Project README: Real-time Finance Data Pipeline

Index

Overview

This project focuses on creating a real-time data pipeline for finance data, enabling interactive querying and analysis. The project involves infrastructure setup, data collection, transformation, analysis, and visualization. The primary technologies used include AWS services such as EC2, Kinesis, S3, Glue, Athena, and Jupyter Notebooks.

Technology Stack

EC2 Application (DataTransformer):

Utilizes the `yfinance` module to collect stock data.

Sends the collected data to a Kinesis Delivery Stream.

Kinesis Stream (DataCollector):

Acts as an intermediary for storing and processing finance data.

Provides a scalable and real-time data streaming solution.

DataAnalyzer (AWS Glue and Athena):

AWS Glue is used to catalog and transform data stored in S3.

Athena allows interactive querying of the transformed data.

Data Visualization (Jupyter Notebook):

`Analysis.ipynb` uses **`results.csv`** to create visualizations.

Visualizations include trends in maximum volatility per company and daily average volatility per company.

Instructions for Recreating Results

Run **`data_transformer.py`** on an EC2 instance to collect and send real-time stock data to Kinesis.

Configure Kinesis Stream and Delivery Stream for data storage.

Set up AWS Glue and Athena to transform and query the collected data.

Use **`Analysis.ipynb`** to generate visualizations from the queried results.

Infrastructure

EC2 Application (DataTransformer):

data_transformer.py gathers actual finance data using the yfinance module.

It pushes data to the Kinesis Delivery Stream with a slight delay between data points.

Kinesis Stream (DataCollector):

Configured to hold finance data collected by the DataTransformer.

Monitored using kinesis_monitor.jpeg in the assets folder.

DataAnalyzer (AWS Glue and Athena):

AWS Glue is configured to point to the S3 Bucket in DataCollector.

Athena is used to interactively query the S3 files generated by DataTransformer.

results.csv and query.txt in the Athena folder contain the query output and the query itself.

Data Transformation (EC2, yfinance)

The **`yfinance`** Python library is used to collect pricing information for e-commerce stocks.

Containerization is employed to install the library and run the program.

Gathers ten days of stock HIGH and LOW prices for each company between Dec 4th, 2023, and Dec 15th, 2023, at a five-minute interval.

Each data point is transformed into a JSON object with high, low, volatility, timestamp (**`ts`**), and company name (**`name`**).

The algorithm sleeps for 0.05 seconds after sending each data point to the Kinesis Stream.

Data Analysis (AWS Glue, Athena)

Configures Glue to point to the S3 Bucket in DataCollector.

Runs a Glue Crawler to enable Athena queries against the collected data.

Athena query calculates average, highest, and lowest volatility per company per day.

Results stored in **`results.csv`**, and the query is in **`query.txt`**.

Data Visualization (Jupyter Notebook)

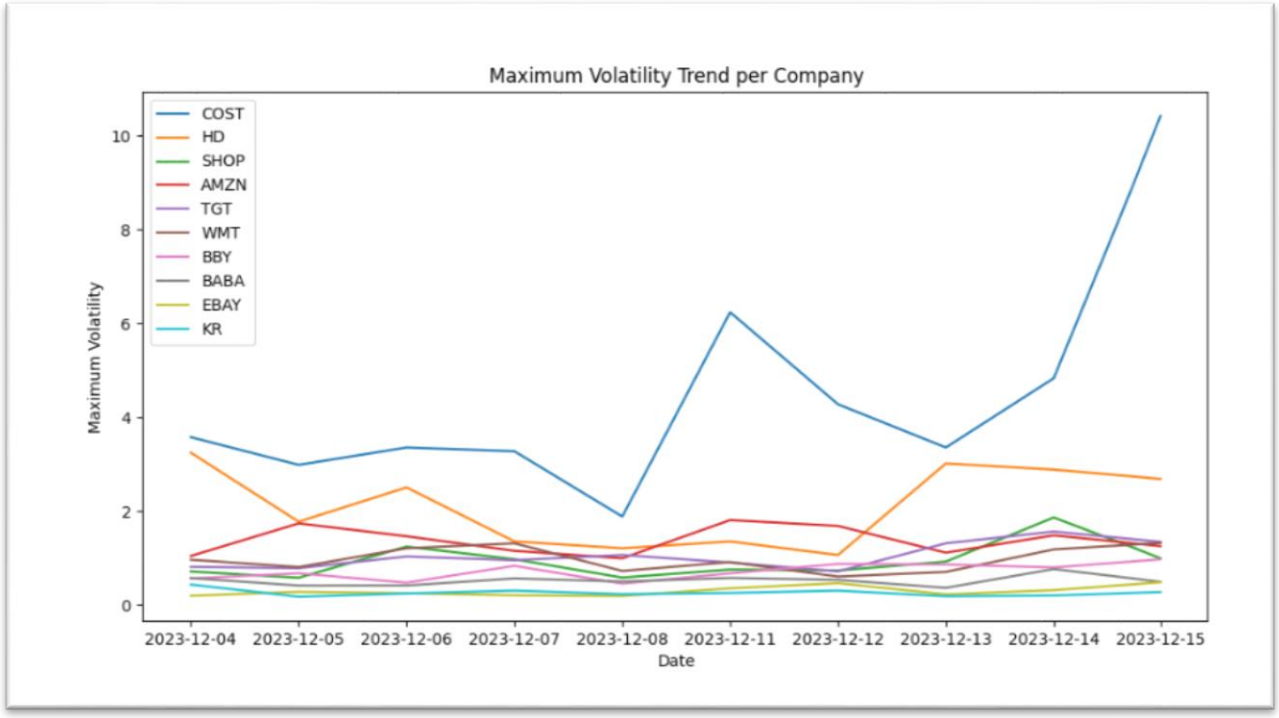
`Analysis.ipynb` generates visualizations using **`results.csv`**.

Maximum Volatility Trend per Company (Line Chart):

Each line represents a company.

Illustrates the maximum volatility trend over the specified time frame.

Screenshot:

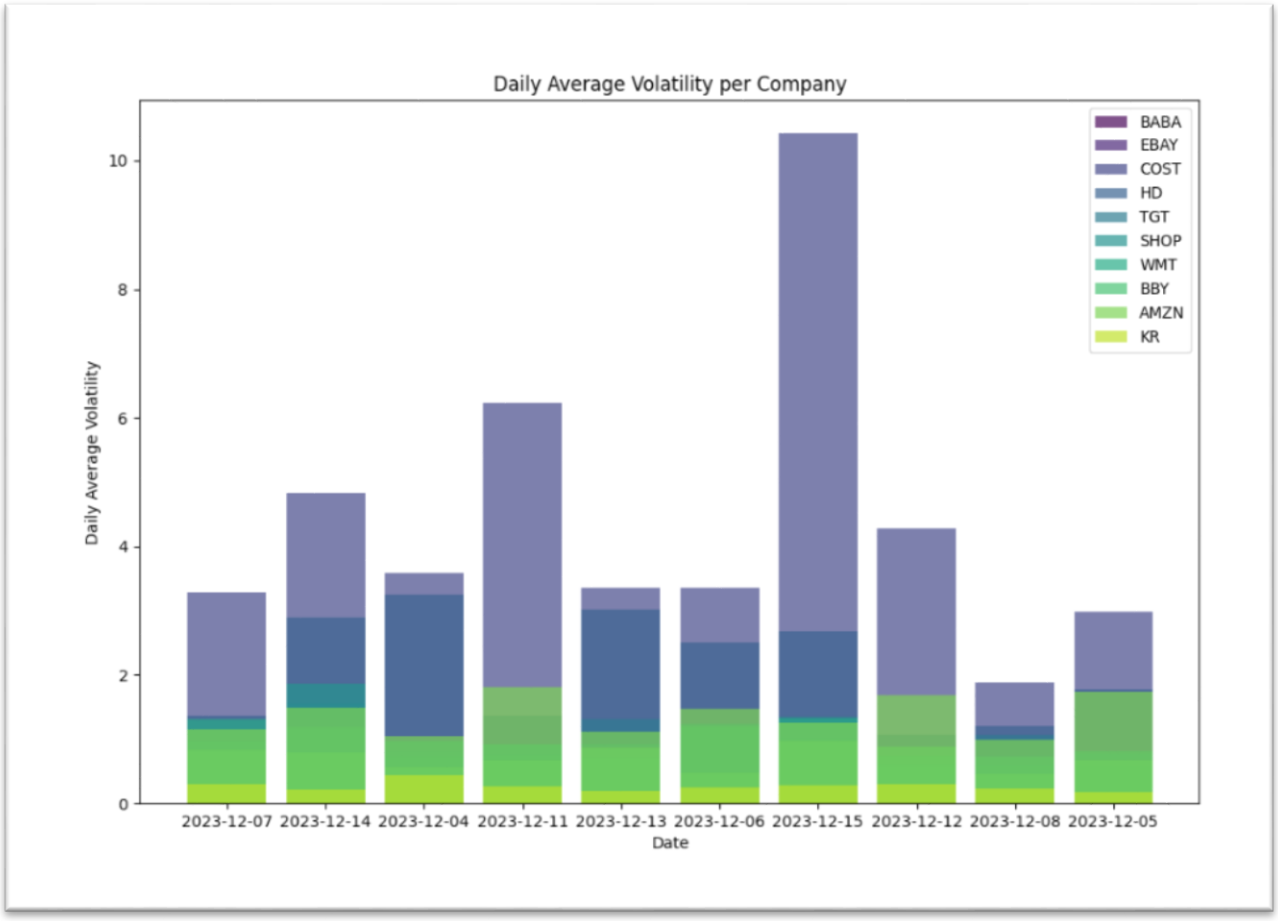


Daily Average Volatility per Company (Grouped Bar Chart):

Each group represents a company, and bars represent daily highest volatility.

Compares daily average volatility trends across different companies.

Screenshot:



Visualization Answers

A] The most volatile company is Costco (COST).

B] Yes! The findings from the second graph supports my conclusion from the first graph that the company Costco has highest volatility.