Monte Carlo Independent Project 1

Introduction

In this independent project, I will be examining the 3 convergence diagnostics outlined in the lecture notes and expand them according to my understanding.

Part 1: Gelman and Rubin (1992)

General idea

The Gelman and Rubin method uses ideas from ANOVA to detect the lack of convergence of the chain. We run a total of J chains, which is analogous to having J treatment groups in ANOVA. We start off with starting points from an overdispersed estimate of the target distribution. Each chain is then run for 2N iterations where the first N iterations are discarded as burn-in. We then compute the variance of the means of the chain B, which can be interpreted as the between chain variance that is analogous to the between group variance in ANOVA.

$$B = \frac{1}{J-1}\sum_{j=1}^{J}(I_f^j[h] - \bar{I}_f^j[h])$$

where $I_f^j[h]$ is the ergodic average of the jth chain based on the last half of N iterations and $\bar{I}_f^j[h])$ is the average of the ergodic averages of J chains.

And the within chain variance W, which is the mean of the variance of each chain,

$$W = \frac{1}{J}\sum_{j=1}^{J}\frac{1}{N-1}\sum_{n=N+1}^{2N}(h(X_j^{(n)}) - I_f^j[h])^2$$

where $h(X_j^{(n)})$ is the nth iterate of the jth chain. We note that W will tend to underestimate the true within chain variance early on as $X_j^{(n)}$ has yet to traverse the entire stationary distribution.

Let V be the weighted average between B and W such that $V = \frac{N-1}{N}W + B$ and we can then calculate the potential scale reduction factor R:

$$R = \sqrt{\frac{V}{W}} = \sqrt{\frac{N-1}{N} + \frac{B}{W}}$$

We note that R has a F-distribution as it can be shown that similarly to ANOVA, that $\frac{B}{W}$ has a F-distribution with $d_1 = J-1$ and $d_2 = \frac{2W^2}{((1/J)*var(s_i^2))}$, where s_i^2 is the average of the within-chain variance. (Gelman and Rubin (1992)) R tends to starts off as bigger than the true variance which is 1, especially for slowly mixing chains, due to the overdispersed starting points. But as $n \to \infty$, $R \to 1$. Because $B \to 0$ and W gets bigger as $X_j^{(n)}$ traverses the entire stationary distribution. We can then test the hypothesis of R=1 under the null hypothesis of no lack of convergence and then repeat the procedure by increasing the number of iteration 2N until R is statistically equivalent to 1 under the null hypothesis.

Algorithm

1) Obtain an overdispersed estimate of the target distribution and sample the starting points from it.
2) Run J chains, each with 2N iterations and use the last N iterations to compute B,W,R.
3) Test R=1 using the F distribution and if a lack of convergence is detected, increase number of iterations N and return to step 1.

Advantages

1) Simple to compute as we are only computing sample averages and sample variances.

Disadvantages

1) Requires multiple chains with large iterations. Computationally expensive.
2) Discards a large number of samples, which is inefficient.
3) Relies on the ability to find an overdispersed estimate of the target distribution.

Part 2: Geweke (1991)

General idea

Geweke's method can be used to detect the non convergence of Markov Chains and estimate an appropriate burn-in period for the ergodic average to converge. It is a time-series approach similar to the two-sample test of means that compares the means and variances of non-overlapping samples from the beginning and end (usually the first 0.1 and last 0.5 proportions) of a Markov Chain.

Consider samples X^n of a Markov Chain and test function h, the process $\{h(X^n)\}$ can be treated as a time series and the spectral density of the time series is defined as $S(\omega) = \frac{1}{2\pi}\sum_{k=-\infty}^{\infty}\gamma_k exp(ik\omega)$, where $\gamma_k = Cov(h(X^{(0)}), h(X^{(k)})$.

Also, define the ergodic averages of the initial and final segments of the chain as

$I_f^A[h] := \frac{1}{N_A}\sum_{n=M+1}^{M+N_A} h(X^{(n)})$ $I_f^B[h] := \frac{1}{N_B}\sum_{n=N-N_B}^{N} h(X^{(n)})$

If assumptions for the existence of a spectral density with no discontinuities at frequency 0 hold, and the ratios N_A/N and N_B/N are fixed with $(N_A + N_B)/(N - M) < 1$. Then by the Central Limit Theorem,

$$\frac{I_f^A[h] - I_f^B[h]}{\sqrt{\frac{1}{N_A}\hat{S}_A(0) + \frac{1}{N_B}\hat{S}_B(0)}} \to N(0,1)$$

where $\frac{1}{N_A}\hat{S}_A(0)$ and $\frac{1}{N_B}\hat{S}_B(0)$ are the asympotitc variances of $I_f^A[h]$ and $I_f^B[h]$ respectively. The hypothesis of $I_f^A[h] - I_f^B[h] = 0$ can then be tested using this asympotic relationship.

Discussion

The main idea is that if the chain converges to a stationary distribution, the ergodic averages will be similar. But we can only prove non convergence and cannot guarantee convergence of the Markov Chain. Using this idea, we can also determine the number of burn-in to use by taking the smallest number of iteration where the ergodic average of the initial segment is statistically the same with the ergodic average of the final segment.

Advantages

1) Addresses both bias and variance of the estimator, we can also build confidence intervals using the asymptotic relationship.
2) Requires only one chain
3) Can be applied to any MCMC method
4) Intuitive way to determine the burn-in number

Disadvantages

1) Sensitive to the specification of the spectral window n_A and n_B (Cowles and Carlin 1996)

Minor notes

1) The Gelman and Rubin method uses multiple chains in contrast to only one chain in the Geweke method.
2) The Gelman and Rubin method throws away much more samples than the Geweke method and is therefore much more inefficient.
3) The Gelman and Rubin method gives only an estimate of the bias while the Geweke method estimates both bias and variance.
4) The Geweke method needs to compute spectral densities which could be computationally expensive whereas the Gelman and Rubin method needs to compute multiple chains of 2N iterations. So it is hard to tell which one is more computationally expensive.

Part 3: Raftery and Lewis (1991)

General idea

When we are interested in estimating posterior quantiles, we can use the Raftery and Lewis method to give us the number of iterations to perform such that we can achieve the estimate of the posterior quantile "q" within a certain user specified accuracy "r" with a certain user specified probability "s".

The method is based on 2 state Markov Chain theory which is as follows:

If we have samples $X^{(n)}$ of a Markov Chain, we can construct a 2 state (binary) Markov Chain $\{Z_n\} = 1(X^{(n)} <= u_q)$ where u_q is the quantile value for the quantile q which the user specified . The main idea is that the serial dependency of $\{Z_n\}$ falls with lag. So while the process $\{Z_n\}$ is not a Markov Chain, we can form a new process $\{Z_n^{(k)}\}$ where $Z_n^{(k)} = Z_{1+(n-1)k}$, where k is the smallest value where a first order Markov Chain model of $Z_n^{(k)}$ is preferable to the second order Markov Chain. Details of how k is chosen can be found in Raftery and Lewis (1991).

We then use $Z_n^{(k)}$ to find the number of burn-in samples M for $Z_n^{(k)}$ to approach within "r" of its stationary distribution. Let

$$P = \begin{pmatrix} 1-\alpha & \alpha \\ \beta & 1-\beta \end{pmatrix}$$

be the transition matrix for $Z_n^{(k)}$. Then through the computation details described in Raftery and Lewis (1991), we can show that $M = \frac{log(\frac{\epsilon(\alpha+\beta)}{max(\alpha,\beta)})}{log(1-\alpha-\beta)} * k$

We then use k to determine the number of iterations to run using the formula:

$N = \frac{\frac{\alpha\beta(2-\alpha-\beta)}{(\alpha+\beta)^3}}{(\frac{r}{\Phi(\frac{1+s}{2})})^2} * k$

The last number that we have to determine is the number of iterations for the initial chain N_{min}:

$N_{min} = \Phi^{-1}(\frac{1+s}{2})^2 q(1-q)/r^2$

Thus the algorithm will be as follows

Algorithm

1) Specify the quantile you want to estimate q, with probability s, of estimating q within a tolerance r.
2) Calculate N_{min}
3) Run the MCMC algorithm for N_{min} iterations. You will get $X^{(n)}$ for n=1 ,..., N_{min}
4) Calculate the process $\{Z_n\}$ for each X_n and find k.
5) Use k to get the transition matrix of process $\{Z_n^{(k)}\}$
6) Calculate the number of iterations N and burn-in M.

Discussion

1) A large value of M suggests slow convergence to the stationary distribution of the Markov Chain, while a value of N much larger than N_{min} and/or "k" greater than 1 suggest strong autocorrelations within the Markov Chain. (Cowles and Carlin 1996)

Advantages

1) Robust summaries of the posterior distribution that are calculated using quantiles become available. e.g: median, scaled version of interquartile range.

Disadvantages

1) Prone to initiation bias. Given different initial chains, the number of iterations N outputted by the algorithm can vary a lot. (Cowles and Carlin 1996)
2) Need to rerun the method for every q that you want to estimate.

Part 4: Implementing the Gelman and Rubin method

```
#target is beta(0.5,0.5) which is a bimodal distribution
target = function(x){
  if(x<0 | x>1){
    return(0)}
  else {
    return( (x^(-0.5))*((1-x)^(-0.5)) )
  }
}

#metropolis hastings algorithm
easyMCMC = function(niter, startval, proposalsd){
```

```
  x = rep(0,niter)
  x[1] = startval
  for(i in 2:niter){
    currentx = x[i-1]
    proposedx = rnorm(1,mean=currentx,sd=proposalsd)
    A = target(proposedx)/target(currentx)
    if(runif(1)<A){
      x[i] = proposedx       # accept move with probabily min(1,A)
    } else {
      x[i] = currentx        # otherwise "reject" move, and stay where we are
    }
  }
  return(x)
}
```

```
#Drawing Samples from the Metropolis Hastings sampler
set.seed(1)
N = 10000
J = 100
ergodic_average_list = c()
W_list = c()

for (j in 1:J){
  z = easyMCMC(20000,sample(c(0.9,0.1),1),0.1) #starting values randomised to be 0 or 1
  z = z[(N+1):(2*N)] #discarding first N iterations
  ergodic_average_list = c(ergodic_average_list, mean(z)) #collecting ergodic average for each chain
  W_list = c( W_list, sum((z - mean(z))^2) ) #computing W
}
```

```
#Computing results

B = sum((ergodic_average_list - mean(ergodic_average_list))^2)/(J-1)
W = sum(W_list)/(J*N - J)
V = (((N-1)/N) * W) + B
R = sqrt(V/W)
df2 = (2*W^2)/(var(W_list/(N-1)) * (1/J)) #df2 calculated as given in gelman and rubin (1991)
R
```

```
## [1] 1.005845
```

```
qf(0.95, J-1, df2)
```

```
## [1] 1.244865
```

R is statistically similar to 1 under the null distribution, with alpha level = 0.95. We conclude that no lack of convergence is detected.

Discussion and Bibliography

1) Mary Kathryn Cowles; Bradley P. Carlin. Markov Chain Monte Carlo Convergence Diagnostics: A Comparative Review. Journal of the American Statistical Association, Vol. 91, No. 434. (Jun., 1996), pp. 883-904.

2) J. Geweke. Evaluating the accuracy of sampling-based approaches to the calculation of posterior moments, volume 196. Federal Reserve Bank of Minneapolis, Research Department Minneapolis, MN, 1991.
3) A. Gelman and D. B. Rubin. Inference from iterative simulation using multiple sequences. Statistical science, 7(4):457–472, 1992.
4) A. E. Raftery and S. Lewis. How many iterations in the Gibbs sampler? Technical report, Washington University Seattle Department of statistics, 1991.