Finding Solution to Linear System of Equations using Gibbs Sampling

Abstract

In this work, we explore Gibbs Sampling as a method to solve a linear system of equations evading computationally intensive matrix multiplications. We evaluate the proposed method using synthetic data generated randomly.

1 Introduction

There are many methods to solve a linear system of equations like the Matrix Inversion method [5] and Cramer's method [2], involves calculating the product of two matrices. Matrix Multiplication is computationally intensive since the complexity of multiplying two matrices, say $A(m \times n)$ with $B(n \times p)$ is of Order $O(m \times n \times p)$. So for two square matrices of Order $n \times n$ is $O(n^3)$. This becomes very inefficient in large dimension matrices when using methods like matrix inversion to find the solution to the linear system of equations.

Methods like the steepest descent [6], conjugate gradient [6] and Jacobi Method [1] solve this issue by multiplying vectors instead of matrices which reduces the complexity to $O(n^2)$. In this work, we explore Gibbs Sampling as an alternative method to solve a linear system of equations evading computationally intensive matrix multiplications.

2 Method

Before solving the actual problem we tried using Gibbs sampling to sample from a Multi-Variate Normal(MVN) given mean and inverse of covariance matrix without finding the covariance matrix. Say we are dealing with n-dimensional multivariate which means Mean(μ) is a $n \times 1$ matrix and covariance (Σ) is a symmetric positive definite matrix of order $n \times n$ so Σ^{-1} say A is also symmetric and positive definite of order $n \times n$. Our matrix X is a vector $[x_1, x_2, ..., x_n]$, we can break X into two parts X_1, X_2 where X_2 contains only one element whereas X_1 contains $X - X_2$ i.e all elements in X except X_2. Let $X_1 \sim N(\mu_1, \Sigma_{11})$ and $X_2 \sim N(\mu_2, \Sigma_{22})$ then we can split Σ into block matrices,

$$\Sigma = \left[\begin{array}{c|c} \Sigma_{11} & \Sigma_{12} \\ \hline \Sigma_{21} & \Sigma_{22} \end{array}\right].$$

and let $x = [x1, x2..x_{i-1}, x_{i+1}, ...x_n]$ then,

$$E[X_2|X_1 = x] = \mu_2 + \Sigma_{21}\Sigma_{11}^{-1}(x - \mu_1) \tag{1}$$

$$\Sigma_{22|1} = \Sigma_{22} - \Sigma_{21}\Sigma_{11}^{-1}\Sigma_{12} \tag{2}$$

But we don't have Σ we have only A so to get the block matrices of Σ from A we use Schur complement of matrix Σ. Schur Complement [7] of a sample block matrix P, below $S = D - CA^{-1}B$

$$P = \begin{bmatrix} A & B \\ C & D \end{bmatrix}^{-1} = \begin{bmatrix} A^{-1} + A^{-1}BS^{-1}CA^{-1} & -A^{-1}BS^{-1} \\ -S^{-1}CA^{-1} & S^{-1} \end{bmatrix}.$$

31st Conference on Neural Information Processing Systems (NIPS 2017), Long Beach, CA, USA.

Here in place of P, we have our Σ. We can use the matrix $A = \Sigma^{-1}$ to get the terms $\Sigma_{21}\Sigma_{11}^{-1}$ and $\Sigma_{22} - \Sigma_{21}\Sigma_{11}^{-1}\Sigma_{12}$ directly without doing any matrix multiplications. After getting those terms we can update X_2,

$$X_2 \sim N(E[X_2|X_1 = x], \Sigma_{22|1}) \tag{3}$$

and thus we will iteratively modify each dimension of X and generate our samples.

3 Experiments

Many experiments were done starting from comparing the steepest descent and conjugate gradient to find the solution of the linear system of equations to sampling from a multivariate normal distribution with known mean and inverse of the covariance matrix using Gibbs sampling without actually computing the inverse of the covariance matrix

3.1 Experiment for comparing steepest descent and conjugate gradient

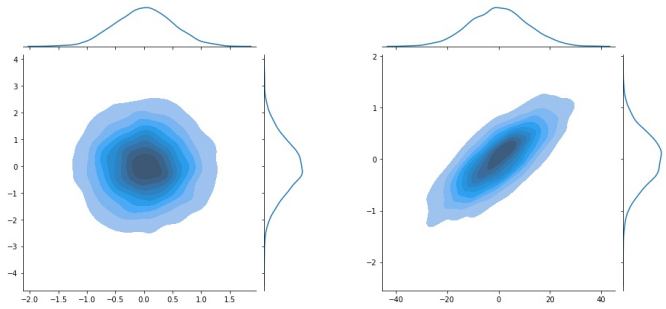
we compared the two algorithms in the number of iterations and time they take before converging to a solution with an error less than 10^-9 while solving the linear system of equations $Ax = b$ where A, b, x_0(initial starting point) were generated randomly. For a fixed dimension, the value of A, b, x_0 were kept the same for both the steepest descent and Conjugate Gradient algorithm. Here A is of Order $n \times n$,b is of Order $n \times 1$ and x is order $n \times 1$

	Steepest Descent		Conjugate Gradient	
n	iterations	time (in sec)	iterations	time (in sec)
10	507682	10.5	10	338μ
50	5419350	117	50	$1.61m$
100(sparse)	2490	$95.3m$	64	$16.5m$

Methods like steepest descent and conjugate gradient work for large dimensions but the conjugate gradient is better because it converges in less number of iterations and time, especially in the case of sparse matrices whereas the positive side of steepest descent is it can converge a bit closer to the actual value than conjugate gradient due to more number of iterations and time taken per iteration is less in steepest descent. Plain steepest descent takes many iterations but with certain optimizations, we can make the steepest descent converge in less number of iterations. [4]

3.2 Experiment to sample from bivariate and draw the KDE contour plots

We sampled from a bivariate with a Mean(μ) and covariance(Σ) using the Cholesky decomposition [3]. For any affine transformation of $X \sim N(0, I)$ to $Y = ZX + M$ then $Y \sim N(M, Z.(Z.T))$ but here we want samples with Mean μ so we keep $M = \mu$ and covariance Σ so $Z.(Z.T) = \Sigma$ which is nothing but Cholesky decomposition [3] of Σ. Hence here $Z = Cholesky(\Sigma)$. In this experiment for simplicity, we took Mean as zero, $\mu = 0$. We get circular contours for bivariate when the two random variables are uncorrelated. When they are correlated, we get elliptical contours.



(a) the plot of bivariate with zero mean and zero covariance

(b) the plot of bivariate with zero mean and non-zero covariance

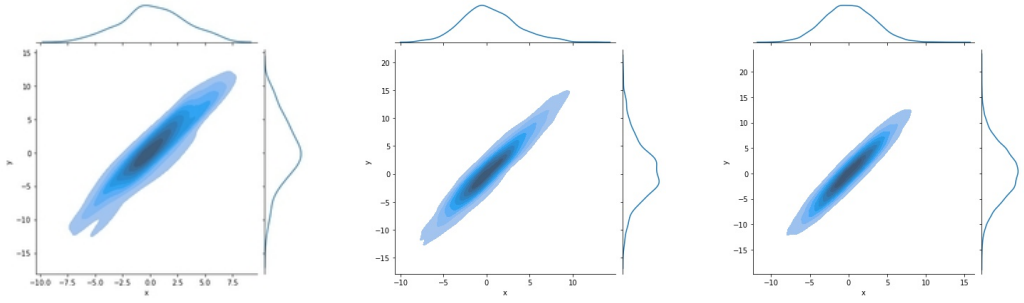
Figure 1: bivariate joint density KDE plots

Contour KDE plots of two gaussian random variables which are uncorrelated are circular. But when we add correlation between those two random variables, the KDE contour plots become elliptical. We can also say that each circular contour C with a radius r has its corresponding elliptical contour E related by $E = LC$, where L is the Cholesky decomposition [3] of Σ

3.3 Experiment verifying the Gibbs sampler

The samples drawn from the Gibbs sampler were plotted to check whether they match with the KDE contour plots for a normal bivariate. We took 100,1000,10000 samples and plotted the respective KDE contour plots for all the samples. For the bivariate, we took Mean $\mu = 0$ and the covariance matrix

$$\Sigma = \begin{bmatrix} 10.10549468 & 15.08509136 \\ 15.08509136 & 24.89129378 \end{bmatrix}.$$



(a) joint bivariate KDE plot of scatter points(100) sampled using Gibbs sampler

(b) joint bivariate KDE plot of scatter points(1000) sampled using Gibbs sampler

(c) joint bivariate KDE plot of scatter points(10000) sampled using Gibbs sampler

Figure 2: Gibbs sampler plots

The KDE contour plots for all the three plots match with the standard bivariate KDE plots which shows that our sampler is indeed generating samples from bivariate normal with a given distribution but the error in the covariance matrix generated from the samples drawn to the actual covariance matrix varies with the number of samples.

References

[1] F Naeimi Dafchahi. A new refinement of jacobi method for solution of linear system equations ax= b. *Int. J. Contemp. Math. Sciences*, 3(17):819–827, 2008.

[2] Mehdi Dehghan, Behnam Hashemi, and Mehdi Ghatee. Computational methods for solving fully fuzzy linear systems. *Applied mathematics and computation*, 179(1):328–343, 2006.

[3] Nicholas J Higham. Cholesky factorization. *Wiley Interdisciplinary Reviews: Computational Statistics*, 1(2):251–254, 2009.

[4] Oleksandra Osadcha and Zbigniew Marszaek. Comparison of steepest descent method and conjugate gradient method. In *CEUR Workshop Proceedings, System*, 2017.

[5] Ivan V Oseledets and Sergey V Dolgov. Solution of linear systems and matrix inversion in the tt-format. *SIAM Journal on Scientific Computing*, 34(5):A2718–A2739, 2012.

[6] Jonathan Richard Shewchuk et al. An introduction to the conjugate gradient method without the agonizing pain, 1994.

[7] Wikipedia contributors. Schur complement — Wikipedia, the free encyclopedia. `https://en.wikipedia.org/w/index.php?title=Schur_complement&oldid=1041999862`, 2021. [Online; accessed 21-April-2022].