Cody Kimmins

JFreeChart Program Report

JFreeChart is a free Java chart library that allows users to easily generate professional quality graphs and charts in applications. It can be used to create a pie charts, bar charts, line charts, and scatter plots to name a few. To use this library, you must import it in your IDE. In my case, I imported the library to IntelliJ by downloading the jar files and adding it to my module within the program. The tutorial I followed used the XYSeriesCollection function that represents an XYSeries object on a graph. The following graph shows the slope function y = 2x + 5 as the red line, salted data as the blue line, and the smoothed data as the green line.

Chart, line chart

Description automatically generated

Figure 1

I created the title as “Plot, Salt, & Smooth Graph” with the x and y axis labelled as X and Y values respectively. In this result the X values are incrementing by 2 because the slope is 2. The salting method will take the y values and either add or subtract to that value based on a random integer 0 to 2. I chose to do it this way because I wanted the salter to be as random as possible but not all over the place. The amount added or subtracted is also random with bounds including 25 and 30. I used Apache’s library to smooth the data by using the mean function. I used this because of the rolling average method that takes 3 or more values and divides that by 3. However, this also makes the dataset smaller by 3 as it takes away values to find the average. The green line represents the smoothed data, and it mostly corrects the high and low spiking of the salted data line.

Chart, line chart

Description automatically generated

Figure 2

The second graph I generated shows the rolling average taking 5 values this time instead of 3 as shown in Figure 1. By increasing it to 5, the smoothing results are better and more like the original line. Taking a larger number of numbers to average means a better result although it also cuts off the last 5 numbers and stops short of the original red line. To make things more interesting I went and generated a final graph which is shown in Figure 3.

Chart, line chart

Description automatically generated

Figure 3

This time I increased the X values from 50 to 100 and added points to each line representing the values. The customization options to add the markers at each line makes it easier to see the lines and gives a better understanding of where each value is for the three lines. Playing around with the customization choices I generated a graph just to see how the different methods work as shown in Figure 4.

Chart

Description automatically generated

Figure 4