Markov Chain Monte Carlo Sampling Methods

Harshal Priyadarshi
UTEid - hp7325

March 6, 2016

1 Overview

Both rejection and importance sampling though quite effective in smaller dimensions, suffer from the curse of dimensionality in higher dimensions. Rejection sampling sufferes from the issue of almost all samples being rejected, because of huge amount of region above the actual distribution blankedted by the proposal distribution. Importance sampling also suffers from the similar issue in higher dimensions. Thus a more general purpose and popular framework called Markov chain Monte Carlo (MCMC) was developed, which allows sampling from large class of distributions and is robust to higher dimensionality. The three major MCMC algorithms are:

- Metropolis Algorithm[1]
- Metropolis - Hastings Algorithm[1]
- Gibbs Sampling[2]

In this report we will look at the 1^{st} and 3^{rd} algorithm.

2 Algorithm

2.1 Metropolis Algorithm

1. Choose the number of samples required, T ,and the proposal distribution to be used, $q(x)$. The original distribution is $p(x)$
2. Initialize the first sample vector x randomly.
3. $innov$ $\leftarrow$sample from the proposal distribution q using transformational, rejection or importance sampling, whichever is appropriate.
4. $can \leftarrow x + innov$
5. $p\_can = p(can)$
6. $p\_x = p(x)$
7. Update $x \leftarrow can$ with probability $min(1, \frac{p\_can}{p\_x})$ otherwise keep the new sample as the previous sample.
8. Store the new x and repeat from **Step 3**

2.2 Gibbs Sampling

1. Obtain the conditional distribution of the actual distribution $p(x)$ with repsect to all the variables of $x = (x_1, x_2, \ldots, x_M)$ i.e. $p(x_i|x) \forall i \in 1 \ldots M$
2. Initialize $x_i : i = 1, \ldots\ldots, M$
3. for sample in range(0, num_samples):
   (a) for $i = 1, \ldots\ldots, T$:
      i. Sample $x_1^{(i+1)} from\ p(x_1|x_2^{(i)}, x_3^{(i)}, \ldots\ldots, x_M^{(i)})$
      ii. Sample $x_2^{(i+1)} from\ p(x_2|x_1^{(i)}, x_3^{(i)}, \ldots\ldots, x_M^{(i)})$
      iii. ...
      iv. Sample $x_j^{(i+1)} from\ p(x_j|x_1^{(i)}, x_2^{(i)}, \ldots, x_{j-1}^{(i)}, x_{j+1}^{(i)} \ldots, x_M^{(i)})$
      v. ...
      vi. Sample $x_M^{(i+1)} from\ p(x_M|x_1^{(i)}, x_2^{(i)}, \ldots\ldots, x_{M-1}^{(i)})$
   (b) Store the samples $(x_1, x_2, \ldots, x_M)$ obtained after the inner loop of T iterations.

3 Experiment

3.1 Sampling Distribution

The toy sampling distribution chosen for experimentation is a mixture of Gaussians for both 1D case:

$$p(x) = 0.3(p(x| - 25, 10)) + 0.7(p(x|20, 10)) \tag{1}$$

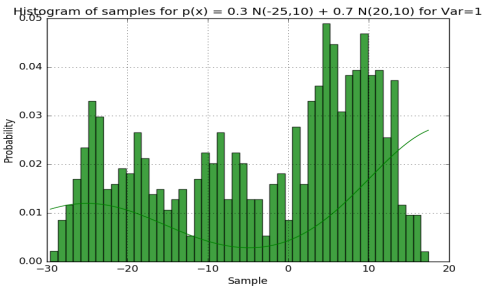
and for the 2D case

$$p(x, y) = 0.3(p(x, y| - 25I, \Sigma)) + 0.7(p(x, y|20I, \Sigma)) \tag{2}$$

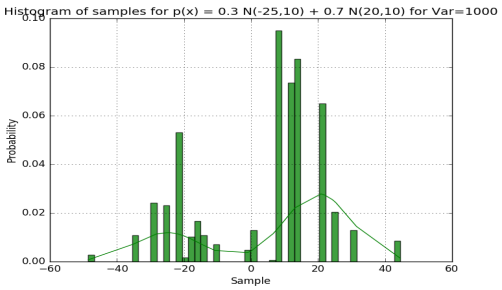
where $I = 2x2$ Identity matrix and $\Sigma = \begin{bmatrix} 10 & 0 \\ 0 & 10 \end{bmatrix}$

3.2 Metropolis Algorithm -1D

The proposal distribution chosen was a 1D gaussian with 0 mean and tunable variance. The alogirthm was run for various values of proposal variance, and the obtained sampled distribution obtained are shown in Fig. 1.

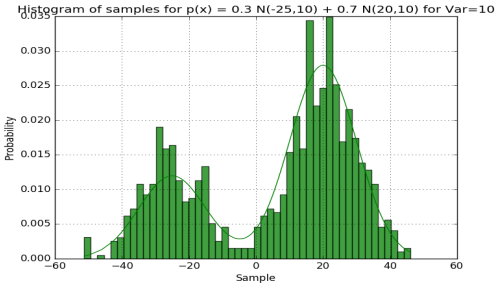


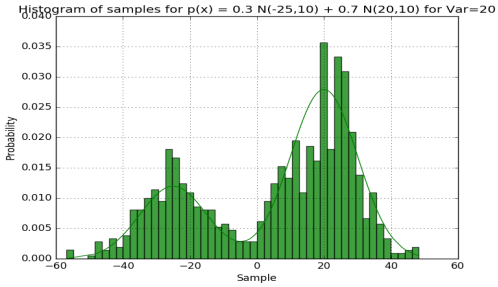
(a) variance = 1 (Very Low)



(b) variance = 1000 (Very High)

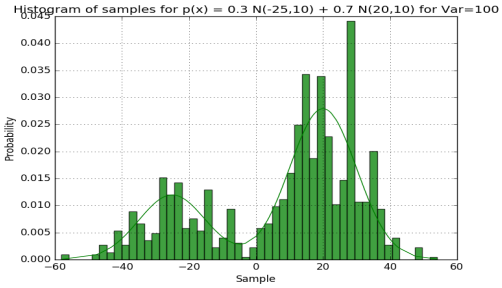
Figure 1: Bad Variances



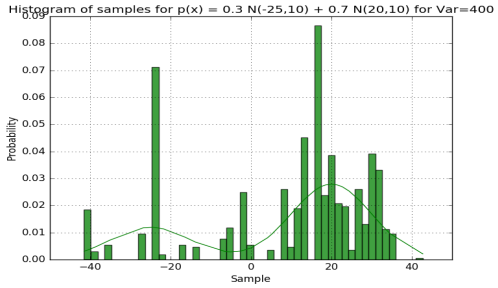


(a) Variance = 10 (Good Reconstruction) (b) Variance = 20 (Good Reconstruction)

Figure 2: Optimal Variances



(a) Variance = 100

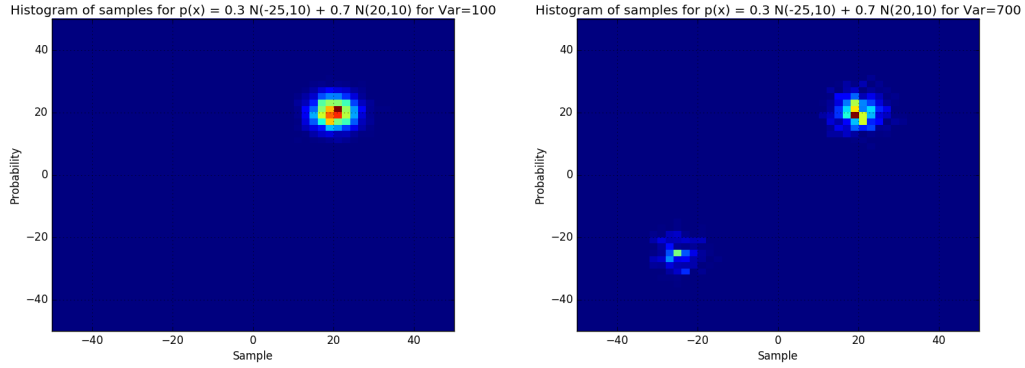


(b) Variance = 400

Figure 3: Additional Plots

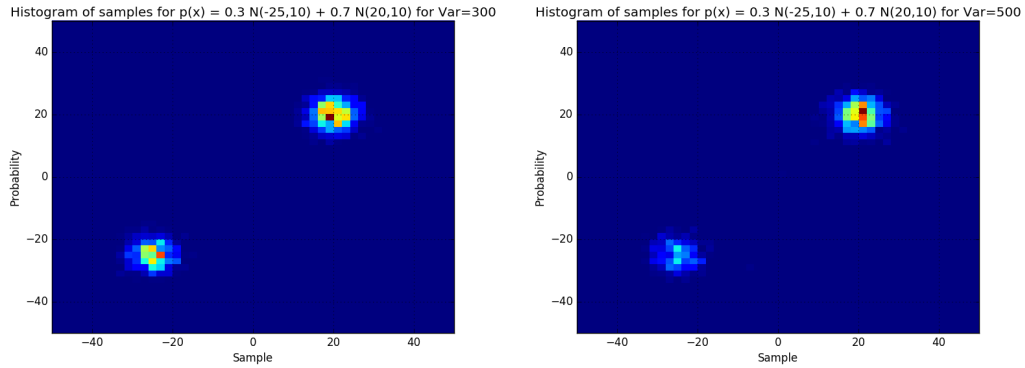
3.3 Metropolis Algorithm - 2D

The proposal distribution chosen here was a bivariate normal distribution with zero mean matrix, and diagonal covariance matrix with same tunable variance across each dimension. The obtained sampling is as shown below.



(a) variance = 100 (Very Low) (b) variance = 700 (Very High)

Figure 4: Bad Variances

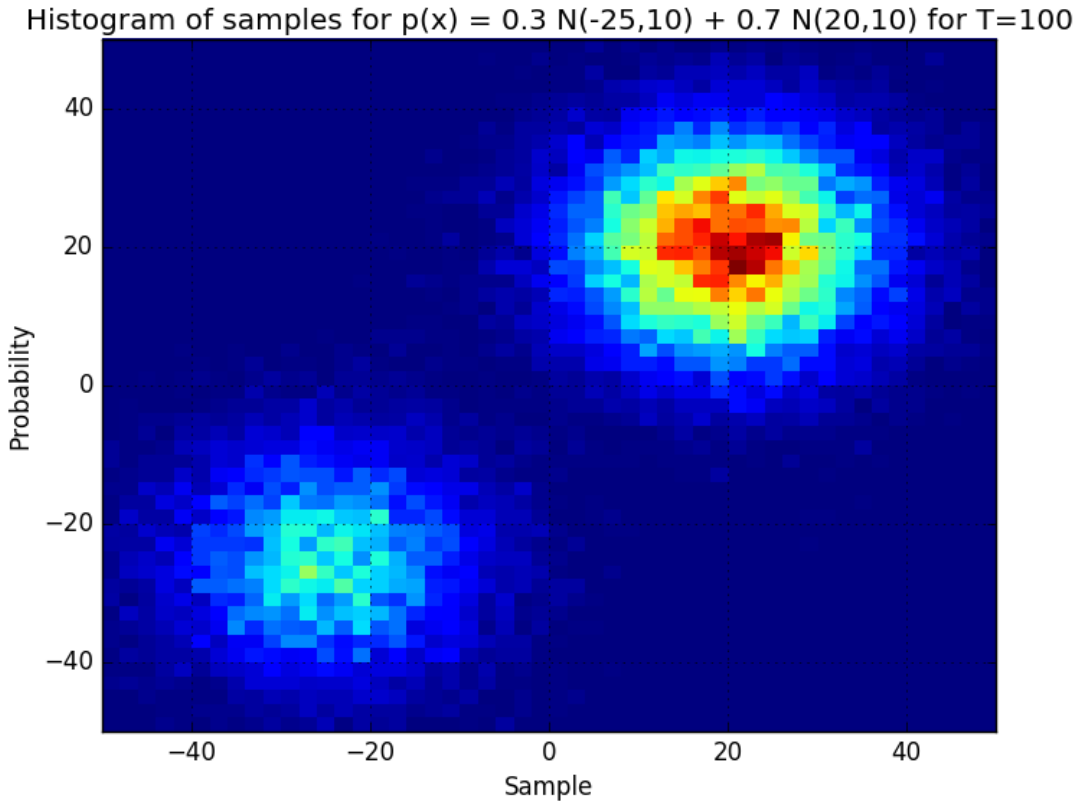


(a) variance=300(Good Reconstruction) (b) variance=500(Avg. Reconstruction)

Figure 5: Good Variances

3.4 Gibbs Sampling - 2D

We were saved by the fact that conditional of a gaussian distribution is also a gaussian distribution. Thus the appropriate gaussian distribution was obtained and then gibbs sampling was done. The obtained plot for Gibbs Sampling is shown below:



4 Observation & Justification

- For the 1D case of Metropolis algorithm, we can see that if the variance is **too low**, then the sampling is done in a very restricted portion of the entire distribution. Most of the samples are close to mean and the maxima are not obtained at $x = -25$ and $x = 20$. Rather they are obtained close to $x = 10$ and $x = -20$, which is not the true reconstruction.

- For the 1D case of Metropolis algorithm, we can see that if the variance is **too high**, then the sampling is not smooth, rather is sparse. Most of the samples are chosens at certain values of x while certain x have not received even one sample. This is justified, because of the arbitrariness of the sampling for large variance proposal distribution.
- Similar trend is obtained for the 2D gaussian case, where for small variance, Fig 4(a), the sampling totally ignores the sampling of the gaussian (mean=-25, var = 10) from the mixture. Thus the sampling is local.
- For the variance being very high, we see that the gaussian sampling curve is not smooth, as there are many points of sampling maxima instead of just one at $x, y = (20, 20)$ and $(x, y) = (-25, -25)$
- For the case of Gibbs Sampling we see that the construction is very good, though the variance of the sample obtained is a bit more than the case of metropolis algorithm for the same number of samples (30000 samples in each algorithm). This factor will decrease with the increase in T as the samples will converge. It seems that the samples have not converged yet. However, training on large T with same number of samples desired was taking a lot more computation time than expected. Doing this can be a future scope of work.
- In all the sampling algos, we see that more of samples are taken from the second gaussian in the mixture, which has higher weight (0.7), this shows that our results are consistent with the desired result.
- The maxima was also close to the mean in all sampling methods, which also shows consistency.

References

[1] Chib, Siddhartha, and Edward Greenberg. "Understanding the metropolis-hastings algorithm." The american statistician 49, no. 4 (1995): 327-335.

[2] Casella, George, and Edward I. George. "Explaining the Gibbs sampler." The American Statistician 46, no. 3 (1992): 167-174.