# Design and Implementation of Machine Learning Based Multi Factor Quantitative Trading Strategy

Yu Zhang(✉)

School of Computer Science, Beijing University of Posts and Telecommunications, Beijing 100876, China
OctieZhang@bupt.edu.cn

**Abstract.** Quantitative trading is a trading method that combines finance, mathematics, and computer science to achieve a goal. This method can help investors to filter out negative emotional influences effectively so that it is becoming more and more widely used in the Chinese stock market. Traditional quantitative trading strategies predict the trend of stock prices by analyzing fundamental indicators or technical indicators and building formulas quantitatively. However, this paper will use the emerging machine learning technologies to analyze the influence of multiple factors which impacts the stock price, then predict the return of particular stocks and stress the trading strategy.

This research's main work consists of obtaining financial data from third-party platforms and defining indicators; using Support Vector Machine, Random Forest, and XGBoost machine learning algorithms to build prediction models to predict which stock can bring excess return; generating the stock holding list; and designing the trading strategy accordingly. The outcomes are a multiple factors quantitative trading strategy based on machine learning which brings a steady excess return ratio while bearing low risk. The research achievement solves some problems in the current quantitative trading strategy: the selection of indicators is biased, a single machine learning model is not effective through a long period of time, the process of strategy research is not convenient enough.

**Keywords:** Quantitative trading · Machine learning · Stock indicators · Strategy optimization

## 1 Introduction

Considering the stock market investment, traditional methods are evaluating the company's intrinsic value thoroughly, which will consume lots of research and social time. Also, these methods are pretty subjective and may be negatively affected by the investor's emotions. That's why quantitative trading methods appear. Quant trading methods gain a comprehensive utilization of mathematics, finance and computer technologies, which can filter out emotional influences effectively. There are two main streams of quant trading methods: quantitative stock selection and quantitative trading moment selection.

F. Balli et al. (Eds.): ESFCT 2022, AEBMR 226, pp. 977–984, 2022.
https://doi.org/10.2991/978-94-6463-052-7_111

However, due to the Securities Law of The People's Republic of China, the latter one is restricted by the T+1 legislation [1]. This paper will mainly focus on the quantitative stock selection.

Factor means the indicator to identify the trend of stock price, it can be roughly divided into the fundamental factors which represent the intrinsic value of the company and technical factors which are mainly calculated by the stock price data and represent the momentum. If we consider multiple factors while doing the analysis, then we call it multi factor quantitative trading analysis. However, traditional multi factor quantitative trading analysis methods are not efficient enough to take more than ten factors, not to mention their weakness in dealing with non-linear problems. The power of machine learning enables investors to consider tens of indicators at one time. Furthermore, investors can find effective factors which indicate the stock price faster, which is a considerable strength.

The current quant trading methods utilizing machine learning are not perfect. There are still some problems that can be optimized. Firstly, the selection of indicators is biased, most researchers implement their experiments based on either fundamental factors or technical factors. Secondly, one single machine learning model is not effective for a long period. Due to the fast-moving stock market data, it is nearly impossible to expect one static model to predict the trend with a pretty accuracy in the long run. Lastly, the process of strategy research is not convenient enough. When we implement the experiments, it is hard to configure the strategy immediately, so this research also defines a group of configurable parameters.

## 2 Relevant Technics

### 2.1 SVM

Support Vector Machine (SVM) is a machine learning algorithm which solves classification problems. The main idea of the model is to find the best hyperplane which separates the data sample into different labels. This research focuses on the stock classification problem, so the classifier is called Support Vector Classifier (SVC). The hyperplane in N-dimension can be represented by:

$$y = w_0 + w_1x_1 + w_2x_2 + \cdots = b + w^T X \tag{1}$$

The key hyperparameter that needs to be tuned is the kernel, gamma, and C. The kernel, which is the most important hyperparameter, indicates the different kernel functions used for SVC. The gamma controls the sensitivity between two classes. If the gamma is big, then two points need to stay really close to be identified as the same class. The C tells SVC how much to ignore the misclassification. If the C is big, then every misclassified sample will significantly influence the margin.

SVC's strengths are: It is efficient in the high dimension where is easier to find the best hyperplane. It is not influenced by the outliers seriously since the margin is dependent on support vectors.

### 2.2 Ensemble Learning

The decision is formed by ensemble learning by combining several weak classifiers [2]. Generally speaking, ensemble learning can outperform weak classifiers with smaller deviation and variance. Two main stream of ensemble learning is Bagging (parallelized) and Boosting (serialized). In this research, it uses both Random Forest (Bagging) and XGBoost (Boosting). They both use a decision tree as the weak classifier.

Due to the random feature selection process, the random forest can still perform a nice prediction when lacking some features' data. XGBoost has decision tree pruning and special hardware level optimization, which accelerated the speed of training.

### 2.3 Data Source and Back Test Platform

Financial data is the heart of this research, this paper will acquire data from JQData and TuShare. JQdata is developed by the JoinQuant team, who focuses on providing local quantitative financial data services for financial institutions, academic groups and individual quantitative finance investors. TuShare is an open platform, it uses high speed database to provide high quality data to users with low cost. Technical indicators will be calculated by using NumPy. It is a Python package that contains various practical science calculation functions.

### 2.4 Quant Trading Method Review

Traditional multi factor quantitative trading methods track the value of several indicators, and buy in stocks when the correlated indicators emit buy signals. One of the most famous strategies is Fama-French three factor strategy, which tracks the market excess return, the outperformance of small verses big companies, outperformance of high book versus low book company [3]. These methods suffer from the laborious indicator exploration process not to mention the indicator will change frequently due to the fast-moving nature of financial markets.

Different from traditional methods, quantitative trading strategies utilize the machine learning technologies which unleash the power of the computer. The process of excavating indicators is easier and faster. People can also consider more factors at one time, solving the nonlinear problems reasonably. With the development of machine learning, new strategies are also on the rise. Van-Dai Ta and his team used LSTM neural network to analyze the financial data of S&P 500 index constituent stocks and predicted the trend successfully [4].

## 3 Strategy Forming Process

In order to make the whole process coordinate with the data flow, designing a layer model is a preferred way. In this research, layers are as follows: data acquisition, data cleaning, modeling, strategy forming and back testing. These layers make up the whole process of strategy generating, and allowed each layer remain transparent with each other.

**Table 1.** Indicators Selected

| Indicator Type | Specified Type | Indicator Name |
|---|---|---|
| Fundamental | Volume | Market Value |
| | | Circulation Market Value |
| | Emotion | Turnover Rate |
| | Value | Price Earnings Ratio |
| | | Price to Book Ratio |
| | Operation | Net Profit Margin |
| | | ROE |
| | | Month on Month Net Profit Growth |
| Technical | Momentum | Rate of Change (ROC) |
| | | Relative Strength Index (RSI) |
| | | Moving Average (MA) |
| | | Money Flow Index (MFI) |

### 3.1 Data Acquisition

As the saying goes, Garbage in, Garbage out, the effectiveness and accuracy of input data dramatically influences the performance [5]. R.Rosilo and J.Giner selected Bollinger Band, RSI, MACD, Momentum as indicators to analyze the stock price in the US market [6]. Their research in 2015 mainly focused on technical indicators. However, Weinan Zhang, Tongyu Lu, Jianming Sun used operation indicators such as market value and ROE to analyze the trend in the Chinese stock market. Their research is bundled with fundamental indicators [7]. In this paper, the research aimed to utilize both fundamental and technical indicators. The following table shows the indicators selected from various sectors (Table 1).

### 3.2 Data Cleaning

After gathering all the data, make a table for each trading date. Every row represents a stock in CSI300, every column stores the value of one indicator. For each table, apply the data cleaning process: Use the average value of Shenwan industry classification to impute the missing value. Then apply the winsorize method to get rid of outliers by setting the upper bond to two variances above average [8]. Finally use a z-score to standardize the data, enabling different columns comparable.

We also need to define the class label. For every stock, define Yield as the future return if the investors buy the stock today and sell the stock on the next position adjusting date. The formula is as follows:

$$\mathrm{Yield_i} = (\mathrm{close_{i+1}} - \mathrm{close_i})/\mathrm{close_i} \tag{2}$$

close stands for the close price of the particular stock on date i. In order to better separate the different classes, the research sorted descending by Yield. Set the top 30% stocks (which is configurable, such as top 20% or 40%) as class 1, which make more money over the period. Set the bottom 30% as class 0. To wrap up, we got a table for every trading date between March 15, 2021 and March 22, 2022. Each table has 90 (30% of CSI300 constituent stocks) rows labeled as 1 and 90 labeled as 0.

### 3.3 Modeling

Financial markets are moving fast, it is impossible to apply one single model throughout the time. In this research, the research made up a model group, which contains SVC, random forest, XGBoost. For every position adjusting day, these three models will go through the training process: read in the training set, tune hyperparameters. The Grid-searchCV tool is useful which helps researcher to tune the models to the best in an efficient way. After tunning, choose the model with the highest AUC value and predict stock performance in the test set.

For each position adjusting date, the training set is all the data from the past half-year (which is configurable, it can be changed to past one year), while the test set is the table for the position adjusting day. It works as a sliding window. Every stock in the test sets will be predicted as either class 1 or 0. Also, using predict_proba function to figure out the probability of being class 1 as well. With the probability, select the top 10 stocks (which is configurable, it depends on how many stocks investors aim to hold at one time) which most likely to be class 1, add it to the buy list.

Take the training result for August 25, 2021 as an example, the AUC value for every model in the model group are as follows (Table 2):

As XGBoost has the most significant AUC value, so for the training set of August 25, 2021, the strategy should use XGBoost to make a prediction. The process for other trading days is similar, but the model with the highest AUC value may vary. SVC and random forest will perform well in some specific period.

### 3.4 Strategy Forming and Back Testing

The buy list generated in Sect. 3.3 indicates which stock to buy in on a specific trading date. The main idea of the strategy is buying and selling stocks according to the buy list. From August 23, 2021 to January 28, 2022, every ten trading days, which lasts approximately two weeks, is defined as a position adjusting day. On the adjusting day, Buy the 20 stocks, or keep them which appear in the buy list and sell those stocks which are not. Also, in order to minimize the risk, this research uses dynamic stop-loss. If the

**Table 2.** AUC for SVC, RF and XGBoost

| Model | AUC Before Tuning | AUC After Tuning |
|---|---|---|
| SVC | 0.7672 | 0.7697 |
| Random Forest | 0.7412 | 0.7623 |
| XGBoost | 0.7395 | 0.7700 |

current stock price is lower than the highest price during holding times one minus the stop-loss rate (which is 5% in this instance), the strategy will trigger a sell signal [9].

Lastly, this research uses JoinQuant platform to back test the strategy. Their engine is super secure and can generate the back test report for the given condition. This research will focus on the following metrics: benchmark yield (CSI300 index), total annualized returns, excess returns, and max drawdown.

## 4 Back Test Result

Back test results in this research are based on the following conditions: the back test period is from August 23, 2021 to January 28, 2022. CSI300 Index is set as a benchmark. The initial capital is 100000 Chinese Yuan. Reading the financial report provided by JoinQuant platform, it is convenient to track strategy performance over time. The metrics for the quantitative trading strategy before and after adding stop-loss optimizing algorithm are as below (Table 3):

As with the metrics shown in the table, the stop-loss algorithm effectively increased the excess return and decreased the max drawdown, which means the stop-loss algorithm can help reduce risk and bring more risk.

In order to better illustrate the performance of this strategy, the research compared these metrics with the classic Double Moving Average Strategy [10] and a common benchmark strategy – Random Selection Strategy, which randomly buy in stocks on the position adjusting day. The results are as follows (Table 4):

**Table 3.** Strategy Back Test Result Before and After Optimization

| Metrics | After Optimization | Before Optimization |
|---|---|---|
| Total Annualized Return | 22.97% | 18.54% |
| Benchmark Yield | −4.31% | −4.31% |
| Excess Return | 14.17% | 12.40% |
| Max Drawdown | 2.73% | 4.42% |

**Table 4.** Strategy Back Test Result Comparison

| Metrics | Multi Factor Machine Learning Strategy | Double Moving Average Strategy | Random selection Strategy |
|---|---|---|---|
| Total Annualized Return | 22.97% | −5.66% | −7.94% |
| Benchmark Yield | −4.31% | −4.31% | −4.31% |
| Excess Return | 14.17% | 1.93% | 0.87% |
| Max Drawdown | 2.73% | 11.16% | 7.63% |

The results show the effectiveness of the multi factor machine learning based strategy. It has a higher annualized return and lowers max drawdown compared with other strategies.

## 5 Conclusion

The research in this paper mainly focuses on machine learning-based multi-factor quantitative trading strategies. Through the data acquisition, data cleaning, modeling, strategy forming and back testing layers, the research finally formed an effective strategy. It uses the power of SVC, random forest and XGBoost algorithms to enable researchers and investors quickly track and predict the trend of the Chinese stock market. However, the research is not perfect yet. It yields an effective quantitative trading strategy in the back test environment which focuses on the past. But it still needs months of mock trading which will go through the future, to ensure the effectiveness. Also, the trading strategy should be wrapped by an automatic trading system, which helps investors implement the strategy to the stock market easier.

**Acknowledgments.** In the end, I want to thank my instructor Ke Xu, who gave me a lot of practical advice during the research. He also supported me by sharing his experience in both academic and industry field. Also, I would like to appreciate Beijing University of Posts and Telecommunications, which provides me a wonderful place to complete my research. The JoinQuant platform and TuShare data source have done their job incredibly well. They provide a ton of accurate financial data as well as the useful strategy back test tool, which really saves the time. Last but not least, some of my colleagues at Chinese International Capital Corporation also shared their insights about quantitative finance which broadened my eyesight.

## References

1. Hongbing Zhu, Bing Zhang. The Influence of T+1 Trading System on China's Stock Market. Securities Market Herald. F63. 2020, (08): 24–32.
2. Lilly Chen. Basic Ensemble Learning (Random Forest, AdaBoost, Gradient Boosting)- Step by Step Explained. https://towardsdatascience.com/basic-ensemble-learning-random-forest-adaboost-gradient-boosting-step-by-step-explained-95d49d1e2725.
3. Connor, Gregory and Sehgal, Sanjay (2001) Tests of the Fama and French model in India. Discussion paper (379). Financial Markets Group, London School of Economics and Political Science, London, UK.
4. Van-Dai Ta, CHUAN-MING Liu, Direselign Addis Tadesse. Portfolio Optimization-Based Stock Prediction Using Long-Short Term Memory Network in Quantitative Trading. Applied Science. 2020, 10, 437.
5. L. Todd Rose EdD, Kurt W. Fischer PhD. Garbage In, Garbage Out: Having Useful Data Is Everything. Measurement: Interdisciplinary Research and Perspectives. Volume 9, 2011 - Issue 4: 222–226.
6. Rosillo R., Giner J., De la fuente D., etc. Trading System Based on Support Vector Machines in the S&P500 Index. The Steering Committee of The World Congress in Computer Science, Computer Engineering and Applied Computing (WorldComp). (2012): 1–5.

7. Weinan Zhang, Tongyu Lu, Jianming Sun. Prediction optimization of support vector machine in multi-factor stock selection[J]. Application of Electronic Technique. 2019, 45(9): 22-27.
8. R. Wilcox. Encyclopedia of Biostatistics. 2nd Edition. John Wiley & Sons, Ltd. 2005.
9. Shelton, A. The value of stop-loss, stop-gain strategies in dynamic asset allocation. J Asset Manag 18, 124–143 (2017). https://doi.org/10.1057/s41260-016-0010-y.
10. Gurrib, I. Optimization of the Double Crossover Strategy for the S&P500 Market Index. Global Review of Accounting and Finance. Vol. 7. No. 1. 2016, 3. Pages: 92–107.