ALGORITHMIC TRADING USING PYTHON

DSE-309 PROJECT REPORT

Jayesh Kumpawat | Shubhajit Dey
Indian Institute of Science Education and Research, Bhopal

ABSTRACT

This document serves as the project report for the project titled 'Algorithmic trading using python', which was a part of the course DSE309 offered by the Data Science and Engineering Department at IISER Bhopal.

Various sections are introduced in this paper in a structured manner which would help its readers to get an idea regarding what the project is all about. Broadly, in this project ideas to create investing strategies have been fused with the power and features of python to automate the procedure using algorithms.

- Summarising, we have for the project -

USER INPUT : amount of money to be invested
SYSTEM OUTPUT : detailed sheet containing top 50 companies to invest and how much to invest
MEDIUMS : raw data in csv file and randomized market data, requested from API

KEYWORDS

Algorithmic trading, Black Box trading, API

1 INTRODUCTION

Structure of the paper. This paper has been framed in a certain structure considering the crucial factors like readability etc. In this *Introduction* section, a slight taste of all the key aspects of the project has been given. In order to maintain continuity of the literature, the *Background* section has been put next, where a concise self-contained summary of necessary pre-requisites are provided. Following up, next is the *Methodology* section which contains an organized information regarding how finally the project was built. The *Results* section states what was the output from the project and then inferential justifications and motivations for the project are presented under *Conclusions*. Finally, the literature is put to an end by putting *Discussion* section where major ideas are discussed from an creative point of view. This very same section also contains a 'behind-the-scene' subsection where the journey of the project is explained.

Traditionally, until a last couple of years people used to consult financial analysts or advisers in order to make trading decisions. But gradually after the rise of domains like Data Science and Machine Learning, there was a shift from human advisers to computers. If presented in a broad manner, the **algorithmic trading** basically means using computers to make trading decisions. While there are many types of algorithmic trading, their main difference is their speed of execution. In present scenarios many advanced analytical approaches are being developed for algorithmic trading[1].

Also if placed well, algorithmic trading is also called **Black Box Trading**. It uses mathematical models to code programs for automatic trading. These programs have direct access to market and can order trades from market when certain conditions are met. Thus algorithmic trading is a combination of finance , trading , mathematics and computer science. We use historical data to develop the program.

The Algorithmic Trading Process. The process of running a quantitative investing strategy can be further simplified into further steps as:

1. Collecting data.
2. Developing a hypothesis for a strategy.
3. Back testing the developed strategy.
4. Implementing the strategy into production.

! **Note**: Step 4 involves the investing real world money using the strategies developed in the above steps.

Approach. In this project, random data were used. To be precise, the API tokens used were free versions and hence were dependant of random non-real data. At the end, the results were exported in an external excel file for further usage.

Bird's eye view. Summarizing, we can state that in this project two distinct algorithmic trading strategies were developed using Python programming language. Firstly, a quantitative **Momentum Investing Strategy** was built and then emphasis was given to built a quantitative **Value Investing Strategy**.

Contribution at a glance. The project will serve to be a good reference for any person/entity trying to explore usage and applications of Python programming language in the world of finance. Some salient features are as follows:

- The usage of APIs has been done in this project.
- The traditional machine learning approach has not been used actively, and hence more emphasis has been given to various features of Python programming language.
- Results gained further have been exported and stored as external excel files for any further follow-up works.

2 BACKGROUND

Technology has become an asset in finance: financial institutions are now evolving to technology companies rather than only staying occupied with just the financial aspect. The 1960s: Black and white television. Analogue radio. Telephone trades. It was a halcyon period built around human-to-human interactions: an investor with money and a hot hunch would call his broker, who would enter the order into his system. Done deal. The good old days. Or were

Supervised by Dr. Parthiban Srinivasan.

Algorithmic Trading Using Python, DSE309, IISER-B
2021.

they? Then came the fintech disruptors in the new millennium: algorithmic trading, high-frequency trading.

Why use algorithmic trading?

- Remove human error
- Increased opportunity with instant execution
- Can be backtested using available historical and real-time data to see if it is a viable trading strategy.

3 METHODOLOGY

Basically, two strategies are followed for making the investment decisions.

3.1 API Request

The IEX Cloud API Sandbox is used for requesting the data of the stock market. The data is obtained using the *requests* library.

3.2 Quantitative Momentum Strategy

Traders loosely define momentum as the speed at which the markets move. This is correct to some extent, but that's not all and we should certainly not limit our understanding to just that. Simply put, momentum is the rate of change of returns of the stock or the index. If the rate of change of returns is high, then the momentum is considered high and if the rate of change of returns is low, the momentum is considered low. Momentum investing is a trading strategy in which investors buy securities that are rising and sell them when they look to have peaked. The strategy is to filter out highest momentum stocks i.e., stocks which are stable and perform well in long periods of time. High-quality momentum stocks are more preferred because low-quality momentum can often be caused by short-term news that is unlikely to be repeated in the future. Here, after getting the returns (1-year, 6-month, 3-month, 1-month); we calculate their respective percentiles. A score called HQM (**High Quality Momentum**) Score is defined which is the average of all the percentiles. Now, the stocks for 50-highest values of this score is selected and the investment decision is laid out.

3.3 Quantitative Value Strategy

Value investing is an investment strategy that involves picking stocks that appear to be trading for less than their intrinsic or book value. Value investors actively ferret out stocks they think the stock market is underestimating.Quantitative investment strategies have evolved into complex tools with the advent of modern computers but the strategies' roots go back over 80 years. We are going to build an investing strategy that selects the 50 stocks with the best value metrics. And then, we will calculate recommended trades for an equal-weight portfolio of these 50 stocks. Our metrics comprises of the following indicators:

- **Price-to-Earnings(P/E) Ratio** It is the ratio for valuing a company that measures its current share price relative to its earnings per share (EPS). P/E ratios are used by investors and analysts to determine the relative value of a company's shares in an apples-to-apples comparison. It can also be used to compare a company against its own historical record or to compare aggregate markets against one another or over time.
- **Price-To-Book(P/B) Ratio** Companies use this ratio to compare a firm's market capitalization to its book value. It's calculated by dividing the company's stock price per share by its book value per share (BVPS). An asset's book value is equal to its carrying value on the balance sheet, and companies calculate it netting the asset against its accumulated depreciation.
- **Price-to-Sales(P/S) Ratio** It is a valuation ratio that compares a company's stock price to its revenues. It is an indicator of the value that financial markets have placed on each dollar of a company's sales or revenues.
- **Enterprise Value(EV)** It is a measure of a company's total value, often used as a more comprehensive alternative to equity market capitalization. It includes in its calculation the market capitalization of a company but also short-term and long-term debt as well as any cash on the company's balance sheet.
- **EBITDA** Enterprise Value divided by Earnings Before Interest, Taxes, Depreciation, and Amortization (EV/EBITDA)
- **EV/GP** Enterprise Value / Gross Profit

All the listed metrics are obtained using the API and respective percentiles are also calculated. Now, a **Score** is defined as the mean of these percentiles. The stocks are arranged in the ascending order of this 'Score' and first 50 are extracted.

3.4 Quantity of Shares

Using any of the strategies, a list of 50 companies is obtained. Eventually, the quantity of each share is calculated from the user's portfolio size. An order sheet is obtained which is then saved to a .xlsx(Excel) file.

4 RESULTS

Thus, summarizing, we can state that the when any user provides his/her investing amount as the **input**, our algorithm provides the amount distribution that must be invested in the top 50 company stocks out of 500 present in the **SP 500 Index Fund**. These suggestion is based on various factors which further is dependant on the type of strategy used.

5 CONCLUSIONS

After completion of the project, there are some significant conclusions which can be deduced. They are listed as follows:

- Definitely, after the completion of the project it can be inferred that python can be used to optimize the process of strategy making in financial investments.
  - Although the accuracy of the strategies developed in the project cannot be verified due to constraints in using real time data. To be precise, proper APIs were not available for use and hence the Sandbox version was used, which provides randomized data in place of real time data.
- Various other stock parameters can also taken into account in order to improve performance of the algorithms developed.

6 DISCUSSIONS

Takeaways from the project. While performing this project, several extensive literature readings were done. Many aspects and applications of python were explored which gradually led to the know the usability of python to a further extent. Some of them are discussed below:

1. Involvement of API

(1) In this project, as mentioned earlier, the IEX CLOUD API was used to gather stock market information.
(2) The motivation to complete this project led to exploration of python's features to make http requests from the environment to the API used.
(3) In the same clause, usability and features of **requests** module was explored.

2. File handling with Python

(1) In the scope of this project, a component was to export the results obtained in highly usable format.
(2) Involvement of tabular data as outputs implied the file format for exporting the result must be excel.
(3) This further led to the extensive exploration of various features of the **xlsxwriter** module.

3. Scientific computing with python

(1) Evidently, a lot of scientific computing aspects were involved in this project. Majorly, one of them was to deal with tabular data and perform various mathematical operations on them.
(2) For such case, **pandas** module came to rescue with its Data Frames and Series.
(3) **Numpy** helped by providing speedy solutions to complex mathematical operations.
(4) Gradually, **scipy** module was also used in the project.

Challenges. This project avoided complex data science methodologies like involvement of artificial intelligence component, machine learning algorithms etc., which are very common these days in any generic data science project. This was done to focus more on applications of **python** programming language and hence this project made the spotlight fall only on the extensive features of python. Even being less complex in nature, many hindrances were encountered while shaping this project. Some of them are discussed as follows:

- An effort to make the project India-centric was made. This would have opened some further research opportunities for algorithmic trading based on Indian stock market.
- To be precise, it was decided to use **KiteConnect API** as it would provide access to Indian stock data. But the same API is available in exchange of payment and hence it could not be used in the project.
- Eventually, the Sandbox version of **IEX Cloud API** was used to serve the purpose.

Follow-up. Thus, if the project is followed up then the major next step would be to develop algorithms and investing strategies based on Indian stocks and also to use real-time data so the that strategies/algorithms developed could be actually tested in the market.

ACKNOWLEDGEMENT

This project was supported by Dr. Parthiban Srinivasan from the Department of Data Science and Engineering, Indian Institute of Science Education and Research-Bhopal.

Initial insights to start the project were gathered through a thorough literature survey of the blog by **Datacamp.org**[2]. Youtube preparatory videos by **freecodecamp.org** were also of great value as they helped to gathered crucial knowledge to proceed further in the project work.

Further, some solid projects which were investigated and explored during moving the project ahead are listed as follows:

- **pyalgotrade**
- **ks-orderapi-python**
  * hyperlinked GitHub repositories.

REFERENCES

[1] Sarika K B, Sreekumar R, and Shilja M S. 2013. An Analytical Approach for Algotrading. *IJSER International Journal of Scientific & Engineering Research* 4, 6 (2013), 13–16. https://www.ijser.org/researchpaper/An-Analytical-Approach-for-Algotrading.pdf

[2] Datacamp.org. 2019. Python For Finance: Algorithmic Trading. https://www.datacamp.com/community/tutorials/finance-python-trading?utm_source=adwords_ppc&utm_medium=cpc&utm_campaignid=12492439802&utm_adgroupid=122563403721&utm_device=c&utm_keyword=python%20finance&utm_matchtype=b&utm_network=g&utm_adpostion=&utm_creative=504158804782&utm_targetid=aud-299261629654:kwd-301111479313&utm_loc_interest_ms=&utm_loc_physical_ms=1007745&gclid=Cj0KCQiAkZKNBhDiARIsAPsk0Wj8zBR5zmi1izn55Yu0f1nmIX3lvgbrHZwoFp25kW_d1K7OQWhW_YkaAl__EALw_wcB