Using Octave

Written By: George Camac

I chose to learn Octave about as it interested me and I wanted to learn more about how this language actually worked. Octave can run its commands directly or through a script file. I began to try and learn how these script files worked and the common commands in Octave. I went through a lot of the provided syntax libraries for octave using ctrl-F to try and build what is now a working Plotter, Salter, and Smoother.

**Plotting**

When I began creating the Plotter I tried to figure out common commands, like

“linspace”: using linspace allows us to create an x- range for the graph that we would like to create, in my code I decided to go with was for my cos an sin graphs

“plot”: this command accepts the range, the curve, the color, display name, and a legend

“global”: while global will set the x and y values for all files so you can reference other scripts

“hold on”: allows you to plot another curve on the same graph you have just plotted

**Figure 1-1 sine and cosine plotted across with 20 points**

A screenshot of a computer

Description automatically generated

**Figure 1-2 sine and cosine plotted across with 50 points**

**A screenshot of a computer

Description automatically generated**

Doing this allows us to look at how our data and curve will show in a graph, making it easier to digest the information. Now to make this program better all I can really think of since there is so much ease of use already, is by creating scripts and turning them into methods with configurable parameters.

**Salting**

Using Octave I also was tasked with creating a Salter to be able to hide data and curves from those who may want to take steal or benefit from the information we have saved. To do so you have to go through the values of the curve you have submitted and add or subtract from a random value between 0 and the salt value you would like to use. In this section the most important and common things I need to know was…

**rand: this is Octave’s implementation of a random function to choose the randomized salt values**

**for/end: in Octave you do not use brackets for For loops, but you use end once you finished, this help**

**Figure 2-1 I salted the graph from 1-1 with a value of .3**

**A graph on a screen

Description automatically generated**

**Figure 2-2 I salted the graph from 1-2 with a value of 1**

**A screenshot of a graph

Description automatically generated**

While creating the salter I made sure that the values were being globally saved to be referenced in the future. I believe that this is a pretty good salter that couldn’t be changed too much to improve it.

Smoothing

The smoothing method was probably the hardest to understand how to implement. Luckily I was able to find extensions that have helped me understand the errors I was running into and why things weren’t working. Doing so I was able to figure out how to fully implement the sliding window through octave, however, I finally seemed to be able to find similarities between the java smoother and octave plotter

**Figure 3-1 I smoothed the graph from 2-1**

**A screenshot of a graph

Description automatically generated**

**Figure 3-2 I smoothed the graph from 2-2**

**A screenshot of a graph

Description automatically generated**

As you can see from the graph it hasn’t been smoothed that much, so we will try running the smoother again to see if we can do better!

**Figure 3-3 I smoothed Figure 3-2**

**A screenshot of a graph

Description automatically generated**

**Now that looks a lot better and more similar to the starting graph, however the max and minimum values are getting further from their respective points.**

Overall using octave wasn’t necessarily terribly difficult, the idea of having the built in functions for graphing is amazing. While it was also very helpful for them to include a octave.pdf with a breakdown of all the syntax used within octave, and how to build things from scratch. I was able to effectively use these provided documents to construct a working Plotter Salter and Smoother. I did also reference one other source in the beginning to help construct the beginning plot. Which helped me get a real understanding of how a full script is written.

Sources:

* Octave.pdf
* YouTube. (n.d.-a). *Octave Tutorial #10: Plot different types of plots for Absolute Beginners*. YouTube. https://www.youtube.com/watch?v=OakYztiXKKE