

IE630: Simulation Modelling & Analysis

Random Number Generation

Random Variate Generation

Saurabh Jain

Assistant Professor

Department of Industrial Engineering and Operations Research

IIT Bombay



IIT

BOMBAY

Department of Industrial Engineering and Operations Research



Quick Recap



IIT
BOMBAY

Department of Industrial Engineering and Operations Research



Part I

Random Number Generation



IIT
BOMBAY

Department of Industrial Engineering and Operations Research



Introduction

- Data, parameters are not deterministic
- Example
 - Inter-arrival time, service time
 - Number of students attending the class
 - Balking behaviour
 - Machine break down frequency, machine fix frequency
 - and many more
- Challenges
 - Replication of the Process
- From this statistical distribution (PDF or PMF), how do we get different parameters (variates) each time?



Basic Concept

- We have the parameters represented as **statistical distributions**

—

- UNIF(1,4)
- NORMAL(25, 1)
- TRIANGLE (4,5,6)
- EXPO(0.2)

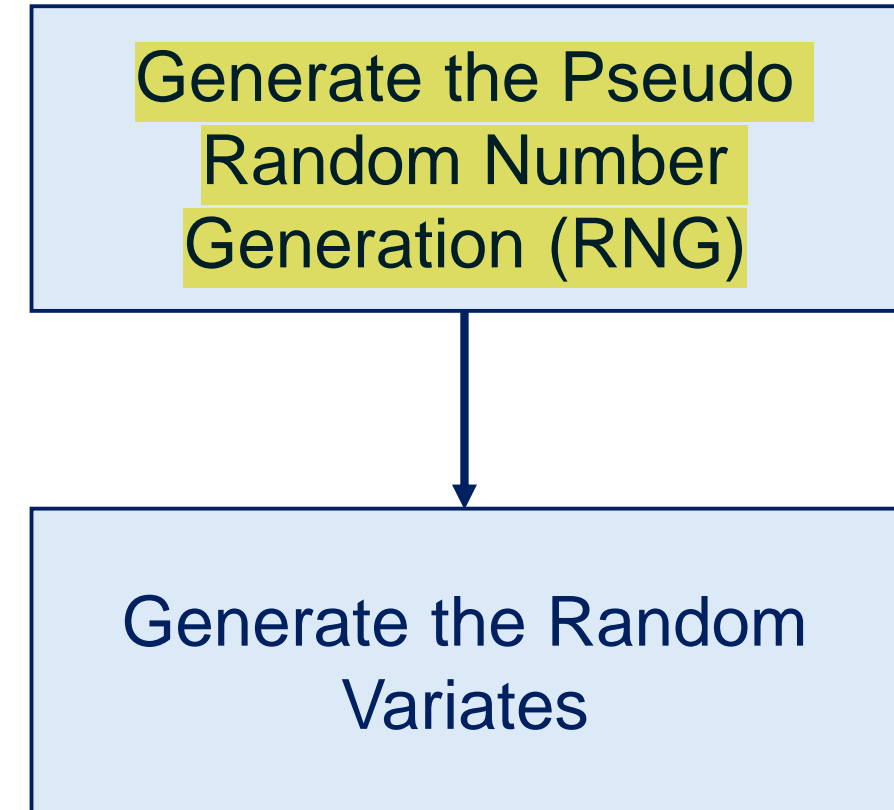
Big Question – How do we draw the appropriate value from these input distributions?

- Random Numbers
- Random Variates



Steps - Generate the Random Variates

- Generate the **INDEPENDENT & IDENTICALLY DISTRIBUTED (I.I.D)** random numbers following **UNIF (0,1)**
- Transform the **I.I.D UNIF (0,1)** random number into _____ from the defined input distribution



Generation of the Random Numbers

- **Physics Processes** –
 - Noise from the analog circuits, Thermal/atmospheric noise
 - Hardware Random Number Generator
- **Computational Methods** –
 - Pseudo-random number generator
 - Recursive methods to generate numbers
 - **Output is predictable, given the seeds**
- Hence, we need to find out -
 - Truly random number generator
 - Computational schemes to generate RN
 - Statistically random



Pseudo Random Number Generator: Properties

- Possess the ideal statistical properties
 - Uniformity
 - Independence
- Should be reproducible the given stream of random numbers exactly or vice versa
- Reasonably fast
- Have sufficiently long cycle
- Produce the separate streams of random numbers



Linear Congruential Method

- To produce a sequence of integers Z_1, Z_2, Z_3, \dots Between 0 and $m-1$ by following a recursive relationship:

$$Z_i = (aZ_{i-1} + c) \bmod m, i = 1, 2, 3, \dots$$

- Assumption: $m > 0$ - Modulus
 $a < m$ - Multiplier
 $c < m$ - Increment
 Z_i - Seed
- Values for a, c, m , and X_1 affects the _____ &



Linear Congruential Method

- To produce a sequence of integers Z_1, Z_2, Z_3, \dots Between 0 and $m-1$ by following a recursive relationship:

$$Z_i = (aZ_{i-1} + c) \bmod m, i = 1, 2, 3, \dots$$

- Define the U_n as shown below –

$$U_i = Z_i/m \text{ for } i = 1, 2, 3, \dots$$

$$\bullet \left(\quad \right) < U_i < \left(\quad \right)$$



Linear Congruential Method

- LCM parameters used by Arena until version 3.0

$$m = 2^{31} - 1 = 2147483647$$

$$a = 7^5 = 16807$$

$$c = 0$$

$$Z_0 = 14561, \dots, 32535$$

- Excel - $Z_i = (1140671485 \cdot Z_{i-1} + 12820163) \bmod 2^{24}$

- Cycle, Full Cycle

- Period (p): the length of a cycle ($p \leq m$)
- Generator is called “full period” if $p == m$



General Congruence

- General Congruences

- $Z_i = g(Z_{i-1}, Z_{i-2}, \dots) \bmod m$

- Composite Generators

- Combination to produce a separate random number generators
 - Let Z_{2i} and Z_{1i} be two separate generators having two different LCGs
 - Then,

- $Z_i = (Z_{2i} - Z_{1i}) \bmod m$

- Then, $U_i = Z_i/m$, if $Z_i > 0$



Composite Generators

- L'Ecuyer (1988)

- $Z_{1,i} = (40,014) Z_{1,i-1} \bmod (2^{31} - 85)$
- $Z_{2,i} = (40,692) Z_{2,i-1} \bmod (2^{31} - 249)$
- $Y_i = (Z_{1,i} - Z_{2,i}) \bmod (2^{31} - 86)$
- $U_i = Y_i / (2^{31} - 85)$

Combined period
 $\sim 2 \times 10^{18}$

- L'Ecuyer (1999)

- $Z_{1,i} = [1,403,580 Z_{1,i-2} - 810,728 Z_{1,i-3}] \bmod (2^{32} - 209)$
- $Z_{2,i} = [527,612 Z_{2,i-1} - 1,370,589 Z_{2,i-3}] \bmod (2^{32} - 22853)$
- $Y_i = (Z_{1,i} - Z_{2,i}) \bmod (2^{32} - 209)$
- $U_i = Y_i / (2^{32} - 209)$

Combined period
 $\sim 3.1 \times 10^{57}$



Random Number Generators: Test

- There are several ways to test RNG:
 - Testing the goodness of fit with IID $U(0,1)$
 - “Die Hard Tests”
 - Basic Statistician Tests
 - Theoretical Tests
- **Die-Hard Tests**
 - Birthday spacing
 - Overlapping permutations
 - Monkey tests
 - Count the 1s
 - Parking lot test
 - The craps test
 - Minimum distance test
 - Random spheres test
 - The squeeze test
 - Overlapping sums test
 - Runs test



Random Numbers: Test

- Two properties – Uniformity & Independence
- Test for **uniformity**:
 - $H_0: R_i \sim U[0,1]$
 - $H_1: R_i \not\sim U[0,1]$
 - Failure to reject the H_0 means that evidence of non-uniformity has been detected
- Test for **Independence**:
 - $H_0: R_i \sim \text{Independent}$
 - $H_1: R_i \not\sim \text{Independent}$
 - Failure to reject the H_0 means that evidence of dependence has been detected



Chi-Square Test

- Test Statistic:

$$\chi_0^2 = \sum_{i=1}^k \frac{(O_i - E_i)^2}{E_i}$$

Expected Frequency

$$E_i = n \times p_i$$

where p_i is the theoretical prob. of the i -th interval.

O_i : Observed frequency in the i -th class

- Test Conditions:

$$\chi_0^2 \leq \chi_{k-1, 1-\alpha}^2$$

Unable to reject Null Hypothesis

$$\chi_0^2 > \chi_{k-1, 1-\alpha}^2$$

Reject Null Hypothesis



Kolmogorov-Smirnov Test (K-S Test)

- Test fitted cdf with empirical cdf
 - Step 1: Rank the data smallest to largest
 - Step 2: Compute D^+ and D^-
 - Step 3: $D = \text{Max}\{D^+, D^-\}$
 - Step 4: Test Statistics

Case	Adjusted Test Statistics	$1-\alpha$				
		0.850	0.900	0.950	0.975	0.990
All parameters known	$\left(\sqrt{n} + 0.12 + \frac{0.11}{\sqrt{n}}\right)D$	1.138	1.224	1.358	1.480	1.628
Normal ($\hat{\mu}, \hat{\sigma}^2$)	$\left(\sqrt{n} - 0.01 + \frac{0.85}{\sqrt{n}}\right)D$	0.775	0.819	0.895	0.955	1.035
Expo($\hat{\lambda}$)	$\left(\sqrt{n} + 0.26 + \frac{0.5}{\sqrt{n}}\right)\left(D - \frac{0.2}{n}\right)$	0.926	0.990	1.094	1.190	1.308



Part II

Random Variate Generation



IIT
BOMBAY

Department of Industrial Engineering and Operations Research



Random Variates Generation

- Generate IID $U(0,1)$ random number
- Transformation of Random Number to Variates using Input Distribution
- Methods –
 - Inverse-Transform
 - Acceptance-Rejection
 - Convolution
 - Composition
 - Special



Inverse Transform Method

- **Intuition:**

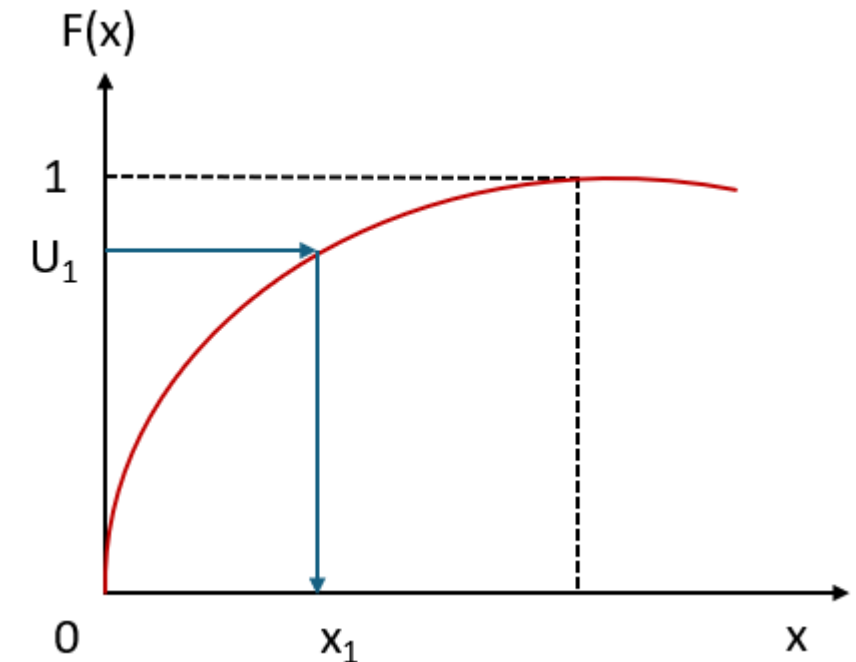
- Uses the inverse of the cdf $F(x)$
- Mostly useful when F^{-1} can be computed

- Steps to find generator

- Compute CDF of the desired random variable X
- Set $F(X) = U$, where $U \sim \text{IID Uniform}(0,1)$
- Solve $F(X) = U$ for X in terms of U

- Steps to use generator

- Generate U_i
- Return $X_i = F^{-1}(U_i)$



Inverse Transform Method

- Can be used for sampling from **continuous** as well as **discrete** distribution
 - Uniform
 - Exponential
 - Triangular
 - Weibull
 - Log-logistic
 - Empirical Continuous
 - Discrete Uniform
 - Geometric
 - Bernoulli
 - Empirical Discrete
- **Advantages:**
 - Valid for mixed discrete-continuous distributions
 - Facilitates variance reduction techniques
 - Ease of use in truncated distributions
- **Disadvantages:**
 - Need to evaluate $F^{-1}(U)$
 - May not be the fastest way!



Inverse Transform Method – Some Distributions



Inverse Transform Method – Empirical Continuous Distribution

- Suppose the data collected for 100 broken-widget repair times are:

Interval (Hours)	Frequency	Relative Frequency	Cumulative Frequency	Slope
[0.25,0.5]	31			
[0.5, 1.0]	10			
[1.0, 1.5]	25			
[1.5, 2.0]	34			



Inverse Transform Method – Empirical Discrete Distribution

- Suppose the number of shipments, x , on the loading dock of a company is either 0, 1, or 2
- Data - Probability distribution:

x	$P(x)$	$F(x)$
0	0.5	
1	0.3	
2	0.2	



Inverse Transform Method

- The most common random number generator in use today is called a linear congruential generator. The linear congruential generator that we use is as follows:

$$Z_i = (2Z_{i-1} + 3) \bmod 5$$

Using this generator, manually generate 3 consecutive random variates (i.e. $i = 1, 2, \& 3$) following the Uniform distribution, Unif (4, 6). Assume your seed value is 3. Show all the steps of your answers



Inverse Transform Method

- For the same random numbers, derive three random variates following a discrete distribution: 10% of combo 1, 80% of combo 2, 10% of combo 3



Convolution method

- **Convolution**

- Sum of IID random variables
- Can be readily generated than direct generation of main Random variate

$$Y = X_1 + X_2 + X_3 + X_4 + \dots + X_m$$

where X_i are independent R.V's from same distribution

Procedure:

- Step 1: Generate X_1, X_2, \dots, X_m IID each with distribution function
- Step 2: Return

$$Y = X_1 + X_2 + X_3 + X_4 + \dots + X_m$$



Convolution method

- $Y \sim \text{Erlang}(k, \theta)$
- $Y \sim \text{Binomial}(n, p)$



Composition method

- **Composition**

- Lets say we have a distribution function F for generating the random variate
- F can be expressed convex combination of other distribution functions F_1, F_2, F_3, \dots
- The goal is to be able to sample more easily from F_j 's compared to original F

$$F(x) = \sum_{j=1}^{\infty} p_j F_j(x)$$

where

$$p_j \geq 0$$

$$\sum_{j=1}^{\infty} p_j = 1$$



Composition method

- Continuous Random Variate
- Discrete Random Variate



Acceptance-Rejection Technique

- Useful particularly when inverse CDF does not exist in closed form
 - Thinning
- Illustration: Generate random variates, $X \sim U(\frac{1}{4}, 1)$

Procedure:

- Step 1: Generate $R \sim U(0, 1)$
- Step 2: If $R \geq \frac{1}{4}$, accept $X = R$
- Step 3: If $R < \frac{1}{4}$, reject R , return to step 1

- Efficiency heavily depends on the ability to minimize the number of rejections



Acceptance-Rejection Technique: Poisson Distribution

- Probability mass function of a Poisson Distribution

$$P(X = n) = \frac{\alpha^n}{n!} e^{-\alpha}$$

- Apply the A-R technique to generate R.V. for Pois(2)

