ECE3522 Practicum 1: Testing Microprocessors

Introduction

MATLAB can be an excellent tool to perform calculations and simulations for various engineering applications. In this practicum, I will use MATLAB to test a sample of 1,000 and a sample of 10,000 microprocessors which each processor will have a grade and probability independent of any other processor based upon the following requirements:

Grade	X=4	X=3	X=2	X=1
Probability	0.4	0.3	0.2	0.1

Procedure

A sample code was provided within the appendix of the practicum manual which determines a toss of an unfair coin that has a 70% probability to outcome a head (Y=0) or a 30% probability to outcome a tail (Y=1). The sample code can be found within the appendix of this report. I started by modifying the sample code to generate a discrete sequence of 1,000 integers between the grades 1 to 4. I denoted n as the number of microprocessors and used MATLAB's random integer function randi to set up the range 1 to 4. Using a for loop I was able to iterate across the microprocessors and used a series of if/else statements to assign x as the grade and y as the probability of each microprocessor.

Iterating across 1000 microprocessors with various grades

```
n=1e3; %Number of Microprocessors
for ii=1:n
    processors=randi(4); %The sample of processors to be tested
    x(ii)=processors;
    if x(ii)==1      %Grade=1 with probability of 0.1
        y(ii)=0.1;
    elseif x(ii)==2 %Grade=2 with probability of 0.2
        y(ii)=0.2;
    elseif x(ii)==3 %Grade=3 with probability of 0.3
        y(ii)=0.3;
    else            %Grade=4 with probability of 0.4
        y(ii)=0.4;
    end
end
```

Figure 1. MATLAB code iterating over 1000 Microprocessors

Using the data generated, I computed the average value as well as the standard deviation.

```
% Calculations by MATLAB
mean_y_sim=sum(y)/n
std_y_sim=sqrt(sum((y-mean_y_sim).^2)/n)
```

Figure 2. Calculating Mean and Std in MATLAB

To find the conditional probability P(X=3|X≥2), I derived how many processors were given each grade using MATLAB's sum functions and assigned them to generic variables for gate keeping. Knowing that the conditional probability P(X=3|X≥2) will mean the intersection of P(X=3) and P(X≥2) will be when X=3, I multiplied the respective probability to each total amount of processors given a grade 3 and given grades 2 or more. Simplifying the equation, P(X=3|X≥2)=P[X=3∩X>=2]/P[X>=2]=P(X=3)/P(X≥2), I was able to calculate the conditional probability.

```
%The conditional probability P[X=3|X>=2]=P[X=3∩X>=2]/P[X>=2]
%The intersection of X>=2 and X=3, will be X=3
%Processors out of 1000
g3=sum(x==3); %how many processors are grade 3
g2=sum(x==2); %how many processors are grade 2
g4=sum(x==4); %how many processors are grade 4
g1=sum(x==1); %how many processors are grade 1


d=g3+g2+g4; %how many processors are grade 2 or more
P_3=g3*0.3; %Probability P(X=3)
P_2orMore=(g2*0.2)+(g3*0.3)+(g4*0.4); %Probability P(X>=2)
%Conditional Probability calculation
P_32=P_3/P_2orMore;
```

Figure 3. Conditional Probability equation

Finally, I outputted the data as a histogram to represent the grade versus relative frequency of each microprocessor by using MATLAB's hist and bar functions, where relative frequency was calculated as a percent of the probability for each microprocessor that was given a respective grade.

```
%Plotting Histograph and Analysis
figure(1)
hist_x = hist(x, range_x);
bar(range_x,((hist_x*10)/n));
title('Testing 1000 Microprocessors')
xlabel('Grade')
ylabel('Relative Frequency')


fprintf('Mean of Y by MATLAB: %f\n',mean_y_sim);
fprintf('Standard Dev of Y by MATLAB: %f\n',std_y_sim);
fprintf('There are %d Grade 3 and %d Grade 2 or higher Microprocessors\n',g3,d);
fprintf('Therefore, P[x=3|x>=2]=P[X=3∩X>=2]/P[X>=2], which equals %d\n',P_32);
```

Figure 4. Graphing and Outputting Results

Using the data, I derived to find conditional probability I was able to hand calculate values for mean, standard deviation, and the conditional probability P(X=3|X≥2), this can be seen in the results section. To compare my results, I generated a simple script in MATLAB to determine the percent difference between my calculated results and simulated results.

```
%for 1000 Microprocesors
MeanSim=mean_y_sim;
```

```
MeanCalc=0.2498;
stdSim=std_y_sim;
stdCalc=7.8915;
probSim=P_32;
probCalc=0.3534;


%Percent Differences
MeanDiff=(abs(MeanSim-MeanCalc)/((MeanSim+MeanCalc)/2))*100
stdDiff=(abs(stdSim-stdCalc)/((stdSim+stdCalc)/2))*100
probDiff=(abs(probSim-probCalc)/((probSim+probCalc)/2))*100
```

Figure 5. Percent Difference Calculations

Finally, the script was copied with different variable names and ran again with only changing the number of microprocessors from 1,000 to 10,000 to determine if calculated values and simulated values will be closer, given a larger sample space of 10,000 microprocessors. This can be seen within the appendix.

Results

Testing 1,000 Microprocessors

MATLAB Simulation results and Graph

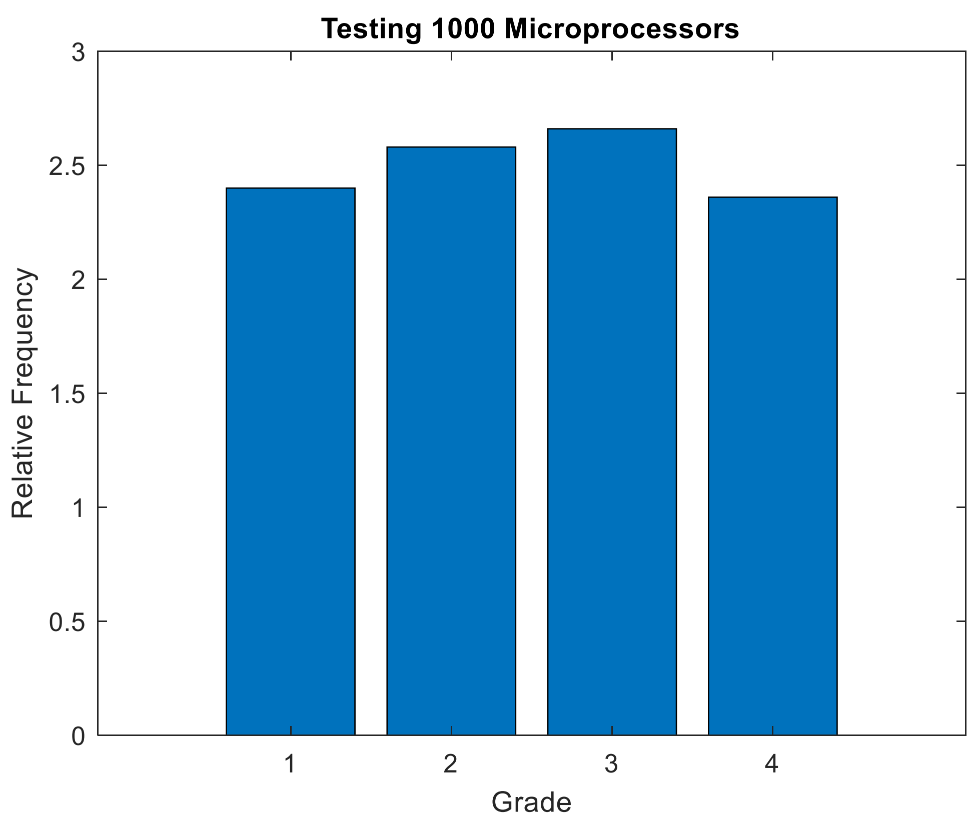

Figure 6. Histogram for 1000 Microprocessors

```
Mean of Y by MATLAB: 0.249800
Standard Dev of Y by MATLAB: 0.109636
There are 266 Grade 3 and 760 Grade 2 or higher Microprocessors
Therefore, P[x=3|x>=2]=P[X=3∩X>=2]/P[X>=2], which equals 3.534101e-01
```

Hand calculations dependent on MATLAB's random 1,000 sample:

Mean Calculation:

Practicum1 Hand calculations

Sunday, February 28, 2021 2:28 PM

To find the average, the mean equation would be:

$$\text{mean} = \frac{\text{Sum of all grades}}{\text{number of microprocessors Tested}}$$

From MATLAB I determined the total processors grades given by the generated sample as:

Grade 1: 240	Probability (y)
Grade 2: 258	Grade 1: 240 · .1 = 24
Grade 3: 266	Grade 2: 258 · .2 = 51.6
Grade 4: 236	Grade 3: 266 · .3 = 79.8
	Grade 4: 236 · .4 = 94.4
	y = 249.8

Therefore the mean is

$$\text{mean}(y) = \frac{249.8}{1000} = 0.2498$$

Standard Deviation Calculation:

Based on these calculations, the standard deviation can be calculated as follows:

$$\sigma = \sqrt{\frac{\Sigma (Y-\mu)^2}{n}}$$

where, σ = standard deviation
Σ = sum of
μ = popular mean
n = number of microprocessors

$Y = \Sigma y = 249.8$ so,

$$\sigma = \sqrt{\frac{(249.8-0.2498)^2}{1000}} = \sqrt{\frac{(249.5502)^2}{1000}} = \sqrt{62.2753} \approx 7.8915$$

Conditional Probability Calculation:

The conditional Probability of $P[X=3 \mid X \geq 2]$

$$P[X=3 \mid X \geq 2] = \frac{P[X=3 \cap X \geq 2]}{P[X \geq 2]}$$

Based on the data generated,

$P[X=3] = 266 \cdot 0.3 = 79.8$

$P[X \geq 2] = (258 \cdot 0.2) + (266 \cdot 0.3) + (236 \cdot 0.4) = 225.8$

The intersection of $P[X=3]$ and $P[X \geq 2]$ is $P[X=3]$

$\therefore P[X=3 \mid X \geq 2] = \frac{79.8}{225.8} = .3534$

Testing 10,000 Microprocessors

MATLAB Simulation results and Graph

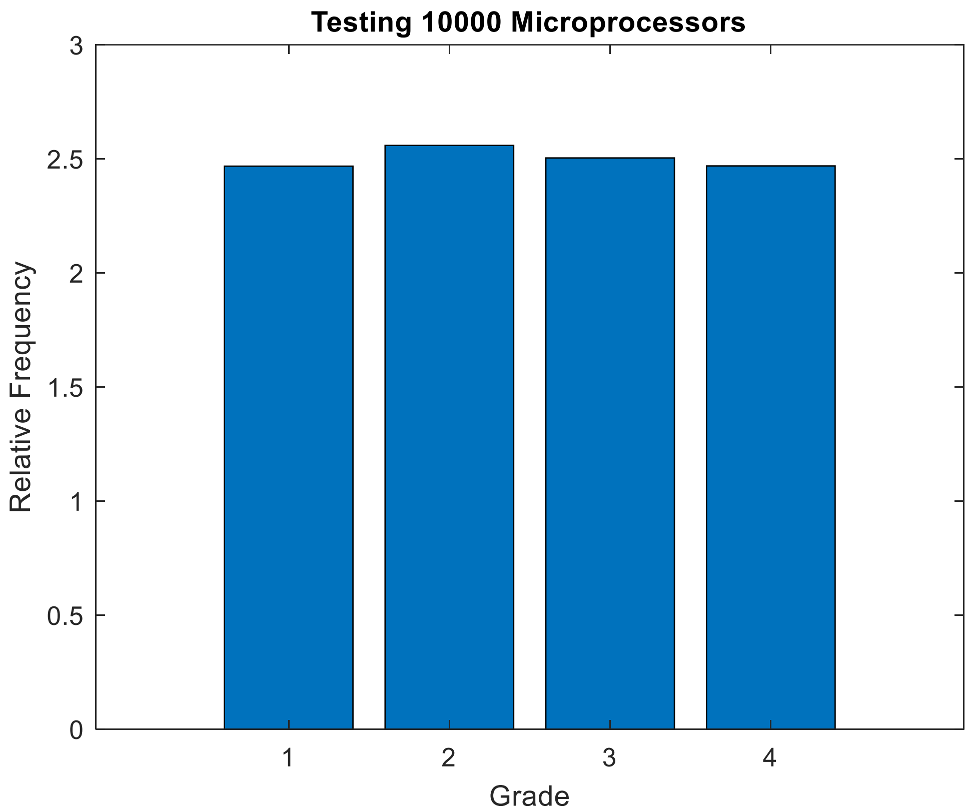

Figure 7. Histogram for 10,000 Microprocessors

```
Mean of Y by MATLAB: 0.249740
Standard Dev of Y by MATLAB: 0.111238
There are 266 Grade 3 and 760 Grade 2 or higher Microprocessors
Therefore, P[x=3|x>=2]=P[X=3∩X>=2]/P[X>=2], which equals 3.534101e-01
```

Hand calculations dependent on MATLAB's random 10,000 sample:

Mean Calculation:

To test 10,000 Micro processors, the same calculations can be done but using a larger sample space:

From MATLAB's randomly generated sample space

	Probability(y)
Grade 1: 240	Grade 1: 240 $\cdot 0.1 = 24$
Grade 2: 258	Grade 2: 258 $\cdot 0.2 = 51.6$
Grade 3: 266	Grade 3: 266 $\cdot 0.3 = 79.8$
Grade 4: 236	Grade 4: 236 $\cdot 0.4 = + 94.4$
	$Y = 249.8$

$$\therefore \text{Mean}(y) = \mu = \frac{249.8}{10{,}000} = 0.0250$$

Standard Deviation Calculation:

For Standard Deviation:

$$\sigma = \sqrt{\frac{(Y-\mu)^2}{n}} = \sqrt{\frac{(249.8-0.0250)^2}{10{,}000}} = \sqrt{\frac{(249.775)^2}{10{,}000}} = \sqrt{6.2388}$$

$$\sigma \approx 2.4977$$

Conditional Probability Calculation:

For Conditional Probability $P[x=3 \mid x \geq 2]$

$$P[X=3 \mid x \geq 2] = \frac{P[x=3 \cap x \geq 2]}{P[x \geq 2]} = \frac{P[x=3]}{[x \geq 2]}$$

since the intersection is simply $P[x=3]$

From the data:

$$P[x=3] = 266 \cdot 0.3 = 79.8$$

$$P[x \geq 2] = (258 \cdot 0.2) + (266 \cdot 0.3) + (236 \cdot 0.4) = 225.8$$

$$\therefore P[x=3 \mid x \geq 2] = \frac{79.8}{225.8} = 0.3534$$

Analysis

To determine if generating a larger sample size, that is to test 10,000 microprocessors, will have a more accuracy, upon running each script to find percent difference. It is apparent that I have probably made some error when hand calculating the mean as I have a larger percent difference from my simulated value to my calculated value when using a larger sample space, which should lead to more values to be iterated and calculated upon, and by having more values to calculate from, there should be an increase in accuracy. Despite the error within mean, there is less of a difference between the calculated values and simulated values for standard deviation:

```
MeanDiff = 2.2222e-14
 stdDiff = 194.5190
probDiff = 0.0029

MeanDiff1000 = 163.6020
stdDiff10000 = 182.9451
probDiff10000 = 0.0029
```

Figure 8. Percent Difference Calculations using MATLAB

This especially does not make sense when comparing to the arrays generated within the workspace which clearly show upon 10k iterations there are 10k values generated:

hist_x	[240,258,266,236]
hist_x1	[2468,2559,2504,...
ii	1000
iii	10000

Figure 9. Proving my error through the workspace

Conclusion

I calculated and simulated the results for 1,000 and 10,000 microprocessors each given their own parameters that classify their grade and probability. By increasing the sample space, I was able to achieve a smaller percentage difference between my calculated and simulated values, though I may have made an error within my analytical calculations and MATLAB calculations that I did not have time to fully debug. Additionally, I attempted to graph relative frequency as a percentage based upon the grade given to each microprocessor, which did show a significant change when iterating from 1,000 microprocessors to 10,000 microprocessors. Hovering over each bar the y-value corresponding to the respective grade differs in the 10,000 microprocessors figure, as the values are extended out 2 decimal places. When testing 1,000 microprocessors the relative frequency for grades 1-4 yielded: 2.4, 2.58, 2.66, and 2.36, while testing 10,000 microprocessors yielded: 2.468, 2.559, 2.504, and 2.469.

Appendix MATLAB CODE:

Note I program this code using MATLAB's Live editor and then converted it into a .m script for the attached MATLAB file. The appendix is based on the live script version.

```
%% ========== Matlab code sample for Practicum 1 Spring2021===========
%Program by Robert L. Bara  TUID:915614617  Email: tuj22026@temple.edu
% Simulation: Testing Microprocessors
close all
clear all
clc
range_x = [1:4]; %Grade Range
```

Iterating across 1000 microprocessors with various grades

```
n=1e3; %Number of Microprocessors
for ii=1:n
    processors=randi(4,1); %The sample of processors to be tested
    x(ii)=processors;
    if x(ii)==1      %Grade=1 with probability of 0.1
        y(ii)=0.1;
    elseif x(ii)==2 %Grade=2 with probability of 0.2
        y(ii)=0.2;
    elseif x(ii)==3 %Grade=3 with probability of 0.3
        y(ii)=0.3;
    else            %Grade=4 with probability of 0.4
        y(ii)=0.4;
    end
end
```

```
% Calculations by MATLAB
mean_y_sim=sum(y)/n
std_y_sim=sqrt(sum((y-mean_y_sim).^2)/n)
```

```
%The conditional probability P[X=3|X>=2]=P[X=3∩X>=2]/P[X>=2]
%The intersection of X>=2 and X=3, will be X=3

%Processors out of 1000

g3=sum(x==3); %how many processors are grade 3

g2=sum(x==2); %how many processors are grade 2

g4=sum(x==4); %how many processors are grade 4

g1=sum(x==1); %how many processors are grade 1


d=g3+g2+g4; %how many processors are grade 2 or more
P_3=g3*0.3; %Probability P(X=3)

P_2orMore=(g2*0.2)+(g3*0.3)+(g4*0.4); %Probability P(X>=2)

%Conditional Probability calculation

P_32=P_3/P_2orMore;



%Plotting Histograph and Analysis
figure(1)

hist_x = hist(x, range_x);

bar(range_x,((hist_x*10)/n));

title('Testing 1000 Microprocessors')

xlabel('Grade')

ylabel('Relative Frequency')


fprintf('Mean of Y by MATLAB: %f\n',mean_y_sim);
fprintf('Standard Dev of Y by MATLAB: %f\n',std_y_sim);

fprintf('There are %d Grade 3 and %d Grade 2 or higher Microrocessors\n',g3,d);

fprintf('Therefore, P[x=3|x>=2]=P[X=3∩X>=2]/P[X>=2], which equals %d\n',P_32);
```

For 10000 processors

```
n1=10e3; %Number of Microprocessors

for iii=1:n1

    processors_10000=randi(4);

    x1(iii)=processors_10000;

    if x1(iii)==1      %Grade=1 with probability of 0.1

        y1(iii)=0.1;

    elseif x1(iii)==2 %Grade=2 with probability of 0.2

        y1(iii)=0.2;

    elseif x1(iii)==3 %Grade=3 with probability of 0.3

        y1(iii)=0.3;

    else            %Grade=4 with probability of 0.4

        y1(iii)=0.4;

    end

end
```

```
% Calculations by MATLAB
mean_y_sim_10000=sum(y1)/n1

std_y_sim_10000=sqrt(sum((y1-mean_y_sim_10000).^2)/n1)


%The conditional probability P[X=3|X>=2]=P[X=3∩X>=2]/P[X>=2]
%The intersection of X>=2 and X=3, will be X=3

%Processors out of 10000

g3_1=sum(x==3); %how many processors are grade 3

g2_1=sum(x==2); %how many processors are grade 2

g4_1=sum(x==4); %how many processors are grade 4

g1_1=sum(x==1); %how many processors are grade 1


d1=g3_1+g2_1+g4_1; %how many processors are grade 2 or more
P_3_1=g3_1*0.3; %conditional probability P(X=3)
```

```
P_2orMore_1=(g2_1*0.2)+(g3_1*0.3)+(g4_1*0.4); %Conditional Probability P(X>=2)

%Conditional Probability calculation

P_32_1=P_3_1/P_2orMore_1;


%Plotting Histograph and Analysis
figure(2)

hist_x1 = hist(x1, range_x);

bar(range_x,hist_x1/n1);

title('Testing 10000 Microprocessors')

xlabel('Grade')

ylabel('Relative Frequency')


fprintf('Mean of Y by MATLAB: %f\n',mean_y_sim_10000);
fprintf('Standard Dev of Y by MATLAB: %f\n',std_y_sim_10000);

fprintf('There are %d Grade 3 and %d Grade 2 or higher
Microrocessors\n',g3_1,d1);

fprintf('Therefore, P[x=3|x>=2]=P[X=3∩X>=2]/P[X>=2], which equals %d\n',P_32_1);
```

Calculating Percent Differences between Simulation and Calculations

```
%for 1000 Microprocesors
MeanSim=mean_y_sim;

MeanCalc=0.2498;

stdSim=std_y_sim;

stdCalc=7.8915;

probSim=P_32;

probCalc=0.3534;


%Percent Differences
MeanDiff=(abs(MeanSim-MeanCalc)/((MeanSim+MeanCalc)/2))*100

stdDiff=(abs(stdSim-stdCalc)/((stdSim+stdCalc)/2))*100
```

```
probDiff=(abs(probSim-probCalc)/((probSim+probCalc)/2))*100


%for 10000 Microprocesors
MeanSim10k=mean_y_sim_10000;

MeanCalc10k=0.025;

stdSim10k=std_y_sim_10000;

stdCalc10k=2.4977;

probSim10k=P_32_1;

probCalc10k=0.3534;


%Percent Differences
MeanDiff1000=(abs(MeanSim10k-MeanCalc10k)/((MeanSim10k+MeanCalc10k)/2))*100

stdDiff10000=(abs(stdSim10k-stdCalc10k)/((stdSim10k+stdCalc10k)/2))*100

probDiff10000=(abs(probSim-probCalc10k)/((probSim10k+probCalc10k)/2))*100
```

Workspace

Name	Value
ans	0.3534
d	760
d1	760
g1	240
g1_1	240
g2	258
g2_1	258
g3	266
g3_1	266
g4	236
g4_1	236
hist_x	[240,258,266,236]
hist_x1	[2468,2559,2504,...
ii	1000
iii	10000
mean_y_sim	0.2498
mean_y_sim_1...	0.2497
MeanCalc	0.2498
MeanCalc10k	0.0250
MeanDiff	2.2222e-14
MeanDiff1000	163.6020
MeanSim	0.2498
MeanSim10k	0.2497
n	1000
n1	10000
P_2orMore	225.8000
P_2orMore_1	225.8000
P_3	79.8000
P_32	0.3534
P_32_1	0.3534
P_3_1	79.8000
probCalc	0.3534
probCalc10k	0.3534
probDiff	0.0029
probDiff10000	0.0029
probSim	0.3534
probSim10k	0.3534
processors	1
processors_10...	2
range_x	[1,2,3,4]
std_y_sim	0.1096
std_y_sim_100...	0.1112
stdCalc	7.8915
stdCalc10k	2.4977
stdDiff	194.5190
stdDiff10000	182.9451
stdSim	0.1096
stdSim10k	0.1112
x	*1x1000 double*
x1	*1x10000 double*
y	*1x1000 double*
y1	*1x10000 double*

Figure 10A and 9B. MATLAB Workspace Generated by my selected run of Random Values

<u>Sample Code given that was modified:</u>

```
%% ========= Matlab code sample for Practicum 1 ===========

% Simulation: Tossing an unfair coin

clear

n = 1000; % Number of tosses


range_y = [0:1];


for ii=1:n

   y(ii) = (rand>0.7);

end


figure(1)

hist_y = hist(y, range_y);

bar(range_y,hist_y/n);

xlabel('Value of face Y')

ylabel('Relative frequency')


mean_y = sum(y)/n;

std_y = sqrt(sum((y-mean_y).^2)/n);


fprintf('The mean value of Y is %5.3f\n',mean_y);

fprintf('The standard deviation of Y is %5.3f\n', std_y);
```