_0 that is consistent with our intuitive feeling what an appropriate burn-in estimate should be (of course bearing in mind the limited steps for which they test convergence).



(a)

(b)

(c)

(d)

(e)

(f)

Figure 6: Examples of trace plots for $\pi(G)$ and p_{inv}. (a)- (b) for data set primates, (c)- (d) for data set anolis and (e)- (f) for data set adh. b shows observable burn-in periods.

(a)

$\pi(G)$ primates				
Chain	R-L	H-W	Geweke	ESS_{max}
1	840	10 000	10 000	**17 000**
2	720	10 000	10 000	**3 000**
3	1440	10 000	10 000	**3 000**
4	1400	10 000	10 000	**68 000**
5	1120	10 000	20 000	**42 000**
6	1080	20 000	20 000	**24 000**
7	1000	10 000	10 000	**2 000**
8	980	10 000	10 000	**76 000**
9	980	10 000	10 000	**3 000**
10	1400	10 000	10 000	**3 000**

(b)

p_{inv} primates				
Chain	R-L	H-W	Geweke	ESS_{max}
1	1 120	10 000	10 000	**8 000**
2	1 120	0	20 000	**22 000**
3	1 320	0	20 000	**18 000**
4	980	0	10 000	**28 000**
5	1 080	**0**	**0**	38 000
6	1 120	0	0	**7 000**
7	1 320	0	10 000	**12 000**
8	1 280	20 000	**0**	**0**
9	1 260	0	0	**24 000**
10	1 680	40 000	no conv.	**14 000**

(c)

$\pi(g)$ anolis				
Chain	R-L	H-W	Geweke	ESS_{max}
1	4 900	50 000	100 000	**10 000**
2	**5 600**	0	0	0
3	5 600	50 000	50 000	**20 000**
4	6 400	50 000	50 000	**30 000**
5	**5 600**	no conv.	0	0
6	4 000	50 000	0	**45 000**
7	7 200	**50 000**	0	0
8	7 200	50 000	100 000	**35 000**
9	7 200	200 000	0	**10 000**
10	7 200	50 000	0	**15 000**

(d)

p_{inv} anolis				
Chain	R-L	H-W	Geweke	ESS_{max}
1	800	0	0	**170 000**
2	**400**	0	0	**400**
3	800	**0**	**0**	100
4	**600**	0	0	500
5	800	**0**	**0**	**0**
6	600	0	0	**10 000**
7	**500**	0	0	400
8	**800**	0	0	300
9	600	**0**	**0**	100
10	800	**50 000**	0	0

(e)

$\pi(G)$ adh				
Chain	R-L	H-W	Geweke	ESS_{max}
1	3 000	0	0	**30 000**
2	2 800	0	0	**1 600**
3	2 800	0	0	**20 000**
4	3 200	0	100 000	**1 800**
5	2 800	0	0	**1 200**
6	**2 800**	0	0	3 600
7	2 800	0	0	**1 800**
8	**2 800**	0	100000	70 000
9	2 800	0	0	**2 400**
10	2 800	0	100 000	**30 000**

(f)

p_{inv} adh				
Chain	R-L	H-W	Geweke	ESS_{max}
1	14 400	100 000	100 000	**70 000**
2	21 600	no conv.	300 000	**700 000**
3	19 000	**100 000**	**100 000**	**100 000**
4	14 000	100 000	100 000	**60 000**
5	18 200	no conv.	200 000	**60 000**
6	9 000	100 000	100 000	**50 000**
7	10 000	200 000	200 000	**700 000**
8	10 000	100 000	100 000	**60 000**
9	12 000	100 000	100 000	**50 000**
10	8 000	100 000	200 000	**60 000**

Table 2: (a)-(f) shows estimated burn-in values of the different chains for each data set and parameter. The numbers in bold shows the burn-in estimates that gives the smallest values of the SEM. H-W and R-L are shorthand notation for the Heidelberger-Welch and the Raftery-Lewis diagnostics. A 0 indicates an assessment that no burn-in is needed and "no conv." indicates an assessment that convergence in distribution is not reached in the first half of the chain.

4.2.2 Summary

Table 3 shows the increased level of accuracy that we get when using the different burn-in estimators (normalized with respect to the the smallest SEM). Naturally, the cases where we have gained the most amount of information compared with discarding no burn-in are the cases when our parameters had a significant (clearly visual) burn-in period. Table 7 in C.2 shows the actual SEM values.

data set	parameter	Average SEM accuracy				
		R-L	H-W	Geweke	ESS_{max}	no burn-in
primates	$\pi(G)$	1.25	1.12	1.13	1	2.09
	p_{inv}	1.09	1.11	1.08	1	1.10
anolis	$\pi(G)$	1.04	1	1.47	1.12	1.77
	p_{inv}	1.04	1.06	1.06	1	1.06
adh	$\pi(G)$	1.01	1.02	1.03	1	1.02
	p_{inv}	1.66	1.07	1.09	1	1.84

Table 3: Relations between the average SEM values obtained by discarding burn-ins suggested by the four diagnostics. Values shows ratios between the smallest SEM for each data set-parameter combination by normalizing with the smallest SEM for the given data set-parameter combination.

We saw in section 4.2.1 that the ESS_{max} suggestions of n_0 gave the smallest SEM most of the times. This is no coincidence since the ESS is related to the SEM. We have through the formulas in B.4 that

$$ESS = \frac{\widehat{s}^2_{ML}}{SEM^2}.$$

We see from the formula above that the ESS and the SEM have a negative relation. We can thus make the somewhat unprecise but intuitive claim that, as long as the $\widehat{s}^2_{ML}$ does not vary too much between different burn-in estimates, the maximum ESS value will often be obtained where the SEM has its minimum. We can furthermore deduce that in the cases where some of the other diagnostics above have estimated a burn-in that gave a lower SEM, we must have that the relative difference in $\widehat{s}^2_{ML}$ was greater than the relative difference in SEM between the two burn-in estimations.

In data set anolis for parameter $\pi(G)$ the ESS_{max} does not obtain the lowest average mean even though ESS_{max} estimates a burn-in that gives the smallest SEM in most of the ten chains (Table 1(c)). This is due to chain 5 where the burn-in suggested by ESS_{max} gets a SEM of $0.51 \cdot 10^{-3}$, a far greater value than the average SEM for this parameter (see Table 7). Raftery-Lewis burn-in estimate however gets a value of $0.38 \cdot 10^{-3}$, thus does slightly better. The Heidelberger-Welch diagnostic assessed that this chain had not converged and this SEM was therefore not averaged over, this contributes largely to why its burn-in estimates shows the smallest average SEMs. Here we note the importance of applying different kinds of diagnostics to a chain. While Raftery-Lewis and ESS_{max} are pure burn-in "optimizers", thus cannot detect a case of non convergence, we have that the Heidelberger-Welch and Geweke diagnostics are convergence diagnostics which do not optimize burn-in but instead give us an indication if the trace seems to have failed to converged. Both these tools are important.

4.3 Run-length estimation

We saw in previous section that ESS_{max} were the burn-in estimator that gave the best burn-in estimates most of the times (according to our measure). We will here examine the run length of these chains with the burn-in suggested by ESS_{max} discarded from these chains.

4.3.1 Results

We see in Table 4 and Table 5 that most chains (53 out of 60) passed the true mean- test. There seems to be no relationship between a confidence interval not containing μ_{ref}, and failure in sufficient run length estimated by the run length diagnostics. Intuitively it seems plausible that the chains that do not fulfill a robustness test (such as the run length diagnostics) may also have a parameter estimate that varies more around the true mean leading to a higher probability of failing the true mean-test. Unfortunately, we get a wider confidence interval for these estimates so these intervals may still contain the true mean. In fact, a quick test showed that 55 chains passed the "true mean"- test even when no burn-in was discarded due to the larger confidence intervals of the mean estimates.

(a)

	$\pi(G)$, primates									
	Sample size test			Stability test			R-L			$\mu_{ref} \in CI$
Chain	100	200	300	0.01	0.025	0.05	0.025	0.05	0.075	
1	✓	✓	✓	✓	✓	✓	-	✓	✓	✓
2	✓	✓	✓	✓	✓	✓	-	✓	✓	✓
3	✓	-	-	✓	✓	✓	-	✓	✓	✓
4	✓	✓	-	✓	✓	✓	-	-	✓	-
5	✓	-	-	✓	✓	✓	-	-	✓	-
6	✓	✓	✓	✓	✓	✓	-	-	✓	✓
7	✓	✓	-	✓	✓	✓	-	-	✓	✓
8	✓	✓	-	✓	✓	✓	*[2]	-	✓	-
9	✓	✓	-	✓	✓	✓	-	✓	✓	✓
10	✓	✓	-	✓	✓	✓	-	✓	✓	✓
% of correct	70	70	60	70	70	70	20	80	70	100

(b)

	$\pi(G)$, anolis									
	Sample size test			Stability test			R-L			$\mu_{ref} \in CI$
Chain	100	200	300	0.01	0.025	0.05	0.025	0.05	0.075	
1	✓	✓	-	✓	✓	✓	-	-	✓	✓
2	✓	-	-	✓	✓	✓	-	-	✓	✓
3	✓	-	-	✓	✓	✓	-	✓	✓	✓
4	✓	-	-	✓	✓	✓	-	-	✓	✓
5	✓	-	-	✓	✓	✓	-	-	✓	✓
6	✓	✓	✓	✓	✓	✓	-	✓	✓	✓
7	✓	✓	-	✓	✓	✓	-	-	✓	✓
8	✓	✓	-	✓	✓	✓	-	-	✓	✓
9	✓	-	-	✓	✓	✓	-	-	✓	✓
10	✓	-	-	✓	✓	✓	-	-	✓	✓
% of correct	100	40	10	100	100	100	0	20	100	100

(c)

	$\pi(G)$, adh									
	Sample size test			Stability test			R-L			$\mu_{ref} \in CI$
Chain	100	200	300	0.01	0.025	0.05	0.025	0.05	0.075	
1	✓	✓	✓	✓	✓	✓	-	✓	✓	✓
2	✓	✓	✓	✓	✓	✓	-	✓	✓	✓
3	✓	✓	✓	✓	✓	✓	-	✓	✓	-
4	✓	✓	✓	✓	✓	✓	-	✓	✓	✓
5	✓	✓	✓	✓	✓	✓	-	✓	✓	✓
6	✓	✓	✓	✓	✓	✓	-	✓	✓	✓
7	✓	✓	✓	✓	✓	✓	-	✓	✓	✓
8	✓	✓	✓	✓	✓	✓	-	✓	✓	✓
9	✓	✓	✓	✓	✓	✓	-	✓	✓	-
10	✓	✓	✓	✓	✓	✓	-	✓	✓	✓
% of correct	80	80	80	80	80	80	20	80	80	100

Table 4: (a)-(c) shows if the different chains passed the run length criterion for parameter $\pi(G)$.

[2] A * indicates that we have too few samples in order for the Raftery-Lewis diagnostic to estimate a sufficient run length for the given value of r. This estimation is done by formula B.1 in Section B.3

(a)

	p_{inv}, primates									
	Sample size test			Stability test			R-L			$\mu_{ref} \in CI$
Chain	100	200	300	0.01	0.025	0.05	0.025	0.05	0.075	
1	✓	-	-	-	-	✓	-	-	✓	✓
2	✓	✓	-	-	-	✓	-	✓	✓	✓
3	✓	-	-	-	-	✓	-	-	✓	✓
4	✓	✓	-	-	-	✓	-	✓	✓	✓
5	✓	-	-	-	-	✓	-	-	✓	✓
6	✓	-	-	-	-	✓	-	-	✓	✓
7	✓	-	-	-	-	✓	-	-	✓	✓
8	✓	-	-	-	-	✓	-	-	✓	✓
9	✓	-	-	-	-	✓	-	-	✓	✓
10	✓	-	-	-	-	✓	-	✓	✓	✓
% of correct	100	20	0	0	0	100	0	30	100	100

(b)

	p_{inv}, anolis									
	Sample size test			Stability test			R-L			$\mu_{ref} \in CI$
Chain	100	200	300	0.01	0.025	0.05	0.025	0.05	0.075	
1	✓	✓	✓	✓	✓	✓	-	✓	✓	✓
2	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
3	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
4	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
5	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
6	✓	✓	✓	✓	✓	✓	✓	✓	✓	-
7	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
8	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
9	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
10	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
% of correct	90	90	90	90	90	90	80	90	90	100

(c)

	p_{inv}, adh									
	Sample size test			Stability test			R-L			$\mu_{ref} \in CI$
Chain	100	200	300	0.01	0.025	0.05	0.025	0.05	0.075	
1	✓	✓	✓	-	-	✓	✓	✓	✓	✓
2	✓	✓	✓	-	✓	✓	*	✓	✓	-
3	✓	✓	-	-	-	✓	-	✓	✓	✓
4	✓	✓	✓	-	-	✓	✓	✓	✓	✓
5	✓	✓	✓	-	✓	✓	✓	✓	✓	✓
6	✓	✓	-	-	-	✓	-	✓	✓	✓
7	✓	✓	✓	-	✓	✓	*	✓	✓	✓
8	✓	✓	✓	-	-	✓	✓	✓	✓	✓
9	✓	✓	✓	-	✓	✓	✓	✓	✓	✓
10	✓	✓	✓	-	-	✓	✓	✓	✓	✓
% of correct	90	90	70	10	30	90	60	90	90	100

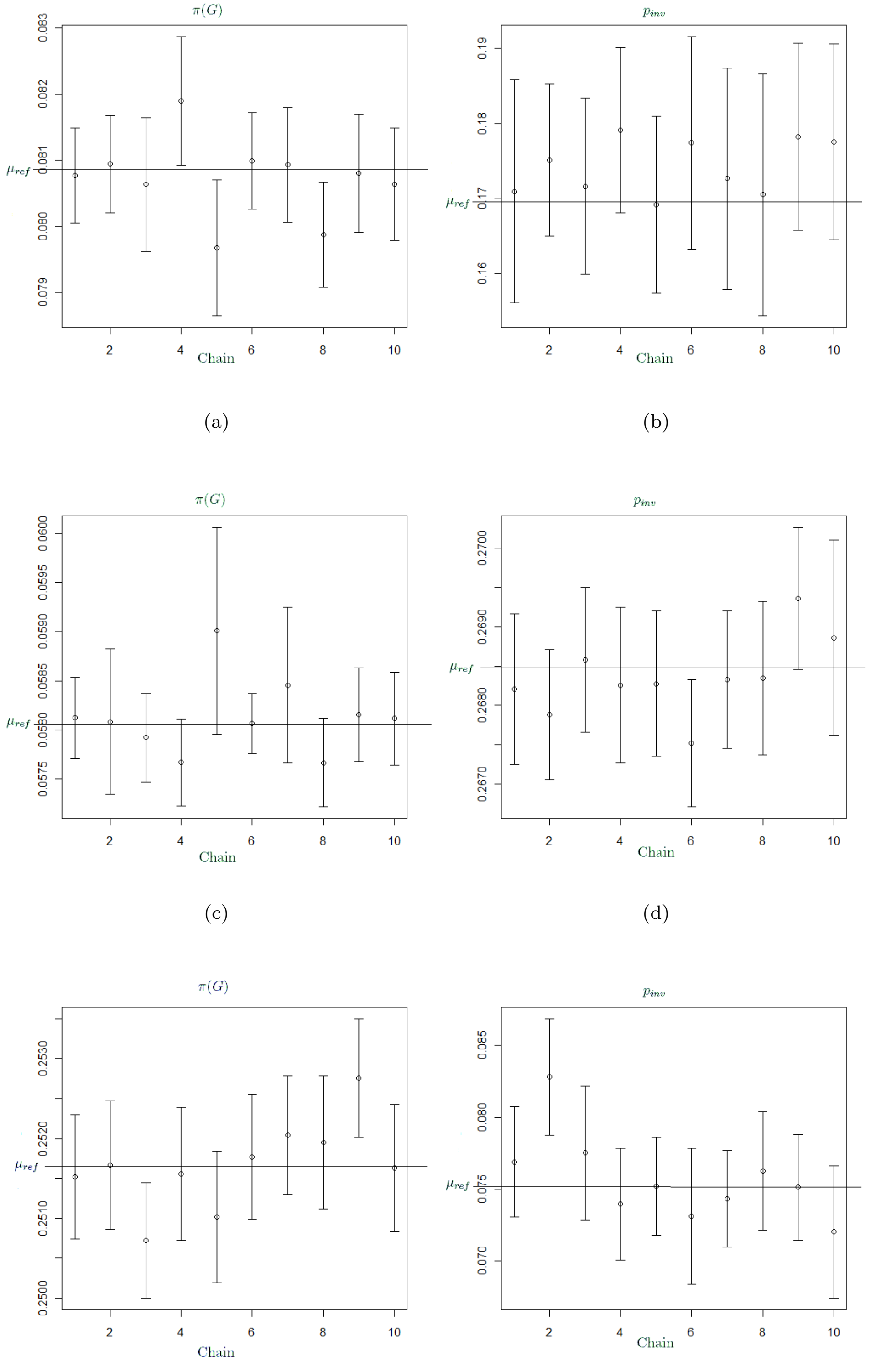
Table 5: (a)-(c) shows if the different chains passed the run length criterion for parameter p_{inv}.

4.3.2 Summary

	Summary									
	Sample size test			Stability test			R-L			$\mu_{ref} \in CI$
	100	200	300	0.01	0.025	0.05	0.025	0.05	0.075	
Assessment error	7	6	5	6	7	7	1	4	7	0
Average %	88	65	52	58	62	88	30	65	88	100

Table 6: Assessment error is the case where run length is claimed to be sufficient but μ_{ref} is outside the confidence interval of the sample mean. Average % is the number of % that the run length diagnostic agrees with the true mean-test. We do not have any run length diagnostic with a specific threshold value that scores high in both cases. Since the assessment error is important to avoid, one candidate for an optimal for giving the best total score is R-L together with 0.025 as the threshold value. However, there seems to be no result pointing in one direction here.

A possible explanation for the high frequency of chains that passed the true mean-test is that most of the chains, in fact, have converged and we could expect some chains to give a confidence interval that does not contain μ_{ref} just by the randomness of a sample. That is, under the assumption that all 60 chains have in fact converged, we could expect (in average) 3 of these chains to have a confidence interval of the sample mean not containing μ_{ref} (for confidence intervals created with 95% confidence level). Thus, our result of 7 failing chains and the expected amount of confidence intervals that do not contain μ_{ref} is a bit too close to be able to draw any significant conclusions. Figure 7 shows the sample means and their corresponding confidence intervals (obtained from formula B.4 in Section B.6) compared to the location of μ_{ref}. There



(a) (b)

(c) (d)

(e) (f)

Figure 7: Sample means and there 95% confidence intervals compared to μ_{ref}. (a)- (b) for data set primates, (c)- (d) for data set anolis and (e)- (f) for data set adh.

seems to be no sample means indicating a clear occurrence of a chain that has not yet reached stationary distribution (has spend time in a different location in the parameter space); Most sample means seems to vary naturally (normally distributed) around μ_{ref}. However, one borderline case is chain 2 Figure 7(f) where both the mean and the 95% confidence interval is somewhat secluded. It is also worth mentioning that Figure 7(a) shows that three of the ten chains have a confidence interval not including μ_{ref}. Under the hypothesis that these ten chains have converged, if we think of a confidence interval of a chain containing μ_{ref} as a success in a binomial distribution with number of trials equal to 10 and probability of success to 0.95, then the probability of obtaining 7 or less successes sums up to approximately 0.015. A value that rejects our hypothesis on 95% level. Thus there are non the less some indications of cases of non-convergence among the 60 chains.

Looking at the summary of the run length analysis in Table 6 we see that a high acceptance ratio of sufficient run length is prone to many assessment errors. There seems to be no clear result here which run length diagnostic-threshold value combination that seems to be the best (as similar to the true mean-test as possible).

5 Discussion

We will here go through what we believe are the strengths and weaknesses of the diagnostics. Also, we discuss problematic issues with the analysis and propose methods for further, more exhaustive analysis.

5.1 On the burn-in diagnostics

The experience we made with the Geweke diagnostic in this analysis is that it does not reveal more information about the burn-in periods than trace plots do. Since it is merely a transformation of the values in the trace plots to a comparison of the location of means, the Geweke diagnostic does not investigate any other properties of the trace that might be more difficult to discover by eye. The advantage, however, in contrast to trace plots, is that it gives a quantity (the Z-score) that can easily be verified against the standard normal distribution table for testing convergence in distribution of the chain. Another advantage of the Geweke diagnostic is that it is very easy to interpret, we basically compare mean values of two different part of the chain. However, to choose the appropriate window sizes (which are specified by the user) requires some profound knowledge about trace plots (see C.1).

The Heidelberger-Welch diagnostic is investigating properties of the trace in a more sophisticated way compared to the Geweke diagnostic, both means and variation between different parts of the chain affects the diagnostic. The cost of this may be loss of the "easy to interpret" factor that the Geweke diagnostic has. We should also mention once again that we used the Heidelberger-Welch and Geweke diagnostic without correction for repetitive hypothesis testing (briefly explained in Section 3.1.1), an issue one may want to deal with before further usage/investigation of these diagnostics.

When evaluating the Raftery-Lewis diagnostic we noted that the burn-in estimate depended on the quantile investigated (thorough presentation of this was skipped). This dependence indicates that the diagnostic have one or several maxima of the estimated burn-in (with respect to different quantiles). We saw that the burn-in estimations given by the Raftery-Lewis diagnostic were often too short compared to the burn-ins suggested by the plots in Figure 6. A possibility for this may be that the maximum n_0 simply was not found. An extended analysis with maxima seeking could be done if one is interested in how precise this diagnostic could estimate the burn-in compared to the visual burn in. However, since estimates must be "re-diagnosed" for every quantile of interest, an analysis based on samples of this might be very impractical. Also often, the question of interest is not how precise estimate we can get on the true n_0 but rather which n_0 that gives precise estimate of a parameter of interest (although these two questions have a close relation).

We saw that the ESS_{max} diagnostic performed the best in our burn-in estimation evaluation. But even this diagnostic (referring to the 0 estimations in Table 1(c)). Then we must again ask us what our main interest is; the true burn-in period (the point we the chain looses all information about its starting point) or the parameter estimate.

Since we defined a good burn-in estimate n_0 to be such that n_0 gives a low SEM of the mean estimate, one possible burn-in diagnostic could be to find a point on the chain where we have the minimum SEM. The SEM quantity has the same computational complexity as the ESS and since it is the quantity we want to minimize, it might be easier to work with it directly. However, if one is interested in some other ergodic estimate, the "minimum SEM" diagnostic might need to be replaced (e.g. different quantiles have different asymptotic distributions). We have seen in the analysis section the strong relation between the ESS and the SEM quantities (the ESS_{max} often gave the burn-in estimations with the smallest SEM). A reason to use the ESS_{max} instead of the minimum SEM would be that ESS is more general. That is, it applies to all quantiles and other test statistics while the SEM only applies to the mean (although a accurate estimate of the mean often imply an accurate estimation of the complete sample distribution). Also, SEM in comparison to ESS are sensitive to areas with low sample variation. If a parameter gets stuck somewhere in an area giving very low variation of the parameter the SEM would give a low value in this area. The ESS would set the SEM in comparison to s^2_{ML}, which are in fact also small in that case, thus giving a proportional measure.

5.2 On the run-length evaluation

In the run length evaluation part we were mainly interested in the cases where a diagnostic would conclude sufficient run length while the true mean-test stated that the parameter estimate was significantly differing from μ_{ref}. We want to choose thresholds such that these cases occurs as rarely as possible. However, this turns out to be difficult to test. First, since we are focusing our analysis on the trace, we can never be sure that the chain has in fact converged. The likelihood function may still not have reached its maximum yet. Second, the true mean test are not as "efficient" as we may want for testing these diagnostic error assessments. The test can tell no difference between a chain that has a mean estimate close to μ_{ref} but with a narrow confidence interval that does not contain μ_{ref} and a chain with a mean estimate far from μ_{ref} but with a wide confidence interval that contains μ_{ref}. Still the first chain will pass some run length diagnostic test while the other won't. So, when testing some diagnostic together with a given threshold, say for example $ESS > 200$ in Table 3(b), we may want a more "precise" test to actually verify if there are any differences between the four chains that passed the test and the six chains that did not. Comparing Table 3(b) with 7(c) in fact shows this insensitivity; the true mean-test cannot tell any difference between chain 5 and 6 (both passed) while we may want to make difference between the two chains when looking at the difference in the location of mean and confidence interval width.

Here we propose an alternative way to analyze the sample size test and the stability test based on an different approach. It was not performed here due to time constraints. We suggest the following procedure.

- From the reference chains, obtain μ_{ref} and s^2_{ref} (s^2_{ref} can be obtained from either V in B.4 or by using a thinned out subchain and assume independent samples). Here, we make the trivial assumption that μ_{ref} and s^2_{ref} give close to true values of the true mean and variance.
- We have by CLT for Markov chains ([21], p.102-103) that a sample mean $\overline{X}_k$ with $ESS = k$ of an MCMC chain in stationary distribution is distributed as $N\left(\mu_{ref}, \sqrt{s^2_{ref}/k}\right)$ (it actually suffice that the sample variables in the test chain have the same mean value and variance as the sample variables in the stationary distribution).
- Thus, we get the following probability

$$P\left(\mu_{ref} + t_{k-1,\frac{\alpha}{2}}\sqrt{s^2_{ref}/k} \leq \overline{X}_k \leq \mu_{ref} + t_{k-1,1-\frac{\alpha}{2}}\sqrt{s^2_{ref}/k}\right) = 1 - \alpha$$

- The above probability suggests that a sample mean $\overline{X}_k$ from a chain in stationary distribution with $ESS = k$ should fall within this interval $1 - \alpha$ number of times. This means that testing n chains, all with $ESS > k$, we should expect that *at least* $(1 - \alpha)n$ chains will have their sample mean within this interval. If a significantly larger ratio of chains than $(1-\alpha)n$ falls outside this interval, we can conclude that the sample size test with k as a threshold value may not be a good run length diagnostic (the test simply let to many chains without a mean that has converged "pass" the run length test).

We have here incorporated both the mean and the variance of the stationary distribution, thus exploring more of the information we have about the true distribution in this test. Note however that this test assumes that the estimated quantity ESS is a good approximation of the independent sample size for this sample.

A thing that is worth mentioning again with the run length test used in this thesis is that we have chosen a 95% confidence interval on our true mean-test, thus forcing us to deal with a certain amount of type I errors. Also, we have used μ_{ref} rather than the true mean value. Even though it is estimated with high precision, (we have *at least* 50 times the iteration span from the stationary part of the chain compared to a full test chain) small deviances may still occur between μ_{ref} and the true mean.

6 Conclusions/Future work

First, what is then a proper method for finding burn-in? As mentioned in other papers dealing with this topic, e.g.[5], we encourage to use as many different burn-in diagnostics as possible. Also, it is always better to analyze the output yourself, rather than with some automated procedure. However, this may be infeasible due to time constraints. We therefore, suggest that a potential method (taking into account that the method should both be accurate and easy to interpret) could be some algorithmic use of the ESS_{max} diagnostic together with the Geweke diagnostic.

The ESS_{max} finds a burn-in estimate n_0 and the Geweke diagnostic verifies that the chain seems to have converged at this point (the Heidelberger-Welch diagnostic could of course also be used here). If convergence is not achieved according to this diagnostic simply apply the ESS_{max} on the remaining part of the chain and proceed in the same way. The implementation of the ESS_{max} diagnostic can be done in several ways. Perhaps a significant amount of time can be saved by stopping the algorithm that calculates the ESS_{max} when the value has gone below some given threshold of a previous ESS peak. The accuracy of the ESS_{max} diagnostic can also be adjusted for, the more accurate the slower the algorithm.

Second, how do we determine if the chain has run sufficiently long? Our tests did not reveal much in order for us to have any good answers to this question. This question also differs depending on the preferences of the person simulating the chains; which accuracy are needed on the estimates? We intended to give some lower bound on the constants for which the different run length diagnostics would give a too high assessment error. The results were however not sufficiently informative to be able to answer this question.

The run-length analysis can be extended and more thoroughly examined. For example, different data sets/parameters or chain length/sample frequencies can be investigated where a higher percentage of chains has not converged. This would give more significant results since we then have a sufficient amount of chains, both converged and non converged on which we can make inference on. Also the test procedure explained in section 5 could be used possibly give more informative results.

We however briefly outline a possible algorithm for analyzing MCMC output taking into account both burn-in and run length estimation. The way the diagnostics could be implemented in each of the steps in the algorithm could be further analyzed but the general structure is outlined here.

1. **Estimation:** Estimating burn-in with ESS_{max}
2. **Consistency:** Check for homogenity of different parts of the chain after the burn-in has been removed. Preferably you want to test the beginning of the remaining chain against the end of the chain (as the Geweke diagnostic does), but you can also split up the remaining chain into several batches and compare the locations of these.
   - If homogenity achieved, go to 3.
   - If not, go to step one.

3. **Stability:** Use some sort of run length diagnostic to see if we have met some accuracy threshold on our estimate.
    - If accuracy achieved, go to 4.
    - If not, longer simulation is needed.
4. **Reproducibility:** In the case of multiple chains, one can test for equality of sample means between different chains with some statistical test such as the F-test, taking into account the dependence of the samples (within a chain).
    - If hypothesis of equality of sample means not rejected, stop.
    - If hypothesis of equality of sample means rejected, longer simulation is needed. In this case however, one might need to do manual analysis of the chain.

References

[1] http://artedi.ebc.uu.se/course/x3-2004/phylogeny/phylogeny-criteria/phylogeny-criteria.html.

[2] Rambaut A and Drummond AJ. *Tracer v1.4.* Available from http://beast.bio.ed.ac.uk/Tracer, 2007.

[3] Allen G. Rodrigo Wiremu Solomon Alexei J. Drummond1, Geoff K. Nicholls. Estimating mutation parameters, population history and genealogy simultaneously from temporally spaced sequence data, 2001.

[4] M.A. Newton Bob Mau and Bret Larget. Bayesian phylogenetic inference via markov chain monte carlo methods. *Biometrics, 55, 234-249*, 1999.

[5] Cowles M.K. Carlin B.P. Markov chain monte carlo convergence diagnostics: A comparative review. *J. Amer. Statist. Assoc. 91, 883-904*, 1996.

[6] Stephen P. Brooks and Gareth O. Roberts. *Convergence assessment techniques for Markov chain Monte Carlo.* Kluwer Academic Publishers, Dordrecht NL-3300 AA Netherlands, 1998.

[7] L.L. Cavalli-Sforza and A.W.F. Edwards. Phylogenetic analysis: models and estimation procedures. *American Journal of Human Genetics*, 1967.

[8] Joseph Felsenstein. *Inferring phylogenies.* Sinauer Associates, Sunderland, Mass, 2004.

[9] W. M. Fitch. Toward defining the course of evolution: minimum change for a specified tree topology. *Systematic Zoology 20: 406-416*, 1971.

[10] W. M. Fitch and E. Margoliash. Construction of phylogenetic trees. *Science 155: 279-284*, 1967.

[11] A.F.M. Smith Gelfand A.E. Sampling based approaches to calculating marginal densities. *Journal Amer. Stat. Assoc., 85, 398-409*, 1990.

[12] Geman D. Geman S. Stochastic relaxation, gibbs distribution and bayesian restoration of images. *IEE Transactions on Pattern Analysis and Machine Intelligence 6: 721-741*, 1984.

[13] Charles J. Geyer. Practical markov chain monte carlo. *Statistical Science, Vol. 7, No. 4*, 1992.

[14] C.J Geyer. Markov chain monte carlo maximum likelihood. *In Computing Science and Statistics: Proceedings of the 23rd Symposium of the Interface (ed. E.M. Keramidas),*, 1991.

[15] Welch P.D. Heidelberger P. Simulation run length control in presence of an initial transient. *Operations Research 31, 1109-1144*, 1983.

[16] Geweke J. Evaluating the accuracy of sampling based approaches to the calculation of posterior moments. *Bayesian Statistics 4, pp. 169-193*, 1992.

[17] Guihenneuc-Jouyaux C Mengersen K., Robert C. *MCMC convergence diagnostics: a review. In:Bayesian Statistics 6.* Oxford University Press, Oxford, 1999.

[18] Martyn Plummer, Nicky Best, Kate Cowles, and Karen Vines. coda:convergence diagnosis and output analysis forMCMC, 2010.

[19] R Development Core Team. *R: A Language and Environment for Statistical Computing.* R Foundation for Statistical Computing, Vienna, Austria, 2010.

[20] Lewis S.M. Raftery A.E. *How Many Iterations in the Gibbs Sampler?* Oxford University Press, 1992.

[21] Christian P. Roberts. *Discretization and MCMC convergence assessment.* New York Springer-Verlag., 1998.

[22] F Ronquist and J.P. Hulsenbeck. *MRBAYES 3:Baysian phylogenetic inference under mixed models*. Bioinformatics 19:1572-1574, 2003.

[23] Favre Anne-Catherine Bernard Bobee Salaheddine, E.A. Comparison of methodologies to assess the convergence of markov chain monte carlo methods. *Computational Statistics and Data Analysis 50 (2006) 2685-2701*, 2005.

[24] Hastings W.K. Monte carlo sampling methods using markov chains and their applications. *Biometrika 57 (1): 97-109 doi:10.1093/biomet/57.1.97*, 1970.

[25] Z. Yang and B. Rannala. Bayesian phylogenetic inference using dna sequences: A markov chain monte carlo method. *Molecular Biology and Evolution 14: 717-724*, 1997.

A Section 3

A.1 Metropolis hastings algorithm

By construction, the Metropolis Hastings algorithm avoids calculating the normalizing factor that for evolutionary trees is growing exponentially in size with the number of taxa. We see this by expanding the ratio in equation (2.2),

$$\frac{Q(\tau_t|\tau^*)}{Q(\tau^*|\tau_t)}\frac{\pi(\tau^*|\mathbf{X})}{\pi(\tau_t|\mathbf{X})} = \frac{Q(\tau_t|\tau^*)}{Q(\tau^*|\tau_t)}\frac{f(\mathbf{X}|\tau^*)f(\tau^*)}{f(\mathbf{X}|\tau_t)f(\tau_t)}\frac{\sum_{j=1}^{n} f(\mathbf{X}|\tau_j)f(\tau_j)}{\sum_{j=1}^{n} f(\mathbf{X}|\tau_j)f(\tau_j)} = \frac{Q(\tau_t|\tau^*)}{Q(\tau^*|\tau_t)}\frac{f(\mathbf{X}|\tau^*)f(\tau^*)}{f(\mathbf{X}|\tau_t)f(\tau_t)}.$$

Thus the acceptance ratio essentially depends on the probability that a certain tree τ^* is proposed given that we are currently in τ_t, times the likelihood ratios of the proposed tree and the current tree.

A.2 Gibbs sampler

The Gibbs sampler is a Metropolis-Hastings algorithm where the proposed random vector $(\theta^*_{(1)}, \ldots, \theta^*_{(n)})$ is accepted with probability 1. To see this we calculate the acceptance ratio for every parameter $\theta_{(k)}$ being at iteration t for the Metropolis-Hastings algorithm.

$$\frac{Q(\theta^t|\theta^*)}{Q(\theta^*|\theta^t)}\frac{p(\theta^*)}{p(\theta^t)} = \frac{p(\theta^t_{(k)}|\boldsymbol{\theta}^*_{(-k)})}{p(\theta^*_{(k)}|\boldsymbol{\theta}^t_{(-k)})}\frac{p(\theta^*_{(k)}|\boldsymbol{\theta}^*_{(-k)})p(\boldsymbol{\theta}^*_{(-k)})}{p(\theta^t_{(k)}|\boldsymbol{\theta}^t_{(-k)})p(\boldsymbol{\theta}^t_{(-k)})} = \frac{p(\theta^t_{(k)}|\boldsymbol{\theta}^*_{(-k)})}{p(\theta^*_{(k)}|\boldsymbol{\theta}^*_{(-k)})}\frac{p(\theta^*_{(k)}|\boldsymbol{\theta}^*_{(-k)})p(\boldsymbol{\theta}^*_{(-k)})}{p(\theta^t_{(k)}|\boldsymbol{\theta}^*_{(-k)})p(\boldsymbol{\theta}^t_{(-k)})} = 1 \quad \text{(A.1)}$$

We have here used that $\boldsymbol{\theta}^t_{(-k)} = \boldsymbol{\theta}^*_{(-k)}$ when updating parameter k for all t. This is true since we only update one parameter at a time, and when updating parameter θ^t_k, all other parameters $\boldsymbol{\theta}^t_{(-k)}$ being fixated since they are marginalized over. Thus, they are the same before and after the proposal of θ^t_k.

B Section 4

B.1 Stationary/Weakly stationary process

Stationary process. *A stochastic process $\{X(t), t \geq 0\}$ is said to be a stationary process if for all $n, s, t_1, \ldots, t_n$ the random vectors $X(t_1), \ldots, X(t_n)$ and $X(t_1 + s), \ldots, X(t_n + s)$ have the same joint distribution.* □

Weakly stationary process. *A stochastic process $\{X(t), t \geq 0\}$ is said to be a weakly stationary process if $E[X(t)] = c$ and $Cov[X(t), X(t + s)]$ does not depend on t.* □

Notice that a weakly stationary process implies a stationary process.

B.2 Finite state first order Markov chain

The conditions of being a finite state first order Markov chain is that

1. The process have a finite number of states and can only be in one state at every given time.
2. The transition probability P_{ij} of transition from state i to j is given for any combination of i and j and the transition probabilities are assumed to be stationary (unchanging over any time period and independent of how state i was reached).
3. Either the initial state of the process or the probability distribution of the initial state is known.

B.3 Raftery-Lewis

The minimal number of iterations needed for the analysis of burn-in and run length estimation are calculated as in equation B.1 where Φ is the standard cumulative distribution function.

$$n_{min} = \frac{q(1-q)}{(\Phi(0.5(p-1))r)^2} \tag{B.1}$$

The number of iterations to discard as burn-in is ms where s is the thinning coefficient of the thinned out sequence Z_{st} and m is given by B.2.

$$m = \frac{\ln \frac{(R_{01}+R_{10})\epsilon}{\max(R_{01},R_{10})}}{\ln(1-R_{01}-R_{10})} \tag{B.2}$$

The number of iteration suggested for a sufficient run length of the chain is given by nk where n is given by B.3.

$$n = \Big(\frac{(2-R_{01}-R_{10})R_{01}R_{10}}{(R_{01}+R_{10})^3}\Big)\Big(\frac{\Phi(0.5(p+1))}{r}\Big) \tag{B.3}$$

B.4 ESS/SEM/ACT

Let $x_1, \ldots, x_n$ be observations in a sample obtained from an MCMC run, $\overline{x}$ the mean value of the output and m the number of iterations between each sample (sample frequency). If we let

$$s = \max(n-1, 2000) \quad \text{and} \quad \gamma(i) = \frac{1}{n-i}\sum_{j=i}^{n-i}(x_j - \overline{x})(x_{j+i} - \overline{x}).$$

These $\gamma(i)$:s are calculated as long as $\gamma(i-1)+\gamma(i) > 0$. Let k be the integer for which $\gamma(k-1)+\gamma(k) \leq 0$ then we obtain the modified sample variance as

$$V = \gamma(0) + 2\sum_{i=1}^{k-2}\gamma(i)$$

Note that $\gamma(0)$ is the estimated sample variance $\widehat{s}^2_{ML}$ obtained from ML estimation. Now we obtain the following quantities from this modified variance statistic where ACT is the autocorrelation time.

$$SEM = \sqrt{\frac{V}{n}}, \qquad ACT = m\frac{V}{\gamma(0)}, \qquad ESS = \frac{mn}{ACT}.$$

These formulas are obtained from Tracer[2], with the theory found in[3],[13].

B.5 Absolute/relative error

Absolute error. *An estimate ρ has absolute error ϵ for $P(\mathbf{y})$ if*

$$|P(\mathbf{y}) - \rho| = \epsilon.$$

□

Relative error. *An estimate ρ has relative error η for $P(\mathbf{y})$ if*

$$\left|\frac{P(\mathbf{y}) - \rho}{P(\mathbf{y})}\right| = \eta.$$

□

B.6 True mean-test

The true mean-test is a "run length verifier" where we have obtained an accurate estimate of the true mean μ_{ref}. To estimate μ_{ref} we run ten long chains for every data set, here called reference runs. From these runs we will get our estimations of the true values of mean for the two different parameters in each of the three data sets. Only the second part of the reference runs are examined and joined together in order to have unbiased samples from the stationary distribution.

These subchains are compounded into one chain and from this concatenated chain we calculate the estimator of the true mean for the parameters. More formally, if x_{ij} denotes a sample in reference chain i at position j, n is the number of reference chains and m is the number of iterations in each chain, then

$$\mu_{ref} = \frac{\sum_{i=1}^{n} \sum_{j=(\frac{m}{2}+1)}^{m} x_{ij}}{nm/2}$$

This "reference value" is our estimator for true mean and it could be considered to be close to the true mean value.

The underlying theory for the true mean-test is based on the validity of the *Central Limit Theorem.* This theorem is also shown to hold for Markov chains ([21],p.102-103).

Central Limit Theorem 1. *If $X_1, X_2, X_3, \ldots$ are independent identically distributed random variables from an infinite population with $E[|X_1|]^2 < \inf$, $E[X_1] = \mu$ and $Var(X_1) = \sigma^2$ then, we have*

$$\frac{X_1 + X_2 + \ldots X_n - \mu n}{\sqrt{n\sigma^2}} \rightarrow N(0,1) \qquad n \rightarrow \infty.$$

□

The sample mean parameter $\overline{X}$ is a specific number for a specific sample. Thus, it can be considered as a random variable that varies from one random sample to another. Provided that the sample size is sufficiently large, we get by the *Central limit Theorem* that the sampling distribution of the sample mean is approximately normal (regardless of the parent population distribution), with mean equal to the mean of the underlying parent population μ and variance equal to the variance of the underlying parent population σ^2 divided by the sample size n. That is, if $\overline{X} = \frac{\sum_{i=1}^{n} X_i}{n}$ (the sample mean), we have

$$\frac{\overline{X} - \mu}{\sqrt{\frac{\sigma^2}{n}}} \rightarrow N(0,1) \qquad n \rightarrow \infty.$$

Since we do not have independent samples in our chains, if n is the total number of samples we have obtained, we cannot use $\sqrt{s^2/n}$ as an estimation of the standard error of the mean value (this is also known as the naïve standard error). As we described before, the correlation between samples will affect the information we have about the sample distribution. Therefore, we instead use the modified SEM given in B.4. (also known as the time-series standard error). Furthermore, the confidence interval quantiles are based on ESS as degrees of freedom.

We verified that these confidence intervals gave reliable confidence intervals for the mean of a sample. This verification was performed on some of the chains, by comparing our confidence intervals with the confidence intervals that was calculated with independent samples[3]. The ratio of the width of these confidence intervals was verified to be close to one, thus indicating that our confidence intervals gave good estimate of the true "measure of insecurity".

[1] The independent samples was created by considering thinned out subchains. The thinning interval $t > 1$ was set to be the autocorrelation-time (ACT) estimated from the chains. Autocorrelation-time is an estimation of how many iterations that needs to be discarded between two successive samples such that the samples could be considered independent. These estimated values was obtained from the program Tracer [2]. For a test chain x of n samples, we got the estimated ACT t. Taking every tth sample from x yielded the new sub chain y of x containing only $\lfloor n/t \rfloor$ samples which could be considered to be independent.

The true mean-test verifies that the 95% confidence interval of $\widehat{\mu}$ (the mean estimate of the test chain) contains μ_{ref}. More formally, if s^2 is the sample mean and variance of the test chain (after burn-in is discarded), $t_{0.975,ESS-1}$ is the 95% two sided confidence interval quantile of the t-distribution with $ESS-1$ degrees of freedom, the "True mean"-test is defined as

$$\begin{aligned}&\text{- Do not reject sufficient runlength if } \mu_{ref} \in \big[\widehat{\mu} - t_{0.975,ESS-1}SEM, \widehat{\mu} + t_{0.975,ESS-1}SEM\big].\\&\text{- Reject sufficient runlength if } \mu_{ref} \notin \big[\widehat{\mu} - t_{0.975,ESS-1}SEM, \widehat{\mu} + t_{0.975,ESS-1}SEM\big].\end{aligned} \tag{B.4}$$

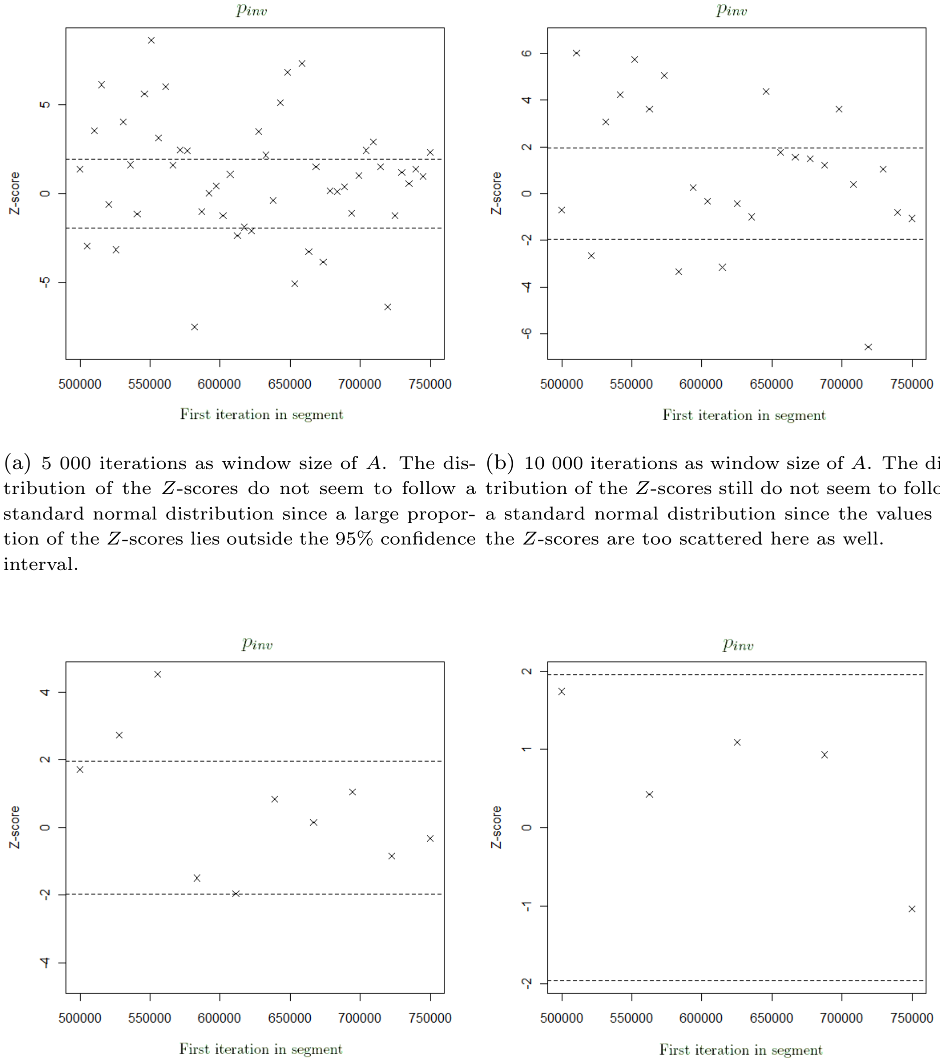
C Section 5

C.1 Geweke - Choosing proper window sizes

We used the default coda settings of the window sizes for the parameters in this paper. However, to choose appropriate window sizes for a more accurate analysis of the burn-in with the Geweke diagnostic, considering different chain lengths and slow-mixing properties for a parameter, can be a whole project in itself. As we mentioned in 3.1.1, choosing to wide first window may "hide" a smaller burn-in period in this window since only the means are compared. On the other side, a too narrow first window in relation to how slow mixing the parameter is may give a lot of Z-scores outside different ends of the confidence interval. This can happen since the Geweke statistic compares means, and the parameter may be stuck in one area of the parameter space for a longer time than the size of the window before jumping to another "area". This will cause very significant differences in means.

Plot 8 show how the Z-scores for p_{inv} in data set primates are distributed along the second part of a reference run (i.e. when the chain is in its stationary phase) and how they vary in distribution as the window sizes of the first window increases. The Z-scores are calculated for disjoint first windows of different sizes 5000, 10 000, 25 000 and 50 000 iterations. We see that a large proportion of Z-scores lies outside of the 95% confidence interval for the plots with the small window sizes. It is clear that in the first two plots the Z-scores do not seem to be normally distributed since about half of the Z-scores lies outside the 95% confidence interval (indicated with the dashed lines). As window sizes increases we se how the Z-scores seems to look more like samples form a normal distribution. Since the plots shows no sign of a positive or negative pattern, the Z-score variable seems to vary around its mean. These properties of these Geweke plots is an indication that at least the two window sizes of 5 000 and 10 000 iterations may be too narrow to give proper convergence assessment for p_{inv}.

The method with the Geweke statistic we have used in this project suffers from other flaws as well. For example figure 8 (a) and (b) also show that after one Z-value within the confidence interval (which is where we conclude that the burn-in phase is over) there may follow many values that lies outside this interval. Thus a stationary mode of the chain might not have been reached. This can be related to the issue of repetitive hypothesis testing mentioned in 3.1.1.



(a) 5 000 iterations as window size of A. The distribution of the Z-scores do not seem to follow a standard normal distribution since a large proportion of the Z-scores lies outside the 95% confidence interval.

(b) 10 000 iterations as window size of A. The distribution of the Z-scores still do not seem to follow a standard normal distribution since the values of the Z-scores are too scattered here as well.

(c) 25 000 iterations as window size of A. Most of the Z-scores lies inside the 95% confidence interval and it starts to look more as if the values would come from a standard normal distribution compared to the two other plots with smaller window size of A.

(d) 50 000 iterations as window size of A. There is now not any indication contradicting that the Z-scores could not be obtained from a standard normal distribution although it could be argued that we have too few observations to say for sure.

Figure 8: Geweke plots for p_{inv} parameter when chain is in stationary phase. The pictures illustrate values of the Geweke Z-score for disjoint first windows A of length given in the sub plots. x-axis shows the start iteration for A. y-axis shows the corresponding Z-score. The two dashed lines across the plot are boundaries for a 95% confidence interval of the Z-score when testing for equal means of windows A and B.

C.2 Average SEM estimates

Data set	Parameter	Average SEM estimates				
		R-L	H-W	Geweke	ESS_{max}	no burn-in
primates	$\pi(G)$	0.545	0.488	0.494	**0.437**	0.913
	p_{inv}	7.16	7.32	7.09	**6.59**	7.26
anolis	$\pi(G)$	0.264	**0.253**	0.371	0.284	0.467
	p_{inv}	0.498	0.509	0.505	**0.478**	0.505
adh	$\pi(G)$	0.405	0.411	0.413	**0.401**	0.411
	p_{inv}	3.41	2.19	2.23	**2.05**	3.78

Table 7: The average SEM values (in 10^{-3}) of the chains that remains after burn-in is discarded. Smallest values showed in boldfaced numbers.