Generation of Gamma Random Variables

Siddhant Garg
150711

February 2018

1 Introduction

The Gamma distribution is well-known and widely used in many signal processing and communications applications. All of the aforementioned applications need the generation of independent Gamma random variables (RVs), X, with arbitrary values of α and λ, i.e., $X \sim \Gamma(\alpha, \lambda)$.

PDF $f(x)$ of $\Gamma(\alpha, \lambda)$ is given by:

$$f(x) = \frac{\lambda^\alpha}{\Gamma(\alpha)} x^{\alpha-1} e^{-\lambda x}$$

where α is shape parameter and λ is the rate parameter.

CDF $F(x)$ is given by

$$\begin{aligned} F(x) &= \int_0^x f(x)dx \\ &= \frac{\gamma(\alpha, \beta x)}{\Gamma(\alpha)} \end{aligned}$$

where $\frac{\gamma(\alpha,\beta x)}{\Gamma(\alpha)}$ is the incomplete gamma function.

Note that inverse of the function $F(x)$ cannot be obtained explicitly. So, various other methods and algorithms are being used to generate from gamma distribution.

In this project, I have provided 7 algorithms found from various resources(cited) to generate 1000 samples from $\Gamma(\alpha, \lambda)$ for different values of α and λ : {(1.5,1), (3.2,1), (100.7,1)}.

A Little Preproccesing

Most of the methods described below are used to generate from $\Gamma(\alpha, \lambda)$, when $0 < \alpha < 1$ and $\lambda = 1$.

To generate from general Gamma Distribution, we will use some properties of the Gamma random variable

$$\alpha = \alpha_i + \alpha_f$$

Where α_i is the integral part of α and α_f is the fractional part.
Note that, if $X \sim \Gamma(\alpha, \lambda)$

$$X \stackrel{d}{=} Y_1 + Y_2$$

where, $Y_1 \sim \Gamma(\alpha_i, \lambda)$ and $Y_2 \sim \Gamma(\alpha_f, \lambda)$, Y_1 and Y_2 are independent.
Further,

$$Y_1 \stackrel{d}{=} Z_1 + \cdots + Z_{\alpha_i}$$

where $Z_i \sim exp(\lambda)$ are independent.
Generation from exp(λ)

$$g(y) = \lambda e^{-\lambda x}$$

$$G(x) = 1 - e^{-\lambda x}$$

Generate $u \sim Uniform(0, 1)$
Return $y = -\frac{1}{\lambda}\ln(1-u)$
Algorithm to Generate from General Gamma Distribution
1. Generate $Z_i \sim exp(\lambda)$, $i = 1, \ldots, \alpha_i$
2. Generate $Y \sim \Gamma(\alpha_f, 1)$ from the proposed algorithm.
3. Return $X = \sum_{i=1}^{\alpha_i} Y_i + \frac{Y}{\lambda}$
Here is the sample from Gamma(α, λ)

2 Methods adopted to generate from Gamma Distribution

2.1 Acceptance-Rejection Method(As done in class)

We start by assuming that the F we wish to simulate from has a probability density function $f(x)$. The basic idea is to find an alternative probability distribution G, with density function $g(x)$, from which we already have an efficient algorithm for generating from (e.g., inverse transform method or whatever), but also such that the function $g(x)$ is "close" to $f(x)$. In particular, we assume that the ratio $f(x)/g(x)$ is bounded by a constant $c > 0$; $\sup_x f(x)/g(x) \leq c$. (And in practice we would want c as close to 1 as possible.)

We can use acceptance-rejection method to generate from Gamma distribution when $0 < \alpha < 1$ and $\lambda = 1$, where λ is rate parameter.

PDF for the above parameters is given by

$$f(x) = \frac{1}{\Gamma(\alpha)} x^{\alpha-1} e^{-x}$$

Note that for

$$g(x) = \begin{cases} kx^{\alpha-1} & 0 < x < 1 \\ ke^{-x} & x \geq 1 \end{cases}$$

Here k is such that $\int_0^\infty g(x)dx = 1$. Therefore, $k^{-1} = \alpha^{-1} + e^{-1}$.
For $c = \frac{1}{\Gamma(\alpha)k}$, $f(x) \leq cg(x)$.
Generation from PDF g()

$$\begin{aligned} G(x) &= \int_0^x g(x)dx \\ &= \begin{cases} \frac{k}{\alpha}x^\alpha & 0 < x < 1 \\ \frac{k}{\alpha} + k(e^{-1} - e^{-x}) & x \geq 1 \end{cases} \end{aligned}$$

Generate $u \sim Uniform(0, 1)$
If $u \leq \frac{k}{\alpha}$, then return $Y = \left(\frac{\alpha u}{k}\right)^{1/\alpha}$
Else return $Y = -\ln\left(\frac{1}{e} - \frac{u}{k} + \frac{1}{\alpha}\right)$
Generation from f(x)
1. Generate $u \sim Uniform(0, 1)$
2. Generate $y \sim g()$
3. If $u \leq \frac{f(x)}{cg(x)}$, then return $x = y$
4. Else goto step 2.

Here x returned, is a sample from $f()$

Here the sample was generated from Gamma(α,λ) when $0 < \alpha < 1$ and $\lambda = 1$.

To generate from general Gamma Distribution, use the algorithm for general Gamma Distribution provided at the last of the introduction section.

2.2 Other Efficient Algorithms [2]

Below are the 3 three algorithms that generate from $\Gamma(\alpha, \lambda)$, when $0 < \alpha < 1$ and $\lambda = 1$

Algorithm 1
1. Generate u from uniform(0,1)
2. Compute $x = -2\ln(1 - u^{1/\alpha})$
3. Generate v from uniform(0,1) independent of u.
4. If $v \leq \frac{x^{\alpha-1}e^{-x/2}}{2^{\alpha-1}(1-e^{-x/2})^{\alpha-1}}$, accept x otherwise goto 1.

Algorithm 2
1. Set $a = \frac{(1-e^{-1/2})^\alpha}{(1-e^{-1/2})^\alpha + \frac{\alpha e^{-1}}{2^\alpha}}$ and $b = (1 - e^{-1/2})^\alpha + \frac{\alpha e^{-1}}{2^\alpha}$
2. Generate u from uniform(0,1)
3. If $u \leq a$, then $x = -2\ln[1 - (ub)^{1/\alpha}]$, otherwise $x = -\ln\left[\frac{2^\alpha}{\alpha}b(1 - u)\right]$.

4. Generate v from U(0,1). If $x \leq 1$, check whether $v \leq \frac{x^{\alpha-1}e^{x/2}}{2^{\alpha-1}(1-e^{-1/2})^{\alpha-1}}$. If true, return x, otherwise goto 1. If $x > 1$, check whether $v \leq x^{\alpha-1}$. If true, return x, otherwise go back to 1.

Algorithm 3

Set $d = 1.0334 - 0.0766e^{2.2942\alpha}$, $a = 2^{\alpha}(1 - e^{-d/2})^{\alpha}$, $b = \alpha d^{\alpha-1}e^{-d}$ and $c = a + b$.

1. Generate u from uniform(0,1)
2. If $u \leq \frac{a}{a+b}, then x = -2\ln\left[1 - \frac{(cu)^{1/\alpha}}{2}\right]$, otherwise $x = -\ln\left[\frac{c(1-u)}{\alpha d^{\alpha-1}}\right]$.
3. Generate $v \sim U(0,1)$. If $x \leq d$, check whether $v \leq \frac{x^{\alpha-1}e^{-x/2}}{2^{\alpha-1}(1-e^{-x/2})^{\alpha-1}}$. If true return x, otherwise go back to 1. If $x > d$, check whether $v \leq \left(\frac{d}{x}\right)^{1-\alpha}$. If true, return x, otherwise goto 1.

To generate from general Gamma Distribution, use the algorithm for general Gamma Distribution provided at the last of the introduction section.

2.3 Ratio of Uniform [1]

In this section we will provide 2 algorithms. First algorithm is used when $\alpha > 1$ and other when $\alpha \leq 1$.

2.3.1 Algorithm for $\alpha > 1$

$$f(x) = cx^{\alpha-1}e^{-x}$$

where c is the normalising constant.

$$s_1 = \sup_x f(x)/c = (\alpha - 1)^{\alpha-1}e^{1-\alpha}$$

$$s_2 = \sup_x x^2 f(x)/c = (\alpha + 1)^{\alpha+1}e^{-1-\alpha}$$

Note that,
1. $f(x)$ is bounded.
2. $x^2 f(x)$ is also bounded.

Algorithm
1. Generate $u \sim U(0, s_1)$.
2. Generate $v \sim U(0, s_2)$.
3. If $u^2 \leq f(\frac{v}{u})$, then accept $x = \frac{v}{u}$, otherwise goto step 1.

2.3.2 Algorithm for $\alpha \leq 1$

If $X \sim \Gamma(\alpha, 1)$, we consider the transformation $T = \alpha \ln X$. The density g(t) of T is

$$g(t) = \frac{\exp(t - e^{t/\alpha})}{\alpha\Gamma(\alpha)}, -\infty < t < \infty$$

Let

$$h(t) = \exp(t - e^{t/\alpha})$$

$$C = \{(u, v) : 0 \leq u \leq \sqrt{h(t)}, -\infty < t < \infty\}$$

From the above we have,

$$0 \leq u \leq \left(\frac{\alpha}{u}\right)^{\alpha/2}$$

$$-\frac{2}{e} \leq v \leq \frac{2\alpha}{e(e-\alpha)}$$

Algorithm
1. Generate $u \sim U(0, (\frac{\alpha}{u})^{\alpha/2})$
2. Generate $v \sim U(-\frac{2}{e}, \frac{2\alpha}{e(e-\alpha)})$.
3. $t = \frac{v}{u}$ and $t_1 = e^{t/\alpha}$.
4. If $2 \ln u \leq t - t_1$, then return t_1, if $\alpha \geq 0.01$; otherwise deliver t/α
5. Return $x = \exp(t/\alpha)$

2.4 TGGD Algorithm [3]

Set $a = \alpha$ and $b = \sqrt{2\alpha - 1}$.

1. Generate $u_1 \sim U(0, 1)$ and $u_2 \sim U(0, 1)$ independently.
2. Set $v = \frac{1}{b} \ln \left[\frac{u_1}{1-u_1}\right]$
3. If $\ln(u_1 u_2(1 - u_1))) \leq \alpha - 1.3862944\alpha v - \alpha e^v$, return $x = ae^v$, otherwise goto 1.

3 Numerical Results

Results for Gamma(1.5,1)

Method of Generation	Proportion of Rejection	Time Taken
Acceptance Rejection	0.240699	0.148
Ratio of Uniform	0.264165	0.1350
TGGD Algorithm	0.246420	0.00456
Algorithm 1	0.269006	0.0098
Algorithm 2	0.136442	0.0118
Algorithm 3	0.151824	0.0114

Results for Gamma(3.2,1)

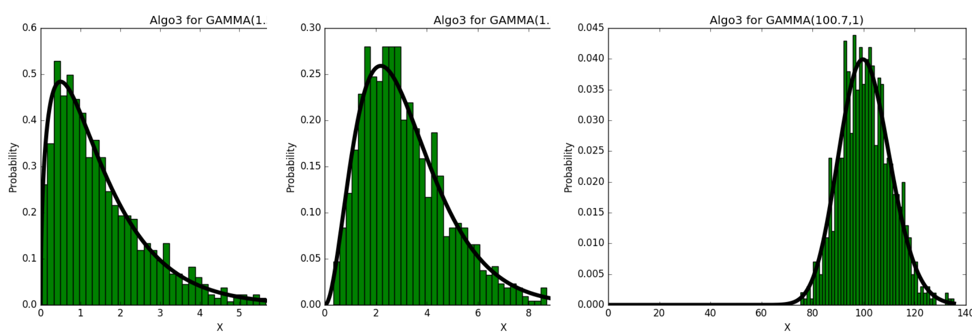
Method of Generation	Proportion of Rejection	Time Taken
Acceptance Rejection	0.144568	0.1321
Ratio of Uniform	0.385749	0.1538
TGGD Algorithm	0.174917	0.0074
Algorithm 1	0.187652	0.0128
Algorithm 2	0.100719	0.0124
Algorithm 3	0.099910	0.0123

Results for Gamma(100.7,1)

Method of Generation	Proportion of Rejection	Time Taken
Acceptance Rejection	0.277457	0.1550
Ratio of Uniform	0.876953	0.7908
TGGD Algorithm	0.111111	0.0035
Algorithm 1	0.303136	0.0115
Algorithm 2	0.145299	0.0171
Algorithm 3	0.120493	0.0156

4 Conclusions

From the above data we can conclude that the **Algorithm 3** of section 2.3 is best among the given algorithms because it has the **least rejection proportion**, for the generation of random samples from the gamma distribution



Gamma (1.5,1) Gamma (3.2,1) Gamma (100.7,1)

Samples Generated using **Algorithm 3** for given Gamma distributions

References

[1] Chuanhai Liu Bowei Xi, Kean Ming Tan. Logarithmic transformation-based gamma random number generators. *ournal of Statistical Software*, 2013.

[2] D. Kundu and R.D. Gupta. A convenient way of generating gamma random variables using generalized exponential distribution. *Computational Statistics and Data Analysis*, 51(120):2796–2802, 2007.

[3] P. R. Tadikamalla. Random Sampling from the Generalized Gamma Distribution. *Computational Statistics and Data Analysis*, 23:199–203, 1979.