Th6: Algorithms for Random Variates Generation

Federico Detomaso

1 Introduction

In simulation, pseudo random numbers serve as the foundation for generating samples from probability distribution models. We will now assume that the random number generator has been rigorously tested and that it produces sequences of $U_i \sim U(0,1)$ numbers. We now want to take the $U_i \sim U(0,1)$ and utilize them to generate from probability distributions.

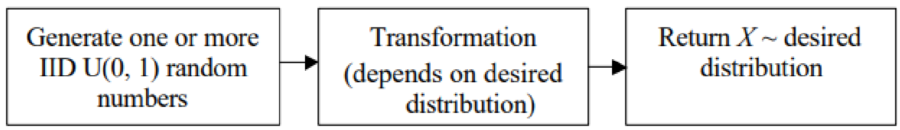

Figure 1: General form of Algorithms

The realized value from a probability distribution is called a random variate. Simulations use many different probability distributions as inputs. Thus, methods for generating random variates from distributions are required. Different distributions may require different algorithms due to the challenges of efficiently producing the random variables. Therefore, we need to know how to generate samples from probability distributions. In generating random variates the goal is to produce samples X_i from a distribution $F(x)$ given a source of random numbers, $U_i \sim U(0,1)$.

There are four basic strategies or methods for producing random variates:

1. Inverse transform or inverse cumulative distribution function (CDF) method.
2. Convolution.
3. Acceptance/Rejection.
4. Mixture and Truncated Distributions.

2 Algorithms

2.1 Inverse Transform Method

The inverse transform method is the preferred method for generating random variates provided that the inverse transform of the cumulative distribution function can be easily derived or computed numerically. The key advantage for the inverse transform method is that for every U_i use a corresponding X_i will be generated. That is, there is a one-to-one mapping between the pseudo-random number u_i and the generated variate x_i.

The inverse transform technique utilizes the inverse of the cumulative distribution function as illustrated in Figure 2, will illustrates simple cumulative distribution function. First, generate a number, u_i, between 0 and 1 (along the U axis), then find the corresponding x_i coordinate by using $F^{-}1(.)$. For various values of u_i the x_i will be properly 'distributed' along the x-axis. The beauty of this method is that there is a one to one mapping between u_i and x_i. In other words, for each u_i there is a unique x_i because of the monotone property of the CDF.

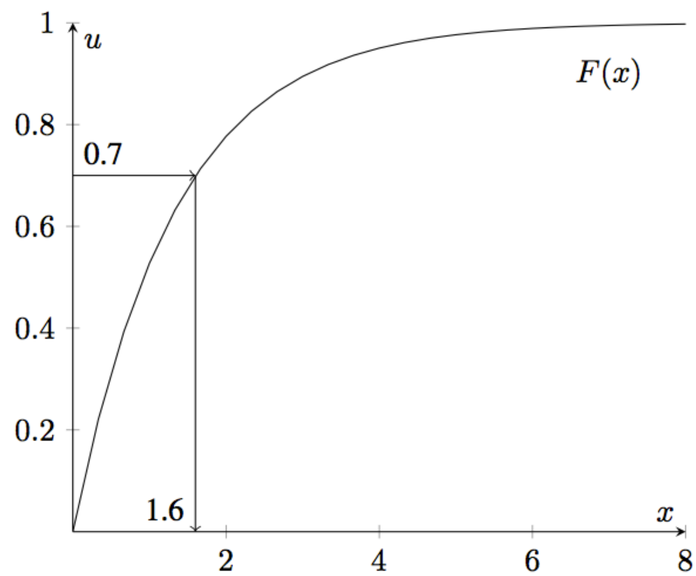

Figure 2: Inverse Transform Method

2.2 Convolution

Many random variables are related to each other through some functional relationship. One of the most common relationships is the convolution relationship. The distribution of the sum of two or more random variables is called the convolution. Let $Y_i \sim G(y)$ be independent and identically distributed random variables. Let $X = \sum_{i=1}^{n} Y_i$. Then the distribution of X is said to be the n-fold convolution of Y. Some common random variables that are related through the convolution operation are:

- A binomial random variable is the sum of Bernoulli random variables.
- A negative binomial random variable is the sum of geometric random variables.
- An Erlang random variable is the sum of exponential random variables.
- A Normal random variable is the sum of other normal random variables.
- A chi-squared random variable is the sum of squared normal random variables.

The basic convolution algorithm simply generates $Y_i \sim G(y)$ and then sums the generated random variables. Let's look at a couple of examples. By definition, a negative binomial distribution represents one of the following two random variables:

- The number of failures in sequence of Bernoulli trials before the r^{th} success, has range $\{0, 1, 2, ...\}$
- The number of trials in a sequence of Bernoulli trials until the r^{th} success, it has range $\{r, r + 1, r + 2, ...\}$

The number of failures before the r^{th} success, has range $\{0, 1, 2, ...\}$. This is the sum of geometric random variables with range $\{0, 1, 2, ...\}$ with the same success probability.

If $Y \sim NB(r, p)$ with range $\{0, 1, 2, ...\}$, then

$$Y = \frac{1}{n}\sum_{i=1}^{r} X_i$$

when $X_i \sim Geometric(p)$ with range $\{0, 1, 2, ...\}$, and X_i can be generated via inverse transform with:

$$X_i = \lfloor \frac{\ln(1-U_i)}{\ln(1-p)} \rfloor$$

f we have a negative binomial distribution that represents the number of trials until the r^{th} success, it has range $\{r, r+1, r+2, ...\}$, in this text we call this a shifted negative binomial distribution.

A random variable from a "shifted" negative binomial distribution is the sum of shifted geometric random variables with range $\{0, 1, 2, 3, ...\}$. with same success probability. In this text, we refer to this geometric distribution as the shifted geometric distribution.

If $T \sim NB(r, p)$ with range $\{r, r+1, r+2, ...\}$, then

$$T = \frac{1}{n} \sum_{i=1}^{r} X_i$$

when $X_i \sim ShiftedGeometric(p)$ with range $\{0, 1, 2, 3, ...\}$, and X_i can be generated via inverse transform with:

$$X_i = 1 + \lfloor \frac{\ln(1-U_i)}{\ln(1-p)} \rfloor$$

Notice that the relationship between these random variables as follows:

Let Y be the number of failures in a sequence of Bernoulli trials before the r^{th} success. Let T be the number of trials in a sequence of Bernoulli trials until the r^{th} success,

Then, clearly, $T = Y + r$. Notice that we can generate Y, via convolution, as previously explained and just add r to get T.

$$Y = \frac{1}{n} \sum_{i=1}^{r} X_i$$

Where $X_i \sim Geometric(p)$ with range $\{0, 1, 2, ...\}$, and X_i can be generated via inverse transform with:

$$X_i = \lfloor \frac{\ln(1-U_i)}{\ln(1-p)} \rfloor$$

2.3 Acceptance/Rejection

In the acceptance-rejection method, the probability density function (PDF) $f(x)$, from which it is desired to obtain a sample is replaced by a proxy PDF, $w(x)$, that can be sampled from more easily. The following illustrates how $w(x)$ is defined such that the selected samples from $w(x)$ can be used directly to represent random variates from $f(x)$. The PDF $w(x)$ is based on the development of a majorizing function for $f(x)$. A majorizing function, $g(x)$, for $f(x)$, is a function such that $g(x) \geq f(x)$ for $-\infty < x < +\infty$.

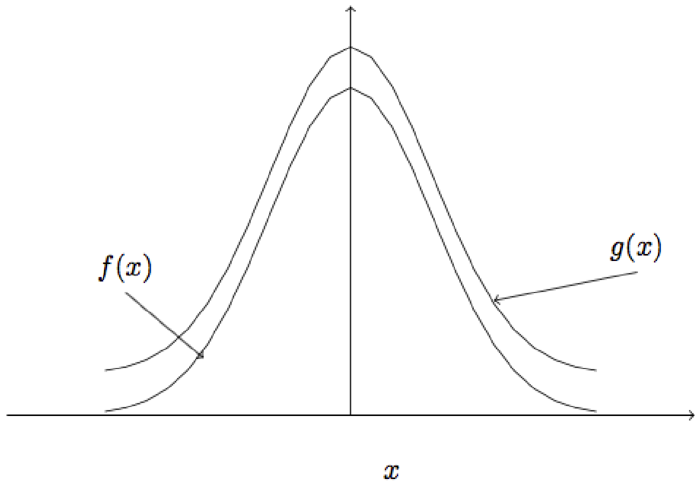

Figure 3: Concept of a Majorizing Function

Figure 3 illustrates the concept of a majorizing function for $f(x)$, which simply means a function that is bigger than $f(x)$ everywhere.

In addition, to being a majorizing function for $f(x)$, $g(x)$ must have finite area. In other words,

$$c = \int_{-\infty}^{+\infty} g(x)\, dx$$

If $w(x)$ is defined as $w(x) = g(x)/c$ then $w(x)$ will be a probability density function. The acceptance-rejection method starts by obtaining a random variate W from $w(x)$. Recall that $w(x)$ should be chosen with the stipulation that it can be easily sampled, e.g. via the inverse transform method. Let $U \sim U(0, 1)$. The steps of the procedure are as provided in the following algorithm. The sampling of U and W continue until $U \times g(W) \leq f(W)$ and W is returned. If $U \times g(W) > f(W)$, then the loop repeats.

1. REPEAT
2. Generate $W \sim w(x)$
3. Generate $U \sim U(0, 1)$
4. UNTIL $(U \times g(W) \leq f(W))$
5. RETURN W

The validity of the procedure is based on deriving the cumulative distribution function of W given that the $W = w$ was accepted, $P\{W \leq x \mid W = w \text{ is accepted }\}$.

The efficiency of the acceptance-rejection method is enhanced as the probability of rejection is reduced. This probability depends directly on the choice of the majorizing function $g(x)$. The acceptance-rejection method has a nice intuitive geometric connotation.

2.4 Mixture Distributions, Truncated Distributions, and Shifted Random Variables

This section describes three random variate generation methods that build on the previously discussed methods. These methods allow for more flexibility in modeling the underlying randomness.

Definition (Mixture Distribution)

The distribution of a random variable X is a mixture distribution if the CDF of X has the form:

$$F_X(x) = \sum_{i=1}^{k} \omega_i F_{X_i}(x)$$

where $0 < \omega_i < 1$, $\sum_{i=1}^{k} \omega_i = 1$, $k \geq 1$ and $F_{X_i}(x)$ is the CDF of a continuous or discrete random variable X_i, $i = 1, ..., k$.

Notice that the ω_i can be interpreted as a discrete probability distribution as follows. Let I be a random variable with range $I \in \{1, ..., k\}$ where $P[I = i] = \omega_i$ is the probability that the i^{th} distribution $F_{X_i}(x)$ is selected. Then, the procedure for generating from $F_X(x)$ is to randomly generate I from $g(i) = P[I = i] = \omega_i$ and then generate X from $F_{X_I}(x)$. The following algorithm presents this procedure.

1. Generate $I \sim g(i)$
2. Generate $X \sim F_{X_I}(x)$
3. RETURN X

Because mixture distributions combine the characteristics of two or more distributions, they provide for more flexibility in modeling. For example, many of the standard distributions that are presented in introductory probability courses, such as the normal, Weibull, lognormal, etc., have a single mode. Mixture distributions are often utilized for the modeling of data sets that have more than one mode.

As an example of a mixture distribution, we will discuss the hyper-exponential distribution. The hyper-exponential is useful in modeling situations that have a high degree of variability. The coefficient of variation is defined as the ratio of the standard deviation to the expected value for a random variable X. The coefficient of variation is defined as $c_v = \sigma/\mu$, where $\sigma = \sqrt{Var[X]}$ and $\mu = E[X]$. For the hyper-exponential distribution $c_v > 1$. The hyper-exponential distribution is commonly used to model service times that have different (and mutually exclusive) phases. An example of this situation is paying with a credit card or cash at a checkout register. The following example illustrates how to generate from a hyper-exponential distribution.

References

[1] https://rossetti.github.io/RossettiArenaBook/app-rnrv-rvs.html#AppRNRV:subsec:MTSRV

[2] https://www.cyut.edu.tw/~hchorng/downdata/1st/SS8_Random%20Variate.pdf