Data Engineering Project

Historical End of Day Option Price Widget

By William (Wei-Lun) Mai

Abstract

This project explores the solution of presenting historical end-of-day details of an option. The current solutions on the market are all table-based, the charts are clumsy and old-fashioned, or better ones come with a hefty price, such as Bloomberg terminal. The price of an option is dependent on many different variables, such as strike price, expiration date, current stock price, implied volatility, etc. A beginner investor or a student studying options would benefit from this widget since it illustrates the complex relationship between these variables.

Design

First, I implemented a data extraction script to extract the ten most active options traded on NYSE and NASDAQ. The script is run every day after the market closes, and it saves the end-of-day data into a MongoDB atlas database. On GitHub, a streamlit python script retrieves data from MongoDB based on user inputs and presents the insights on the Streamlit web-based platform.

Data

I use EOD Historical Data API to retrieve detailed option data for major US stocks from NYSE and NASDAQ. Each call is for one ticker, and data is from the last trading date. JSON responses from API were later flattened into a pandas DataFrame for plotting and presenting as a table. Each document (row) consists of options of around 20 different expiration dates. There are two different types of options for each expiration date, call and put, and for each type of option, there are from 20 to 50 different strike prices. We have detailed info on the option for each strike price, such as bid price, ask price, trading volume, greeks, etc.

Algorithms

This project currently does not involve any machine learning modeling. The work mainly focuses on how to effectively extract data from multiple levels of nested lists and dictionaries. In the end, the end-users can choose the ticker, and the app will show the available historical trading dates of this ticker, along with the available expiration dates for that specific trading date dynamically.

Tools

- Numpy and Pandas for manipulating data
- Requests and json to work with API
- Pymongo to work with Mongo DB
- MongoDB Atlas to store data on the cloud
- Streamlit for app deployment
- Matplotlib for plotting

Communications

Presentation pdf, write-up, code scripts can be found on this GitHub page:
https://github.com/zyzzyva1423/option_price_widget

Link to App:
https://share.streamlit.io/zyzzyva1423/option_price_widget/main