Name: Thuan Tran
Date: April 8th, 2017
CSS 490

Homework 1

1)

// First I manually input the data by using the command:

```
data =
{753,955,957,661,768,806,775,730,939,952,946,558,655,775,653,905,935,940,918,856,70
8,498,835,898,918,909,809,869,960,693,820,870,878,704,562};
```

// Then I changed the data type of "data" in a matrix

```
dataMatrix = cell2mat(data);
```

// Then I calculated the mean of the matrix

```
meanValue = mean(dataMatrix);
```

// The formula to calculate the mean is:

```
//The result is 810.51
```

```
Then I calculate the standard deviation of the matrix

standardDev = std(dataMatrix);

//The formula to calculate the sample standard deviation is :
// The result is 128.32


 // I then plot the data,change the size to make the data more noticeable

plot(dataMatrix,0,"markersize", 12);

// Hold, not plotting the point yet

hold on;

 // I then plot the mean value in the same plot but this time make the point bigger
```

plot(meanValue,0, 'o',"markersize", 17);

Based on the sample mean, we can interpret that : over different days, the average solar intensity is: 810.51

2.) NOTE : For this assignment, I used an additional package of octave called "statistics"
To load this package, type pkg load statistics
To install this package, install package io first

Just like the one above, first I load in the text file (I need to locate octave to the directory that has the text file first)

y = load("HW1_problem2_data.txt","r")

pkg load statistics // Load the additional package that has the tabulate method

// The tabulate method return a matrix that divide the file into 8 bins and the 4th column return the cumulative frequency

fre = tabulate(y,8)

5329.0000	2.0000	2.0000	2.0000
5348.7500	3.0000	3.0000	5.0000
5368.5000	15.0000	15.0000	20.0000
5388.2500	14.0000	14.0000	34.0000
5408.0000	22.0000	22.0000	56.0000
5427.7500	21.0000	21.0000	77.0000
5447.5000	14.0000	14.0000	91.0000
5467.2500	9.0000	9.0000	100.0000

// **Question 2a**
// Plot the cumulative frequency plot
bar(fre(:,1),fre(:,4),"hist")

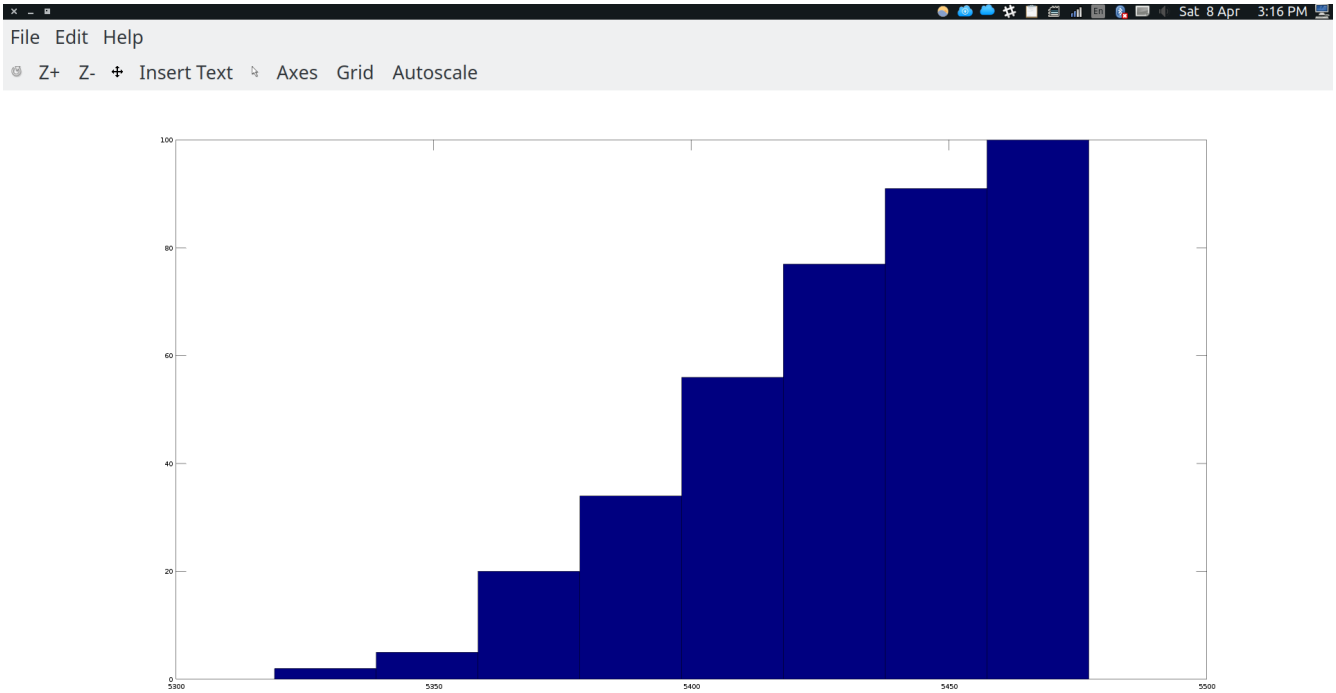

Illustration 1: cumulative frequency plot using 8 bins

// Question 2b

// Plot the histogram with 8 bins
hist(y,8)

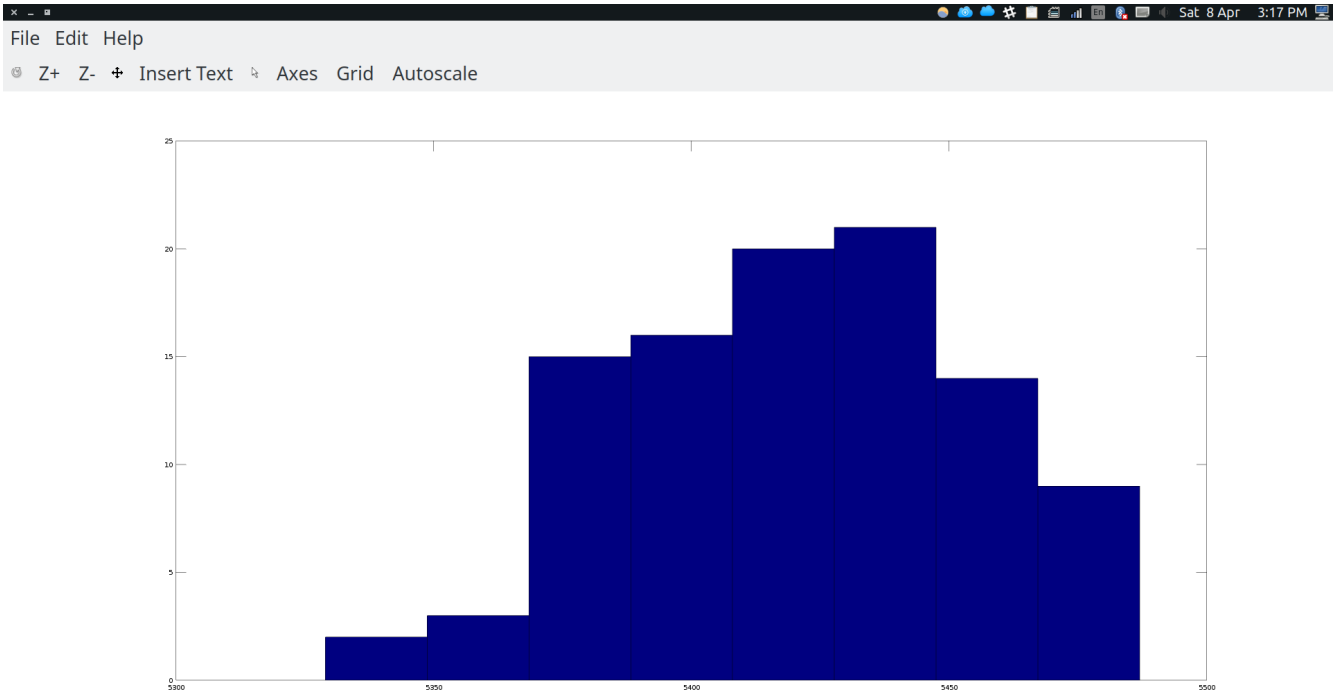

Illustration 2: Histogram using 8 bins

// **Question 2c**
// Plot cumulative frequency with 16 bins
// Need to use the tabulate function with 16 bins this time

fre = tabulate(y,16)

5329.0000	1.0000	1.0000	1.0000
5338.8750	1.0000	1.0000	2.0000
5348.7500	2.0000	2.0000	4.0000
5358.6250	1.0000	1.0000	5.0000
5368.5000	4.0000	4.0000	9.0000
5378.3750	11.0000	11.0000	20.0000
5388.2500	3.0000	3.0000	23.0000
5398.1250	11.0000	11.0000	34.0000
5408.0000	10.0000	10.0000	44.0000
5417.8750	12.0000	12.0000	56.0000
5427.7500	11.0000	11.0000	67.0000
5437.6250	10.0000	10.0000	77.0000
5447.5000	7.0000	7.0000	84.0000
5457.3750	7.0000	7.0000	91.0000
5467.2500	4.0000	4.0000	95.0000
5477.1250	5.0000	5.0000	100.0000

bar(fre(:,1),fre(:,4),”hist”)

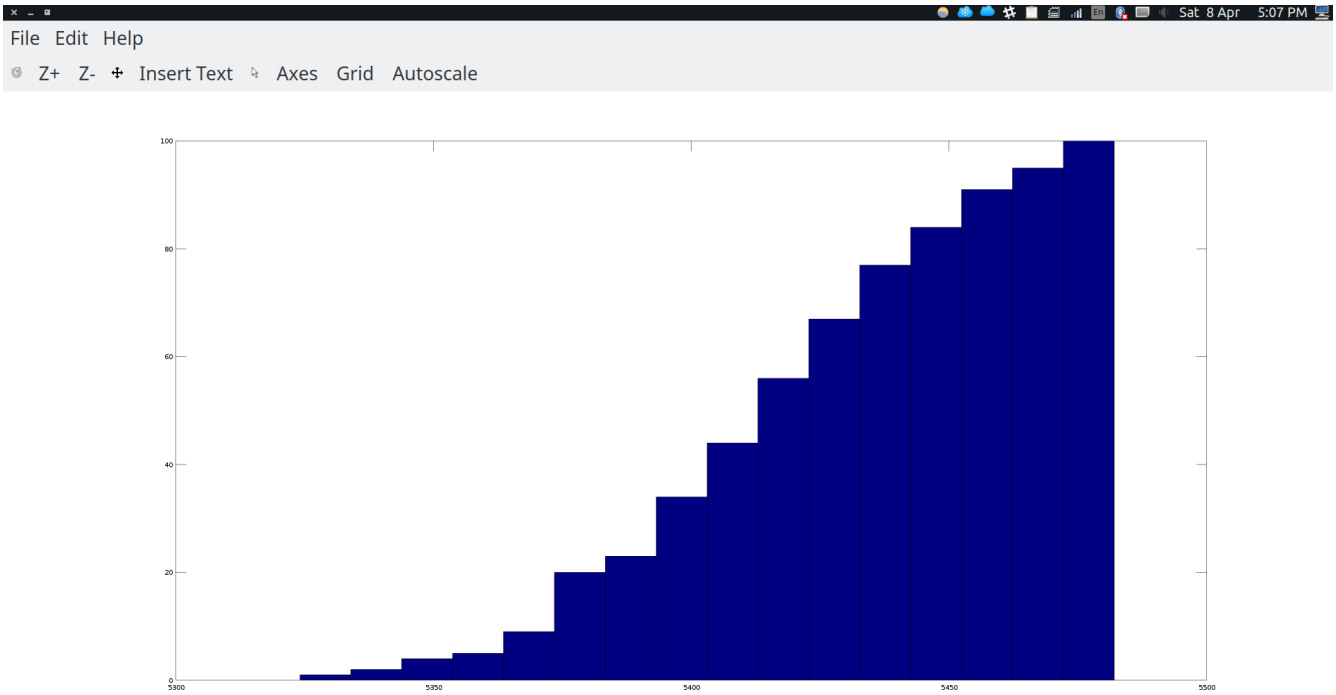

Illustration 3: cumulative frequency plot using 16 bins

// **Question 2d**
// Histogram with 16 bins
hist(y,8)

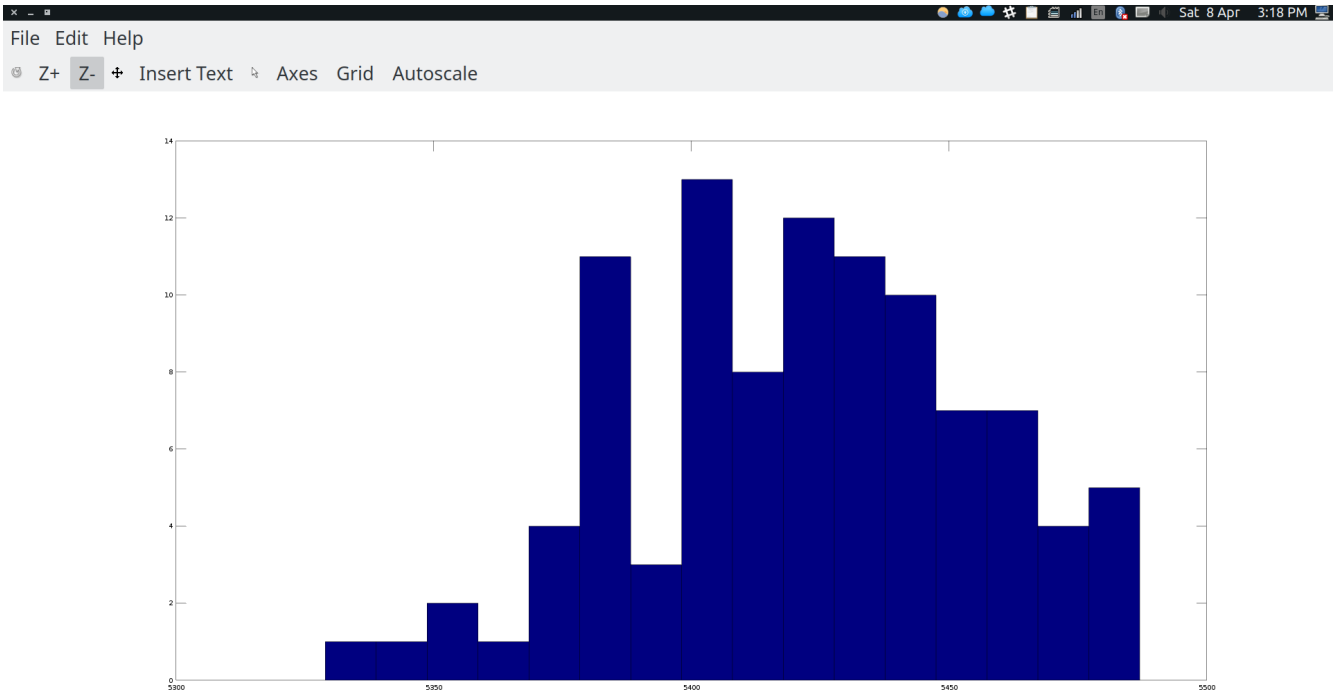

Illustration 4: Histogram with 16 bins

e) Difference between using the histogram vs cumulative frequency plot
For histogram, it will be easy to indicate the frequency of the variables in the data. We can easily pick up how many time this data (or data lies in interval) appear
For cumulative frequency plot, it is good for determining the percentile. At at certain point, we can determine how much that value “accumulate”
When using different bins, it provided different values to interpret, the more bins, the more detailed value. Depending on circumstances, we might want smaller bins to generalize or a larger bins for more details. The cumulative plot with 16 bins is more detailed than the one with 8 bins
For example, the histogram with 16 bins vs the histogram with 8 bins. The one with 16 bins shows that in the nearly middle there is a data point that has a small frequency. However, in the 8 bins, all the data in that interval has been grouped so there appear a lot of data in that interval.

3)
a) A probability that a package weight less than 10 lbs is the integral of the function from 1 to 10

```
// Defined the function in Octave
fun = inline("(70/(69*x^2))")


// Compute the function from 1 to 10
quad(fun,1,10)

// The answer is : 0.91304. Meaning that 91.3% that the package weight less than 10lbs
```

b)

```
// Define the function for the mean (expected value)
fun = inline("x*(70/(69*x^2))")

// Compute the mean from 1 to 70
me = quad(fun,1,70)
// The answer is : 4.3101

// Now we have the mean, we can plug it into another function to calculate the sample variance
// Here we defined the a function with the input parameter x and we plug in the variable that represent the mean
f = @(x) (x-me)^2 *(70/(69*x^2))

// Compute the variance
quad(f,1,70)

// The answer is 51.423
```

c) Since we know the average of a package is 4.3101. We can take that amount times the cost per lbs
4.3101 * 3 = $ 12.930