MCMC Diagnostic

The theoretical convergence properties of MCMC (that the generated samples using MCMC will be able to approximate the true posterior) are based on some assumptions on the generated MCMC samples. These assumptions are stationarity, reversibility and ergodicity. Sufficient conditions for ergodicity are (1) irreducibility: the chain can reach each possible outcome whatever the starting position; (2) aperiodicity: there is no cyclic behavior in the chain and (c) positive recurrence: the chain visits every possible outcome an infinite number of times and the expected time to return to a particular outcome, irrespective of where we start in the chain, is finite. In practical terms, ergodicity means that the chain will explore the posterior distribution exhaustively.

Convergence diagnostics thus essentially involve testing whether the generated samples satisfy those conditions: 1) stationarity, 2) reversibility and 3) ergodicity. Reversibility is automatically satisfied, as the expression for the acceptance probability relies on the detailed balance condition.

Detailed balance condition: Given a transition function, it is possible to define an acceptance probability $a(\theta \to \theta')$ that gives the probability of accepting a proposed mutation from θ to θ' in a way that ensures that the distribution of samples is proportional to $f(\theta)$, our target density, which in our context is $P(\theta)$. If the distribution is already in equilibrium, the transition density between any two states must be equal by detailed balance condition:

$$P(\theta)T(\theta \to \theta')a(\theta \to \theta') = P(\theta')T(\theta' \to \theta)a(\theta' \to \theta)$$

. Note that in our previous definitions, $T(\theta \to \theta') = q(\theta' \mid \theta)$ and $T(\theta' \to \theta) = q(\theta \mid \theta')$.

The solution of $a(\theta \to \theta')$ that maximizes the rate at which above equilibrium is reached is $a(\theta \to \theta') = \min\{1, \frac{P(\theta')q(\theta|\theta')}{P(\theta)q(\theta'|\theta)}\}$

Why Gibbs sampling is a special case of MH sampling with acceptance probability $a(\theta \to \theta') = 1$? To prove the above, let's go step by step.

- What is proposal distribution in case of a Gibbs sampler? (Meaning, how are we generating a candidate?)
- What is the transition probability? (q(Candidate | Current) and q(Current | Candidate)?)

There are plenty of diagnostics tools. We will just cover the most popular ones. MCMC chains share commonalities with time series datasets. Hence, most of the diagnostic tools are borrowed from exploratory analysis methods of time series datasets.

1 Trace plots

Trace plot is essentially to plot the postburn samples like `plot(xp[,1])`. This will tell us to visually detect any pattern in the generated samples. Independent time series are also called "White noise". Our ultimate goal is to obtain independence posterior samples. Hence, the trace plot of the posterior samples should look like white noise.

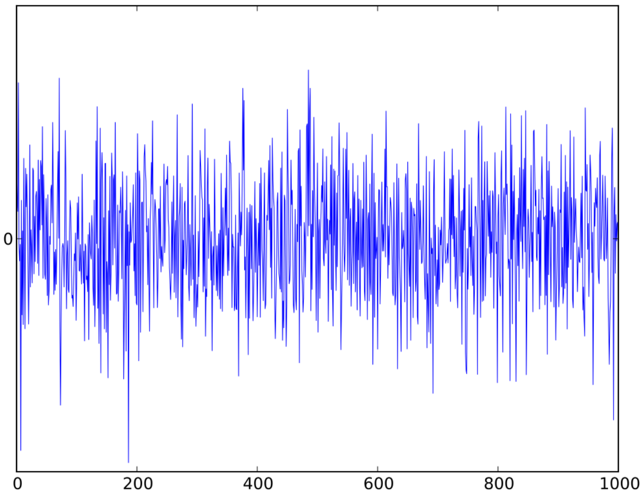

Figure 1: Typical trace plot of white noise and thus an ideal or desired plot of the MCMC samples.

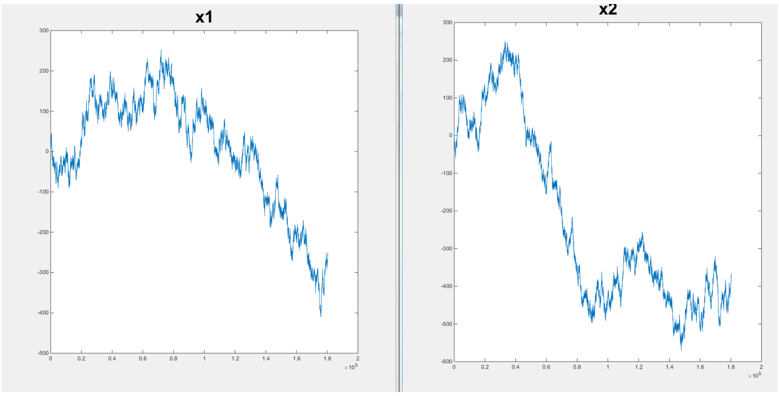

Figure 2: An example of bad convergence.

2 Autocorrelation

MCMC samples by construction are autocorrelated. In case of Gibbs sampling, every alternate sample is expected to be independent. Hence, thinning by 2 is always kind of default. However, for general MH, it is not generally true. We need to make acf() plots to ascertain appropriate the lag beyond which there is no significant dependence.

The main motivation of the diagnostic test is to check whether there is any discernible pattern.

3 Diagnostic tests

Four diagnostic tests for assessing stationarity and/or accuracy are introduced here. The first three tests assess convergence on a single chain and are based on the time-series or stochastic process properties of a Markov chain. The fourth diagnostic evaluates the discrepancy between multiple Markov chains to detect nonstationarity. Two diagnostics evaluate stationarity (size of burn-in part κ_0) and accuracy (number of extra iterations κ_1).

Geweke diagnostic: Geweke (1992) suggests to formally test the stationarity of a Markov chain by comparing the means of an early and a late part of the chain using a (frequentist) significance test. If the n values θ were i.i.d. and split up into two different parts: A (early part) with n_A elements and B (late part) with n_B elements, then their respective (posterior) means $\bar{\theta}_A$ and $\bar{\theta}_B$ could be compared with a Z-test given by $Z = \frac{\bar{\theta}_A - \bar{\theta}_B}{s_A^2/n_A + s_B^2/n_B}$. Here s_A and s_B are sample standard deviations. However, the elements of the Markov chain are dependent. Hence, the two means and the standard deviations are all dependent. The means however are asymptotically unbiased. One can use spectral methods (using Fourier transformation) to obtain better estimates for the variances.

Geweke (1992) however suggested taking for A the initial 10% of the iterations ($n_A = n/10$) and for B the last 50% ($n_B = n/2$) to create a distance between the two parts. Then increase n so that the overall Z-test becomes insignificant at say, $\alpha = 0.05$. When the overall Z-test is significant at $\alpha = 0.05$, either the burn-in part (i.e. κ_0) was taken too small and/or the total chain is too short. A dynamic version of Geweke diagnostic might also help to find a better value for κ_0. For the dynamic version of the test, the Z-test is applied on $100(K-m)/K\%(m = 0, \ldots, K)$ last iterations of the chain. This produces $Z_m(m = 0, \ldots, K)$ test statistics that are plotted in a time-series plot.

Brooks–Gelman–Rubin (BGR) diagnostic: When the posterior is multi-modal, a single Markov chain might get stuck for a very long time in an area around a local mode. Convergence to a local mode is a well-known problem in classical optimization routines. To circumvent this problem it is advised to start up the optimization routine from various initial positions. In the same spirit, Gelman and Rubin (1992) suggested a convergence diagnostic based on multiple chains with ‘overdispersed’ (relative to the posterior distribution) starting positions. Their diagnostic

Geweke's diagnostic	`geweke.diag`
Heidelberger–Welch (HW) diagnostic	`heidel.diag`
Raftery–Lewis (RL) diagnostic	`raftery.diag`
Brooks–Gelman–Rubin (BGR) diagnostic	`gelman.diag`

Table 1: Diagnostic codes for different methods are available in R package `coda`. The required functions are listed above.

is based on an ANOVA idea where the chains play the role of groups.

The algorithm first requires to run the MCMC with several starting values in parallel. The samples from a fixed MCMC chain can be regarded as a group. Hence, if we run M many parallel chains for a parameter θ and get the posterior samples $(\theta_{m,i})_{1\leq i\leq n}$ for each $m = 1, \ldots, M$. Here n stands for total number of samples. We can apply classical within- and between-group ANOVA analysis to check if there is any group difference. That's the main idea.

There are two more diagnostic tools. 1) Heidelberger–Welch (HW) diagnostic which kind of automates Geweke diagnostic. The above convergence diagnostics are all based on the posterior mean. There is 2) Raftery–Lewis (RL) diagnostic which implements a quantile based diagnostic measure, not posterior mean.

4 Effective sample size

This number essentially tells us our generated MCMC samples correspond to how many independent samples. It is given by (Total postburn samples)$/(1 + 2\sum_{j=1}^{M} |\rho_j|)$, where ρ_j is the correlation associated with *j-th* lag. In R outout `acf`, it is $j+1$-*th* values. Hence if $M = 7$ and (Total postburn samples) $= 5000$. It will be `(Total postburn samples)/(1+2*sum(acfvals$acf[2:(M+1)]))`.