Entropy
Final Round Section 2
Report

Le Tran Ngoc Minh
minh.le2018@qcf.jvn.edu.vn

May 7, 2020

1 Introduction

In the modern world working on market economics, the investment is the most energized activity in the markets which could lead to the economic development of the country. Besides the earn profits, the contingent and phenomenal driven properties of stock market making the challenge in the forecasting. In addition, the occurrences of strong power computer, the data-driven approach making high-frequency trading become more popular and outperform the normal stockers. However, there are exist the approach aim to winning in the long term. Those methods are based on Portfolio Modern Theory of Markowitz.

As the main approach of final round entropy contest, I propose the method which combined the **Mean-Variances Method** and **Bayesian Optimization** which could apply in the Vietnamese Market and gain the high value of Sharpe Ratio.

2 Problem Statement

The objective of this problems is optimizing the sharpe ratio

$$\text{Sharpe Ratio } = \sqrt{n} * \frac{\mathbb{E}\left[R - R_f\right]}{\sqrt{\text{var}\left[R - R_f\right]}} = \sqrt{252} * \frac{\mathbb{E}\left[R\right]}{\sqrt{\text{var}\left[R\right]}}$$

where :

- $R_f = 0$
- $n = 252$
- R are equally distributed by the returns of N porfolio $R = \frac{1}{N} \sum i^N R_i$

Therefore the feature that we mainly using is the returns prices of each stocks.

$$R_i = \frac{P_i^T}{P_i^t} - 1$$

3 Data Preprocessing

Transforming Data

The given data are organized as a data frame with ticker, date, close, volume and columns. Because we mainly using the date, closing price and ticker, the data are transformed into the $T \times N$ data frame with the row's index is the T trading dates and columns index is N ticker ids.

Cleaning Missing Values

There is a lot of missing value in the data set. Mainly, that behaviour could belong to one of three causes:

- Data is missing in long interval till the end of stock. This means that the company left the stock market. We could delete them with hesitation.
- Data is missing from the beginning. This means that the company still not join the stock market until the days that data was available. We also do not fill these values.
- Data is missing in the middle. We could fill this type of data by using **Linear Interpolation** to make sure it will reflect the trends and provide the non-zero daily returns (which is important in our models)

4 Approach

The model including Three step.

- The **first step**, which was introduced in round 1, is using **Non-Negative Least Squares** to solving Markowitz Mean-Variance problems in order to remove the short positions.
- In the **second step**, I use the **Simple Moving Average** to filter the increasing stock prices.
- The **final step** using **Bayesian Optimization** to searching the best average.

4.1 Removing Short Positions

Based on Markowitz Modern Portfolio Theory, the maximizing Sharpe Ratio is also the normalization solution of minimization variance problems, called tangency portfolio:

$$\begin{aligned} \min_{\mathbf{w}} \quad & \frac{1}{2}\mathbf{w}^T\mathbf{\Sigma}\mathbf{w} \\ \text{subject to} \quad & \mathbf{w}^T\hat{\mu} = p \end{aligned}$$

where :

- $\mathbf{w}$ is the row weight vector of the assets in portfolio
- $\hat{\mu}$ is the expected return row vector of assets in portfolio
- p is the scalar expected return of all portfolio

The explicit solution of tangency portfolio is:

$$\mathbf{w}^* = \frac{\sum^{-1}\hat{\mu}}{1_N^T\Sigma^{-1}\hat{\mu}}$$

Apply the formula: $\mathbf{\Sigma} = \mathbb{E}(RR^T) - \mathbb{E}(R)(\mathbb{E}(R))^T$ we could form the minimization variances into least squared:

$$\begin{aligned} \min_{w} \quad & \mathbb{E}[|p - R\mathbf{w}|^2] \\ \text{subject to} \quad & \mathbf{w}^T\hat{\mu} = p \\ \text{and} \quad & \mathbf{w}^T\mathbf{1}_N = 1 \end{aligned}$$

or:

$$\begin{aligned} \min_{w} \quad & \frac{1}{T}\left\|p\mathbf{1}_T - R\mathbf{w}\right\|_2^2 \\ \text{subject to} \quad & \mathbf{w}^T\hat{\mu} = p \\ \text{and} \quad & \mathbf{w}^T\mathbf{1}_N = 1 \end{aligned}$$

where:

- T is the numbers of return prices (the number of days if using daily returns)
- R is the $T \times N$ matrix with N is the number of assets
- $\mathbf{1}_z$ is the column vector ones with size z.

However, they require a portfolio is the simple average of $k \leq N$ portfolio. Thus our problems could be solved by select k portfolio that maximizes the Sharpe Ratio with no short position. Moreover, the portfolio expected return p is not required in the solution (because of normalization of weight), therefore we could choosing an arbitrary value of p. To be more capture the fluctuate of market, we set the $p\mathbf{1}_T = \tilde{\mu}$ where each row t th of $\tilde{\mu}$ is the expected value of daily return market calculate by simple average all asset at time $t-1$. Finally, because or daily index in one final year is smaller than a number of stocks, we facing the shrinkage problem as $T < N$.

From the observation above, I propose the method using Non-Negative Least Squared:

$$\min_{\mathbf{w} \succeq 0} \frac{1}{T}\left\|\tilde{\mu} - R\mathbf{w}\right\|_2^2$$

4.2 Filtering Positive Trend by Simple Moving Average

Because of the past data does not project the future situation of the stocks, some stock in the list may have negative return trend. To filter that, we use the Simple Moving Average trading rules: If the stock price path is cross over the Simple Moving Average line, the stock will be considered to buy or sell. Therefore, if the stock price is smaller than its SMA, the trend of the stock is increasing and vice versa. Thus we will reject all stock that have the final day prices smaller than the SMA.

4.3 Finding Best Average by Bayesian Optimization

We could rewrite the problem in the matrix form:

$$\begin{aligned} & \text{maximize} \quad \sqrt{n}\frac{\mathbb{E}\left[R\right]}{\sqrt{\text{var}\left[R\right]}} \\ \Leftrightarrow \quad & \text{maximize} \quad \frac{\mathbf{w}^T\mu}{(\mathbf{w}^T\mathbf{\Sigma}\mathbf{w})^{\frac{1}{2}}} \qquad (n \text{ is annuallization constant factors}) \end{aligned}$$

Because the expect return are compute by the mean of all return of K stock (which $K < N$) we select, then we could rewrite the weight as form:

$$\mathbf{w}^T = \frac{1}{|K|}\mathbf{w}'^T$$

where

- $\mathbf{w'}^T = [w_1, w_2, ...w_i] \quad w_i = 0 \text{ or } 1, \quad i \in [0, N]$
- $|K| = \mathbf{w'}^T \mathbf{1}_N$ ($|K|$ is the number of selected assets)

Then our final optimization problems is:

$$\begin{aligned} \mathbf{w'} &= \arg\max_{\mathbf{w'}} \quad \frac{\frac{1}{|K|}\mathbf{w'}^T\mu}{(\frac{1}{|K|}\mathbf{w'}^T\mathbf{\Sigma}\frac{1}{|K|}\mathbf{w'})^{\frac{1}{2}}} \\ &= \arg\max_{\mathbf{w'}} \quad \frac{\mathbf{w'}^T\mu}{(\mathbf{w'}^T\mathbf{\Sigma}\mathbf{w'})^{\frac{1}{2}}} \end{aligned}$$

From here, when the number of remaining stocks is relevant small, we could use the best subset selection by exhausted search. However, there still be a very expensive task. Instead, I proposed using Bayesian Optimization method for searching the parameters.

Bayesian optimization is the optimization algorithm that incorporates prior belief about function f and updates the prior with samples drawn from f to get a posterior that better approximates f. The model used for approximating the objective function is called surrogate model. Bayesian optimization also uses an acquisition function that directs sampling to areas where an improvement over the current best observation is likely.

In this report the surrogate model is $f(\mathbf{w'}) = \frac{\mathbf{w'}^T\mu}{(\mathbf{w'}^T\mathbf{\Sigma}\mathbf{w'})^{\frac{1}{2}}}$ and we using the acquisition function as upper confidence bound (UCB).

In the implement step, it is hard to generate all combination of $\mathbf{w'}$. However, we could mimic this by generated the decimal integer x, then convert x from decimal to N bits binary. This binary form of x is using to constructing the weigth vector $\mathbf{w'}$.

For example, if we have 5 assets, and we generate x in domain $[0, 2^5 - 1]$. If $x_{10} = 3_{10}$, then $x_2 = 00010_2$ so the weight is $\mathbf{w'}^T = [0, 0, 0, 1, 0]$.

Therefore, our surrogate model could be rewritten as follow:

$$f(x) = \frac{\mathbf{g}(x)^T\mu}{(\mathbf{g}(x)^T\mathbf{\Sigma}\mathbf{g}(x))^{\frac{1}{2}}} \quad , \quad x \in [0, 2^{|N|} - 1]$$

where :

- $$g(x)^T = [w_N, w_{N-1}, \ldots, w_1]$$
- $$x = 2^{|N|-1} \times w_N + 2^{|N|-2} \times w_{N-1} + \cdots + 2^1 \times w_2 + 2^0 \times w_1$$

In this project, I obtain the maximum result when observing that there no improvement after 200 iterations with the $\kappa = 5$.

5 Result

I acquired the list of **20** stocks and receive the testing result from the API at 03/09/2019 equal **11.132025544144525**.

The list of stocks is: **ABT, AST, BCE, HII, JVC, KMR, L10, LIX, PGD, ROS, SBV, SFC, SJF, TCL, TDC, TMS, TNT, VCF, VIS, VNE**