# Probability Methods in Engineering

Dr. Safdar Nawaz Khan Marwat
DCSE, UET Peshawar

Lecture 26

# Random Experiments

- Generate outcomes of eight Bernoulli trials with success probability of 0.5

# Random Experiments (cont.)

```
X = rand(1,8);
% Generate 1 row of Bernoulli trials with p = 0.5
Y = X < 0.5;
Y =
1 0 0 1 1 1 0 0
```

- `X = rand(m,n)`
  - Returns m-by-n matrix
  - Contains pseudorandom values
  - Drawn from standard uniform distribution on interval (0,1)
  - If number produced by `rand` greater than $p = 0.5$, outcome is 1

# Random Experiments (cont.)

- Generate outcomes of 1000 repetitions of a random experiment that counts the number of successes in 16 Bernoulli trials with probability of success 0.5. Plot the relative frequencies of the outcomes in the 1000 experiments and compare to binomial probabilities with $n = 16$ and $p = 0.5$.

# Random Experiments (cont.)

```
X = rand(1000,16) < 0.5;
% Generate 1000 rows of 16 Bernoulli trials with p = 0.5
Y = sum(X,2);
% Add the results of each row to obtain the number of
% successes in each experiment. Y contains 1000 outcomes.
K = 0:16;
H = hist(Y,K);
S = sum(H,2);
Z = H./S;
% Find the relative frequencies of the outcomes in Y.
bar(K,Z)
% Produce a bar graph of the relative frequencies.
hold on
% Retains the graph for next command.
stem(K,binopdf(K,16,0.5))
% Plot the binomial probabilities along
% with the corresponding relative frequencies.
```

# Random Experiments (cont.)

- `B = sum(A,dim)`
  - Sums along dimension of $A$ specified by scalar $dim$
  - $dim$ input is integer value from 1 to $N$
    - where $N$ is number of dimensions in $A$
  - Set $dim$ to 1 to compute sum of each column, 2 to sum rows, …
- `X = 0 : n`
  - Gives a row matrix with $n + 1$ columns
  - Start value is 0, end value is $n$
  - Default step size is 1
    - Other step sizes using `0 : step : n`
- `H = hist(Y,K)`
  - Where $K$ is a vector, returns distribution of $Y$ among `length(K)` bins
  - Bin centers specified by $K$

# Random Experiments (cont.)

- Write code to simulate tossing a fair coin a) 10 b) 100 and c) 1000 times and illustrate how the law of averages for large numbers works.

**Source**: H. Pishro-Nik, Introduction to Probability, Statistics, and Random Processes, *Kappa Research*, LLC, 2014

# Random Experiments (cont.)

```
n = 10;
U = rand(n, 1);
toss = (U < 0.5);
total = zeros(n + 1,1);
avg = zeros(n,1);
for i = 2 : n + 1
      total(i) = total(i - 1) + toss(i - 1);
      avg(i - 1) = total(i)/(i - 1);
end
plot(avg)
```

**Source**: H. Pishro-Nik, Introduction to Probability, Statistics, and Random Processes, *Kappa Research*, LLC, 2014

- Give an algorithm for simulation of generating the value of a random variable $X$ such that
  - $P(X = 1) = 0.35$
  - $P(X = 2) = 0.15$
  - $P(X = 3) = 0.4$
  - $P(X = 4) = 0.1$

**Source**: H. Pishro-Nik, Introduction to Probability, Statistics, and Random Processes, *Kappa Research*, LLC, 2014

# Random Variables (cont.)

- Divide unit interval $[0, 1]$ into subintervals
  - $A_0 = [0, 0.35)$
  - $A_1 = [0.35, 0.5)$
  - $A_2 = [0.5, 0.9)$
  - $A_3 = [0.9, 1)$
- Subinterval $A_i$ has length $p_i$
  - Obtain uniform number $U$
  - If $U$ belongs to $A_i$, then $X = x_i$
    - $P(X = x_i) = P(U \in A_i)$
    - $= p_i$

**Source**: H. Pishro-Nik, Introduction to Probability, Statistics, and Random Processes, *Kappa Research*, LLC, 2014

# Random Variables (cont.)

```
P = [0.35, 0.5, 0.9, 1];
X = [1, 2, 3, 4];
counter = 1;
r = rand
while(r > P(counter))
      counter = counter + 1;
end
X(counter)
```

**Source**: H. Pishro-Nik, Introduction to Probability, Statistics, and Random Processes, *Kappa Research*, LLC, 2014

# Discrete Distribution

- Binomial distribution
  - `Y = binopdf(X,n,p)`
    - Computes binomial pdf
    - At each value in vector $X$
    - $n$ number of trial
    - Success probability $p$
  - `P = binocdf(X,n,p)`
    - Computes binomial cdf
  - `R = binornd(n,p)`
    - Generates random number from binomial distribution

**Source**: H. Pishro-Nik, Introduction to Probability, Statistics, and Random Processes, *Kappa Research*, LLC, 2014

# Continuous Distribution

- Exponential distribution
  - `Y = exppdf(X,mu)`
    - Computes exponential pdf
    - At each value in vector $X$
    - Mean value $(1/\lambda)$ given as $mu$
  - `P = expcdf(X,mu)`
    - Computes exponential cdf
  - `R = exprnd(mu)`
    - Generates random number from exponential distribution

**Source**: H. Pishro-Nik, Introduction to Probability, Statistics, and Random Processes, *Kappa Research*, LLC, 2014

# Tasks

- Write a MATLAB program to generate Geometric random variable values with $p = 0.2$.

# Tasks (cont.)

- Write MATLAB programs to generate 100 geometric random variable values with $p = 0.2$. Compare the relative frequencies with geometric probabilities.

# The End!

- Thank you!