

# Efficient Generation of Gaussian Variates Via Acceptance-Rejection Framework

Anamul Haque Sajib

Department of Statistics, Dhaka University, Dhaka-1000, Bangladesh

(Received: 25 March 2019; Accepted: 23 June 2019)

## Abstract

The Gaussian distribution is often considered to be the underlying distribution of many observed samples for modelling purposes, and hence simulation from the Gaussian density is required to verify the fitted model. Several methods, most importantly, Box-Muller method, inverse transformation method and acceptance-rejection method devised by Box and Muller<sup>1</sup>, Rao et al.<sup>7</sup> and Sigman<sup>8</sup> respectively, are available in the literature to generate samples from the Gaussian distribution. Among these methods, Box-Muller method is the most popular and widely used because of its easy implementation and high efficiency, which produces exact samples<sup>2</sup>. However, generalizing this method for generating non-standard multivariate Gaussian variates is not discovered yet. On the other hand, inverse transformation method uses numerical approximation to the CDF of Gaussian density which may not be desirable in some situations while performance of acceptance-rejection method depends on choosing efficient proposal density. In this paper, we introduce a more general technique by exploiting the idea invented by Wakefield<sup>9</sup> under acceptance rejection framework to generate one dimensional Gaussian variates, in which we don't require to choose any proposal density and it can be extended easily for non-standard multivariate Gaussian density. The proposed method is compared to the existing acceptance-rejection method (Sigman<sup>8</sup> method), and we have shown both mathematically and empirically that the proposed method performs better than Sigman<sup>8</sup> method as it has a higher acceptance rate (79.53 %) compared to Sigman (76.04 %) method.

**Keywords:** Gaussian distribution, Monte Carlo integration, Ratio-of-Uniforms method

## I. Introduction

Gaussian distribution is one of the commonly used distributions in Statistics as it is often assumed the distribution of underlying many observed phenomena. For example, in a classification or clustering problems it is assumed that observed samples come from the mixture of  $K$  component Gaussian distributions. Under this circumstance, to investigate the efficiency of the fitted model simulating from the  $K$  component Gaussian distributions is required (which requires simulation from each component). The above mixture distribution could be a mixture of  $K$  component multivariate Gaussian distributions. For another example, generating sample from Gaussian distribution is required to approximate an intractable integral which involves Gaussian density under Monte Carlo method. In this paper, our work is limited to the problem of generating from one dimensional Gaussian density.

There are several methods available in the literature to generate samples from Gaussian density. For example, Box-Muller method invented by Box and Muller<sup>1</sup>, CDF approximation based approach invented by Rao et al.<sup>7</sup> and Hastings<sup>3</sup> and central limit theorem based approach are the most important. All the methods mentioned here generate standard Gaussian variates at first, and then transform them to get  $X \sim N(\mu, \sigma^2)$ . Among these methods, Box-Muller method is widely used because of its easy implementation procedure and high efficiency, which produces independent and exact samples from Gaussian density. However, generating non-standard multivariate Gaussian density ( $d \geq 3$ ),  $d$  is the dimension, under the current setting of Box-Muller method is not discovered yet<sup>2</sup>.

On the other hand, the CDF approximation based approaches use numerical approximation to the CDF of Gaussian density, which may not be desirable where no such approximations are desired. Furthermore, to the best of our knowledge, approximation to the CDF of non-standard multivariate Gaussian distribution is not available in the literature. Like the CDF approximation based approach,

central limit theorem based approach produces sample which are approximate as well. Acceptance-rejection algorithm produces independent and exact samples from arbitrary target densities provided that it is applicable. This is because it uses another density called proposal density to simulate from the target densities. However, the acceptance rate of acceptance-rejection method dramatically reduces if appropriate proposal is not chosen. Generating Gaussian variates using the basic acceptance-rejection algorithm, Sigman<sup>8</sup> used exponential(*rate*,  $\lambda = 1$ ) proposal which accepts 76.01 % proposals. Using standard Cauchy density as a proposal to generate Gaussian variates accepts 65 % proposals, which is available in the literature. In general, choosing better proposal to simulate from an arbitrary probability density, especially densities belong to unknown family, under acceptance-rejection framework is challenging<sup>9</sup>.

Motivated by the difficulties observed among the above mentioned methods, our aim is to introduce a new technique in this paper to generate samples from one dimensional Gaussian density. This new method will have the important features: (i) produce exact and independent sample (ii) can be easily extendable in multivariate case and does not need any approximation.

Under the above circumstances, we got the idea (generating random variates efficiently using the modified ratio-of-uniforms method from an arbitrary density) invented by Wakefield<sup>9</sup> to implement our aims. Wakefield<sup>9</sup> method is a variant of acceptance-rejection method in which choosing a proposal density is not required and it can be extended easily in non-standard multivariate case.

We organize the rest of the paper as follows: Section 2 presents the description of Gaussian distribution and some relevant terminologies used in this paper such as Ljung-Box test, acceptance-rejection method and its different variants. Section 3 discusses the procedure of acceptance-rejection method and then its different variants are applied to

\* Author for correspondence. e-mail: sajibstat@yahoo.co.uk

produce Gaussian variates. Section 4 presents the results and the discussion which will be followed by a future work presented in sections 5.

## II. Gaussian Distribution and Related Terminologies

The form of the one dimensional Gaussian density which has two parameters mean ( $\mu$ ) and variance ( $\sigma^2$ ) is given by

$$f(x; \mu, \sigma^2) = (2\pi\sigma^2)^{-0.5} e^{-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2}; -\infty < x < \infty$$

where  $-\infty < \mu < \infty$ , and  $\sigma > 0$ . From standard Gaussian variate  $Z$  a Gaussian variate  $X$  with mean  $\mu$  and variance  $\sigma^2$  can be obtained using the transformation  $X = \mu + \sigma Z$ . On the other hand, the form of the  $d$  dimensional Gaussian density is given by

$$f(x; \mu, \Sigma) = (2\pi)^{-d/2} |\Sigma|^{-0.5} e^{-0.5[(x-\mu)^T \Sigma^{-1}(x-\mu)],}$$

where  $\mu$  and  $\Sigma$  are the mean vector and variance-covariance matrix respectively. The above form reduces to standard multivariate Gaussian density when all the covariance elements of variance-covariance matrix are zero. Otherwise it will be known as non-standard multivariate Gaussian density.

### Ljung-Box Test

To test the randomness of a time series Ljung-Box test is widely used in the field of Econometrics and other time series analysis. This test is jointly developed by Ljung and Box<sup>5</sup>. According to them, the algorithm of Ljung-Box test is:

(i)  $H_0$  : the data are independently distributed against  $H_a$  : the data possess some serial correlation up to a certain lag  $h$  (ii) The quantity  $Q = n(n+1) \sum_{k=1}^h [(n-k)^{-1} r_k^2]$ , which is a function of sample autocorrelation  $r$  at lag  $k$  and sample size  $n$ , denotes the test statistic (iii)  $Q \sim \chi_h^2$  under  $H_0$  and reject the null hypothesis if  $Q > \chi_{(1-\alpha, h)}^2$  where  $\chi_{(1-\alpha, h)}^2$  is the  $(1-\alpha)^{th}$  quintile of the  $\chi^2$  distribution with  $h$  degrees of freedom.

### Acceptance-Rejection (AR) Method

Generating a random variate  $X$  using the inversion method requires inverting its distribution function i.e.  $F_X^{-1}(U)$  where  $U \sim Uniform(0, 1)$ . Most of the time inverting is not possible. Even though inverting is available in inversion method, alternative methods such as acceptance-rejection method can be used to simulate  $X$  that could be more efficient than the inversion method. The basic idea of acceptance-rejection method proposed by Neumann<sup>6</sup> is to choose a proposal density  $g(x)$  to simulate  $X$  from  $f(x)$ . The proposal density  $g(x)$  is chosen in such a way that  $g(x)$  is very close to  $f(x)$ , and there is an efficient method available to simulate from  $g(x)$ . A proposal  $Y$  simulated from the proposal density  $g(y)$  is considered to follow  $f(x)$  i.e.  $Y \sim f(x)$  under acceptance-rejection method if  $U \leq \frac{f(y)}{Mg(y)}$ , where  $Sup_x \left\{ \frac{f(y)}{g(y)} \right\} \leq M$  and  $U \sim Uniform(0, 1)$ . The detailed procedure of acceptance-rejection method to simulate a sample of size  $n$  is discussed in Algorithm 1.

### Ratio-of-Uniforms Method

Ratio-of-uniforms is one of the random variates generation techniques from an arbitrary probability density, often

specified up to proportionality, under acceptance-rejection framework which was proposed by Kinderman and Monahan<sup>4</sup>. Unlike, the conventional acceptance-rejection method, this technique doesn't require any proposal density to sample from an arbitrary probability density.

---

#### Algorithm 1: Acceptance-Rejection algorithm

---

**Input:** Proposed value  $Y$  from  $g(y)$

**Output:** Produce  $X$  from the target density  $g(x)$

---

**Begin**

**For**  $i = 1, 2, \dots, n$  **do**

1. Generate  $Y$  from  $g(y)$

2. Calculate  $Sup_y \left\{ \frac{f(y)}{g(y)} \right\} \leq M$

3. Generate  $U \sim Uniform(0, 1)$

4. **If**  $U \leq \frac{f(y)}{Mg(y)}$  **then**

•  $X = Y$

**Else**

• Go back to step 1

**End If**

**End For loop**

• Return all  $X_1, X_2, \dots, X_n$  as a desired sample

**End Begin**

---

Suppose our aim is to simulate from  $f(x) = \frac{f_1(x)}{\int f_1(x) dx} = \frac{f_1(x)}{c} \propto f_1(x)$ , where  $c = \int f_1(x) dx$  is the normalizing constant. To simulate from  $f(x)$  using the conventional acceptance-rejection method, there is a need to choose an efficient proposal density  $g(x)$  to make the acceptance-rejection method efficient discussed earlier. In some situations, finding an efficient proposal density for an unnormalized probability density is quite difficult or even impossible (densities those don't have familiar forms). These situations arise in posterior distribution quite frequently in the Bayesian framework because of the multiplication of likelihood and prior distribution<sup>9</sup>. Ratio-of-uniforms technique alleviates this problems as it doesn't require to choose any proposal density from an unnormalized probability density. Kinderman and Monahan<sup>4</sup> showed that if the joint density of two uniform random variables is uniformly distributed on

$$R = \left\{ (u, v) : 0 < u \leq \sqrt{f_1\left(\frac{v}{u}\right)} \right\} \quad (1)$$

then the variable  $X = \frac{v}{u}$  has probability density function  $f(x) = \frac{f_1(x)}{c} \propto f_1(x)$ . The proof of the above theorem is considered in this paper as it will be required in the generalized ratio-of-uniforms method, which is our main interest.

**Proof:** To prove the above theorem, Kinderman and Monahan<sup>4</sup> introduce a new variable  $Y$  and make the transformation from  $(U, V) \rightarrow (X, Y)$  via  $X = \frac{V}{U}$  and  $Y = U$  in the first step. In the second step, they find the area of  $R$  and hence obtain the joint density of  $(U, V)$ . Using the joint density obtained in the second step, the joint density of  $(X, Y)$  can be derived with the help of the Jacobian in the penultimate step. Finally, they marginalize the joint density of  $(X, Y)$  with respect to  $Y$  to produce the marginal density of  $X$ , which is the target density. Implementing the first step requires the knowledge of  $|J|$ , where  $J$  is the Jacobian (some time called determinant of the Jacobian matrix) and  $||$  denotes the absolute value sign. The determinant of the Jacobian matrix under this transformation is

$$J = \begin{pmatrix} \frac{\partial u}{\partial x} & \frac{\partial u}{\partial y} \\ \frac{\partial v}{\partial x} & \frac{\partial v}{\partial y} \end{pmatrix} = \begin{pmatrix} 0 & 1 \\ y & x \end{pmatrix} = -y$$

and  $|J| = y$ . Therefore, the area of  $R$  is  $\iint \partial u \partial v = \iint_0^{\sqrt{f_1(x)}} y \partial y \partial x = \int \left[ \frac{y^2}{2} \right]_0^{\sqrt{f_1(x)}} \partial x = \frac{\int f_1(x) \partial x}{2} = \frac{c}{2}$ . Then the form of the joint density of  $(U, V)$  over  $R$  is  $\frac{1}{\text{Area}} = \frac{2}{c}$  as  $(U, V)$  is uniformly distributed on  $R$ . Now the form of the joint density of  $(X, Y)$  is  $\frac{2y}{c}$ , which is the product of the joint density of  $(U, V)$  and  $|J|$ . Finally, integrating the joint density of  $(X, Y)$  with respect to  $Y$  produces the marginal distribution of  $X$ , denoted as  $f_X(x)$ , which is  $f_X(x) = \int_0^{\sqrt{f_1(x)}} \frac{2y}{c} dy = \frac{2}{c} \left[ \frac{y^2}{2} \right]_0^{\sqrt{f_1(x)}} = \frac{f_1(x)}{c} \propto f_1(x)$ . Therefore, the random variable  $X = \frac{V}{U}$  (the ratio of two uniform random variables) has pdf  $f(x) = \frac{f_1(x)}{c} \propto f_1(x)$ . To generate  $(U, V)$  uniformly over  $R$ , the boundary of the region  $R$  needs to be specified firstly. For  $a, b_1$  and  $b_2 \in \mathbb{R}$ , Kinderman and Monahan<sup>4</sup> enclosed  $R$  in a rectangle  $[0, a] \times [b_1, b_2]$  provided that the following theorem is hold:

**Theorem 1:** The region  $R$  will be enclosed in a rectangle  $[0, a] \times [b_1, b_2]$  subject to the conditions that  $f_1(x)$  and  $x^2 f_1(x)$  are bounded where  $a = \sup_x \sqrt{f_1(x)}$ ,  $b_1 = \inf_{x \leq 0} x \sqrt{f_1(x)}$  and  $b_2 = \sup_{x \geq 0} x \sqrt{f_1(x)}$ . The proof of this theorem is not considered in this paper but proof is available in the original paper<sup>4</sup>. The theoretical acceptance probability,  $P_{\text{accep}}$ , of a point generated in the bounding rectangle under the ratio-of-uniforms method is given by

$$P_{\text{accep}} = \frac{\text{Area of } R}{\text{Area of rectangle}} = \frac{c}{2a(b_2 - b_1)} \quad (2)$$

Finally, for symmetrical unimodal densities, Kinderman and Monahan<sup>4</sup> showed that the probability of acceptance  $P_{\text{accep}}$  is maximized when mode of these densities ( $\mu$ ) is relocated to zero which is stated below in Theorem 2.

**Theorem 2:** Without loss of generality, mode ( $x = \mu$ ) of a positive symmetric function  $f_1(x)$  defined on  $\mathbb{R}$  can be rescaled to  $x = 0$ . Furthermore, provided that  $\sup_x [f_1(x)]^{0.5} < \infty$  and  $\sup_x x^2 f_1(x) < \infty$ , then sampling from  $f_1(x)$  is equivalent to sampling from  $f_1(x - u)$ . Under these conditions,  $P_{\text{accep}}$  is maximized when  $u = 0$ .

The proof of the above theorem is not considered here but available in Kinderman and Monahan<sup>4</sup> paper. The detailed procedure of ratio-of-uniforms method to simulate a sample of size  $n$  from an arbitrary probability density  $f(x) = \frac{f_1(x)}{c} \propto f_1(x)$  with bounded  $f_1(x)$  and  $x^2 f_1(x)$  is summarized in algorithm 2.

#### Generalized Ratio-of-Uniforms Method (GRoU)

Wakefield et al.<sup>9</sup> modified the basic version of ratio-of-uniforms method for the sake of increasing the efficiency in terms of acceptance rate of a point generated in the

bounding rectangle. In the generalized ratio-of-uniforms method, a more general version of equation 1 was proposed by introducing a new function  $g$ , which is strictly increasing differentiable function on  $\mathbb{R}^+$  such that  $g(0) = 0$ . The more general version of basic-ratio-uniforms method proposed by Wakefield et al.<sup>9</sup> is stated in Theorem 3.

---

**Algorithm 2:** Algorithm of ratio-of-uniforms method

---

**Input:** Bounding constraints  $a, b_1$  and  $b_2$

**Output:** Produce  $X$  from the target density  $f(x)$

---

**Begin**

**For**  $i = 1, 2, \dots, n$  **do**

1. Generate  $U_1, U_2 \sim \text{uniform}(0,1)$

2. Calculate  $U = a * U_1$  and  $V = b_1 + (b_2 - b_1) * U_2$

3. **If**  $U \leq \sqrt{f_1\left(\frac{V}{U}\right)}$  **then**

•  $X = \frac{V}{U}$

**Else**

• Go back to step 1

**End If**

**End For loop**

• Return all  $X_1, X_2, \dots, X_n$  as a desired sample

**End Begin**

---

**Theorem 3:** For a strictly increasing differentiable function  $g$  defined on  $\mathbb{R}^+$  such that  $g(0) = 0$ , if the joint density of two uniforms random variables is uniformly distributed on  $R = \left\{ (u, v) : 0 < u \leq g^{-1} \left[ k f_1 \left( \frac{v}{g'(u)} \right) \right] \right\}$  (3)

where  $k > 0$  is a constant while  $g'$  and  $g^{-1}$  are the first derivative of the function  $g$  and its inverse function respectively. Then the ratio  $X = \frac{V}{g'(U)}$  has pdf  $f(x) = \frac{f_1(x)}{c} \propto f_1(x)$ . The proof of the above theorem is not shown in original paper<sup>9</sup> but necessary information were given to prove. In this paper, we have proved and presented Theorem 3 in the following as it is necessary to understand how this method works.

**Proof:** The component by which GRoU is different from RoU method is function  $g$ . Wakefield<sup>9</sup> suggested a power function for  $g$  i. e.  $g(u) = u^{r+1}/(r+1)$ , where  $r \geq 0$  and  $k = (r+1)^{-1}$ , for which  $g'(u)$  is strictly increasing function. By considering a new variable  $p = g(u) = u^{r+1}/(r+1)$ , we can find  $g^{-1}(p) = [p(r+1)]^{1/(r+1)}$  while  $g'(u) = u^r$ . By applying both  $g'$  and  $g^{-1}$  to equation 3, we have the following equivalent forms of equation 3:

$$\begin{aligned} R &= \left\{ (u, v) : 0 < u \leq g^{-1} \left[ k f_1 \left( \frac{v}{u^r} \right) \right] \right\} \\ &= \left\{ (u, v) : 0 < u \leq \left[ k f_1 \left( \frac{v}{u^r} \right) \cdot (r+1) \right]^{1/(r+1)} \right\} \\ &= \left\{ (u, v) : 0 < u \leq \left[ f_1 \left( \frac{v}{u^r} \right) \right]^{1/(r+1)} \right\}. \end{aligned} \quad (4)$$

Considering  $x = \frac{v}{u^r}$  and  $y = u$  produce the following Jacobian matrix

$$J = \begin{pmatrix} \frac{\partial u}{\partial x} & \frac{\partial u}{\partial y} \\ \frac{\partial v}{\partial x} & \frac{\partial v}{\partial y} \end{pmatrix} = \begin{pmatrix} 0 & 1 \\ y^r & x y^{(r-1)} \end{pmatrix} = -y^r$$

and hence  $|J| = y^r$ . Therefore, like earlier the area of region  $R$  can be calculated as  $\iint_0^{\sqrt{f_1(x)}} y^r \partial y \partial x =$

$\int_{\left[\frac{y^{r+1}}{r+1}\right]_0^{[f_1(x)]^{r+1}}} \partial x = \frac{\int f_1(x) \partial x}{r+1} = \frac{c}{r+1}$ . The product of the joint density of  $(U, V)$  (reciprocal of the area of  $R$ ) and  $|J|$  produces the joint density of  $(X, Y)$  which is  $\frac{(r+1)y^r}{c}$ . Finally, the marginal density of  $X$  can be obtained by integrating the joint of  $(X, Y)$  out with respect to  $Y$  which is  $\int_0^{[f_1(x)]^{r+1}} \frac{(r+1)y^r}{c} dy = \frac{(r+1)}{c} \left[\frac{y^{r+1}}{r+1}\right]_0^{[f_1(x)]^{r+1}} = \frac{f_1(x)}{c} \propto f_1(x)$ . Therefore, the random variable  $X = \frac{V}{g'(U)}$  has pdf  $f(x) = \frac{f_1(x)}{c} \propto f_1(x)$ . From the above proof, it is observed that the generalized ratio-of-uniforms method reduces to basic form of ratio-of-method when  $r = 1$ . Like the basic ratio-of-uniforms method, the bounding rectangle for the generalized ratio-of-uniforms method exists only when  $f_1(x)$  and  $x^{(r+1)}[f_1(x)]^r$  are bounded, which is stated in Theorem 4.

**Theorem 4:** The region  $R$  will be enclosed in a rectangle  $[0, a] \times [b_1, b_2]$  subject to the conditions that  $f_1(x)$  and  $x^{(r+1)}[f_1(x)]^r$  are bounded where the value of the quantities  $a = \sup_x g^{-1}[k f_1(x)]$ ,  $b_1 = \inf_{x \leq 0} x g'[g^{-1}[k f_1(x)]]$  and  $b_2 = \sup_{x \geq 0} x g'[g^{-1}[k f_1(x)]]$  respectively.

The  $a, b_1$  and  $b_2$ , defined in theorem 4, become  $a = \sup_x [f_1(x)]^{\frac{1}{(r+1)}}$ ,  $b_1 = \inf_{x \leq 0} x [f_1(x)]^{\frac{r}{(r+1)}}$  and  $b_2 = \sup_{x \geq 0} x [f_1(x)]^{\frac{r}{(r+1)}}$  due to simplification by applying both  $g^{-1}$  and  $g'$  (simplification is made like the way equation 4 is derived from equation 3). However, the proof of Theorem 4 which is similar to Theorem 1 is not considered here. Finally, the theoretical acceptance probability under generalized ratio-of-uniform method,  $P_{accept}$ , of a point generated in the bounding is given by

$$P_{accept} = \frac{\text{Area of } R}{\text{Area of rectangle}} = \frac{c(r+1)^{-1}}{a(b_2 - b_1)} \quad (5)$$

---

**Algorithm 3:** Generalized ratio of uniforms method

---

**Input:** Bounding constraints  $a, b_1$  and  $b_2$

**Output:** Produce  $X$  from the target density  $f(x)$

---

**Begin**

**For**  $i = 1, 2, \dots, n$  **do**

1. Generate  $U_1, U_2 \sim \text{uniform}(0,1)$
2. Calculate  $U = a * U_1$  and  $V = b_1 + (b_2 - b_1) * U_2$
3. **If**  $U \leq \left[ f_1 \left( \frac{v}{u^r} \right) \right]^{1/(r+1)}$  **then**

•  $X = \frac{v}{u^r}$

**Else**

- Go back to step 1

**End If**

**End For loop**

- Return all  $X_1, X_2 \dots, X_n$  as a desired sample

**End Begin**

---

The acceptance probability  $P_{accept}$ , defined in equation 5, is a function of  $r$  as all the quantities  $a, b_1$  and  $b_2$  depends on  $r$ . Thus, equation 5 needs to be maximized with respect to  $r$  to get a high acceptance rate, which is the measure of efficiency of an acceptance-rejection algorithm. Finally, relocating the distribution by the mode is suggested by Wakefield et al.<sup>9</sup> before optimizing  $P_{accept}$  over  $r$  in their generalized ratio-of-uniforms method, which yields higher

acceptance rate. The detailed procedure of generalized ratio-of-uniforms method to simulate a sample of size  $n$  from an arbitrary probability density function  $f(x) = \frac{f_1(x)}{c} \propto f_1(x)$  with bounded  $f_1(x)$  and  $x^{(r+1)}[f_1(x)]^r$  is summarized in algorithm 3.

### III. Generating Gaussian Variates via AR Algorithm

#### Exponential Proposal Density

Sigman<sup>8</sup> developed an algorithm to generate standard Gaussian variates under acceptance-rejection algorithm. Sigman<sup>8</sup> considered the exponential density with rate  $\lambda = 1$ ,  $g(x; \lambda) = e^{-x}, x \geq 0$ , as a proposal density to generate from the distribution of absolute value of  $Z$  i.e.  $|Z|$ , which has density  $f(x) = 2(2\pi)^{-0.5} e^{-x^2/2}, x \geq 0$ . The ratio of two densities is bounded by  $c \leq \sqrt{2e/\pi}$ , which can be obtained by solving  $\sup_x [f(x)/g(x)]$ . Finally set proposal value equal to  $|Z|$  if  $U \leq [f(x)/cg(x)] = e^{-0.5(x-1)^2}$ , where  $U \sim \text{Uniform}(0,1)$ . After generating  $|Z|$ , transform  $|Z|$  to  $Z$  using the symmetry property of standard Gaussian distribution. This can be done by  $Z = |Z|$  if  $U \leq 0.5$  and  $Z = -|Z|$  if  $U \geq 0.5$ , where  $U \sim \text{Uniform}(0,1)$ . Using the  $\text{exp}(1)$  proposal, this method requires on average  $c = 1.315$  number of iterations to generate one standard Gaussian variates as the ratio of two densities is bounded by  $c \leq \sqrt{2e/\pi} = 1.315$ . Generating a sample of size  $n$  according to Sigman<sup>8</sup> method is summarized in Algorithm 4a.

Step 4 of the Algorithm 4a happens if and only if  $-\log U \geq 0.5(y-1)^2$ . As  $-\log U \sim \text{exp}(1)$ , steps 1-4 of Algorithm 4a can be simplified as follows: (i) generate two independent variables  $Y_1, Y_2 \sim \text{exp}(1)$  (ii) set  $|X| = Y_1$  if  $Y_2 \geq 0.5(Y_2 - 1)^2$ . Because of the memoryless property of exponential distribution the quantity  $Y \stackrel{\text{def}}{=} Y_2 - 0.5(Y_1 - 1)^2$ , provided that  $Y_1$  is accepted, is itself a random variable which follows  $\text{exp}(1)$  and independent of  $Y_1$  (proof are given in Sigman<sup>8</sup> paper but not considered here)

---

**Algorithm 4a:** Sigman<sup>8</sup> algorithm to generate  $Z$

---

**Input:** Proposed value  $Y$  from  $g(y)$

**Output:** Produce  $X$  from the target density  $f(x)$

---

**Begin**

**For**  $i = 1, 2, \dots, n$  **do**

1. Generate  $Y \sim \text{exp}(1)$
2. Find  $\sup_y \left\{ \frac{2(2\pi)^{-0.5} e^{-y^2/2}}{e^{-y}} \right\} = \sqrt{\frac{2e}{\pi}} \leq c$
3. Generate  $U \sim \text{Uniform}(0,1)$
4. **If**  $U \leq \frac{f(y)}{cg(y)} = e^{-0.5(y-1)^2}$  **then**
  - $|X| = Y$
  - Generate  $U \sim \text{Uniform}(0,1)$
  - **If**  $U \leq 0.5$  **then**
    - $X = |X|$
  - **Else**
    - $X = -|X|$

**End If**

**Else**

- Go back to step 1

**End If**

**End For loop**

- Return all  $X_1, X_2 \dots, X_n$  as a desired sample

**End Begin**

---

**Algorithm 4b:** Sigman<sup>8</sup> algorithm to generate Z

**Input:** Proposed value Y from g(y)

**Output:** Produce X from the target density f(x)

**Begin**

1. Generate  $Y_1, Y_2 \sim \exp(1)$
2. **If**  $Y_2 \geq 0.5(Y_1 - 1)^2$  **then**
  - $|X| = Y_1$
  - Generate  $U \sim \text{Uniform}(0, 1)$
  - **If**  $U \leq 0.5$  **then**
    - $X = |X|$
  - **Else**
    - $X = -|X|$
  - **End If**
- **Else**
  - Go back to step 1
- **End If**
- Return all  $X_1, X_2, \dots, X_n$  as a desired sample

**End Begin**

This by product Y can be used as one of the  $\exp(1)$  variates needed in step (i) for further iterations, which reduces the expected number of uniforms required to generate one Z. The simplified version of algorithm 4a is used in this paper for generating Z, and procedures for generating one Z are given in Algorithm 4b. Sigman<sup>8</sup> considered  $\exp(1)$  to generate standard Gaussian variates. The ratio  $M = \sup_x [f(x)/g(x)]$  becomes  $\sqrt{\frac{2}{\pi}} \lambda^{-1} e^{\lambda^2/2}$  when the form of  $g(x; \lambda)$  is considered as  $\lambda e^{-\lambda x}$ . Using elementary calculus, it can be shown that  $M = \sqrt{\frac{2}{\pi}} \lambda^{-1} e^{\lambda^2/2}$  is minimized at  $\lambda = 1$ .

*Generating Gaussian Variates via RoU Method*

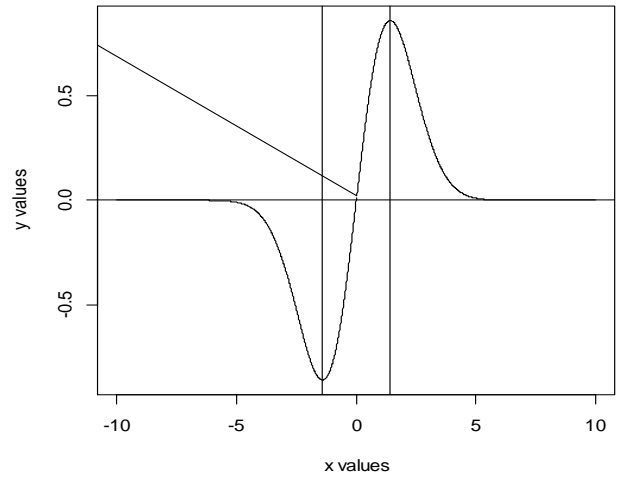
This section describes how RoU method can be used to generate Gaussian variates for different values of location parameters. Suppose our aim is to simulate  $X \sim N(\mu, \sigma^2)$ . In this section, we will show how to generate  $X \sim N(\mu, \sigma^2)$  using the RoU method. To generate  $X \sim N(\mu, \sigma^2)$  using the RoU method, we first generate  $X^* \sim N(\mu, 1)$  and then transform  $X^*$  to X by  $\sigma X^*$  to get  $X \sim N(\mu, \sigma^2)$ . Here we consider different values of  $\mu$  such as  $\mu = 0, 1, 5$  and 10 and show that acceptance rate is decreased when we move away from  $\mu = 0$  (which is stated in Theorem 2).

*Case 1:  $\mu = 0$*

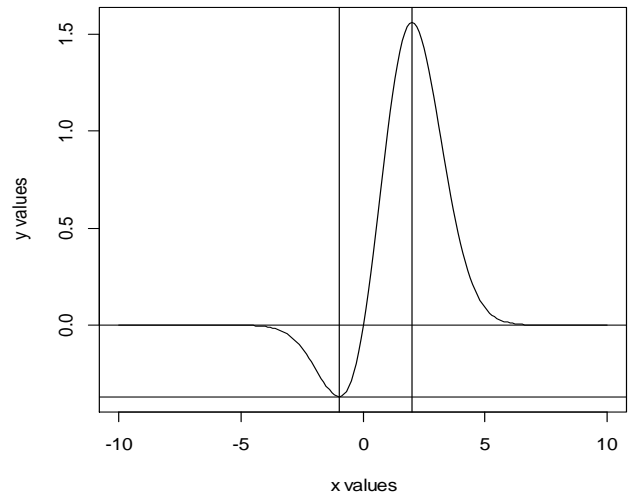
When  $\mu = 0$  and  $\sigma = 1$ , Gaussian density becomes  $f(x) = (2\pi)^{-0.5} e^{-0.5 x^2} = \frac{e^{-0.5 x^2}}{c} = \frac{f_1(x)}{c}$ , where  $f_1(x) = e^{-0.5 x^2}$  and  $c = \sqrt{2\pi}$ . To implement RoU method, we need to find the values of  $a, b_1, b_2$  and  $P_{accept}$ , which can be calculated using the Theorem 1 and the equation 2. (i) First find  $a = \sup_x \sqrt{f_1(x)} = \sup_x e^{-x^2/4}$ . After taking natural logs in both sides, we have  $\log a = \sup_x -x^2/4$ . To maximize  $-x^2/4$  with respect to x requires first and second derivative of  $-x^2/4$  which are  $-x/2$  and  $-1/2$  respectively. Solving  $-x/2 = 0$  yields  $x = 0$  at which  $-x^2/4$  is maximized as the value of second derivative is  $-1/2 < 0$ . Hence  $a = \sup_x e^{-x^2/4} = 1$ . (ii) To find  $b_1$  and  $b_2$ , we need to minimize and maximize  $x e^{-x^2/4}$  for  $x \leq 0$  and for  $x \geq 0$  respectively. From the graph of  $y = x e^{-x^2/4}$  (shown in Figure 1), it is obvious that the magnitudes of y values for  $x \leq 0$  are the same as the

magnitudes of y values for  $x \geq 0$  but have the negative sign. Therefore, finding the maximum of  $x e^{-x^2/4}$  for  $x \geq 0$  will also give us the minimum of  $x e^{-x^2/4}$  for  $x \leq 0$ .

Let's find  $b_2 = \sup_{x \geq 0} x \sqrt{f_1(x)} = \sup_{x \geq 0} x e^{-x^2/4}$  first then. Like earlier, the first and second derivatives of  $\log(x e^{-x^2/4})$  with respect to x are  $(\frac{1}{x} - \frac{x}{2})$  and  $(-\frac{1}{x^2} - \frac{1}{2})$  respectively. Solving for  $(\frac{1}{x} - \frac{x}{2}) = 0$  yields  $x = \sqrt{2}$ , for which the value of second derivative is  $-0.25 < 0$ . Therefore,  $x e^{-x^2/4}$  has a maximum at  $x = \sqrt{2}$ , and the maximum value is  $b_2 = \sqrt{2} e^{-1}$ . Using this  $b_2$  value, we have  $b_1 = -\sqrt{2} e^{-1}$  which is already discussed. Finally, plugging the values of  $a, b_1$  and  $b_2$  into equation 2 yields the theoretical acceptance rate of a point generated in the bounding rectangle which is  $P_{accept} = \frac{c}{2a(b_2 - b_1)} = \frac{\sqrt{2\pi}}{2 * 1 * (2 * \sqrt{2} e^{-1})} = 0.7305$ .



**Fig. 1.** The graph of  $x e^{-x^2/4}$  for  $-10 \leq x \leq 10$ , vertical lines are drawn at  $-\sqrt{2}$  and  $\sqrt{2}$ .



**Fig. 2.** The graph of  $x e^{-(x-1)^2/4}$  for  $-10 \leq x \leq 10$ : vertical lines are drawn at  $-1$  and  $2$ ; horizontal lines are drawn at  $-e^{-1}$  ( $-0.3678$ ) and  $0$ .

Case 2:  $\mu = 1$ : When  $\mu = 1$  and  $\sigma = 1$ , the form of the Gaussian density becomes  $f(x) = \frac{1}{\sqrt{2\pi}} e^{-0.5(x-1)^2} = \frac{e^{-0.5(x-1)^2}}{c} = \frac{f_1(x)}{c}$ , where  $f_1(x) = e^{-0.5(x-1)^2}$  and  $c = \sqrt{2\pi}$ .

The value of  $a = \sup_x \sqrt{f_1(x)} = \sup_x e^{-(x-1)^2/4}$  is 1, which is calculated like the way  $a$  is calculated earlier in case 1 but the detailed calculation are not shown here. Figure 2 plots the function  $y = x e^{-(x-1)^2/4}$  over a domain  $-10 \leq x \leq 10$ , which will be helpful to find the maximum and minimum values for  $x \geq 0$  and  $x \leq 0$  respectively. The first and second derivatives of  $\log(x e^{-(x-1)^2/4})$  are  $(\frac{1}{x} - \frac{x-1}{2})$  and  $(-\frac{1}{x^2} - \frac{1}{2})$  respectively. Solving for  $(\frac{1}{x} - \frac{x-1}{2}) = 0$  yields  $x = -1$  and 2, which are known as stationary points (points at which maximum or minimum can occur). The value of the second derivative at  $x = 2$  is  $-0.25 < 0$  which implies that  $y = x e^{-(x-1)^2/4}$  has maximum at  $x = 2$  and maximum value is  $b_2 = 2 e^{-1/4}$ . On the other hand, at  $x = -1$  the value of the second derivative is  $-1.50$  which also implies that  $y = x e^{-(x-1)^2/4}$  has maximum at  $x = -1$  which appears to contradict the findings from Figure 2 (minimum at  $x = -1$ ). This happens because of there is no such  $x$  values here for which the second derivative will be positive. Therefore, it is not possible to find minimum value of  $y$  using the idea of basic calculus theory. Instead, we will find the infimum of  $y = x e^{-(x-1)^2/4}$  for  $x \leq 0$  using the idea of infimum of a set. From Figure 2, it is obvious that the set of all values of  $y = x e^{-(x-1)^2/4}$  is bounded, bounded from above and below by 0 and  $-e^{-1}$  respectively. Any value which is less than  $-e^{-1}$  is also a lower bound of this set, so the greatest lower bound of this set is  $-e^{-1}$ . Therefore,  $b_2 = \inf_{x \leq 0} x e^{-(x-1)^2/4} = -\frac{1}{e}$ .

Finally, plugging the values of  $a, b_1$  and  $b_2$  into equation 3 yields the theoretical acceptance rate of a point generated in the bounding rectangle which is  $P_{accept} = \frac{c}{2a(b_2 - b_1)} = \frac{\sqrt{2\pi}}{2(2e^{-1/4} + e^{-1})} = 0.6509$ .

Case 3:  $\mu = 5$

When  $\mu = 5$  and  $\sigma = 1$ , the normal density becomes  $f(x) = \frac{1}{\sqrt{2\pi}} e^{-0.5(x-5)^2} = \frac{e^{-0.5(x-5)^2}}{c} = \frac{f_1(x)}{c}$ , where  $f_1(x) = e^{-0.5(x-5)^2}$  and  $c = \sqrt{2\pi}$ . In this case, we have  $a = 1, b_1 = -0.000272$  and  $b_2 = 5.1882$  which are obtained using similar approach used in case 2. Therefore, the theoretical acceptance probability for  $\mu = 5$  is  $P_{accept} = 0.241$ .

Case 4:  $\mu = 10$

When  $\mu = 10$  and  $\sigma = 1$ , the normal density becomes  $f(x) = \frac{1}{\sqrt{2\pi}} e^{-0.5(x-10)^2} = \frac{e^{-0.5(x-10)^2}}{c} = \frac{f_1(x)}{c}$ , where  $f_1(x) = e^{-0.5(x-10)^2}$  and  $c = \sqrt{2\pi}$ . In this case, we have  $a = 1, b_1 = -0.20 e^{-26.01}$  and  $b_2 = 10.098$  which are obtained using similar approach used in cases 2 and 3.

Therefore, the theoretical acceptance probability for  $\mu = 10$  is  $P_{accept} = 0.124$ .

### Generating Gaussian Variates via GRoU Method

In this section, we will show how to generate  $X \sim N(\mu, \sigma^2)$  using the GRoU method. To generate  $X \sim N(\mu, \sigma^2)$  using the GRoU method, we first generate  $X^* \sim N(0, 1)$  i.e. density is relocated via mode  $\mu = 0$  and then transform  $X^*$  to  $X$  by  $\mu + \sigma X^*$  to get  $X \sim N(\mu, \sigma^2)$ .

When  $\mu = 0$  and  $\sigma = 1$ , normal density becomes  $f(x) = (2\pi)^{-0.5} e^{-0.5x^2} = \frac{e^{-0.5x^2}}{c} = \frac{f_1(x)}{c}$ , where  $f_1(x) = e^{-0.5x^2}$  and  $c = \sqrt{2\pi}$ . To implement GRoU method, we need to find the values of  $a, b_1, b_2$  and  $P_{accept}$ , which can be calculated using both Theorem 4 and equation 5. (i) Firstly, we determine the value of  $a$  which is  $a = \sup_x [f_1(x)]^{1/(r+1)} = \sup_x [e^{-x^2/2}]^{1/(r+1)}$ . By taking natural logs in both sides, we have  $\log a = \sup_x -[1/(r+1)]x^2/2$ . Maximize  $-\frac{1}{(r+1)}x^2/2$  with respect to  $x$  requires first and second derivative of  $-\frac{r}{(r+1)}x^2/2$  which are  $-x/(r+1)$  and  $-(r+1)^{-1}$  respectively. Solving  $-x/(r+1) = 0$  yields  $x = 0$  at which  $[e^{-x^2/2}]^{1/(r+1)}$  is maximized as the value of the second derivative is  $-1/(r+1) < 0, \forall r \geq 0$ . Hence  $a = \sup_x [e^{-x^2/2}]^{1/(r+1)} = 1$ .

(ii) To find  $b_1$  and  $b_2$ , we need to minimize and maximize  $y = x[e^{-x^2/2}]^{r/(r+1)}$  for  $x \leq 0$  and for  $x \geq 0$  respectively. From the graph of  $y = x[e^{-x^2/2}]^{r/(r+1)}, \forall r \geq 0$ , (shown in Figure 3), it is obvious that the magnitudes of  $y$  values for  $x \leq 0$  are the same as the magnitudes of  $y$  values for  $x \geq 0$  but have the negative sign. Therefore, finding the maximum of  $x[e^{-x^2/2}]^{r/(r+1)}$  for  $x \geq 0$  will also give us the minimum of  $x e^{-x^2/4}$  for  $x \leq 0$  but will be the negative of maximum value. Figure 3 is plotted for  $r = 0.5$  and considering different values  $r$  doesn't change the shape of  $y$ .

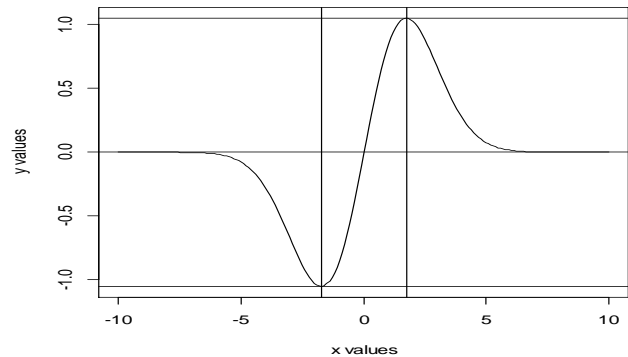


Fig. 3. The graph of  $x[e^{-x^2/2}]^{r/(r+1)}$  over the domain  $-10 \leq x \leq 10$  and  $r = 0.5$ : vertical lines are drawn at  $-\sqrt{(r+1)/r}$  and  $\sqrt{(r+1)/r}$  points; horizontal lines are drawn at  $-\sqrt{(r+1)/r} e^{-0.5}, 0$  and  $\sqrt{(r+1)/r} e^{-0.5}$  points.

Let's calculate the value of  $b_2 = \sup_{x \geq 0} x \sqrt{f_1(x)} = \sup_{x \geq 0} x [e^{-x^2/2}]^{r/(r+1)}$  at first. The first and second derivatives of  $\log(x [e^{-x^2/2}]^{r/(r+1)})$  with respect to  $x$  are  $(\frac{1}{x} - \frac{rx}{r+1})$  and  $(-\frac{1}{x^2} - \frac{r}{r+1})$  respectively. Solving for  $(\frac{1}{x} - \frac{rx}{r+1}) = 0$  yields  $x = \sqrt{(r+1)/r}$ , for which the value of second derivative is  $-2r/(r+1) < 0$  as  $r \geq 0$ . Therefore,  $x [e^{-x^2/2}]^{r/(r+1)}$  has a maximum at  $x = \sqrt{(r+1)/r}$ , and the maximum value is  $b_2 = \sqrt{\frac{(r+1)}{r}} e^{-1}$ .

Using this  $b_2$  value, we have  $b_1 = -\sqrt{\frac{(r+1)}{r}} e^{-1}$  as the magnitude of  $y$  for  $x \leq 0$  is exactly same as  $y$  for  $x \geq 0$  but negative. Finally, plugging the values of  $a, b_1$  and  $b_2$  into equation 5 yields the theoretical acceptance rate of a point generated in the bounding rectangle which is given by

$$P_{accept} = \frac{c(r+1)^{-1}}{a(b_2 - b_1)} = \frac{\sqrt{2\pi}(r+1)^{-1}}{2\sqrt{(r+1)/r} e^{-0.5}} \quad (6)$$

$P_{accept}$  needs to be maximized as high acceptance rate is the measure of efficiency of an acceptance-rejection algorithm. The first and second derivatives of  $\log(P_{accept})$  with respect to  $r$  are  $(\frac{1}{2r} - \frac{3}{2(r+1)})$  and  $(-\frac{1}{2r^2} + \frac{3}{2(r+1)^2})$  respectively. Solving  $(\frac{1}{2r} - \frac{3}{2(r+1)}) = 0$  yields  $r = 0.5$ , for which the value of second derivative becomes  $-1.334 < 0$ . Therefore,  $P_{accept}$  attains its maximum at  $r = 0.5$  and the maximum value is 0.7953. Finally, using the GRoU method to generate Gaussian random variates produces higher acceptance rate compare to Sigman and RoU methods.

#### IV. Results and Discussions

In this section, both theoretical and simulation results obtained under AR, RoU and GRoU methods are presented along with discussions. Here all numerical computations are computed in R on a Samsung XI machine with an Intel (R) Core (TM) i7-4900 (single) processor running at 3.60 GHz. Table 1 shows the theoretical acceptance rate ( $P_{accept}$ ) of a point generated in the bounding rectangle under RoU method for different values of  $\mu$ . From this table, it is clear that  $P_{accept}$  decreases when  $\mu$  moves away from zero, and  $P_{accept}$  is maximized when  $\mu = 0$ .

**Table 1. Theoretical ( $P_{accept}$ ) and simulated ( $\widehat{P}_{accept}$ ) acceptance rates for different values of  $\mu$  under RoU method.**

$\mu$	0	1	5	10
$P_{accept}$	0.7305	0.6509	0.2410	0.1240
$\widehat{P}_{accept}$	0.7272	-	-	-

Therefore, before generating  $X \sim N(\mu, 1)$  the density is relocated to  $X \sim N(0, 1)$  first to achieve maximum acceptance rate. Table 1 also shows the simulated acceptance rate ( $\widehat{P}_{accept}$ ) for  $\mu = 0$  which is very close to

$P_{accept}$ . The simulated acceptance rate is calculated based on a sample of size ten thousand (10,000). We have used random seed number to produce  $\widehat{P}_{accept}$ , and we also observed that using different seed numbers produce approximately similar results. From the above discussions, it is observed that acceptance rate of a point generated in the bounding rectangle decreases when the location parameter moves away from 0, and it is maximized when  $\mu = 0$ . Here we haven't considered the simulated acceptance rate for  $\mu = 1, 5$  and 10 as this is not our main interest. Our main interest is to simulate from Gaussian density via acceptance-rejection method with high acceptance rate, which is achieved when the distribution is relocated via mode  $\mu = 0$ . For the same reason, we have kept variance fixed here (acceptance rate only depends on relocating the density but not rescaling)

A comparison of AR, RoU and GRoU methods for evaluating performance based on acceptance rates is made which is shown in Table 2.

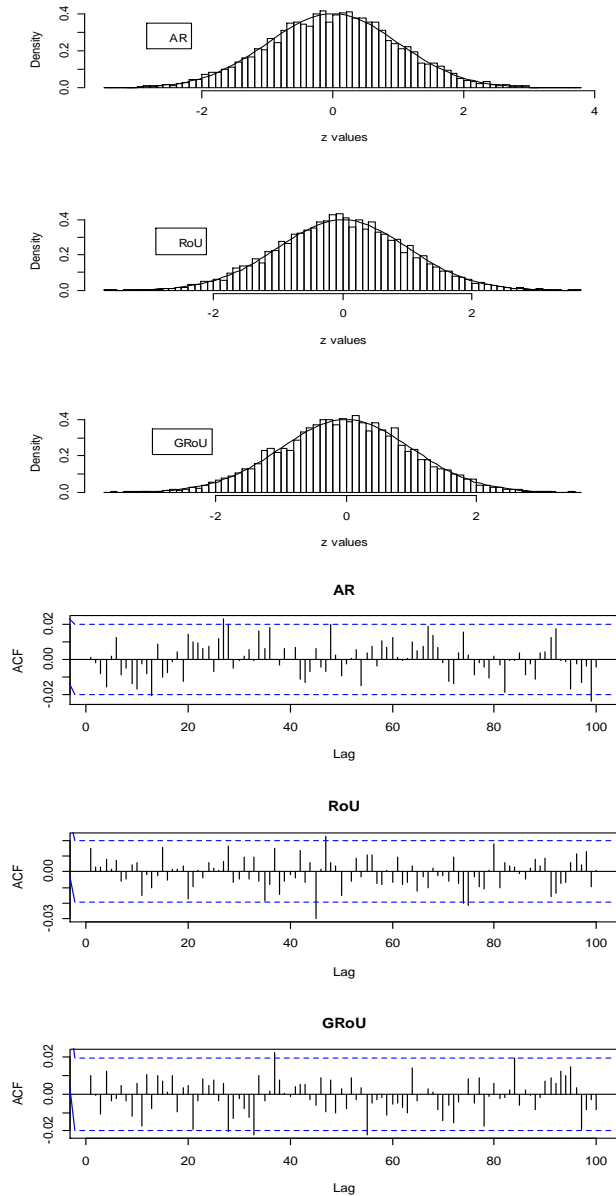
**Table 2. Theoretical and simulated acceptance rate along with computing time for 3 different methods.**

Method	$P_{accept}$	$\widehat{P}_{accept}$	Com. Time (seconds)
AR	0.7604	0.7575	.01
RoU (relocate)	0.7305	0.7272	.0166
GRoU (relocate)	0.7953	0.7968	0.0132

From Table 2, it is observed that GRoU method has the highest acceptance rate among all three methods, and the higher acceptance rate in GRoU method compared to other two methods is achieved without increasing the burden of computational time. All three methods require almost same computing time to generate 10,000 observations.

Normality property of the generated samples for each method is examined through by overlaying the  $N(0, 1)$  densities over their respective histograms, which is shown by the top 3 plots of Figure 4. From these plots, we can see that histograms based on generated samples for each method mimic the true  $N(0, 1)$  density very well, hence we consider the generated samples come from  $N(0, 1)$  density.

On the other hand, both graphical technique (ACF plot) and Ljung-Box test are used to test the randomness of the generated samples obtained under AR, RoU and GRoU methods respectively. The last 3 plots of Figure 1 (ACF plots of generated samples obtained under AR, RoU and GRoU methods respectively), it is observed that some of the ACF at lag around 28 and 98 for AR method, at lag around 45, 49 and 75 for RoU method and at lag around 36 and 55 for GROU method are beyond the significance confidence bands (95%). However, it does not guarantee the presence of autocorrelation, and may happen because of sampling error. To confirm the presence of autocorrelation Ljung-Box test is carried out, and results are shown in Table 3. From Table 2, we observe that the  $P$  Values of Ljung-Box statistic at different lags under all methods considered here are greater than  $\alpha = 0.05$ , which support the null hypothesis of randomness of generated samples obtained under all methods.



**Fig. 4.** Overlaying  $N(0,1)$  density lines over histograms of generated samples (size  $10^4$ ) along with acf plots for AR, RoU and GRoU methods respectively

**Table 2. Ljung-Box test statistic and their corresponding P values in parenthesis at different lags.**

Method	Lag 10	Lag 20	Lag 30	Lag 60	Lag 100
AR	10.398 (0.406)	21.547 (0.365)	35.352 (0.235)	58.651 (0.525)	96.424 (0.582)
RoU	4.5466 (0.919)	13.868 (0.837)	18.999 (0.940)	54.982 (0.661)	89.217 (0.771)
GRoU	6.611 (0.814)	14.331 (0.813)	26.039 (0.673)	52.856 (0.731)	84.80 (0.861)

**V. Conclusion and Future Works**

This paper introduces a new technique to generate Gaussian random variates under acceptance–rejection framework by exploiting the idea (generating random variates efficiently via modified ratio-of-uniform method) invented by Wakefield et al. (1991). The proposed technique for generating Gaussian random variates is considered the best among all methods under acceptance-rejection framework, which has already been shown by mathematically and empirically in terms of acceptance rate. The proposed technique has achieved higher efficiency compared to AR and RoU methods without increasing the computational burden. As a future study one can exploit Wakefield et al.’s idea to simulate from other important univariate and multivariate distributions.

**References:**

1. Box, G. E. P. and M. E. Muller, 1958. A Note on the Generation of Random Normal Deviates. *Annals of Mathematical Statistics*. **29**, 610-611.
2. Dias, J. R., 2010. A Simple Generalization of the Box-Muller Method for Obtaining a Pair of Correlated Standard Normal Variables. *Journal of Statistical Computation and Simulation*. **80(9)**, 953-958.
3. Hastings, C., 1955. *Approximations for Digital Computers*. Princeton University Press, Princeton, N. J.
4. Kinderman, A. J., J. F. Monahan, 1977. Computer Generation of Random Variables Using the Ratio of Uniform Deviates. *ACM Transactions on Mathematical Software* **3(3)**, 257–260.
5. Ljung, G. M. and G. E. P. Box, 1978. On a Measure of Lack of Fit in Time Series Models. *Biometrika*. **65**, 297–303.
6. Neumann, J. V., 1951. Various Techniques Used in Connection with Random Digits. *Monte Carlo Methods*, Nat. Bureau Standards. **12**, 36–38.
7. Rao, K. R., N. K. Boiroju, and M. K. Reddy, 2011. Generation of Standard Normal Random Variables. *Indian Journal of Scientific Research*. **2(4)**, 83-85.
8. Sigman, K., 2007. Simulating Normal (Gaussian) RV’S with Applications to Simulating Brownian Motion and Geometric Brownian Motion in One and Two Dimensions. [www.columbia.edu/~ks20/](http://www.columbia.edu/~ks20/). Sigman/4703-07-Notes- BM-GBM- I.pdf
9. Wakefield, J. C., A. E. Gelfand, A. F. M. Smith, 1991. Efficient Generation of Random Variates via the Ratio-of-Uniforms Method. *Statistics and Computing* **1**, 129–133.