Final Report

Shengyang Luo Zichun Wang

Our report consists of two parts. The first part is Direction Determination part, which gives us a model to determine the direction of movement of stock price.

The second part is Trade Strategy part, which introduces and studies a quantitative method to direct us on how to trade.

(I) Direction Determination part

Zichun Wang

1: Using artificial neural network models

In our experiments, the time series data acquired is applied with exponentially smoothing at the first step. Then we need to extract the indicators. Technical indicators cause expected stock price behavior in future. These technical indicators are used as features to train the machine learning classifiers.

2: Data Preprocessing:

Exponential smoothing grants larger weights to the recent observations and exponentially decreases weights of the past observations. The exponentially smoothed statistic of a series Y can be recursively calculated as:

$$\begin{aligned} S_0 &= Y_0 \\ \text{for} \quad t > 0, \ S_t &= \alpha * Y_t + (1 - \alpha) * S_{t-1} \end{aligned}$$

where alpha is the smoothing factor and 0<alpha<1. Larger values of alpha reduce the level of smoothing. When alpha=1, the smoothed statistic becomes equal to the actual observation. The values of alpha larger, the more the level of smoothing reduced. When alpha=1, the smoothed statistic becomes equal to the actual observation. The smoothed statistic St can be calculated as soon as consecutive observations are available. The smoothing removes random variation or noise from the historical data, which allows the model to easily figure out the long-term price trend from observed stock price behaviors. Technical indicators are then calculated from the exponentially smoothed time series data which will be organized and integrated into a feature matrix. The target to be predicted in the ith day is calculated as follows:

$$target_i = sign(close_{i+d} - close_i)$$

where d is the number of days after which the prediction is to be made. When the value of target is +1, it indicates that there is a positive shift in the price after d days; -1 indicates that there is a negative shift after d days, giving us an idea of the direction of the prices for the respective stock. We assign the target i values as labels to the i_th row in the feature matrix.

3: Feature extraction from the data

In our solution, we consider only the closing price of a stock and we collect these values for many years. Hence, our input data can be considered to be the form (data, closing price). First, I give a brief introduction of the original data here:

Home Depot, JPMorgan Chase&Co., IBM, ARWR, COST and S&P 500, from October 2nd, 2019 to October 2nd, 2020. We will retrieve: High price, low price, adjusted close price and the volume. For the sake of the convenience and the consistency, we will use the daily data from the yahoo finance.

Relative strength index (RSI): RSI is a popular momentum indicator which decides whether the stock is bought or sold overly. A stock is said to be overbought when the demand unjustifiably pushes the price upwards. This condition is generally interpreted as a sign that the stock is overvalued, and the price is likely to go down. A stock is said to be oversold when the price goes down sharply to a level below its true value. This is a result caused due to panic selling. RSI ranges from 0 to 100 and generally, when RSI is above 70, it may indicate that the stock is overbought and when RSI is below 30, it may indicate the stock is oversold.

Stochastic oscillator (SO): Stochastic Oscillator follows the momentum of the price. As a rule, momentum changes before the price changes. It measures the level of the closing price relative to low-high range over a specified period of time.

Williams percentage range(W%R): Williams Percentage Range or Williams %R is another momentum indicator similar in idea to stochastic oscillator. The Williams %R indicates the level of market's closing price in relation to the highest price for the look-back period, which is 14 days. It's value ranges from −100 to 0. When its value is above −20, it indicates a sell signal and when its value is below −80, it indicates a buy signal.

Price rate of change (PROC): The Price Rate of Change (PROC), (Larson, 2015) is a technical indicator which reflects the percentage change in price between the current price and the price over the window that we consider to be the time period of observation.

On balance volume (OBV): OBV utilizes changes in volume to estimate changes in stock prices. We use this technical indicator to find buying and selling trends of a stock, by considering the cumulative volume: it cumulatively adds the volumes on days when prices go up and subtracts the value on the days when prices go down, compared to the price of the previous day.

Machine learning algorithms

In general, here, we mainly use two methods:

Decision trees and random forests. Both methods are popular in machine learning. We could use them to solve a large number of problems in classification. Basic structure is the recursive partitioning of the feature space using a tree structure, where each node is split until pure nodes. In other words, it means that we reached nodes within which samples of a single class are contained. The splitting is done by the means of criteria which focuses on how to maximize the purity of the child nodes relative to their respective parent nodes. Subsequently, we will arrive at the pure nodes.

These pure nodes are not split further and constitute the leaf nodes. When a decision tree is used for the classification of a test sample, it is traced all the way down to to a leaf node of the tree; as the leaf nodes of a decision tree are pure, the respective test sample is assigned the class label of the training samples of leaf node it arrives at.

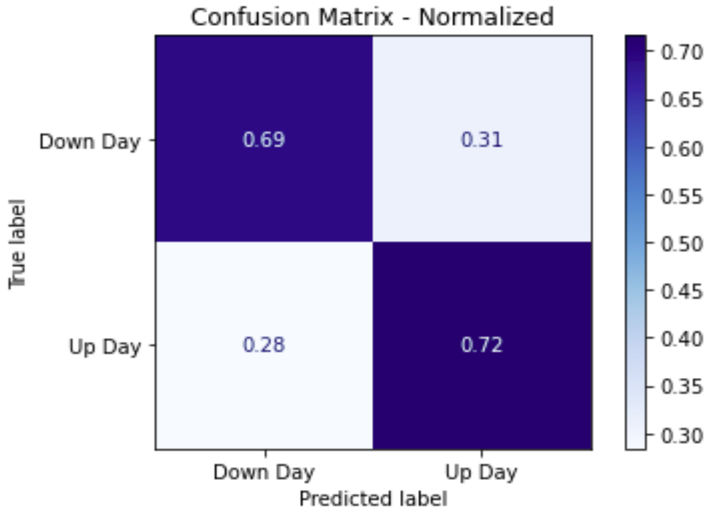
4: Model evaluation:

Confusion Matrix shows us the distribution of true/false positive/negative:

Correct prediction is given by the formula below:

(true positive+true negative)

/(true positive+true negative+false positive+false negative)



Specificity:

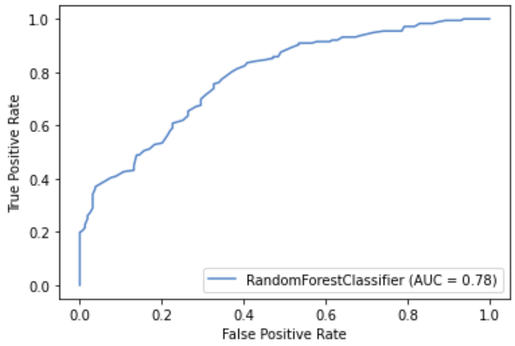
The ability to correct predict negative=true negative/(true negative+true positive)

Feature importance:

Here, we use the gini importance by default.

ROC curve:

A ROC curve (receiver operating characteristic curve) is a graph showing the performance of a classification model at all classification thresholds. This curve plots two parameters: 1: true positive rate 2: false positive rate



Precision

Measures the proportion of all correctly identified samples in a population which are classified as positive labels.

precision=true positive/(true positive+false positive)

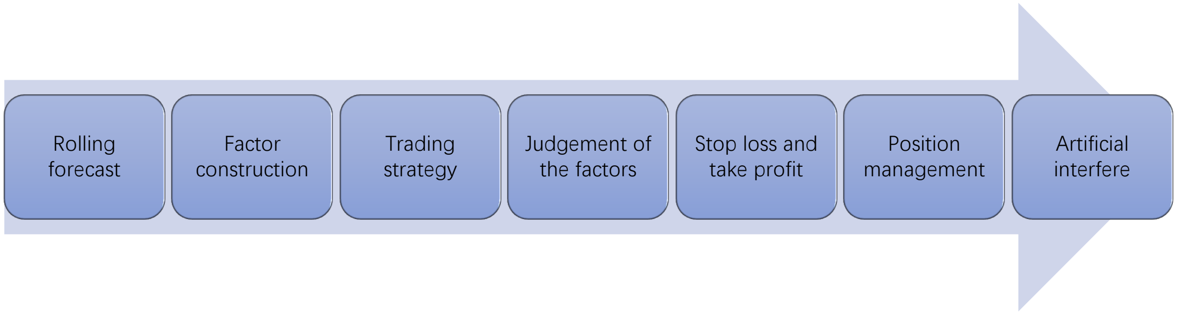
Out-of-bag error score

OOB error, also called out-of-bag estimate, is a method of measuring the prediction error of random forests, boosted decision trees, and other machine learning models utilizing bootstrap aggregating (bagging) to subsample data samples used for training. This is an essential part to evaluate our model.

Trading strategy part

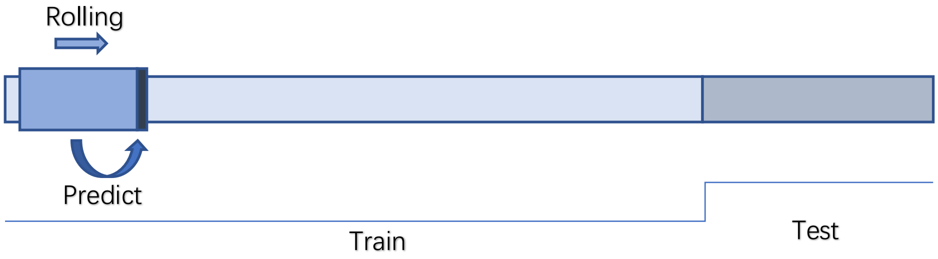
Shengyang Luo

The process of my strategy is as follows:



- Rolling forecast/moving horizon prediction

In our strategy, we divide the data set (containing 1500 data) into two part: train data and test data. And in train data we have 1200 data and 300 data in test part, and I use rolling forecast method that put 200 data into the model and predict one day's result, and then move the time zone. That is to say, the total train set will generate 1000 train results and we can get 300 test results.



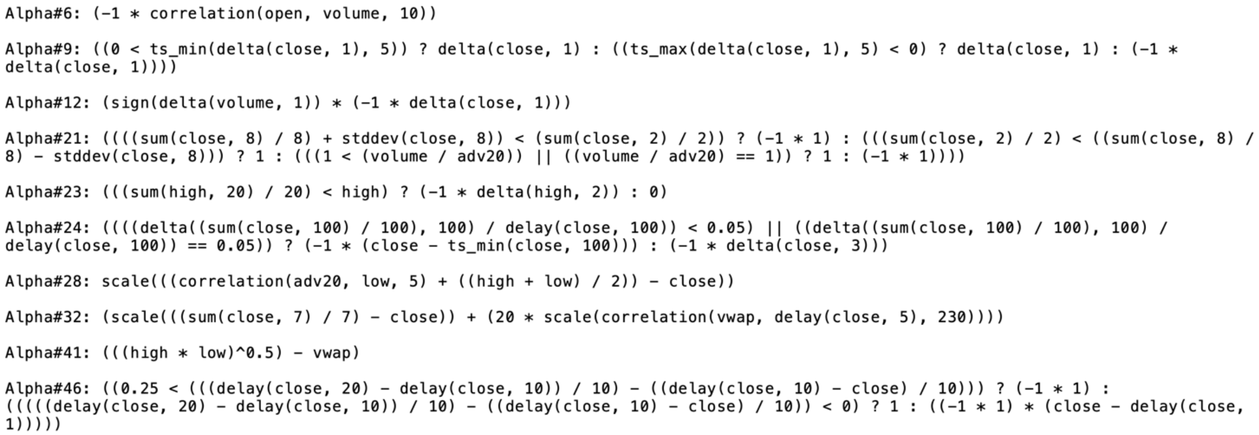
We split the data set into two part mainly to have more possibility that the strategy goes well in the long run. We may get huge profit in the train set but get huge loss in the test set, and this will lead to uncertainty of loss in the future. So the strategy is that we first run the program in the train set and if it reaches the standard we set, we will regard it as a potential good factor to fit the model, and then we give the factor the opportunity to continue running the program in test set. Finally, after the program running in both train set and test set, we can decide if we regard the factor and strategy as good strategy and we will use it in the future.

- Factors construction

Two ways: Factors based on economic meaning & Factors created by Genetic Algorithm.

The first way is to look up the existing factor set on the internet, and the most popular one is 101 Formulaic Alphas by Zura Kakushadze. We tested those factors and adjust the parameter of the factor to fit our model. The factors mentioned in the 101 Formulaic Alphas are really effective before 2016, but with the change of economic around the world, some of them may not useful nowadays and others can be developed by adjusting parameters to fit economical environment. And I used grid search to find out the best parameter for the factor.

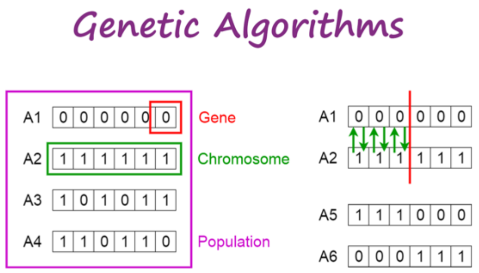
Here are some factors from 101 Formulaic Alphas, and I tried almost all of 101 factors and then test if it works, and the factors below are those can work in some time even now:



```
Alpha#6: (-1 * correlation(open, volume, 10))

Alpha#9: ((0 < ts_min(delta(close, 1), 5)) ? delta(close, 1) : ((ts_max(delta(close, 1), 5) < 0) ? delta(close, 1) : (-1 *
delta(close, 1))))

Alpha#12: (sign(delta(volume, 1)) * (-1 * delta(close, 1)))

Alpha#21: ((((sum(close, 8) / 8) + stddev(close, 8)) < (sum(close, 2) / 2)) ? (-1 * 1) : (((sum(close, 2) / 2) < ((sum(close, 8) /
8) - stddev(close, 8))) ? 1 : (((1 < (volume / adv20)) || ((volume / adv20) == 1)) ? 1 : (-1 * 1))))

Alpha#23: (((sum(high, 20) / 20) < high) ? (-1 * delta(high, 2)) : 0)

Alpha#24: ((((delta((sum(close, 100) / 100), 100) / delay(close, 100)) < 0.05) || ((delta((sum(close, 100) / 100), 100) /
delay(close, 100)) == 0.05)) ? (-1 * (close - ts_min(close, 100))) : (-1 * delta(close, 3)))

Alpha#28: scale(((correlation(adv20, low, 5) + ((high + low) / 2)) - close))

Alpha#32: (scale(((sum(close, 7) / 7) - close)) + (20 * scale(correlation(vwap, delay(close, 5), 230))))

Alpha#41: (((high * low)^0.5) - vwap)

Alpha#46: ((0.25 < (((delay(close, 20) - delay(close, 10)) / 10) - ((delay(close, 10) - close) / 10))) ? (-1 * 1) :
((((delay(close, 20) - delay(close, 10)) / 10) - ((delay(close, 10) - close) / 10)) < 0) ? 1 : ((-1 * 1) * (close - delay(close,
1)))))
```

Then it comes to the second method -- genetic algorithm.

A genetic algorithm is a search heuristic that is inspired by Charles Darwin's theory of natural evolution. This algorithm reflects the process of natural selection where the fittest individuals are selected for reproduction in order to produce offspring of the next generation.



Gene: basic price data (like open price, high price, low price, close price, and some mathematical operator, like plus, minus, multiplication, division, gradient, rank, absolute value, square etc.)

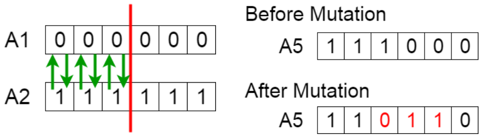
Chromosome: a factor formular constructed by gene

Population: a set of factors

fitness function: the machine learning model and we regard the accuracy as the fitness score

There are several ways to construct a new factor, and let's keep in mind that the chromosome can have any length. We set 101 Formulaic Alphas as our initial population. We have the random forest model as the fitness function and regard the accuracy as the fitness score. The fitness function determines how fit an individual is (the ability of an individual to compete with other individuals). It gives a fitness score to each individual. The probability that an individual will be selected for reproduction is based on its fitness score.

Here's two examples to construct factors.



Some of them can be effective, but there are also many factors that cannot be applied to the model, and we have to drop them by hand. I think my future work will include running the factors by computer after the construction process, keeping those useful factors and dropping those useless ones. The following graphs show some effective factors by gene algorithm:

```
SUM((CLOSE>DELAY(CLOSE,1)?VOLUME:(CLOSE<DELAY(CLOSE,1)?-VOLUME:0)),6)          (NO)

(MEAN(CLOSE,3)+MEAN(CLOSE,6)+MEAN(CLOSE,12)+MEAN(CLOSE,24))/(4*CLOSE)

COUNT(CLOSE>DELAY(CLOSE,1),12)/12*100

COUNT(CLOSE>DELAY(CLOSE,1),20)/20*100

SUM((CLOSE=DELAY(CLOSE,1)?0:CLOSE-(CLOSE>DELAY(CLOSE,1)?MIN(LOW,DELAY(CLOSE,1)):MAX(HIGH,D
 ELAY(CLOSE,1)))),20)

SUM(((CLOSE-LOW)-(HIGH-CLOSE))./(HIGH-LOW).*VOLUME,20)

SMA(MAX(CLOSE-DELAY(CLOSE,1),0),6,1)/SMA(ABS(CLOSE-DELAY(CLOSE,1)),6,1)*100

((HIGH+LOW+CLOSE)/3-MA((HIGH+LOW+CLOSE)/3,12))/(0.015*MEAN(ABS(CLOSE-MEAN((HIGH+LOW+CLOS
 E)/3,12)),12))                              (NO)

(VOLUME-DELAY(VOLUME,5))/DELAY(VOLUME,5)*100

SUM((CLOSE>DELAY(CLOSE,1)?VOLUME:(CLOSE<DELAY(CLOSE,1)?-VOLUME:0)),20)
```

- The trading strategy

Two models: classifier &regression model

There is an important difference between classification and regression problems. Fundamentally, classification is about predicting a label and regression is about predicting a quantity.

We can conclude that we can predict the trend of up and down with Classification predictive modeling and we can predict the accurate price with regression model. And if we want to get the up and down with regression model, we have to do some math to judge whether the price is higher than yesterday or lower than yesterday.

In our strategy, we choose the classifier predicting model to construct the trading strategy. And if the model tells me the stock price will goes up tomorrow, the strategy will buy the stock at the first 3 minutes when tomorrow's stock market open. And will keep the position if it predicts that the stock price won't go down in the next day. And in the market, we can either long the shares or short the shares, which offers more possibility to generate great profit.

As for the frequency of trading, we won't use High Frequency Trading for it has a strict requirement of equipment and device, and we will trade on the day-level, which means we will trade only once a day, with either long position or short position.

- Judgement of the factors

We judge the performance of the prediction model by several targets. The most important one is the Sharpe Ratio. The ratio describes how much excess return you receive for the extra volatility you endure for holding a riskier asset.

$$SharpeRatio = \frac{R_p - R_f}{\sigma_p}$$

I set a target value for Sharpe Ratio. If it's a single factor model, then I need the Sharpe Ratio of the strategy to be above 0.8, and if it's a multiple factor model, then I need the Sharpe Ratio of the model should be above 2.0. And the second target is the Stationarity of the return curve. In the strategy, the strategy may reach the Sharpe Ratio target value as 0.8, but it may get most of the return at the beginning of the horizon and then stay flat in most of the following horizon, which won't guarantee the return in the long run. So, we have to see the return curve and then judge by some mathematical method and then determine whether it can get a steady profit in the long run.

What's more, we have to consider and volume of the underlying asset, in case sometimes we can't get enough share of the stock in some point in the market. And if the volume of the underlying asset is so low that we can't trade enough shares in the same time, we may confront the risk of slippage, which could lead to a wrong result of return curve.

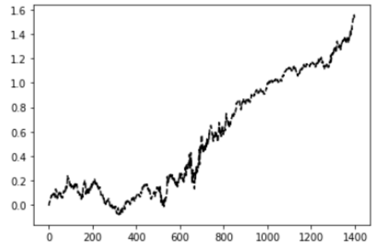
In addition, the turnover rate has impact on the judgement of the strategy as well. If the factor has very strict restriction to select the enter point and easy exit point, the strategy may hold nothing in a long time zone, which is a waste of the assets. To avoid this situation, we have to make sure at least it has at least 1day position in average in one week.

The importance of maximum drawdown can never be overemphasized, and it measures the most loss we may have during the strategy. And investors should be aware that we need a less drawdown for they may not stand the loss during the strategy and they may have an artificial interfere to the strategy, which definitely dose harm to our trading system.

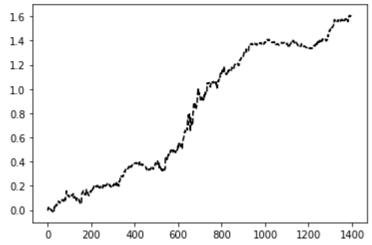
There are some results from different factors:

191Alpha34regression   Alpha53&191Alpha34&60combination

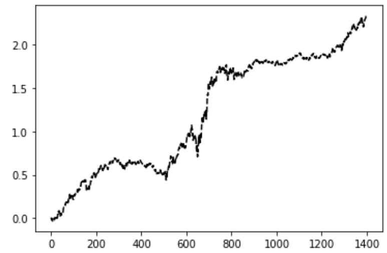
```
1.5557102602218722
0.5544263550151997
Sharpratio:0.9960695794929804
0.3177891952940244
```



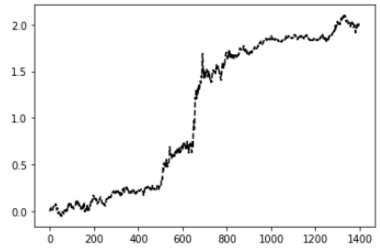
```
1.6162957140495904
0.8161221712776396
Sharpratio:1.7468942810916845
0.14788098014473916
```



```
2.3385908409426044
1.2134448810722047
-0.0322292778090707
Sharpratio:1.504430629967551
0.36481013206036605
```



```
2.105646279849526
1.0586456245680291
Sharpratio:1.2945825571486262
0.3080913304602986
```



191Alpha3Classifier Alpha53regression

As we can see from above, different factors lead to different return, so we have to choose the best factors and apply them to our trading strategy. Among these standard, the most important one is that it has to be profiting continuously and the profit curse need to be going up smoothly.

For example, among these four graphs, the best one is the second one, and others may have different opinions. First of all, it has the highest Sharpe Ratio, and has the least drawdown. What's more, it grows continuously and there's not huge profit or loss in a certain period. As a result, I ought to regard it as the best factor among these four factors, and in my view, it have possibility to continue growing well in the following days.

- Stop loss and take profit

A stop loss (SL) is a price limit entered by a trader. When the price limit is reached the open position will close to prevent further losses.

A take profit (TP) works in a similar way - it automatically closes a position once a profit target is reached to lock in profits.

The advantage of Stop profit and take profit system is that: Stop Losses can limit losses. Take Profits allow the user to maximize profit by exiting a trade as soon as the market is at a favorable price.

Lock in profits

In my opinion, one of the simplest, oldest methods, and most effective ways to help lock in profits and let your winners ride, especially with lower-priced, smaller-cap stocks, is to sell half on a double. This way you take your initial investment off the table and you let your winnings ride. Or you can use a slightly more conservative approach.

Stop loss

I do not want to get whipsawed out of a position because of small and expected pullbacks that can occur in the stock market from time to time. However, limiting large losses can be key to overall long-term performance.

Many thinks using a liberal stop loss as high as 20% is too much. I do not. Stocks fluctuate. A 20% stop loss may not be triggered. This helps prevent getting whipsawed. If you are diligently managing your portfolio positions, you could eliminate weaker performing positions long before the 20% level is hit.

As for more detail about Stop loss and take profit problem, I will introduce more in the later process—position management.

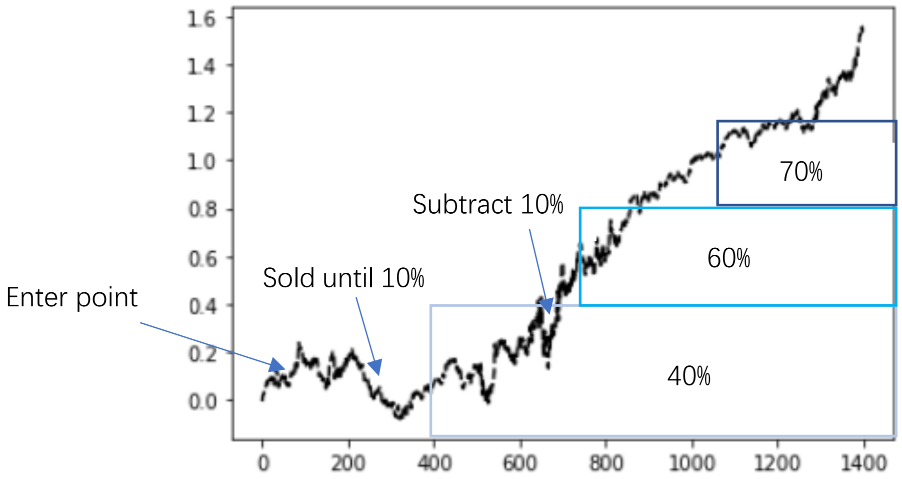
- Position management

Connected to Stop profit and take profit system.

At first, we didn't consider too much about the position management, but when I get some return curve which fall down from some point all the time and then I realized I should add stop-loss and take-profit to my strategy. And the position management comes to be a problem. We not only depend on the result from the model, we also adjust our position every time we trade.

In my opinion, no one position should maintain such a large percentage that it determines the future of the portfolio. Finding proper entry points, trading around core positions, and having a sell discipline can be crucial to increasing the returns of the strategy. Remaining disciplined, unemotional, and mitigating risk are some of the keys to investment success. As we use computer to code the strategy, actually we can avoid our emotion to affect our strategy since we will trust the result of our program in the most of time.

The position management system we used is really simple but effective. If we didn't hold any position yet, we will long or short 20% position when the model gives us the first instruction, and then we follow the following instruction given by the model, if it predicts the same trend as before, we will add 10% position, and if not, we will subtract 10% position instead. The highest level we hold the position of these risky assets if 70%, and we will always hold 10% cash and 20% position of hedge fund as an insurance. So, in our strategy, unless there exists Artificial interfere, the position will be different from each other in different days. And the change won't be too large, in case that it's just some small volatility on that day, and it will keep going up in the following days, which can reduce the commission and tax fee during a long horizon. In this way, we can make sure to capture most of the profit resulted from the price changing in the market.



As we can see from the graph, the most position I have is 70%, and I will add my position if the return curve touched some standard of the guideline.

- Artificial interfere

If there's breakout news or some other extremely serious incidents happing in the day, we can stop our strategy and change to manual trading in the market to avoid some huge loss in advance.

Let me introduce something about manual trading firstly. Manual trading is a trading process that involves human decision-making for entering and exiting trades. Since trades are infrequent, investors are often done manually when an opportunity arises.

In our strategy, I code a web crawler to catch the break-out news on the internet, and the key words of the web crawler are "shut down", "IPO", "refinancing", "at war" … That break-out news definitely has an effect on the stock market, and we can stop the trading and keep the position be zero before it leads to more influence on the market. And we only choose those key words that have really bad influence on the macro-economy, and this is just similar to a stop loss system. We choose not to be speculators that some people may choose the opposite position that the strategy tell you, but I think it's under really high risk and we'd better to stop any action in the market. As for the news that matter only in a few aspects, I think the machine learning model can obtain those changes and those normal change in the market have been reflected in the data we get, so it's wise to just think of worse macro-economy changes as an individual investor.

For example, before the election of American president, the system regard it as a break news and it would have a unexpected influence on the world economy, so I shut down my trading strategy and I will reopen it until the new president has been selected and then the world economic goes normally after that.

Reference:

https://towardsdatascience.com/introduction-to-genetic-algorithms-including-example-code-e396e98d8bf3

https://www.fidelity.com/learning-center/trading-investing/trading/managing-positions

https://planful.com/blog/what-is-a-rolling-forecast/

https://corporatefinanceinstitute.com/resources/knowledge/accounting/rolling-forecast/

https://www.quantopian.com/posts/the-101-alphas-project

https://www.quantilia.com/equity-alpha-strategy/

https://seekingalpha.com/investing-strategy

https://www.pimco.com/en-us/resources/education/portable-alpha-strategies

https://www.sciencedirect.com/science/article/pii/S106294081730400X