Forecasting Monthly Stock Return through Unsupervised K-Means Clustering, Neural Network

Chong Sun[1]

Department of Mathematics
Baylor University, One Bear Place # 97328, Waco, TX 76798

Abstract

In this project, we aim to predict the moving directions of the stock prices in the very next month with unsupervised clustering and neural networks. So essentially we are dealing with a classification problem—the return of next month is positive or negative. Financial data especially stock prices are known for its non-stationarity and nonlinearality which makes it harder to predict the future movement with classical models. A common approach is to combine a method for the partitioning of input space into a number of sub-spaces with a local approximation scheme for each subspace. Unsupervised K-Means clustering algorithm is employed to partition the stock return data and neural networks are used to perform local approximation. Backward propagation is used in training the neural networks. We performed experiments on Apple, Ford, Coca Cola and AT&T stocks. Results of the experiments are analyzed.

Keywords: K-Means Clustering, Neural Network, Backward Propagation

1. Introduction

This project is based on three previous papers [1], [2] and [3].

The objective for this project is to forecast moving direction of stock prices in the next month through unsupervised clustering and neural networks. The reason I am forecasting monthly return as opposed to daily return is because very short term data tend to be really noisy and especially so in the financial market as described in [1].

This paper totally has 3 sections. In section 2, we will explain in detail about the methodology I used in this paper. and the experiment we performed on Apple, Ford, Coca Cola and AT&T stocks. Results will be given and analyzed. In section 3, conclusion will be drawn and future work that may improve the performance of the model will be proposed.

2. Methodology

We are using monthly data from Jan. 1990 to Feb. 2017 for both clustering and neural networks training. We first cluster the 326-month data into four groups based on four macroeconomic

[1]Email Address: Chong_Sun@Baylor.edu

factors—market return, Fed rate, unemployment rate and inflation rate. Then we build a neural network for each group based three technical factors and one company fundamental factor—two-month price moving average, three-month price moving average, current month stock return and size of the company. The details of the designs, ideas behind it and results are discussed below in the following four subsections.

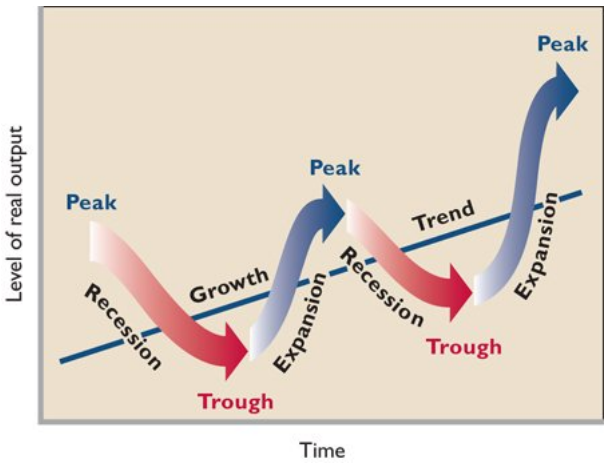
2.1. Experiment of Clustering

The steps to for clustering is as follows,

1. Preprocessing data: we normalize the data by find $z-$score of each of the four types of macro data-market return, Fed rate, unemployment rate and inflation rate, separately.
2. We employed K-means++ clustering algorithm to cluster the 326 months into four groups based on the $z-$scores of four types of data in each month. The data are collected on the website of Bureau of Labor Statistics, official site of Federal Reserve and Yahoo Finance.

The reason for clustering and choosing 4 clusters is because,

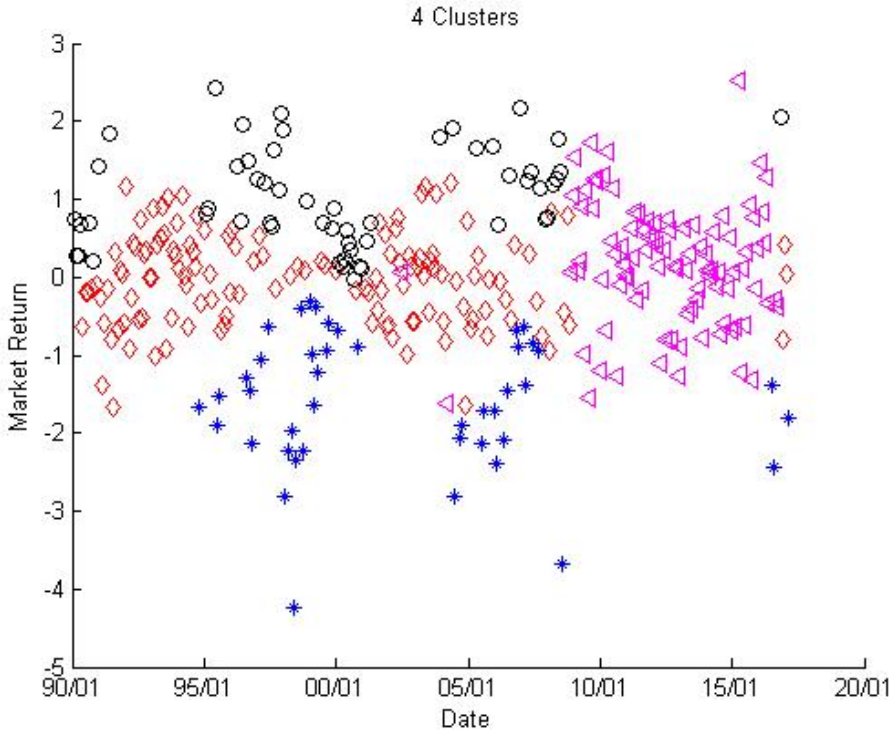
1. According to empirical research on business cycles, business cycle consists of four stages—expansion, peak, contraction and trough. Each stage exhibits different recognizable patterns in macro factors. See the figure below [4],



2. The stock returns exhibit different patterns for different stages.

2.2. Result of Clustering

We implemented the K-means++ algorithm with Matlab, and obtained the following plot and each color with different shape represents one group,



Now if we examine the four groups closely, and compare it to the business cycle figure, we can draw the following conclusion,

1. Group 1 which is represented by red diamond contains 133 points. It matches roughly the recession period;
2. Group 2 which is represented by blue asterisk contains 43 points. This group matches the trough of the economy;
3. Group 3 which is represented by magenta triangle contains 95 points. It matches the expansion period;
4. Group 4 is black circle which matches the peak period. It contains 55 points.

2.3. Neural Networks Training

We built one neural network for each group separately with Matlab. The neural networks have four layers and four nodes in the first hidden layer, three nodes in the second hidden layer. We also used weight constraint for second and fourth cluster due to small sample size in these two groups. The augmented error we used is

$$E_{\text{aug}}(\mathbf{w}) \quad = \quad E_{\text{in}}(\mathbf{w}) + \frac{\lambda}{N}\sum_{l,i,j}(w_{ij}^{(l)})^2. \tag{2.1}$$

Now we discuss in detail the training of each neural network and the ideas behind the setting. The training process of neural networks for each stock is,

1. Preprocessing data: we normalize the data by calculating $z-$score of input data for neural networks—two-month price moving average, three-month price moving average, current month stock return and size of the company.
2. Index the data by the month they belong to.
3. Separate the data points into four groups according to the time index of each point, based on the clustering results we obtained in section 2.1.
4. Build one neural network for each group with forward and backward propagation and regularization. Here we used stochastic gradient descent. We set regularization parameter $\lambda = 0.01$ in the augmented error (2.1) for 2nd and 4th group. The descent rate is set to be $\eta = 0.1$. The stopping criteria is a combination of upper bound for iterations and control of error size. We set upper bound for iteration to be $n = 10,000$. We count 1 round of forward and backward propagation for each point as one round.

The reasons we choose this type of design of neural networks are,

1. Stock market data is highly noisy and nonlinear, neural networks are suitable for modeling nonlinear data.
2. Stock return in different stages of business cycle exhibit different patterns, thus we build one neural network for each group of cluster.
3. The reason we choose two-hidden-layer networks is because I experimented with one-hidden-layer neural network, but it takes a while to converge and they tend to have very large in-sample errors.
4. We chose the setting of nodes in each hidden layer is because of the result from [3].
5. We need regularization is because without regularization, we obtained results that are seriously overfitting. Besides group 2 and 4 have relatively smaller sizes than the other two groups.
6. Descent rate and early stopping settings are due to different experiments on different combinations of these two terms. $\eta = 0.1$ and $n = 10,000$ tend to give us lower validation error within relatively smaller amount of time comparing to other combinations.

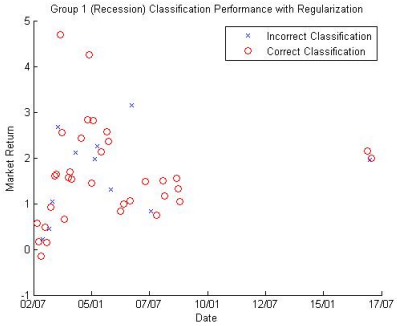
2.4. Results of Neural Networks

We built neural networks for four companies: Apple, Ford, AT&T and Coca-Cola. I chose these four companies from different sectors of economy- tech,industrials,utility and consumer staples respectively.In this way, I can examine whether my model works better for one sector than the other.

Here I only show the plots of classification performances of neural networks on Apple stock. As for other companies, I will only list the relevant data and measurements.

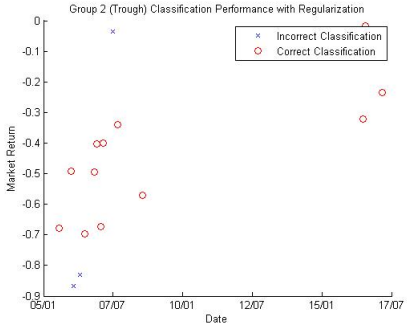
Note each of the following plots denote the performances of the neural networks on each test set. Red circles represent correct classification and blue cross represent incorrect classification. Base rates denote the percentage of months in which the price went up in each test set. In-Sample Error and Out-Sample Error below each plot denote the percentage of correct classification in the training and testing sets respectively.

1. Apple
    - Group 1 (Recession)



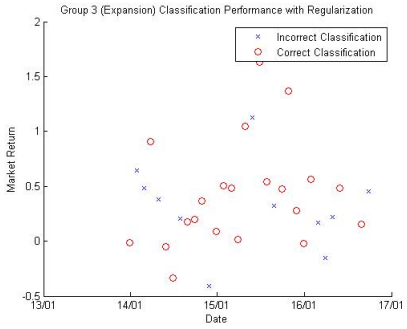
Base Rate: 51%
In-Sample Error: 5%
Out-Sample Error: 24%
Size of the Group: 133 data points

- Group 2 (Trough)



Base Rate: 53%
In-Sample Error: 14%
Out-Sample Error: 20%
Size of the Group: 43 data points

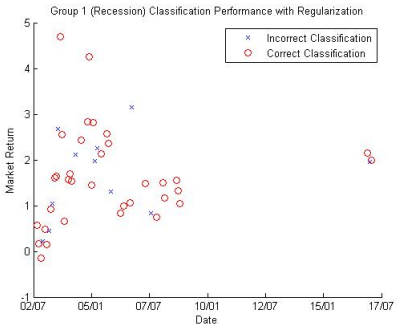
- Group 3 (Expansion)



Base Rate: 51%
In-Sample Error: 16%

Out-Sample Error: 34%
Size of the Group: 95 data points

- Group 4 (Peak)



Base Rate: 46%
In-Sample Error: 0%
Out-Sample Error: 26%
Size of the Group: 55 data points

2. Coca-Cola

	Base Rate	In-Sample Error	Out-Sample Error	Size
Recession	49%	4%	15%	133
Trough	40%	14%	33%	43
Expansion	51%	9%	25%	95
Peak	44%	3%	16%	55

3. AT&T

	Base Rate	In-Sample Error	Out-Sample Error	Size
Recession	51%	5%	15%	133
Trough	42%	24%	27%	43
Expansion	52%	2%	19%	95
Peak	45%	0%	26%	55

4. Ford

	Base Rate	In-Sample Error	Out-Sample Error	Size
Recession	45%	2%	24%	133
Trough	45%	0%	33%	43
Expansion	49%	3%	28%	95
Peak	49%	0%	37%	55

2.5. Analysis

1. The clustering is not robust. Different initialization of first center will result in quite different sizes of groups. When finally chose this group, I did multiple different initializations. I chose these clusters because they have lower error and the size of smallest cluster is acceptable.

2. There is still overfitting in the neural networks despite that I used early stopping and weight restraint to try to remedy that. The gap is large even in group 1 which has largest data sizes.
3. The problem of overfitting is especially severe for the group 2 which only contains 43 points and group 4 which contains 55 data points.As you can see, in all the four stocks, group 2 and group 4 have the high out-sample classification error.
4. However the weight constrains do help cure to some extent the overfitting problem comparing to no weight constraints.
5. One thing I want to point out is that, the base rates are all close to 50% for each group. And our out-sample is relative low comparing to the base rates. So the predicting power of our model is relatively stronger than purely random guess.
6. But however, the base rate poses another question. As we can see the base rate is really close to 50% for all four groups for all four stocks. So does it mean the stock movement is purely random even inside each cluster? And our small out-sample error is only because of the small sample effect? Further research need to be done to answer these questions. Because if the stock movements are purely random, then any type of model will be rendered useless.

3. Conclusions and Future Work

Conclusion

1. Stock market return is highly noisy and hard to find a perfect model;
2. Even within same type of model, different initialization and choosing different factors will yield quite different results;
3. Better to put your money in index funds than in active funds.

Future Work

1. Add more fundamental factors to neural network input set;
2. PCA and regularization to avoid overfitting problems;
3. Cross Validation to fight the problem of small number of data points;
4. Convolutional Neural Network.

[1] Yaser S. Abu-Mostafa, *Financial Applications of Learning from Hints* .
[2] J. Ghosn, Y. Bengio *Multi-Task Learning for Stock Selection*.
[3] N.G. Pavlidis, V. P. Plagianakos, D.K. Tasoulis, M.N. Vrahatis *Financial Forecasting through Unsupervised Clustering and Neural Networks.*
[4] M. Healy *Business Cycles, Unemployment, Inflation*(2017),[Lecture Notes]. Retrieved from http://www.harpercollege.edu/mhealy/eco212i/lectures/ch9-18.htm