**Stock Selection with a Simple Multi-factor Model**

Group J

MB955066, Wu Chan Keong

MB955162, Zeng Jingyi

MB955093, Chen Xi

**1. Topic Introduction**

A multi-factor model is a financial model and it can be used to explain a portfolio of securities. It does so by comparing two or more factors to analyze relationships between variables and the resulting performance. Multi-factor model is an important stock selection model. Because it can consider many aspects of information, its performance is also relatively stable. Even in different market conditions, there are some factors playing roles.

In this project, we used a simple multi-factor model to select stocks on the Hong Kong Stock Exchange. For the sake of simplicity, we ignored the fees in the transaction processes and dividends paid to stockholders. In general, the multi-factor model examines the stock's performance from various aspects, including value factors, growth factors, size factors, momentum factors and volatility factors, etc. In this project, we selected price-to-book ratio (P/B ratio) as value factor, shares outstanding as size factor and earnings quarterly growth as growth factor to select stocks. Then, we chose the best stocks in term of valuation, size and growth in each sector as our portfolio, and each factor was given the same weight. Finally, we tested the performance of this portfolio over the past five years. Once the back testing worked, we may have the confidence to employ it going forward.

**2.Python Library**

We mainly used six libraries: numpy, pandas, yfinance, threading, logging and matplotlib.

The pandas and numpy are used for storing, cleansing, and analyzing the financial records. We can create a DataFrame by loading the datasets from existing CSV file, select best performers by sorting by columns and indexing data, clean data by deleting null values and changing data type, etc.

The yfinance is used for downloading financial historical data from Yahoo Finance. We can download a lot of data directly, such as stock information, historical market data, financial data such as balance sheet, cashflow, and so on.

The threading is used to handle concurrent processing of downloading data with yfinance to boost performance. Using threading to build up program provides design clarity, and it also helps to make the design cleaner and easier to reason about.

The logging is used to record long processes progress into a logfile. The key benefit of having the logging API is that all Python modules can participate in logging, so our application log can include our own messages integrated with messages from third-party modules.

The matplotlib is used to plot the data on the graph, that is, visualization of data. In this project, we use matplotlib to show the returns of our portfolio over the past five years comparing with the Hang Seng Index(HSI).

**3. Data Loading and Cleaning**

We downloaded a list with information of listed companies from Hong Kong Stock Exchange website. And then we filtered the list to keep only the stock symbols of those companies and saved it as csv file.

After that, we used pandas to read the csv file of symbols and downloaded stocks information using yfinance. The information was accumulated to a native Python list of dictionaries. And we converted the list into pandas DataFrame for portfolio selection.

On account of the large amount of data size, we divided several threads for downloading synchronously in the background. Meanwhile, we made use of logging to track the execution of each thread, collected all status information and saved it as a logfile for follow-up review and debugging.

According to the needs of model factors, we selected related columns and used dropna to keep valid data.

**4. Data Processing and Analysis**

A portfolio is a combination of financial assets such as stocks, bonds, etc. Diversification is a process to allocate capital in a way that reduces exposure to risk. With portfolio, investors could achieve diversification and reduce risks.

We selected three factors to construct our portfolio. The first one was P/B ratio, which was a financial ratio used to compare a company's current market price to its book value; the second one was earnings quarterly growth, which measured the increase in a firm's sales from one quarter to another; the third one was shares outstanding, which referred to a company's stock currently held by all its shareholders, including share blocks held by institutional investors and restricted shares owned by the company's officers and insiders. We chose stocks with high P/B ratio, high earnings quarterly growth and high shares outstanding. Usually, high P/B ratio represents that the quality of corporates’ assets is better and corporates have development potentials; high earnings quarterly growth shows that companies we chose are promising; high shares outstanding gives the sign that the company has strong operating capital capacity. And each factor was given the same weight and then we rated each stock. We sorted these scores with descending order, and chose the top 1 stock, group by sector to find 11 stocks from different sectors. Then we used Price-weighted method (add each stock price together and divided by the number of stocks) to put weight for each stock. Finally, with price-weighted, we constructed our recommended portfolio with a real-time market information.

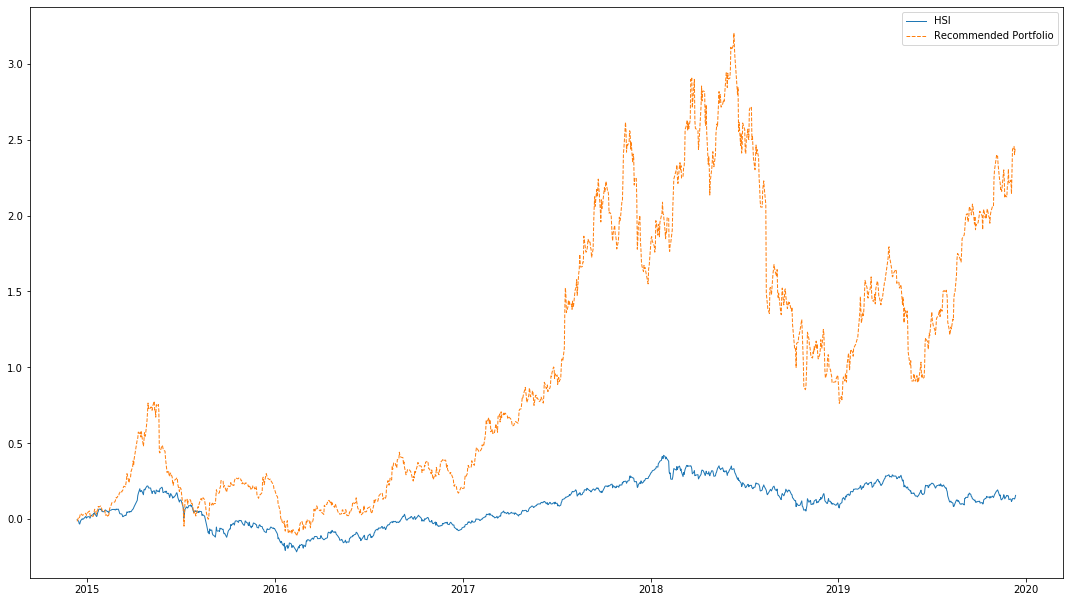
**5. Conclusion**

After using real-time data to construct a portfolio, we did back testing to see whether our choice gave a better result than HSI. We made the following assumptions on the back testing model:

       1. Back testing period: from 12/12/2014 to 12/12/2019;

       2. Each stock was given the same weight.

We used matplotlib to draw a line chart for comparing the rate of returns of the portfolio and HSI for five years, the performance of our portfolio during the back testing period was as follows:



As we could see from the graph, the performance of our portfolio did much better than HSI. Especially, the return of the portfolio had been much higher than that of HSI since 2017. Therefore, we could say that our multi-factor stock selection model is a good model.

**6. Improvement**

In general, our model was simple and easy to operate. And it had performed well in the back testing for the past five years, indicating that the three factors we selected were reliable in the Hong Kong stock market during the period of back testing. So we were also more confident to use this model for selecting stocks. But this model still had a lot of room for improvement.

Firstly, due to the memory space of our computers and the complexity of portfolio composition in practice, we could not build a sophisticated portfolio with frequent updates of stock compositions. Secondly, the performance results we got were based on history, we could not predict the future with certainty. The portfolio could just worked as a reference for investment. Thirdly, we chose factors subjectively, the proper way was to use the scoring method or regression method for factors selection, which was much more difficult and needed much larger memory space to run the code.