Rocco Vulpis

April 25, 2024

Salting Data using Java and Plotting in Excel

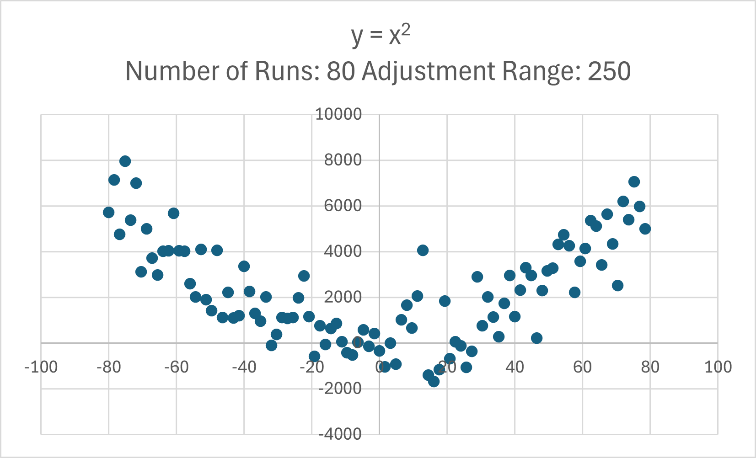
In the salting portion of this project, we randomly altered the y-values of a mathematical function using a process known as "salting". This involved modifying data points read from a CSV file, “data.csv”, before exporting the adjusted data to a new file, “saltedData.csv”. The graph created in Microsoft Excel with this data displays the effects of salting, while still maintaining a resemblance to the original function with noticeable random variations.

The first step is to create the function from the ExcelSalterTester class. This process starts by creating an ExcelSalter object and calling and passing parameters to its **run** method. The first parameter is **numberOfRuns** (The number of times the salter will iterate over the **yValues** ArrayList) and **adjustmentRange** (The range of values that each y value is randomly adjusted up or down by). Inside the **run** method in the ExcelSalter class, the **readFile** method is called. This process included the BufferedReader class for line-by-line reading of “data.csv”, and the parsing of each line's values into doubles. These values are stored in two ArrayLists: one for x-values and one for y-values.

After importing and storing the data, the **saltValues** method is then called to adjust the y-values. This method sets the number of iterations and range of random adjustments specified by the parameters passed to it. Each y-value is randomly increased or decreased within this range, and the adjusted value replaces its original value in the *yValues* ArrayList.

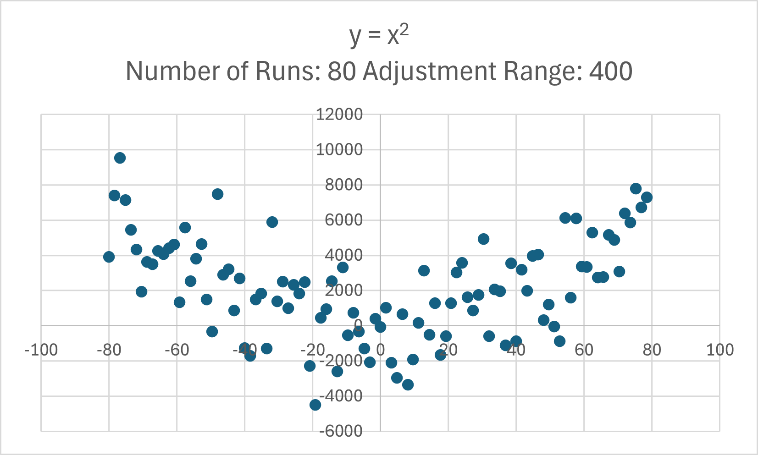
The final step exports the data using the **writeToFile** method. This method writes both x and y values to “saltedData.csv” using the BufferedWriter class. The generated CSV file is stored in the default workspace directory. In Microsoft Excel, the x and y values are stored into columns A and B of the spreadsheet.

Opening the “saltedData.csv” with Excel and using its charting tools, a scatter plot graph is generated. This graph displays a pattern of points that, while scattered, still hint at the original function's form. The graph's appearance changes based on the salting parameter, showing how the random adjustments to the y-values impact the overall data representation. Below is a screenshot displaying the Excel graph with the salted data points.



The result of y = x2 after 80 runs and randomly adjusting the y values between 0 and 250.

Here are additional examples of graphs created with different functions, number of salting iterations, and ranges of random adjustment values.

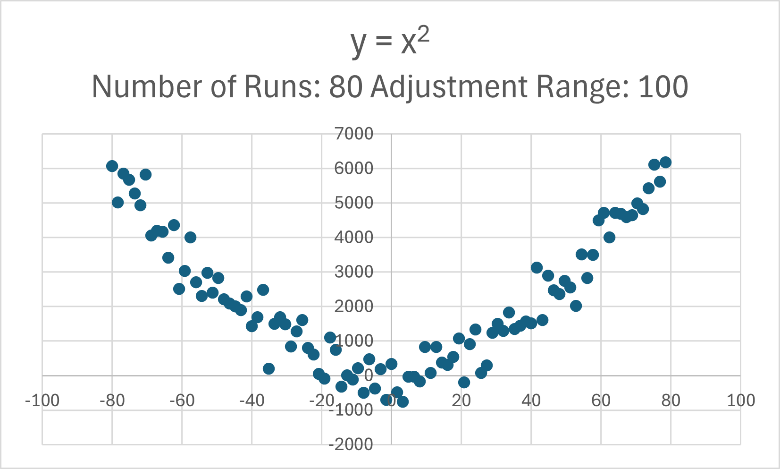


The result of y = x2 after 80 runs and randomly adjusting the y values between 0 and 400.

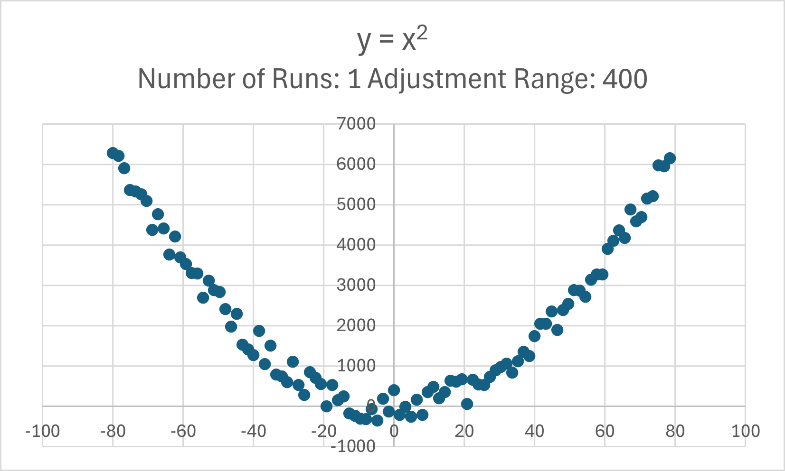
A graph with numbers and dots

Description automatically generated

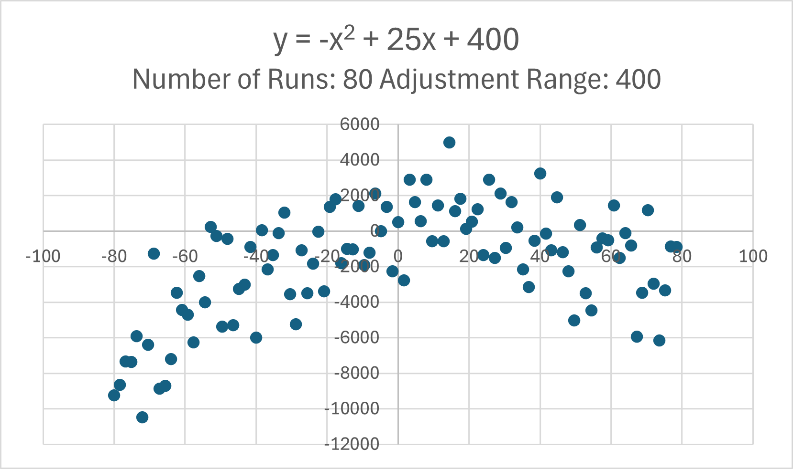
The result of y = x2 after 150 runs and randomly adjusting the y values between 0 and 450.



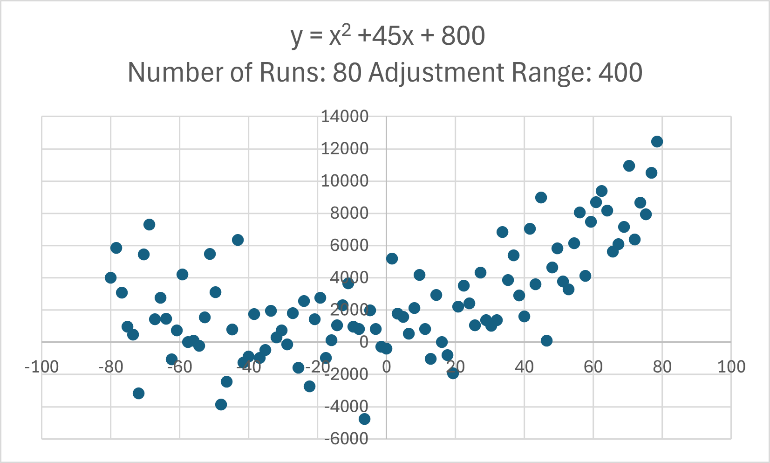
The result of y = x2 after 80 runs and randomly adjusting the y values between 0 and 100.



The result of y = x2 after 1 run and randomly adjusting the y values between 0 and 400.



The result of y = -x2 + 25x + 400 after 80 runs and randomly adjusting the y values between 0 and 400.



The result of y = x2 + 45x + 800 after 80 runs and randomly adjusting the y values between 0 and 400.