Kelsey Nguyen
Matlab 3
November 7, 2017

ECE 352: Gaussian Variable

Gaussian function is the probability density function of the normal distribution follows by the formula $F(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{\frac{-1}{2}(\frac{x-\mu}{\sigma})^2}$. The purpose of this matlab is to compare prototype production lines and current type production lines using Gaussian function.

The provided tensile test data was imported as a variable. The command mean() and std() calculated the mean and standard deviation for the files. The two for loops keep track of the ones that is below 8N. Only 7 can withstand force in current production lines while prototype production lines have 280 can withstand force. Those values can be used to calculate which ones would be better to launch. Using that number divided by 1000 and multiply by a hundred. The probability estimate was calculated using the data in matlab followed by $Z = \frac{(8-Mean)}{Standard\ Deviation}$ formula. Figure 2, shows that ZCurrent is -2.46 which is .0069 and ZProto is -.5526 which is .2912 using the standard normal table. The cdfC and cdfP values are calculated using matlab which is very closed to the values calculated using the standard normal table. The percentage for prototype production lines, PercentP, is higher than current production lines, percentC. All of the values can be found in Figure 2. The two plots were plotted in the same figure. This can be performed by using the subplot () command. Histogram shows how many valuea fall into a certain range, and the grid on command helped human to see the value more precise. Without having to calculate for anything, the graph shows that there is more product withstanding below 8N in the prototype plot. By just observing figure 1 and calculating the data, prototype production lines is the chosen one, and it is ready to be launched.

Overall, the experiment was good. This is a continuous of the previous matlab, which makes it a little easier to start. There were some parts that was a little confusing such as comparing the probability estimate. There was an uncertainty of what compare to what. Even though this is a continuous of the previous matlab, but it was definitely more challenging. It took a while to understand the requirements and see the bigger picture for this matlab. The experiment required a lot of research and knowledge for matlab. This gave students an opportunity to learn more and apply the real data to Gaussian function.

```
clear all

%read the input file
protoLine = importdata('protoLineData.txt');
currentLine = importdata('currentLineData.txt');

X =4:0.1:15;% x-axis to plot agaisnt random variable

% standard normal gausian
protoLineMean = mean(protoLine);
protoLineStand = std(protoLine);
currentMean = mean(currentLine);
currentStand = std(currentLine);


subplot(2,1, 1);
yProto= (1/sqrt(2*pi*protoLineStand*protoLineStand))* exp((-(X-protoLineMean).^2)/(2*protoLineStand*protoLineStand));%plot estimated pdf from the generated data
histogram(protoLine,25, 'Normalization','pdf');
hold on;
plot (X, yProto, 'r--');%plot computed theoretical protoLine
axis([5 13 0 .5])
hold off; grid on;
title('Prototype Production Lines')
legend('Histogram','Theoretical PDF');
xlabel('Resistance')
ylabel('PDF')


subplot(2,1, 2);
%yCurrent = normpdf(X,currentMean,currentStand);%Normal probability density function
yCurrent= (1/sqrt(2*pi*currentStand*currentStand))* exp((-(X-currentMean).^2)/(2*currentStand*currentStand));%plot estimated pdf from the generated data
histogram(currentLine,25, 'Normalization','pdf');
hold on;
plot (X, yCurrent,'r--');%plot computed theoretical currentLine
axis([5 13 0 1])
hold off; grid on;
title('Current Production Lines')
legend('Histogram','Theoretical PDF');
xlabel('Resistance')
ylabel('PDF')
```

```
%Estimating the probability of a catheter testing below 8N
pDataSize= numel( protoLine(:,1));%Number of array elements
PN = 0;% N is less than 8
for i = 1 : pDataSize
   if protoLine(i) < 8.0;
      PN = PN + 1;
         end
end

pDataSize= numel( currentLine(:,1));%Number of array elements
CN = 0;% N is less than 8
for i = 1 : pDataSize
   if currentLine(i) < 8.0;
      CN = CN + 1;
   end
end

%Probability estimate using the input data
ZProto= (8-protoLineMean)/protoLineStand;
ZCurrent= (8-currentMean)/currentStand;


%finding the CDF
cdfP=normcdf(8,protoLineMean,protoLineStand);
cdfC=normcdf(8,currentMean, currentStand);

%CDF in percentage
percentP=cdfP*100;
PercentC=cdfC*100;

%print to the command window
fprintf('ZProto is %d\n', ZProto);
fprintf('ZCurrents is %d\n', ZCurrent);
fprintf('cdfP is %d\n', cdfP);
fprintf('cdfCs is %d\n', cdfC);
fprintf('percentP is %d\n', percentP);
fprintf('percentC is %d\n', PercentC);
```

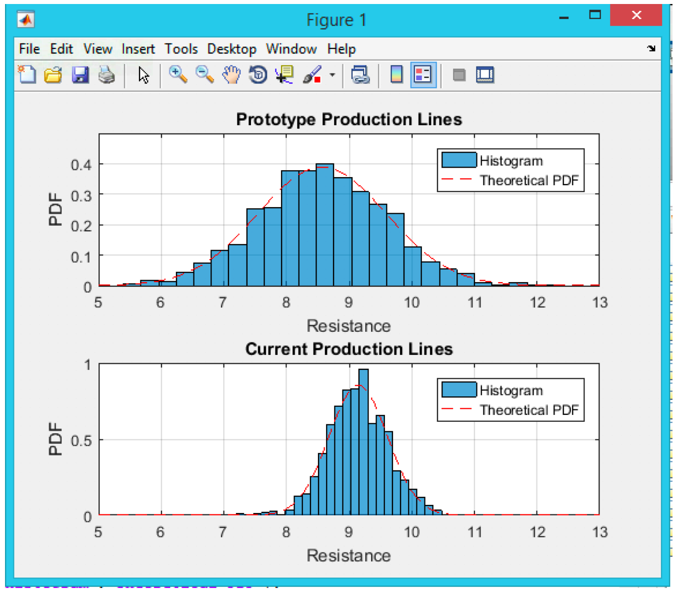

Figure 1

Name	Value
cdfC	0.0070
cdfP	0.2903
CN	7
currentLine	*1000x1 double*
currentMean	9.1454
currentStand	0.4662
i	1000
pDataSize	1000
PercentC	0.7008
percentP	29.0284
PN	280
protoLine	*1000x1 double*
protoLineMean	8.5665
protoLineStand	1.0252
X	*1x111 double*
yCurrent	*1x111 double*
yProto	*1x111 double*
ZCurrent	-2.4568
ZProto	-0.5526

Figure 2