By: Jubayad Hassnat

Probability & Statistics Final Project Documentation

Table of Contents

[Part 1: Datasets and Graphs 2](#_Toc197377999)

[Code 2](#_Toc197378000)

[Quadratic Graph 8](#_Toc197378001)

[Outputs 8](#_Toc197378002)

[Sine Graph 13](#_Toc197378003)

[Outputs 13](#_Toc197378004)

[Absolute Graph 17](#_Toc197378005)

[Outputs 17](#_Toc197378006)

[Part 2: Octave 22](#_Toc197378007)

[Octave Tutorial & Learning 22](#_Toc197378008)

[All Learning Sources 27](#_Toc197378009)

[Octave Code & Outputs 28](#_Toc197378010)

[Main.m 28](#_Toc197378011)

[OriginalFormula.m 45](#_Toc197378012)

[Salter.m 47](#_Toc197378013)

[Smoother.m 48](#_Toc197378014)

[DataHandler 48](#_Toc197378015)

[What is Octave (1 Page Paper) 49](#_Toc197378016)

[Part 3: JFree 50](#_Toc197378017)

Da

# Part 1: Datasets and Graphs

## Code

#### Main.java

This is the output of my csv files filled with values of the original graph, the salted values, and the smoothed values. Each formula, quadratic, sine, and absolute has three sets. Each set has bounds from -10 to 10, -25 to 25, and -50 to 50 namely.

A computer screen shot of text

AI-generated content may be incorrect.

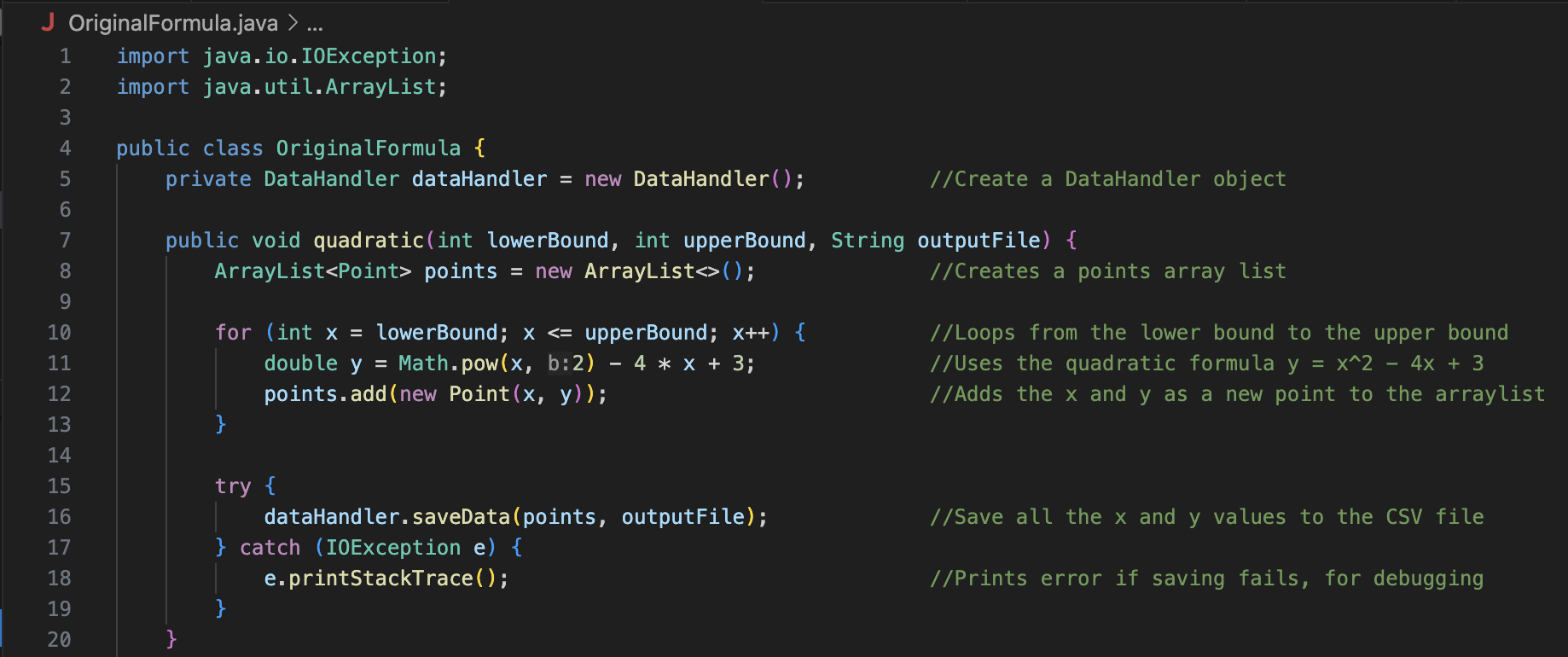
A computer screen with text

AI-generated content may be incorrect.

#### OriginalFormula.java

##### Quadratic

The quadratic class is made by having the parameters of lowerBound, upperBound, outputFile. This allows us to set the upper and lower bounds along with the output file. By having a designated output file, it allows for the program to write over the file rather than creating a new file on every run. The quadratic class loops from the lower bound to the upper bound. It uses the formula y . The values of the points are added to the points array list and that array list gets saved to the outputFile by the DataHandler object. If it fails, it gives us an error and we are able to debug using that.



##### Sine

The sine class is made by having the parameters of lowerBound, upperBound, outputFile. The quadratic class loops from the lower bound to the upper bound. It uses the formula. The values of the points are added to the points array list and that array list gets saved to the outputFile by the DataHandler object. If it fails, it gives us an error and we are able to debug using that.

A computer screen with text

AI-generated content may be incorrect.

##### Absolute

The absolute class is made by having the parameters of lowerBound, upperBound, outputFile. The quadratic class loops from the lower bound to the upper bound. It uses the formula. The values of the points are added to the points array list and that array list gets saved to the outputFile by the DataHandler object. If it fails, it gives us an error and we are able to debug using that.

A computer screen shot of a program code

AI-generated content may be incorrect.

#### Salter.java

The Salter class reads the data of the inputFile that’s given to it through the parameter. After the data is read, it checks if the given salt range is greater than 1. If not, it returns and says for the user to change the salt range. If it goes through, it applies the salt algorithm, which is a random value that’s within the salting range and adds it to the y value. They add this new salted y value to the saltedPoints array, and this array is than written into a new csv file that’s handled by the DataHandler object. If it fails, it gives us an error and we are able to debug using that.

A computer screen shot of a program code

AI-generated content may be incorrect.

#### Smoother.java

The smoother class takes read’s the input file that’s given to it by the parameter. It than takes the y values from the file and applies the smoothing algorithm to it. The smoothing algorithm is finding both the start and ending values of the y values. From the given windowSize variable, it finds the average point by looping through from the start to the end values. This new value is then the new smoothed y value. This value is then written and saved into a csv file which is handled by the DataHandler object.

A screen shot of a computer program

AI-generated content may be incorrect.

#### DataHandler.java

##### loadData.java

The loadData class creates an arraylist variable called data. It then tries to read the csv file. Once it reads the file, it skips the header, which just says “x, y” which is irrelevant to the class. It than makes the values given into it into a coordinate point, in the form of x,y. Then it transforms the x values given to it to a double and lets it be referenced as x. Same thing is done to the y value. Then both of them data values are stored into the file in the form of x,y.A computer screen shot of a program code

AI-generated content may be incorrect.

##### saveData.java

The saveData class writes data into a file. If the file exists already, then it writes on top of it. It writes the header of the csv file as x, y. It then loops through every point and writes the values into the file in the form of x, y. Once done, it prints a message saying this was successfully printed into this new file.

A computer screen with colorful text

AI-generated content may be incorrect.

#### Points.java

The points class creates a class that has two double variables, x and y. These variables can be referenced and used by calling the point method.

A screen shot of a computer code

AI-generated content may be incorrect.

## Quadratic Graph

Formula Used:

### Outputs

#### Set 1:

Bounds: -10 to 10

Salt Range: -10 to 10

Smoother Window Size: 1

#### Set 2:

Bounds: -25 to 25

Salt Range: -25 to 25

Smoother Window Size: 3

#### Set 3:

Bounds: -25 to 25

Salt Range: -25 to 25

Smoother Window Size: 3

## Sine Graph

Formula Used:

### Outputs

#### Set 1:

Bounds: -10 to 10

Salt Range: -10 to 10

Smoother Window Size: 1

#### Set 2:

Bounds: -25 to 25

Salt Range: -25 to 25

Smoother Window Size: 3

#### Set 3:

Bounds: -25 to 25

Salt Range: -25 to 25

Smoother Window Size: 3

## Absolute Graph

Formula Used:

### Outputs

#### Set 1:

Bounds: -10 to 10

Salt Range: -10 to 10

Smoother Window Size: 1

#### Set 2:

Bounds: -25 to 25

Salt Range: -25 to 25

Smoother Window Size: 3

#### Set 3:

Bounds: -25 to 25

Salt Range: -25 to 25

Smoother Window Size: 3

# Part 2: Octave

## Octave Tutorial & Learning

<https://www.youtube.com/watch?v=LhPZwdhutgU>

I started following the first source. I did something very simple and plotted the first 10 values of a sine function. I ran some very simple code and plotted it. I got this from running the following script in the Octave gui:

A screenshot of a computer

AI-generated content may be incorrect.

A screen shot of a graph

AI-generated content may be incorrect.

The code sets the upper bound variable equal to 10. X is the increments from the first value, which is 1, to the variable upper bound, which is 10. Y is equal to the sine of x. The resulted graph depicts exactly that as we can see that x starts at 1 and ends at 10. Y is the values of the sine function.

<https://www.youtube.com/watch?v=NtMOab_nhs0>

https://www.youtube.com/watch?v=WUxImdA7k8E

As I got more familiar with Octave and watching more videos, I started to learn about different ways to structure my code and how to ask for certain data to be printed out into graphs. I also learned about the “hold on” syntax which retains the data that was given and add more data on top of it. Below is my test code based on what I learned and saw from their test code:

A screen shot of a computer program

AI-generated content may be incorrect.

A graph with blue dots and a line

AI-generated content may be incorrect.

I took apart their code and tried to understand what each line does.

Line 1 sets the variable npoints equal to 8.



Line 3 creates a new variable x which equals to increments of 1 through 8 divided by 8. The “ ’ ”, or apostrophe, turns it into columns.



Line 4 creates a new variable y which creates a cosine function with extra values making it not look like a regular cosine wave.



Line 6 uses the plot function to plot the variables of x and y. The ‘ . ’ Changes the markers to become dots or periods. The ‘markersize’, 10 syntax makes the size of the markers bigger so that it’s easier to see.



Line 8 makes a new variable called S which filles a matrix filled with zeros to the length of 8, since npts is equal to 8.



The nested for loop from lines 9-13 fills the matrix that’s filled with zeros with values. It goes row by row.

A black background with numbers and symbols

AI-generated content may be incorrect.

Line 15 creates a new variable called Sinv which uses the inverse function to find the inverse of the S variable matrix that’s filled.



Line 16 creates a new variable called the c which multiples the values of the Sinv variable and the values of the y variable. This variable tells us what numbers to use to find the best curve.

A black background with a black and pink text

AI-generated content may be incorrect.

Line 18 creates a new variable called np2 and sets that equal to 50. The purpose of this variable is to have more points in the graph, making our graph smoother.



Lines19-20 creates new variables called x2 and y2. x2 is equal to the values of 1 to npt2, which is 50. The “ ‘ “, or the apostrophe turns it into columns and then the value is divided by np2, or 50. The variable y2 is equal to the values of x2

A black background with white text

AI-generated content may be incorrect.

Line 21 creates a new variable fj which is equal to the values of c.



Line 22-27 is a nested for loop in which calculates the values of y2 using the formula. The first for loop runs from 1 to np2 which is 50. The second loop runs from 1 to npts which is 8.A black background with white text

AI-generated content may be incorrect.

Line 29 uses the syntax “hold on” which keeps the values plotted from before.



Line 30 plots the values of x2 and y2.

A black background with orange text

AI-generated content may be incorrect.

## All Learning Sources

1. <https://www.youtube.com/watch?v=LhPZwdhutgU> - basics

2. <https://www.youtube.com/watch?v=NtMOab_nhs0> - more in depth

3. <https://www.geeksforgeeks.org/octave-basics-of-plotting-data/> - basics

4. <https://bioweb.pasteur.fr/docs/modules/octave/3.8.2/octave/Formatted-Output.html#XREFsprintf> – title configurations for cleaner output

## Octave Code & Outputs

### Main.m

#### Set 1 Quadratic Code:

Lines 4-7 runs the first set of the quadratic function. It uses the bounds of -10 to 10, uses the salted range of 10, and the window size of 1.

A black background with text

AI-generated content may be incorrect.

#### Output:

A graph with a line drawn on it

AI-generated content may be incorrect.

A graph with a line drawn on it

AI-generated content may be incorrect.

A graph with a line drawn on it

AI-generated content may be incorrect.

#### Set 2 Quadratic Code:

Lines 9-12 runs the second set of the quadratic function. It uses the bounds of -25 to 25, uses the salted range of 25, and the window size of 3.

A computer code with text

AI-generated content may be incorrect.

#### Output:

A graph with a blue line

AI-generated content may be incorrect.

A graph showing a line

AI-generated content may be incorrect.

A graph with green line

AI-generated content may be incorrect.

#### Set 3 Quadratic Code:

Lines 14-17 runs the third set of the quadratic function. It uses the bounds of -50 to 50, uses the salted range of 50, and the window size of 7.

A black background with text

AI-generated content may be incorrect.

#### Output:

A graph of a function

AI-generated content may be incorrect.

A graph showing a line

AI-generated content may be incorrect.

A graph with green line

AI-generated content may be incorrect.

#### Set 1 Sine Code:

Lines 19-22 runs the first set of the sine function. It uses the bounds of -10 to 10, uses the salted range of 50, and the window size of 1.

A computer code with text

AI-generated content may be incorrect.

#### Output:

A graph with blue lines

AI-generated content may be incorrect.

A graph showing a line

AI-generated content may be incorrect.

A graph with green line

AI-generated content may be incorrect.

#### Set 2 Sine Code:

Lines 24-27 runs the first set of the sine function. It uses the bounds of -25 to 25, uses the salted range of 25, and the window size of 3.

A black background with text and numbers

AI-generated content may be incorrect.

#### Output:

A graph of a sine

AI-generated content may be incorrect.

A graph showing a line of red lines

AI-generated content may be incorrect.

A graph with green lines

AI-generated content may be incorrect.

#### Set 3 Sine Code:

Lines 29-32 runs the first set of the sine function. It uses the bounds of -50 to 50, uses the salted range of 50, and the window size of 7.

A black background with text and numbers

AI-generated content may be incorrect.

#### Output:

A graph with blue lines

AI-generated content may be incorrect.

A graph showing a red line

AI-generated content may be incorrect.

A graph with green lines

AI-generated content may be incorrect.

#### Set 1 Absolute Code:

Lines 34-37 runs the first set of the sine function. It uses the bounds of -10 to 10, uses the salted range of 10, and the window size of 1.

A black background with text

AI-generated content may be incorrect.

#### Output:

A graph of a line

AI-generated content may be incorrect.

A graph showing a red line

AI-generated content may be incorrect.

A graph with green lines

AI-generated content may be incorrect.

#### Set 2 Absolute Code:

Lines 39-42 runs the first set of the sine function. It uses the bounds of -25 to 25, uses the salted range of 25, and the window size of 3.

A black background with text and numbers

AI-generated content may be incorrect.

#### Output:A graph with a line drawn on it AI-generated content may be incorrect.

A graph with red lines

AI-generated content may be incorrect.

A graph with green line

AI-generated content may be incorrect.

#### Set 3 Absolute Code:

Lines 44-47 runs the first set of the sine function. It uses the bounds of -50 to 50, uses the salted range of 50, and the window size of 7.

A black background with text

AI-generated content may be incorrect.

#### Output:

A graph with a line drawn on it

AI-generated content may be incorrect.

A graph showing a line of red lines

AI-generated content may be incorrect.

A graph with green lines

AI-generated content may be incorrect.

### OriginalFormula.m

#### Quadratic Function

The code creates a quadratic function that takes in the parameter of lowerbound, upperbound, and outputfile. You can manually change the bounds so its easier to get varied results. The file parameter also allows for the name of the output csv file so it doesn’t continuously make new files every time and just overwrites over one if a csv file is created. It uses the formulaUsing the datahandler class from the DataHandler file, it's able to save the x and y values to the output file which is set by the parameter.

A screenshot of a computer program

AI-generated content may be incorrect.

#### Sine Function

The code creates a sine function that takes in the parameters lowerbound, upperbound, and outputfile. You can change the bounds manually. The outputFile parameter allows for the name of the output csv file so it doesn’t continuously make new files every time and just overwrites over one if a csv file is created. The formula being used is y. Just like the other functions, it uses the dataHandler from the DataHandler file to save the x and y values to the output file that’s passed in.

A screen shot of a computer program

AI-generated content may be incorrect.

#### Absolute Function

The code creates an absolute value function that takes in the parameters lowerbound, upperbound, and outputfile. You can manually adjust the bounds depending on how much of the graph you want to display. The outputFile parameter lets you name the CSV file so it doesn’t create a new one every time — instead, it will overwrite an existing file if one already exists. The formula being used is y = |x| - 2. Like the other functions, it uses the dataHandler from the DataHandler file to save the x and y values into the output file that’s provided.

A screen shot of a computer

AI-generated content may be incorrect.

### Salter.m

The salter function reads the csv file from thanks to the csvread function. It reads the csv file and skips the first line, or header. The header just says “x, y” so it’s not necessary for the program to read it. Afterwords it puts all x values in the first column and all y values in the second column. It applies the salting algorithm, which takes a random number in between the given salt range and adds that to the y. Afterwords, it writes the new salted y value back to the outputFile which is handled by the DataHandler File. It then prints out the salted graph.

A screenshot of a computer program

AI-generated content may be incorrect.

### Smoother.m

The smoother function copies the same method of reading the csv file as the salting function. It reads and ignores the first row. It then checks if the given windowSize variable is greater than 1. This is set through the parameter, but if the windowSize is less then 1 then the code asks you to change the value. If the windowSize is 1, it will return the same values as before since the value hasn’t moved any places. If the windowSize is anything else, it applies the smoothing algorithm to it. It averages the y value with the windowSize, and that becomes the value of the new y. This is then stored into a new csv file with the DataHandler function through the DataHandler file. Afterwords, it plots the graph.

A computer screen shot of a program

AI-generated content may be incorrect.

### DataHandler

The dataHandler function opens a file using the fopen function. The file that’s being opened is determined through the values of the parameter. Afterwords the “w” means to write, so it writes into the file. The fprint function in line 3 writes the heading of the file, which is “x, y”. The fprint function in line 4 writes the values of x and y into the file.

A black screen with white text

AI-generated content may be incorrect.

## What is Octave (1 Page Paper)

Octave is an open-source software and programming language. It is widely used as an alternative to Matlab because it’s free compared to the paid software Matlab. Octave is mostly used to graph mathematical functions in both 2D and 3D rendering. Since it’s an open-source program, it has received many useful updates that people themselves have suggested. One reason for its popularity is because of how similar Octave is to Matlab. So, if you were to able to learn and code in the Octave software, then you have also learned how to work in Matlab. Because of how little the difference between the two are, people tend to use Octave for the majority of their project and only tend to switch to Matlab if Octave is missing some functionality.

Octave is very similar to java and the syntax used. They both use similar syntax, so a feeling of familiarity helps approach people who are new to using the software. One of the biggest differences between java and octave is the use of semicolon. In java, semicolons mark the end of a statement or line. It tells the computer that everything before the semicolon is together and should be read as such. In octave it’s a bit different. The semicolon at the end of a line stops it from being displayed at the system. In the Octave language, every line is run and shown in the system after compiling. The use of the semicolon stops the system from displaying that line. In a sense, it can be seen as a stopper which is completely different from java.

Octave is mainly used for graph creation, data plotting, matrix operations, and numerical calculations. Because of its wide range of uses in math related subject, it’s often used in engineering, physics, and data science. It is especially useful in situations where data or graphs needs to be visualized and shown. It’s proficient in both 2D and 3D data models and rendering. It’s easy and friendly to use for beginners and doesn’t take long to become proficient in. It’s usage of short and easy code to give out the results make it great for beginners as well.

# Part 3: JFree