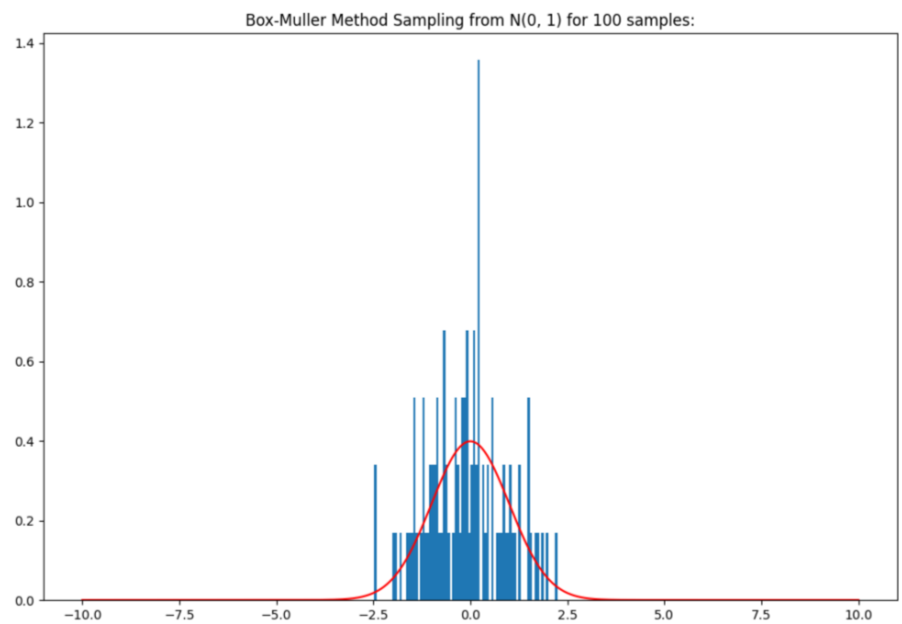
MA323 - Monte Carlo Simulation

Lab - 4

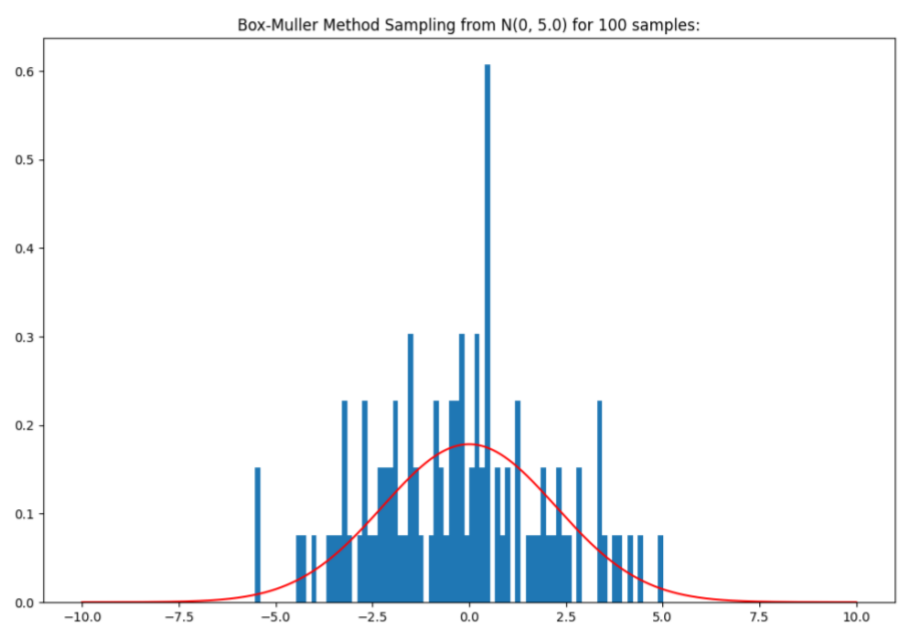
NAVEEN KUMAR A G 210123075

1: Box Muller Method

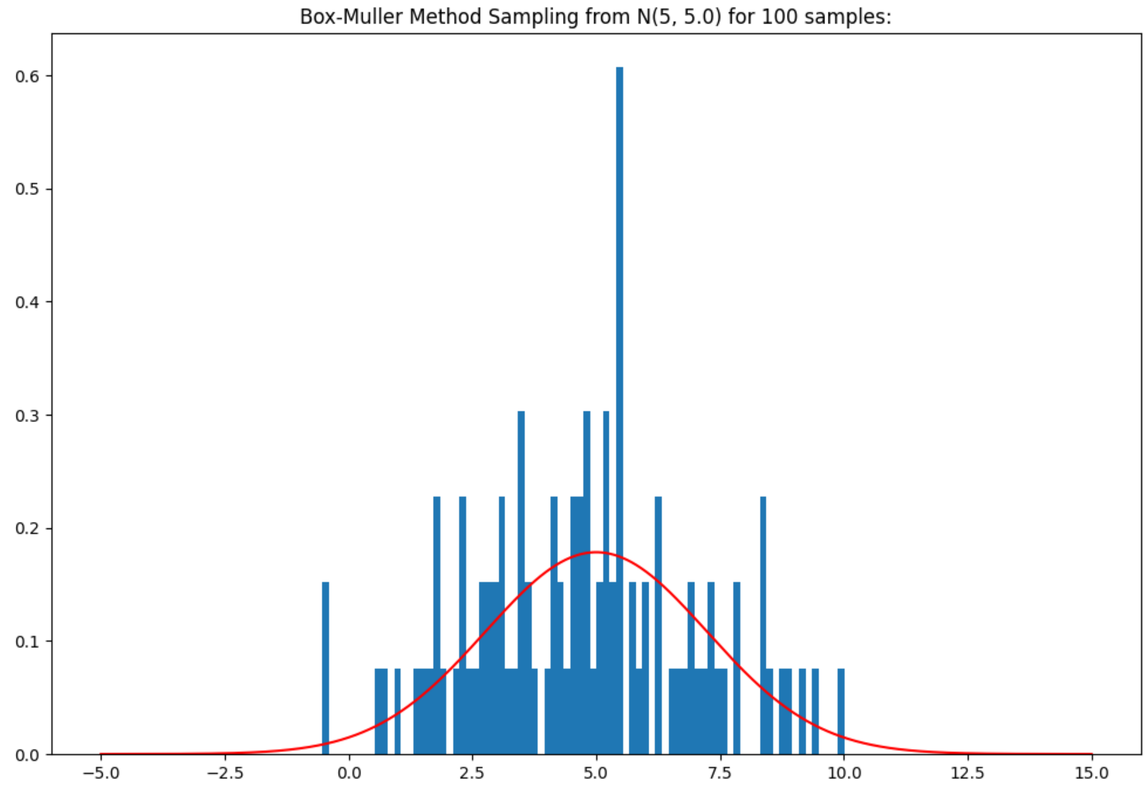
1.1: 100 samples from N(0, 1) distribution:



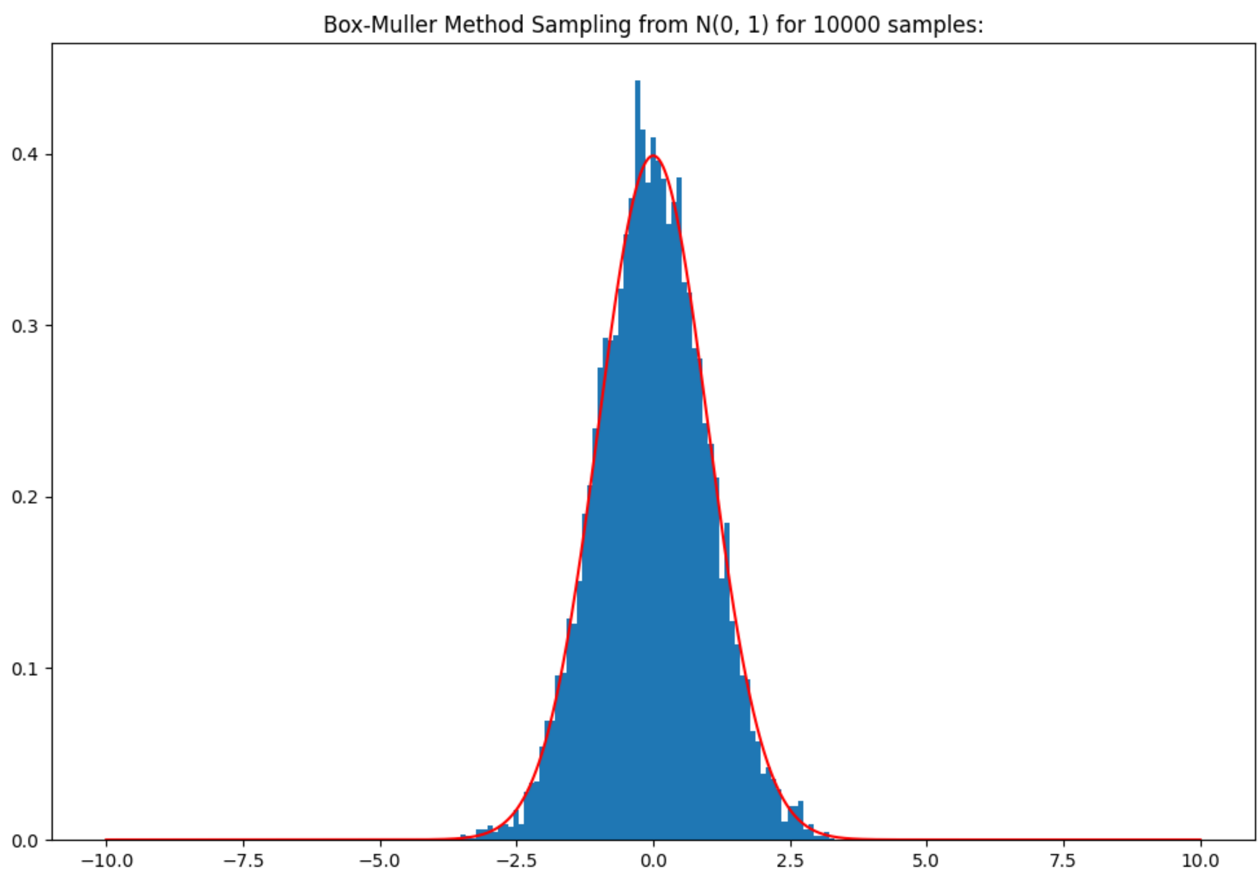
1.2: 100 samples from N(0, 5) distribution:



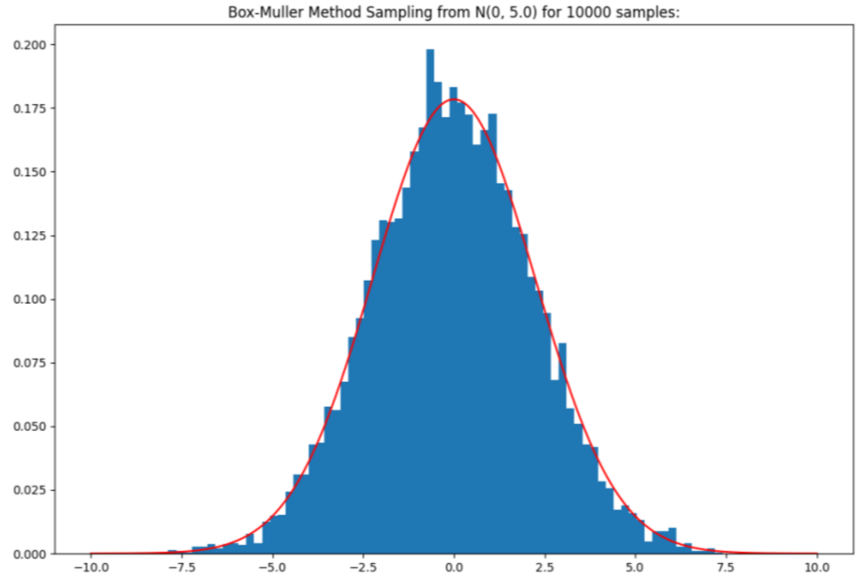
1.3: 100 samples from N(5, 5) distribution:



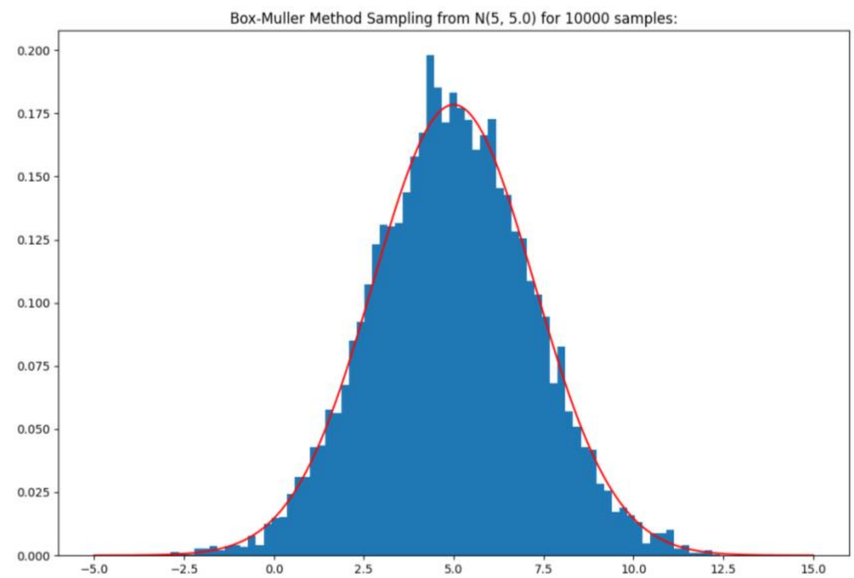
1.4: 10,000 samples from N(0, 1) distribution:



1.5: 10,000 samples from N(0, 5) distribution:



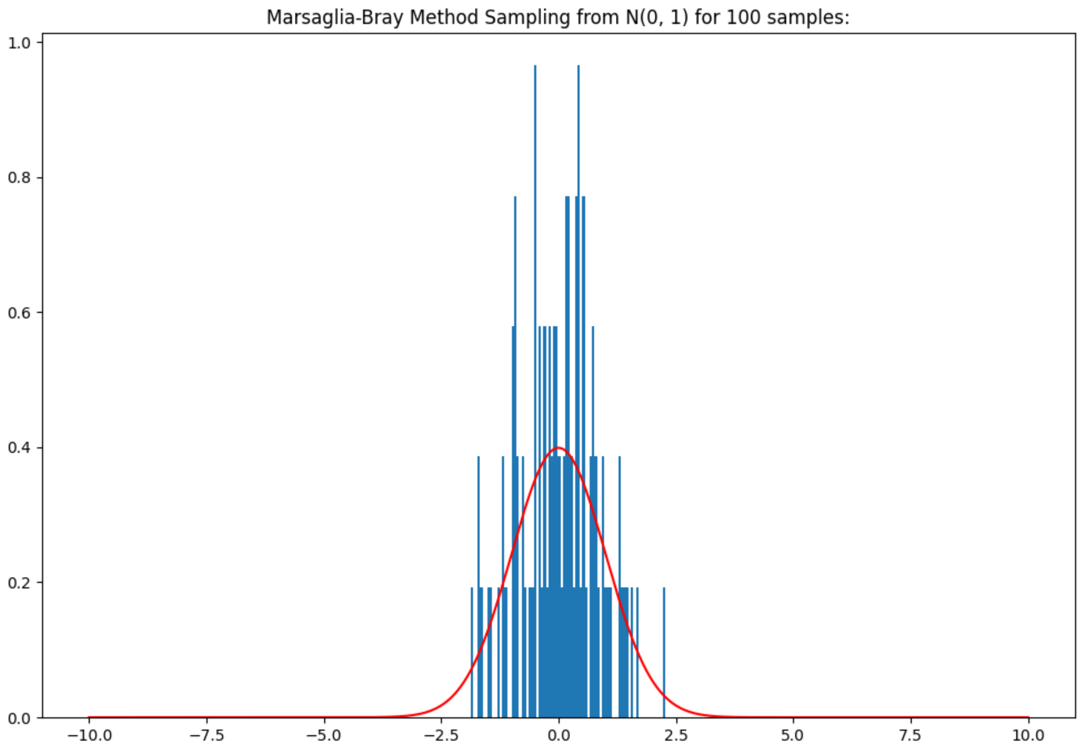
1.6: 10,000 samples from N(5, 5) distribution:



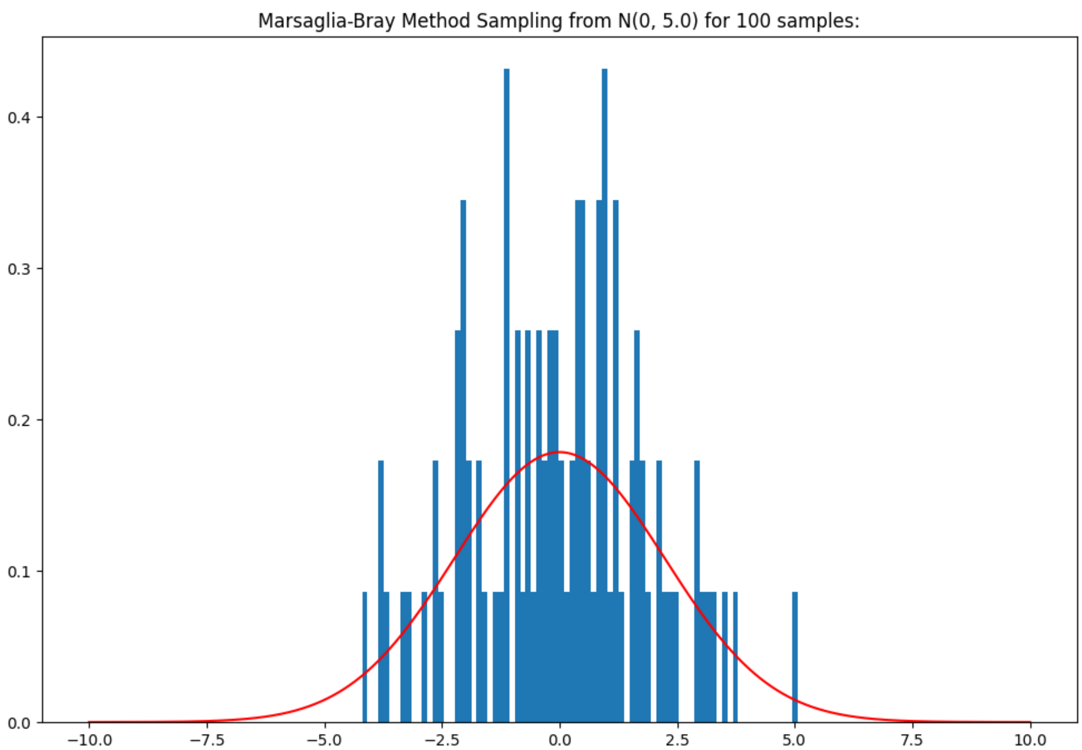
Observations:	**Mean**	**Variance**
1.1: 100 samples from N(0, 1) distribution:	-0.007820909167976146	0.9016163212364471
1.2: 100 samples from N(0, 5) distribution:	-0.0174880845445974	4.508081606182236
1.3: 100 samples from N(5, 5) distribution:	4.98251191545545	4.508081606182237
1.4: 10,000 samples from N(0, 1) distribution:	0.004334843776142069	0.9844307626078244
1.5: 10,000 samples from N(0, 5) distribution:	0.00969300535529554	4.922153813039122
1.6: 10,000 samples from N(5, 5) distribution:	5.009693005355295	4.922153813039122

2: Marsaglia and Bray Method

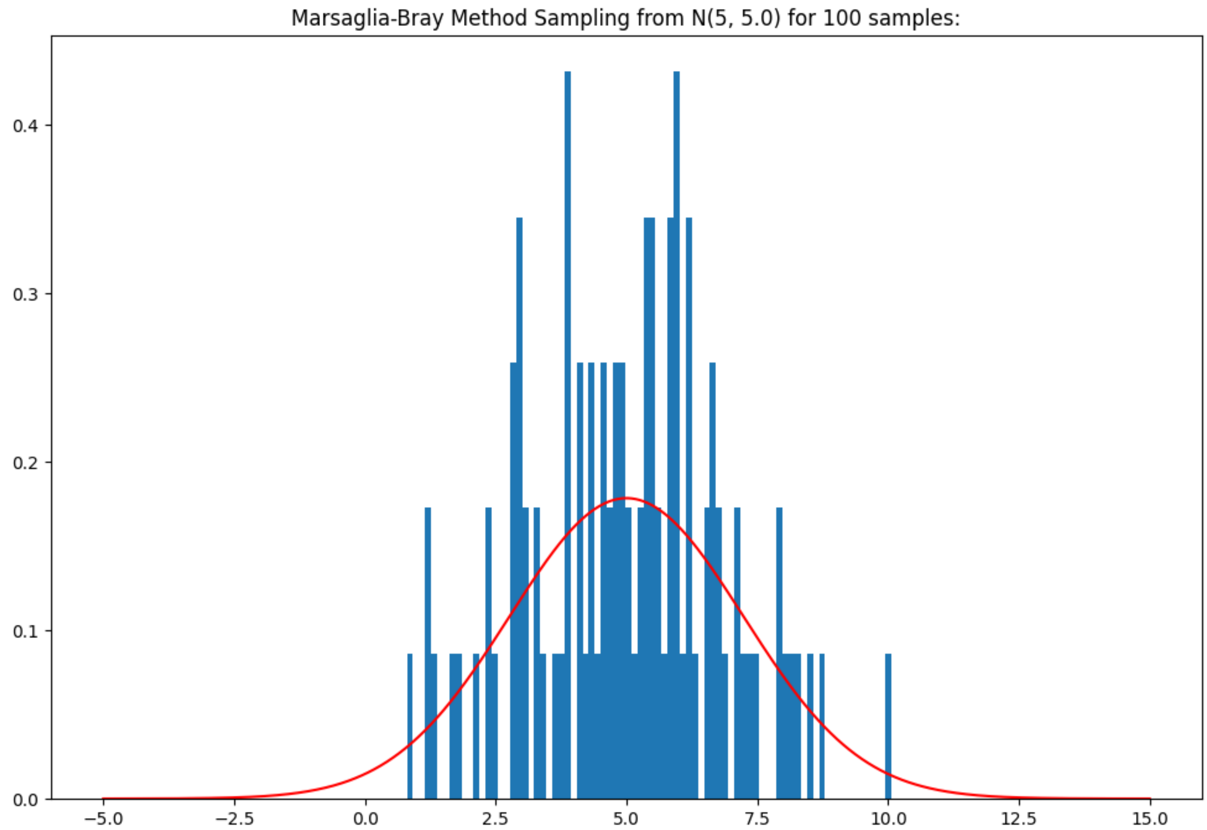
2.1: 100 samples from N(0, 1) distribution:



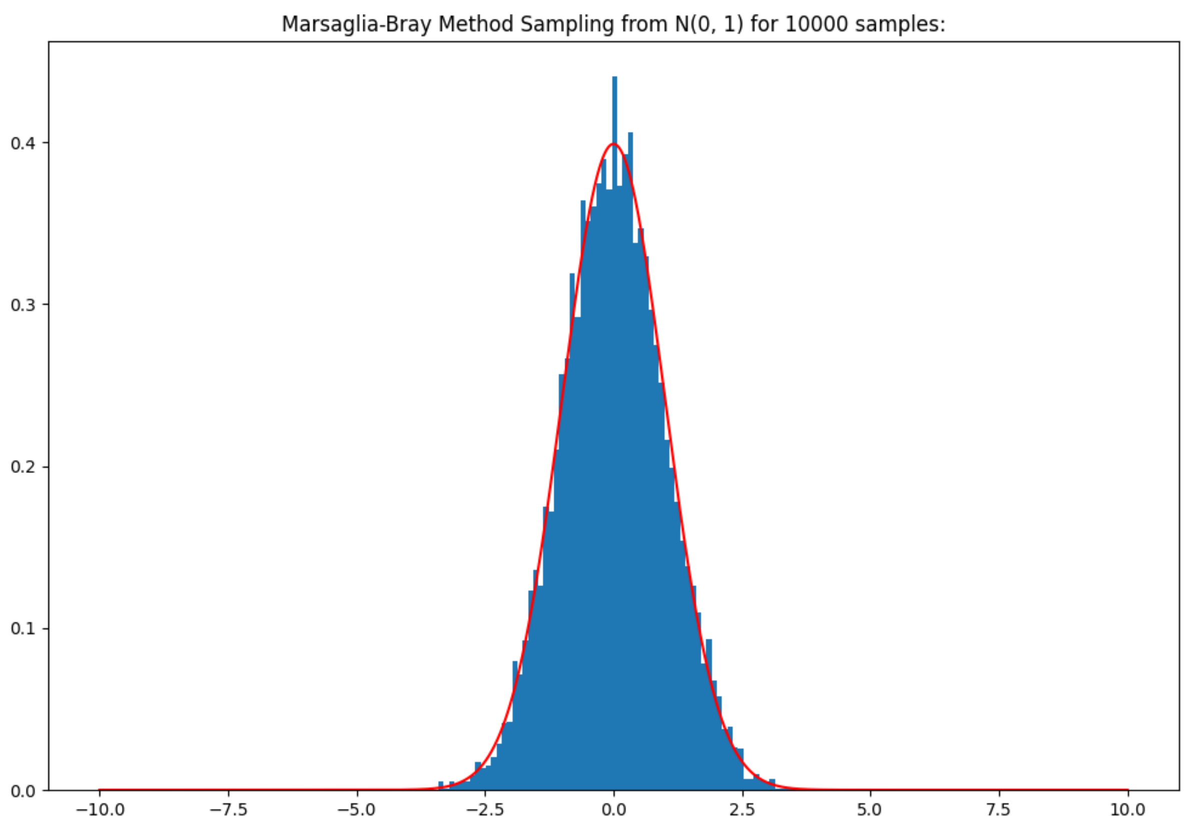
2.2: 100 samples from N(0, 5) distribution:



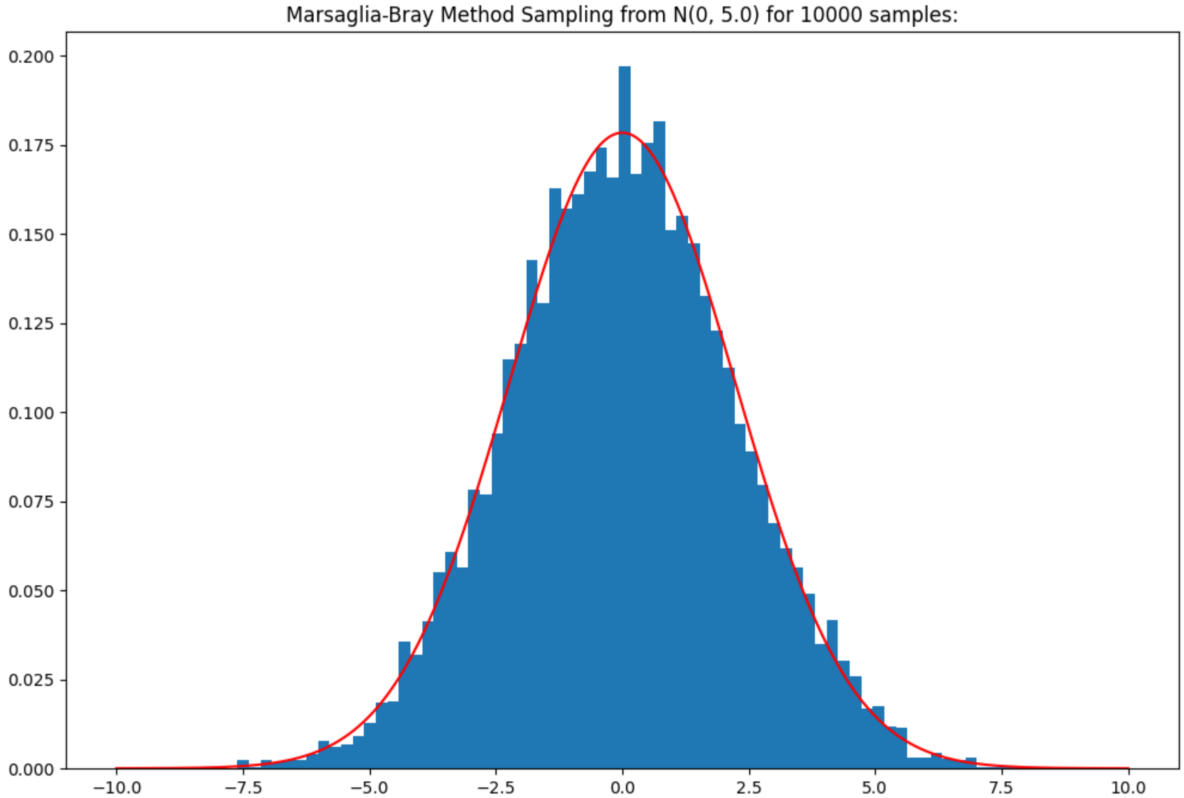
2.3: 100 samples from N(5, 5) distribution:



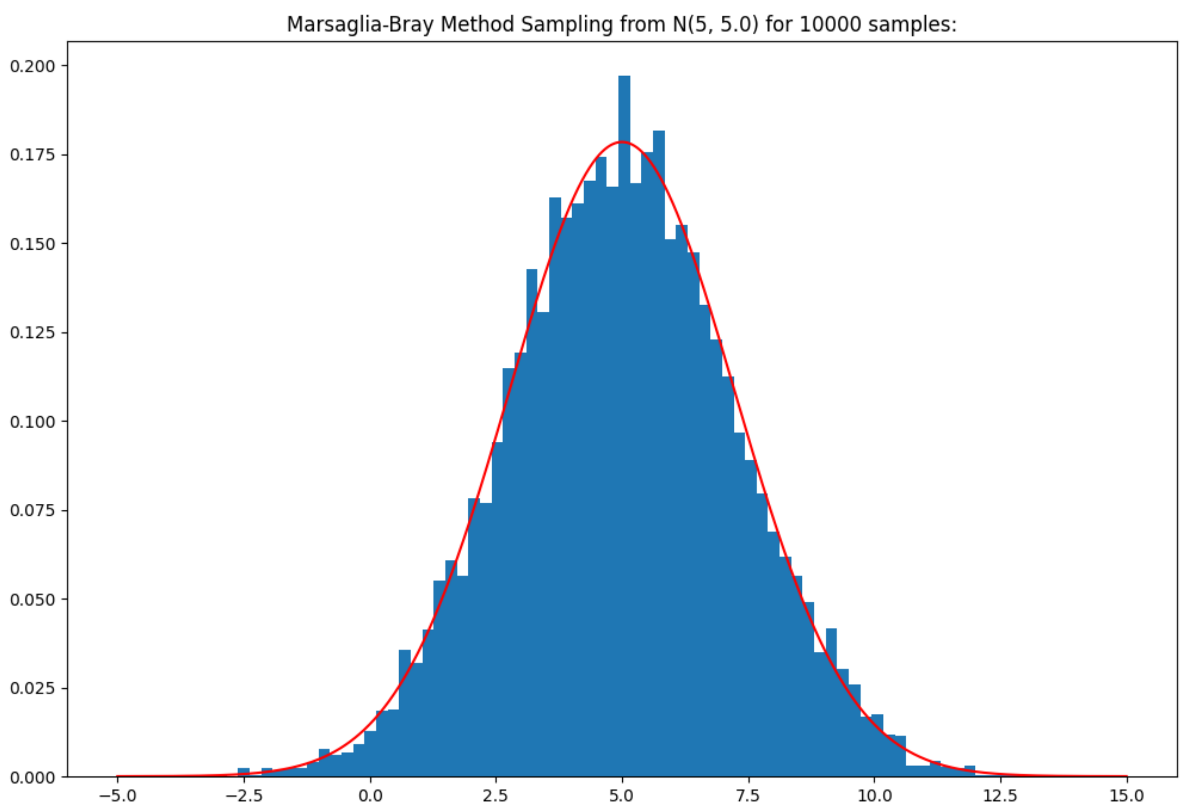
2.4: 10,000 samples from N(0, 1) distribution:



2.5: 10,000 samples from N(0, 5) distribution:



2.6: 10,000 samples from N(5, 5) distribution:



Observations:	**Mean**	**Variance**
2.1: 100 samples from N(0, 1) distribution:	-0.020463256322988482	0.9377005238019894
2.2: 100 samples from N(0, 5) distribution:	-0.0457572321792046	4.688502619009948
2.3: 100 samples from N(5, 5) distribution:	4.954242767820795	4.688502619009948
2.4: 10,000 samples from N(0, 1) distribution:	0.009616830863055573	0.9946974464895022
2.5: 10,000 samples from N(0, 5) distribution:	0.021503887537910236	4.9734872324475115
2.6: 10,000 samples from N(5, 5) distribution:	5.021503887537911	4.9734872324475115

Computational Times

Observations:	**Box Muller**	**Marsaglia and Bray**
100 samples from N(0, 1) distribution:	0.0004911422729492188s	0.0006821155548095703s
100 samples from N(0, 5) distribution:	0.0004911422729492188s	0.0006821155548095703s
100 samples from N(5, 5) distribution:	0.0004911422729492188s	0.0006821155548095703s
10,000 samples from N(0, 1) distribution:	0.024061203002929688s	0.02337479591369629s
10,000 samples from N(0, 5) distribution:	0.024061203002929688s	0.02337479591369629s
10,000 samples from N(5, 5) distribution:	0.024061203002929688s	0.02337479591369629s

Proportion of Values Rejected for Marsaglia and Bray Method.

Observations:	**Rejection Rate**
2.1: 100 samples from N(0, 1) distribution:	0.15254237288135594
2.2: 100 samples from N(0, 5) distribution:	0.15254237288135594
2.3: 100 samples from N(5, 5) distribution:	0.15254237288135594
2.4: 10,000 samples from N(0, 1) distribution:	0.2181391712275215
2.5: 10,000 samples from N(0, 5) distribution:	0.2181391712275215
2.6: 10,000 samples from N(5, 5) distribution:	0.2181391712275215

Note: Since np.random.seed() is not set for the python program, each iteration may give different result.

Observations:

Q-1: It is observed that as the samples size increases, the empirical mean and variance converge to the theoretical mean and variance of the normal density from which it is sampled.

To generate sample **X** from **N (mean, std^2)** from **N (0, 1)**

X = mean + Z * std where **Z ~ N(0, 1)**.

It is observed that as the sample size increases from 100 to 10,000, the empirical density plots become more similar to the density plot generated from the formula. In other words, the two plots overlap more as the sample size increases.

It is observed that the shape of the density plots of **N(0, 5)** and **N(5, 5)** are more distributed than **N(0, 1)** as the variance is higher and the distribution is more spread out. The shape of the density plots of **N(0, 5)** and **N(5, 5)** are same as the plot of **N(5, 5)** is obtained by shifting the plot of **N(0, 5)** by 5 units to the right.

Q-2: The execution times for Box-Muller and Marsaglia and Bray methods were obtained by averaging over 5 simulations.

Empirically, Marsaglia and Bray method runs slower than Box-Muller method for both sample sizes of 100 and 10,000. This is in contrast to the theory according to which Marsaglia and Bray method is quicker as it avoids the computation of *Cosine* and *Sine* functions.

But, it is observed that this trend may not always hold as Acceptance Rejection technique involves accepting only the suitable values and rejecting the unsuitable ones. As the sample size increases, this technique leads to a significant overhead due to which it becomes slower than Box-Muller method with *Cosine* and *Sine* functions.

The method of implementation plays a major role in execution time. If *Cosine* and *Sine* functions are implemented in a very efficient manner, then Box-Muller will be quicker. For n = 100, Box-Muller method beats Marsaglia and Bray method and for n = 10,000, Marsaglia and Bray method beats Box-Muller method.

Q-3: In theory, the proportion of values rejected should be equal to $1 - \pi/4$. This is the area of the discarded region from a box of unit area. Random numbers are chosen such that they lie inside a circle which is inscribed in a square of unit area. As a result, the area of the remaining square is $1 - \pi/4$, which measures the proportion of the values rejected.

It is observed that as the sample size increases from 100 to 10,000, the rejection rate converges to the theoretical rate of rejection i.e. 0.2146.