MG-GY 8411 Data Engineering HW2

Submitted by: Yogya Sharma (ys5250@nyu.edu)

Summary Report: Real-time Forex Data Engineering and Analysis

1. Introduction

This report provides an overview of the methods employed to collect, store, analyze, and export real-time Forex trading data. Utilizing Python for data engineering, we interfaced with the Polygon.io API for data collection, employed MongoDB and SQLite for data storage, performed statistical analysis, and finally exported the analyzed data to a CSV file.

2. Data Collection and Storage

- Polygon.io API Integration: Using the RESTClient from the Polygon.io API, real-time currency conversion data was fetched for the EUR/USD pair.

3. Database Utilization:

- MongoDB: The NoSQL database MongoDB was chosen for its flexibility and scalability to store timestamped Forex rates. A database named "MGGYDE_Spring24" was created with a collection "HW2_forex_aggregates" to hold the Forex data for every 6-minute period.

- SQLite: A relational database management system used for storing structured Forex data, including statistical calculations like maximum, minimum, mean values, volume and fractal dimension for the current 6 min period, in a table named `forex_data`.

4. Data Processing and Analysis
- Statistical Calculations: The application computes real-time statistics, including maximum, minimum, mean rates, and volatility, directly from the collected Forex data in real time continuously for 6 minutes as the data is inserted into MongoDB.

- Keltner Bands Calculation: Utilizing the mean and volatility values, the application calculates Keltner Bands.

- Fractal Dimension (FD) Analysis: A novel approach was introduced to calculate the fractal dimension of the Forex rates, providing insights into the complexity and market behavior over time.

5. Data Export and Finalization

- SQLAlchemy and SQLite3: The final step involved storing the processed data in an SQLite database using SQLAlchemy for ORM support. Subsequently, the data was exported to a CSV file.