Predicting Which Stocks Will Beat the Market

Junlin Liu (junlin93), Yongshang Wu (wuy), Hao Wang (wanghao)

December 12, 2017

1 Introduction

In stock markets, there are usually two types of measurements for a stock or a portfolio: absolute return and relative return. Absolute return is a direct reflection of the profit or loss of a single stock or a portfolio. On the other hand, relative return measures the profit or loss of a single stock or a portfolio relative to the average return of the whole market. For example, if we say FB has a relative return of 15% within the last three months, what we mean is that the profit of FB is 15% more than the market (sometimes represented by an index such as Dow Jones Industrial Average index or S&P 500 index), or its loss is 15% less than the market.

Sometimes, it is hard for investors to make absolute return in a bear market like in 2008. Also, in a long period of bull market like from 2009 to 2017, most investors can make some absolute return. Under such circumstances, relative return becomes more important to judge whether or not an investor is successful or a portfolio manager is good.

If we can predict which stocks can beat the market using machine learning, it will be easier for us to do better than other investors! In this project, we try to predict whether or not a stock can beat the market in the next 1, 2 and 4 quarters, and finally we make it! The most different thing of our project from others is that we predict future returns of stocks not only based on past prices, but also on fundamental data of companies.

2 Data

Our goal is to predict whether or not a stock can beat the market in the next 1, 2 and 4 quarters. The reason we choose a quarter as the unit is that, from a financial point of view, our insight is that future performance of a stock can be affected by two factors: its historical price and the financial status of the company behind the stock. Since a company's financial status is updated every three months (it releases quarterly report every three months), we believe treating a quarter as a unit is a good alignment with that. Therefore, the data we collected is historical prices and quarterly financial status of stocks.

Since there is no open dataset that matches our problem definition, we get them by ourselves. The data is collected from FactSet, which is a famous financial information provider in US. From it, we get historical prices and financial information of all 30 companies in Dow Jones Industrial Average index in recent 15 quarters (from 2013 Q4 to 2017 Q2). After getting raw data, we do some work of data cleaning and preprocessing. Now, for each of the 30 stocks, we have 15 quarters of information of it. Therefore, we have around 450 training samples in total.

3 Features

In our project, we have tried two sets of features. The first is the set of basic features. Here the word basic is just a name we give it. It does not mean it is naive. As a matter of fact, the basic feature set comes from our domain knowledge and reflects our belief that both historical price and financial information of the company can affect the future price of a stock. The second set of features we have tried is an advanced feature set which is built upon the basic feature set. In addition to different feature sets, we also adopt different feature selection methods (Wrapper and Filter). Since Wrapper method is much slower, the way we use them are also slightly different.

3.1 Basic Feature Set

The basic feature set consists of 40 features, which are:

- returns of the market in recent 1, 2, 3 and 4 quarters
- returns of the stock in recent 1, 2, 3 and 4 quarters
- variation from the highest price in recent 1, 2, 3 and 4 quarters
- variation from the lowest price in recent 1, 2, 3 and 4 quarters
- revenue growth of the company in recent 1, 2, 3 and 4 quarters
- operation income growth of the company in recent 1, 2, 3 and 4 quarters
- net income growth of the company in recent 1, 2, 3 and 4 quarters
- by how much earnings of the company beat analysts' expectation in recent 1, 2, 3 and 4 quarters
- dividend ratio of the company in recent 1, 2, 3 and 4 quarters
- dividend growth of the company in recent 1, 2, 3 and 4 quarters

In these 40 features, the first 16 provide information about past prices of a stock and the last 24 provide information about financial status of the corresponding company. In these features, the earliest time we look back is one year ago. The consideration is that we believe information of one recent year is sufficient to affect performance of a stock the next few quarters.

3.2 Advanced Feature Set

We have also tried an advanced feature set, which is built upon the basic feature set. Firstly, for each feature x, we add feature

$$f(x) = \frac{1}{x}$$

Then for each two features x_1 and x_2, we add feature

$$g(x_1, x_2) = x_1 x_2$$

Intuitively, by adding these new features, we want to capture binomial relation between variables represented by basic features and the target category.

3.3 Feature Selection Methods

As both of basic and advanced feature sets are generated by our domain knowledge, there might be some redundant or useless features. In our project, we adopt two different features selection methods to remove them and get better results.

Wrapper method: The method we adopt is backward search. Each time we remove the least significant feature that improves the performance until there is no improvement after removal. This method is implemented by ourselves, but it is rather slow, so we also adopt another filter method.

Filter method: By this method, we calculate an importance value for each feature, and then choose features with highest importance values. The limitation of this method is that only some machine learning models support computation of importance values. For those models which do not, we do not adopt this feature selection method, neither. To use this method, we simply invoke functions implemented by Scikit-learn package.

4 Experiments and Results

Our problem is basically a binary classification problem. Hence, the metric we use is simply accuracy. For each stock at the end of each quarter, we try to predict whether it will beat the market or not in the next 1, 2 and 4 quarters. We divide our data into training set, validation set and testing set. If there is parameter tuning, we choose parameters based on 5-fold cross validation. The accuracy reported below is the result finally obtained on testing set. We have run a lot of experiments, but only present some key results here, which outline our procedure of trials and improvements, due to page limitation.

4.1 Using Basic Features and Wrapper Feature Selection

In the first place, we fit two models (logistic regression and gradient tree boosting) on the data using basic features. It turns out that the results we obtain are just a little bit better than random guess. Then we apply wrapper feature selection method and get better results. The results before and after feature selection are shown in the table below. The number outside of the parenthesis is the average accuracy while that in it is the maximum accuracy by simple parameter tuning. It can be seen that using gradient tree boosting model, we get averagely 55%, 55% and 57% percent of accuracy of prediction for 1, 2 and 4 quarters, respectively after applying wrapper feature selection.

	Logistic Regression		Gradient Tree Boosting	
	Before	After	Before	After
1 Quarter	0.49 (0.61)	0.51 (0.61)	0.52 (0.63)	0.55 (0.65)
2 Quarters	0.50 (0.63)	0.50 (0.67)	0.53 (0.65)	0.55 (0.71)
4 Quarters	0.56 (0.69)	0.54 (0.64)	0.55 (0.69)	0.57 (0.76)

Table 1: Result for basic features and feature selection

4.2 Comparing Performance of Different Models

During our process of experiments, we keep trying a variety of models to see which one has the best performance. The results show that among models we have tried, gradient tree boosting is the best. The comparison is made using basic feature set and wrapper feature selection method as well. As is shown in the figure below, in terms of both average accuracy and maximum accuracy, gradient tree boosting is better than other models.

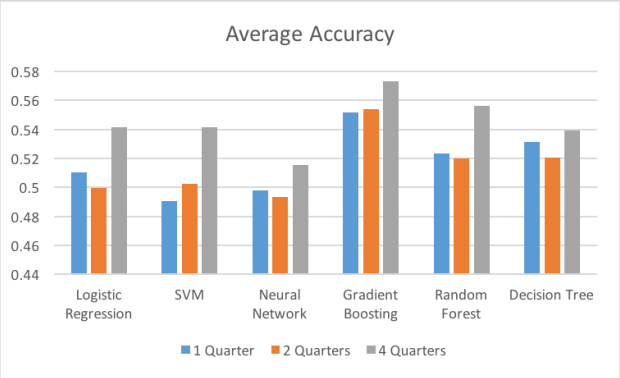

Figure 1: Average accuracy obtained by different models

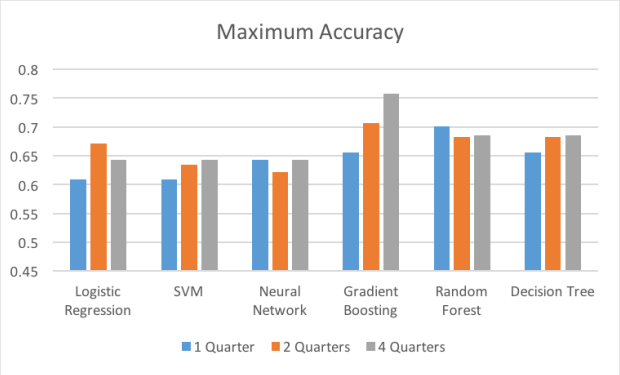

Figure 2: Maximum accuracy obtained by different models

4.3 Using Advanced Features

After a chunk of experiments, we still can not get better performance. We begin to think of adding more advanced features as described in section 3.2. By fitting gradient tree boosting model with new features, we obtain very good results! We have tried two different ways of adding features. The first is performing wrapper feature selection followed by extracting advanced features. The second is extracting advanced features first and then performing filter features selection method. It turns out that the second way is impressively effective. As is shown below, the average accuracies of prediction for 1, 2 and 4 quarters are around 70% percent. The maximum accuracies are even more than 80% percent. From a financial point of view, these are fairly good results as the stock market is a rather chaotic system.

	Origin	Method 1	Method 2
1 Quarter	0.52 (0.63)	0.54 (0.67)	**0.67 (0.80)**
2 Quarters	0.53 (0.65)	0.52 (0.63)	**0.70 (0.82)**
4 Quarters	0.55 (0.69)	0.55 (0.71)	**0.72 (0.86)**

Table 2: Results after adding advanced features

5 Analysis and Discussion

Since the stock market is known as a huge chaotic system, it is really hard to make predictions on the future return of stocks. Furthermore, what we are trying to do is to predict the relative return of stocks instead of the absolute return, which makes it even harder. As is shown in section 4.1 and 4.2, even though we extracted from out domain knowledge, tried

whatever model we can try with feature selection, the best result (for average accuracy) is just around 5% better than random guess. We projected the data onto a 2-D plane using PCA algorithm and found that there is almost no observable boundary that can separate the two categories. Then we realize that features we use may be too simple and begin to consider using polynomial features. As shown in section 4.3, it turns out that this strategy works. That means, the relation between relative returns of stocks and those variables coming from our domain knowledge is not simply linear, which is in accordance with consensus of many real investors.

Another thing we care about is which features are indeed useful for making predictions when we get 70% of accuracy finally. We print the sorted features according to their importance values and then get two observations. First, we find that most features have an importance value of 0, meaning that they are useless in prediction. Only no more than 9% of total features have importance values larger than 0. Secondly, we find that there are no significantly dominant features in prediction for 1, 2 and 4 quarters. The most importance feature has only an importance value of around 0.01 (normalized). However, the important features are different themselves in prediction for 1, 2 and 4 quarters. Below we show a heatmap of feature importance of 6400 features when predicting for 2 quarters.

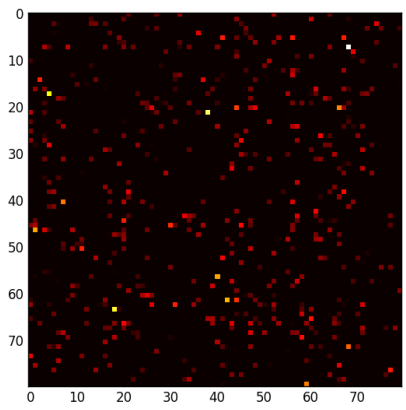

Figure 3: Heatmap of feature importance when predicting for 2 quarters

Last but not least, we find some projects that are similar to ours in the poster session this year and on websites in previous years. However, the results of them are hardly as good as ours. This is because they do not incorporate important domain knowledge (only consider past prices), or they do not use more complicated features, or they do not try to use feature selection methods. Considering the inherent difficulty of our problem (the chaotic nature of financial market and the small size of our dataset), we would say it is not very easy to get the best results we finally have.

6 Acknowledgement

This is a project we really enjoy. Thanks to all of our team members who make efforts to come up with good ideas, discuss possible models and solutions, collect data manually, implement algorithms and try everything to make our project decent! Every one of us shares a considerable amount of work and contributes to our project a lot! Thanks to TAs who give us helpful and insightful feedback. Thanks to Dan and Andrew who give great lectures of this wonderful course.