



Lecture 29

Produced by Dr. Worldwide

Welcome to the 305

Stochastic Simulation



- Simulation is used to “simulate” the operations of various kinds of real-world systems or processes with the aid of a computer
- We make **assumptions** about how a system works
- These assumptions constitute a **model**
- Models are used to gain some understanding of how a system behaves
- In simple models, we can use analytical (mathematical) methods to obtain exact information on questions of interest
- In complex models, use simulation to evaluate the model numerically



Monte Carlo Process



- Large proportion of applications of simulations are for probabilistic models
- **Monte Carlo** is a technique for selecting numbers randomly from a probability distribution
- Monte Carlo is not a simulation model, but a **mathematical process** used within a simulation
- Monte Carlo simulation can also be thought of as **multiple probability simulation**
- Read **Link 1** on course website for more description about Monte Carlo



Random Number Generator

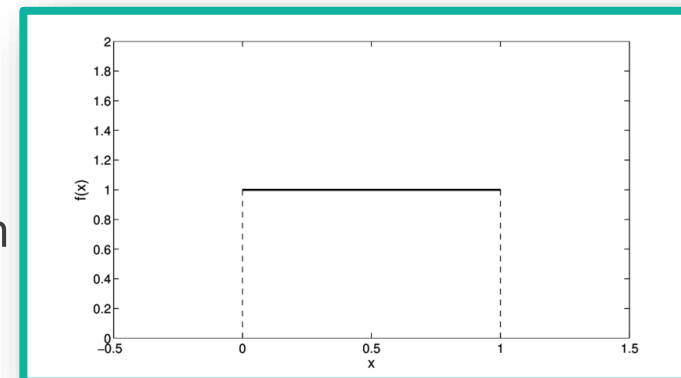


- To run a simulation, we may need “observations” from a random process
- If we “know” the distribution of these observations, we might be able to generate them artificially
- Historically, random numbers have been generated physically
 - Throwing dice
 - Dealing cards
 - Drawing balls from urns
- Not until mid 1950’s where electronic random number generators were used
- Two kinds of random number generators: physical and numerical
- Computer are equipped with numerical random number generators

Random Number Generator



- Q: Other than my teaching, is anything truly random?
- Numerical random number generators are **not** really random
- The most basic type of random numbers that we can artificially generate are from the **Uniform[0,1]** distribution where numbers between 0 and 1 are equally likely
- RAND() function in Excel samples from this distribution
- We can generate random numbers from any other distribution from the RAND() function
- Read **Link 2** on course website for more description about RAND()



Ex: Spinning a Wheel



- Suppose we want to simulate the result of spinning a wheel of fortune



- No physical version of the wheel, but can generate random uniform numbers
- Q: How can we use RAND() to “spin” the wheel?



Ex: Spinning a Wheel



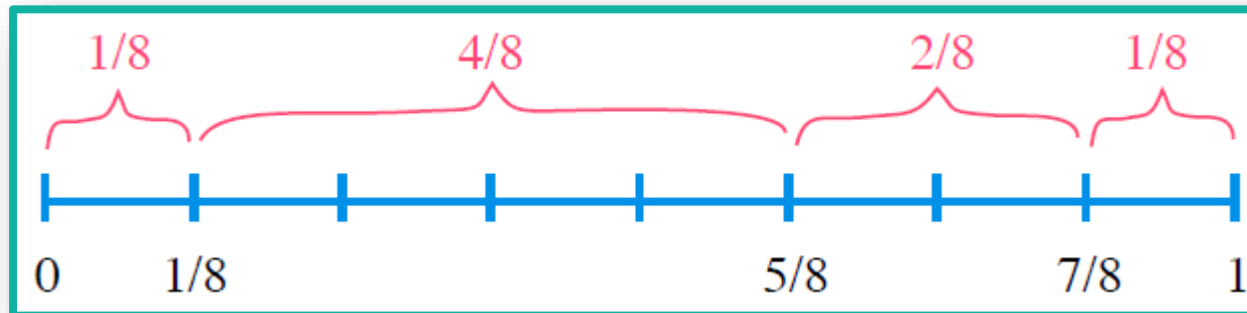
- Compute probabilities of each outcome
- Let X be the result from spinning the wheel once
- Possible outcomes are $X \in \{0, 10, 20, 100\}$
- Each segment of the wheel has the same area
- Probabilities under assumption wheel is “fair”
 - $P(X = 0) = \frac{1}{8}$
 - $P(X = 10) = \frac{4}{8} = \frac{1}{2}$
 - $P(X = 20) = \frac{2}{8} = \frac{1}{4}$
 - $P(X = 100) = \frac{1}{8}$



Ex: Spinning a Wheel



- Represent probabilities within the interval $[0,1]$



- Use Excel RAND() to generate a U from the Uniform $[0,1]$ distribution and set

$$X = \begin{cases} 0, & \text{if } 0 \leq U \leq 1/8, \\ 10, & \text{if } 1/8 < U \leq 5/8, \\ 20, & \text{if } 5/8 < U \leq 7/8, \\ 100, & \text{if } 7/8 < U \leq 1. \end{cases}$$

Ex: Spinning a Wheel



- Download [SpinWheel.xlsx](#) from link [Sheet 1](#) on course website
- When we use the RAND() function repeatedly, the random numbers generated are **independent** of each other
- We can use Excel to simulate as many spins of the wheel as we want
- Random sample located in cells E6:E47
- We are simulating 42 spins of this wheel
- Compare theoretical mean and standard deviation to sample statistics

17	True mean	22.5
18	True variance	893.75
19	True standard deviation	29.89565186

12	Sample mean	26.9047619
13	Sample standard deviation	33.67622707

Ex: Spinning a Wheel



- To transform, the Uniform[0,1] numbers generated by the function RAND() we can create a **look-up** table
 - 1st column contains the values of the function $h(x) = P(X < x)$
 - 2nd column contains the values of x
- Table contain in array B6:C9

Cumulative $P(X < x)$	x
0.00	0
0.13	10
0.63	20
0.88	100

- Table saved into variable named **CDF_Table**

Ex: Spinning a Wheel



- The command **VLOOKUP**(cell, look-up table, 2) will compare the number in the cell to the ranges defined by the first column of the look-up table, and return the value in the second column
- Example of function usage

Cumulative $P(X < x)$	x
0.00	0
0.13	10
0.63	20
0.88	100

Uniform U	X
0.1185088	0
0.2509473	10
0.1383176	10
0.1524281	10
0.1422752	10

- Value $U = 0.1185$ is between 0 and 0.13 so VLOOKUP returns 0

General Simulation for Discrete



- Suppose we want to use simulation to generate samples from a random variable X that takes values

$$\{x_1, x_2, x_3, \dots, x_k\}$$

with probabilities $P(X = x_i) = p_i, i = 1, 2, 3, \dots, k$

- For convenience, we can assume $x_i \leq x_{i+1}$
- We want to generate a random number that has the same distribution as X
- Assume we have a computer that can generate random numbers uniformly distributed on the interval $[0, 1]$

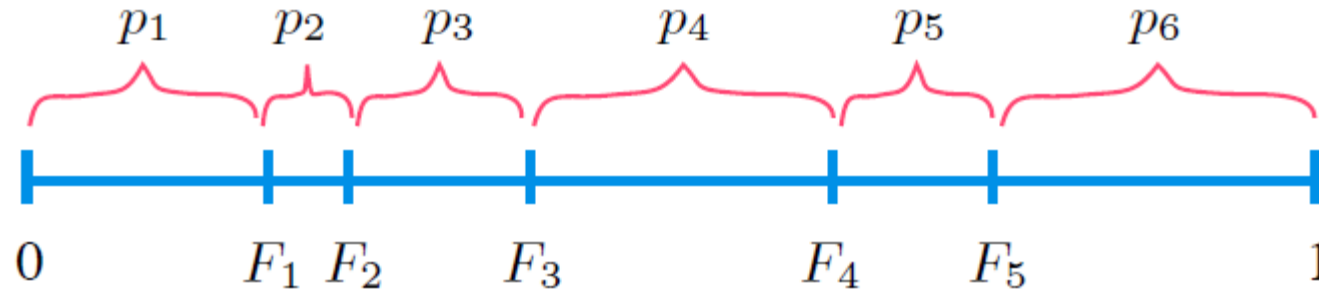


General Simulation for Discrete



- Step 1: Compute the cumulative probabilities

$$F_i = P(X \leq x_i) = p_1 + \cdots + p_i, \quad i = 1, 2, 3, \dots, k$$



- Note that $F_0 = 0$ and $F_k = 1$
- Step 2: Generate a random number $U \sim \text{Uniform}[0,1]$
- Step 3: Return $X = x_i$ if $F_{i-1} < U \leq F_i$

Ex: Cell Phone Production



- Suppose we want to model the production of a cellphone factory
- Factory ships cellphones in boxes containing 100 phones
- Each cellphone has an independent probability $p = 0.01$ of being defective
- We want to simulate a shipment containing 20 boxes
- We are interested in both the number of defective cellphones in a shipment, and the number of boxes containing at least one defective cellphone in a shipment
- If N_i denotes the number of defective cellphones in box i , then $N_i \sim \text{Binomial}(100, 0.01)$
- Related to flipping an unfair coin with probability of heads equal to 0.01

Ex: Cell Phone Production



- Instead of simulating the N_i 's, we try to simulate coin flips, and evaluate

$$N_i = \sum_{j=1}^{100} X_{ij}$$

- where $X_{ij} = \begin{cases} 1 & \text{if the } j\text{th cellphone in box } i \text{ is defective} \\ 0 & \text{otherwise} \end{cases}$
- Download



The End



Dale

