**Logo

Description automatically generated**

**ICE M1012: Stochastic Theory of Communication Lab**

**Assignment**

*Submitted to*

**Dr. Md. Emdadul Haque**

**Professor, Dept. of Information and Communication Engineering,**

**University of Rajshahi, Rajshahi, Bangladesh.**

*Submitted by*

**Name: Shehjad Mobin**

**Roll: 1810677110**

**M. Engineering 2021-22**

**Department of Information and Communication Engineering, University of Rajshahi, Rajshahi, Bangladesh.**

Pdf of Binomial Distribution

**Code:**

n\_trials = 1000;

n\_flips = 21;

values = zeros(1, n\_flips);

% Simulating coin flips

for i = 1:n\_trials

numberOfHeads = sum(randi([0,1], 1, n\_flips)); % Vectorized operation

values(numberOfHeads + 1) = values(numberOfHeads + 1) + 1; % Adjusting index since MATLAB arrays are 1-indexed

end

valuesFloat = values / n\_trials; % Vectorized operation

% Plotting

figure;

bar(valuesFloat);

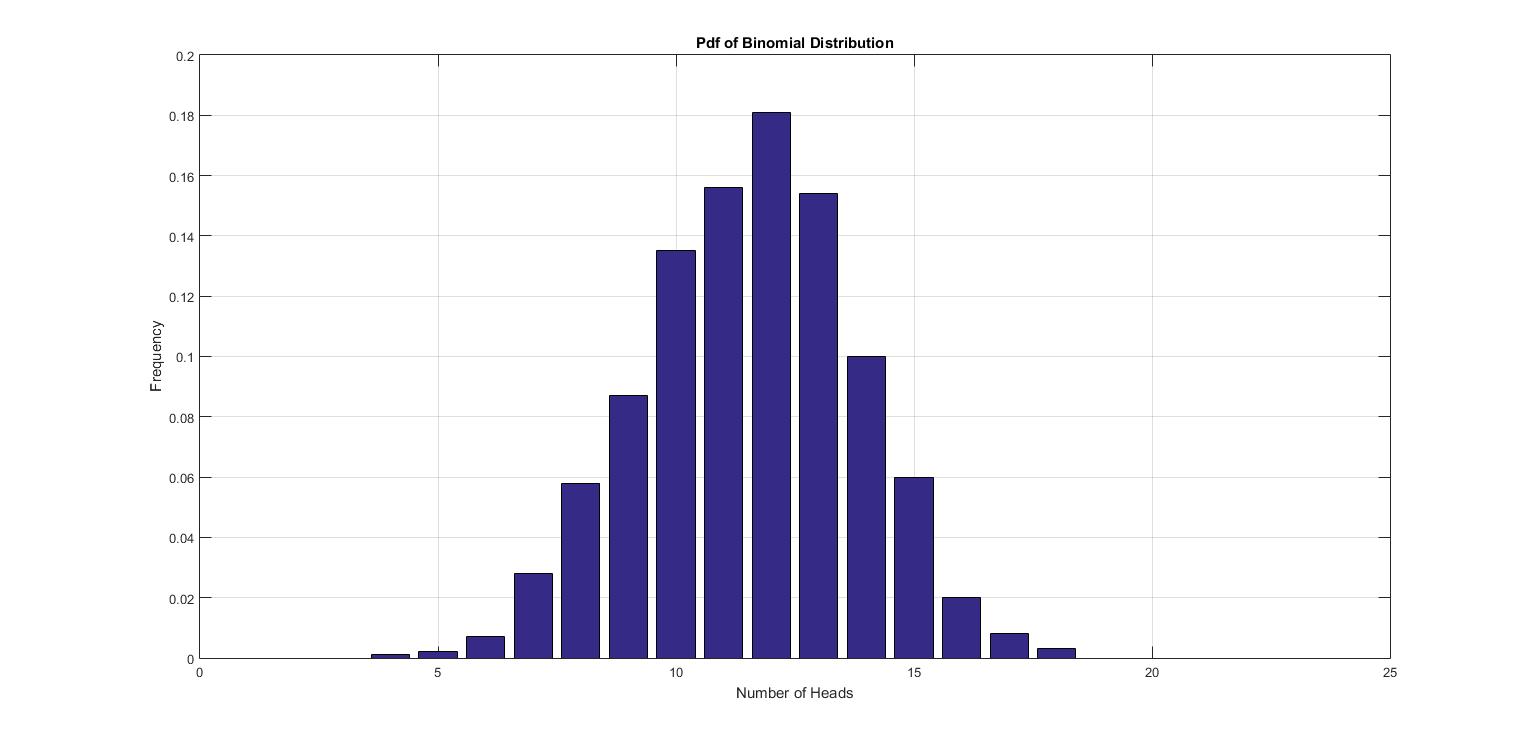
xlabel('Number of Heads');

ylabel('Frequency');

title('Pdf of Binomial Distribution');

grid on;

**Result:**



Pdf of Geometric Distribution

**Code:**

totalExperiment = 1000;

numberOfFlipsUntilFirstHead = zeros(1, 20);

for i = 1:totalExperiment

numberOfTrial = 0;

while randi([0,1]) == 0

numberOfTrial = numberOfTrial + 1;

end

numberOfFlipsUntilFirstHead(numberOfTrial + 1) = numberOfFlipsUntilFirstHead(numberOfTrial + 1) + 1;

end

numberOfFlipsUntilFirstHead = numberOfFlipsUntilFirstHead / totalExperiment;

bar(numberOfFlipsUntilFirstHead);

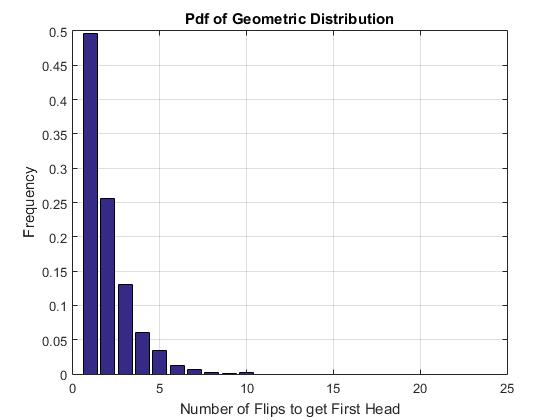
xlabel('Number of Flips to get First Head');

ylabel('Frequency');

title('Pdf of Geometric Distribution');

grid on;

**Result:**



Steady State Matrix Generation

**Code:**

% Example matrix

matrix = [1, 0, 0;

0.002, 0.998, 0;

0, 0.002, 0.998];

% Parameters

threshold = 1e-5;

maxIterations = 5000;

result = matrix;

for i = 1:maxIterations

previousResult = result;

result = result \* matrix;

if norm(result - previousResult, 1) < threshold

disp(i);

break;

end

end

roundedMatrix = round(result, 2);

disp('Steady State Matrix:');

disp(result);

disp('Rounded Steady State Matrix:');

disp(roundedMatrix);

**Result:**

