# STA 371G: Statistics and Modeling

## Review of Basic Probability and Statistics: Introduction to Monte Carlo Simulation

Mingyuan Zhou
McCombs School of Business
The University of Texas at Austin

http://mingyuanzhou.github.io/STA371G

# Why Simulation?

- Extremely useful to model uncertainty
- Can be incorporated into spreadsheet models
- Allow the input variables to be random
- Provide an entire distribution of results, including not only the most likely, but also the worst and the best
- Help analyze a system's sensitivity to changes in operation conditions
- Can be implemented in Excel with the RAND() function or some add-in package (such as @Risk)
- Extremely convenient to implement in R (also in Matlab, Python, ...)

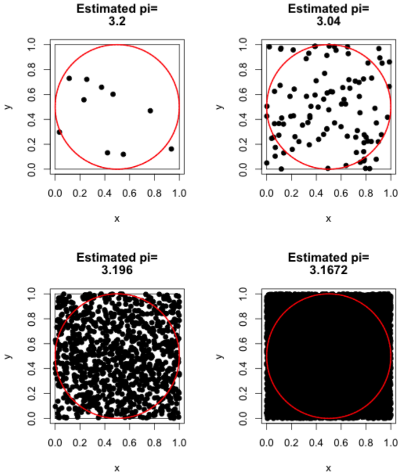
# Example: Simulation of Market Return

- Incorporate uncertainty into spreadsheet models

| | A | B | C | D |
|---|---|---|---|---|
| 1 | Economy | Probability | Market Return | (Market Return - Mean)^2 |
| 2 | Rapid Expanding | 12.00% | 23% | 0.005929 |
| 3 | Moderate Expanding | 40.00% | 18% | 0.000729 |
| 4 | No Growth | 25.00% | 15% | 9E-06 |
| 5 | Moderate Contraction | 15.00% | 9% | 0.003969 |
| 6 | Serious Contraction | 8.00% | 3% | 0.015129 |
| 7 | | | | |
| 8 | Expected Return | 0.153 | Simulated Return | 0.152489879 |
| 9 | Variance of Return | 0.002811 | Simulated variance | 0.002769478 |
| 10 | Standard Deviation of Return | 0.053018865 | Simulated sd | 0.052625831 |
| 11 | | | | |
| 12 | Random # | Simulated Return | Lookup Probability Table | |
| 13 | 0.39065967 | 18.00% | Cumulative probability | Market Return |
| 14 | 0.951676654 | 3.00% | 0 | 23% |
| 15 | 0.902555852 | 9.00% | 12% | 18% |
| 16 | 0.572895871 | 15.00% | 52% | 15% |
| 17 | 0.929102096 | 3.00% | 77% | 9% |
| 18 | 0.428496674 | 18.00% | 92% | 3% |
| 19 | 0.683789167 | 15.00% | | |
| 20 | 0.582831026 | 15.00% | | |
| 21 | 0.362137678 | 18.00% | | |

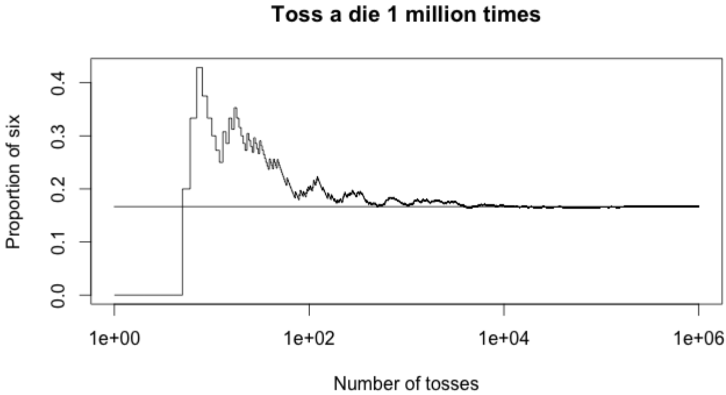
# Example: Estimate $\pi$ with Monte Carlo Simulation

- The area of a circle with the radius of $1/2$ is $\pi/4$
- The area of a unit square is 1
- A uniform random point in the unit square falls within the circle with probability $\pi/4$

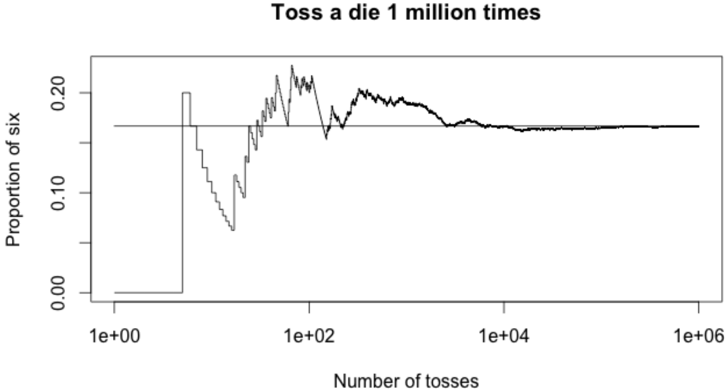


# Law of Large Numbers

- Toss a die one million times
- Record the proportion of die tosses that are 6 after each toss
- The proportion tends to get closer to the probability $1/6$ as the number of tosses increases



Repeat the same experiment: toss a die one million times



# Law of Large Numbers

- Probability: the probability of a random outcome is the proportion of the times that it would occur if the experiment is repeated an infinite number of times.
- Law of Large Numbers: As the number of experiments $n$ increases towards $\infty$, the proportion of a particular random outcome $\hat{p}_n$ converges to the true probability $p$ of that outcome.

# Uniform Random Variable

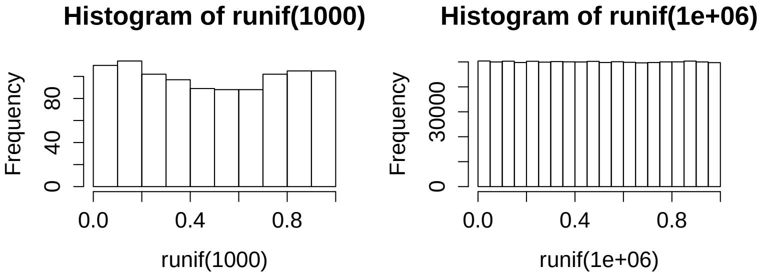
- A random number uniformly distributed between 0 and 1
- Can be used to generate any other random numbers
- In Excel: =RAND()
  - Flip a coin: =IF(RAND()>0.5,"Head","Tail")
  - Roll a die:
    - =RANDBETWEEN(1,6)
    - or use =RAND() and a lookup table
- In R: runif
  - Flip a coin: runif(1)>0.5 or sample(0:1,1)
  - Roll a die: =ceiling(runif(1)*6) or sample(1:6,1)

# Uniform Random Variable

Histogram of uniformly distributed random numbers

In Excel: copy RAND() to n cells , where $n$ is the number of experiments. What if $n = 10^9$?

In R: hist(runif(n))

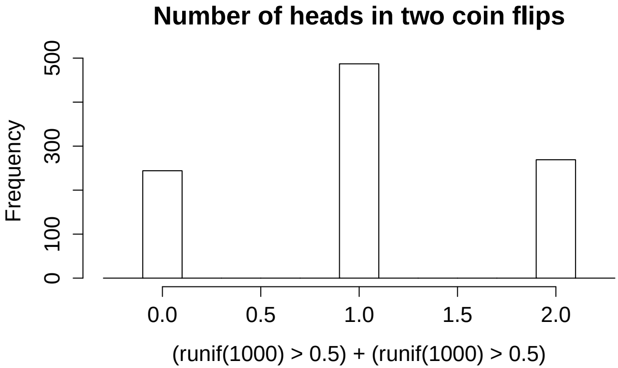


# Example: Flip Two Coins

In Excel: copy =(RAND()>0.5)+(RAND()>0.5) to *n* cells, use scatter plot to visualize the results

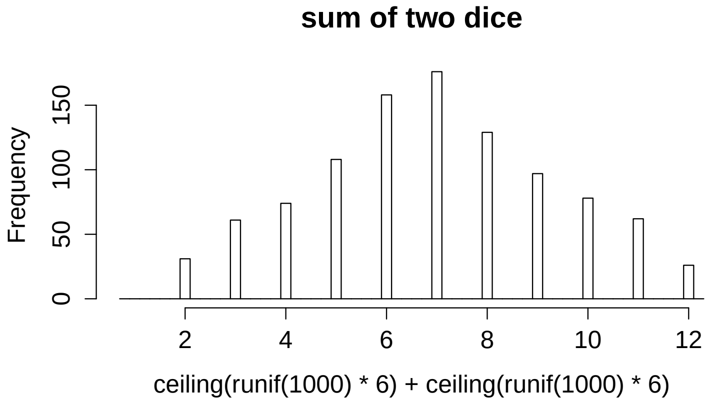
In R: x = (runif(n)>0.5) + (runif(n)>0.5)

use hist(x) and plot(x) to visualize the results



# Example: Toss Two Dice

In R: hist(ceiling(runif(n)*6) + ceiling(runif(n)*6) ) or
hist(sample(1:6,n,replace=TRUE) + sample(1:6,n,replace=TRUE))



# Normal Random Variable

- $X \sim \mathcal{N}(\mu, \sigma^2)$
- In Excel: $=$ NORMINV(RAND(), $\mu$, $\sigma$)
- In R: rnorm(n,$\mu$,$\sigma$) , where *n* is the number of normal random numbers that we ask R to generate
- Let's simulate 1000 standard normal random numbers and visualize the results: