Project 1 Report

By: Bryan Naroditskiy

**Overview**:

This project focuses on the analysis of financial datasets, specifically stock price data from the S&P 500 between 2014 and 2017. The primary goals are to analyze stock trends, detect anomalies, and identify periods of maximum gain. To achieve this I used sorting algorithms, subarray analysis, and anomaly detection techniques and applied them to the stock price data.

**Problem Addressed**:

The project addresses the need for efficient analysis of large financial datasets, aiming to identify important trends and anomalies in stock price movement. Detecting these trends can help investors make informed decisions, while anomalies can indicate irregular market activities that need further investigation.

**Diagram:**

A diagram of a financial analysis system

Description automatically generated

**Class Summary**:

* **FinancialData**: Loads and preprocesses the dataset.
  + Methods: get\_close\_prices, get\_dates, get\_dataframe.
* **Sorting**: Implements the merge sort algorithm to sort the stock prices.
  + Methods: merge\_sort.
* **TrendAnalysis**: Identifies periods of maximum gain in the stock data using the maximum subarray algorithm.
  + Methods: max\_subarray, generate\_report.
* **AnomalyDetection**: Detects anomalies in stock prices by computing the Euclidean distance between data points.
  + Methods: euclidean\_distance, closest\_pair, find\_anomalies

**Steps to Use Code:**

1. Loading Data
   * Use the FinancialData class to load the stock price dataset by passing in your csv file with the data
2. Performing Sort on Data
   * Sort the stock prices by first using the get\_close\_prices() method and then running merge\_sort() method on that data
3. Analyze the Trends
   * Apply the max\_subarray() method in the TrendAnalysis class to analyze daily price changes and find the period of maximum gain in the stock prices.
4. Detect Anomalies
   * Use the find\_anomalies() method from the AnomalyDetection class to identify any unusual patterns or irregularities in stock price movements based on Euclidean distance calculations.
5. Generate Report
   * You can create a report that visualizes the period of maximum gain and highlights any anomalies using the generate\_report() method in the TrendAnalysis class.

**Code Output:**

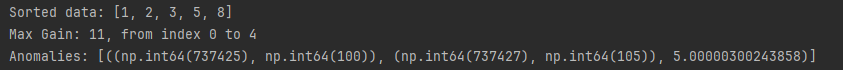
A computer screen with numbers and numbers

Description automatically generated

A screen shot of a graph

Description automatically generated

**Toy examples result:**



**Discussions and Findings:**

The analysis identified a period of maximum gain in the stock prices, highlighting a potentially profitable investment window. Anomalies were also detected, suggesting irregular market behavior or external influences affecting stock movements.

Some of the challenges I faced included efficiently processing large datasets and processing the data correctly for anomaly detection. The current code I have could be refined to better handle financial data patterns, with future improvements potentially incorporating machine learning for more precise predictions.