Project Report

Stock Prediction System using machine learning technique

KUN MEI 145008679
JUNZHE ZHENG

Abstract: In this paper, we discuss a decision making system focused on stock buying and selling prediction using machine learning techniques. It is based on neuron network and logistical regression. We developed a machine learning algorithm and prediction methods for the SPX(standard and poor index 500) prediction system. We find that through this system, we would achieve a accurate result and the testing on this system shows a reasonable profit and beat the market. This report is mainly based on the work from " stock market prediction with modular neural networks"

Introduction and Problem statement

Nowadays, stock price trend has always been one of the hottest topics in research. However, it is always a key problem to find out an accurate prediction system to trace the stock price trend and make the buying & selling decision wisely. Several mathematical models have been developed, but the results have been dissatisfying. We chose this application as a means to check whether neural networks and logistic regression method could produce a successful model in which their generalization capabilities could be used for stock market prediction.

Neural network and logistic regression are being applied widely to classification problems such as email spam filter, face detection, classify proteins according to function, etc. The stock trading decision making system is one of classification

problem. Before Neural network and logistic regression, several mathematical has been developed, but there are some flows of those models created dissatisfying results.

We choose this project to check weather neural network and logistic regression could build a good enough model which could used for stock market prediction. If so, then we will build a trading system based on the models which aim to make buying , holding or selling decisions for a future time period by studying relationship between previous technical index and their corresponding decisions.

Implementation details

The implementation includes the following steps:

1. Data selection. During this step, we select data and preprocess data in order to format data into form that fits the learning model.
2. Model Description. Two learning models will be briefly introduced.
3. We used a technical called Moving window to solve time continuously training and predicating issues.
4. Evaluating. Description of two evaluating criterial.

Data Selection

The first step for a machine learning prediction system is to generate formatted input data. The data should have two separate parts: features and target variables.

We believe stock prices are determined by many technical indexes and those indexes changing over times. In order to minimize random price fluctuation, we used k-days average data on each indexes.

1. Features

The input data used is GSPC, which is a ticker symbols of S&P 500. It includes High, low, close, volume and adjust close as raw parameters. They are not used directly into our model because they are not sufficient to describe the market. Instead, several technical indexes are generated from raw data. All input indexes are shown in table 1.

Table 1. Input indexes

ATR	Average True Range
SMI	Stochastic Momentum Indicato
ADX	Directional Movement Index
Aroon	Aroon Oscillator
BB	Bollinger Bands
ChaikinVolDelt	Chaikin Volatility
CLV	Chaiken Volatility

EMV	Arm's Ease of Movement
MACD	Moving Average Convergence / Divergence
MFI	Money Flow Index
SAR	Parabolic Stop and Reversal
Volat	Stock Volatility

All above indexes are combined to used as features.

2. Target Variables

There are three stock trading decisions: Sell, buy, hold. They are operations based on analyzing performance and trends of the market. The goal is to make profits. Generally speaking, the trading decision should follow a simple criteria: buy at low price and sell at high price.

The target variables are generated with the following step:
Compute daily average price:

$$P_i = \frac{C_i + H_i + L_i}{3}$$

C,H,L and P indicate close price, high price, low price and average price respectively. Subscript i indicates the i-th day.

For i-th day, compute price changing rate for the next k days:

$$V_i = \left\{ \frac{P_{i+j} - C_i}{C_i} \right\}_{j=1}^{k}$$

Note that , Vi is a list with k values.

Define target T to be the summation of price changing rate which is grater than p% or less then -p%

$$T_i = \sum_v \{ v \in V_i : v > p\% \text{ or } v < -p\% \}$$

If T is positive and greater than some criteria value, which means the overall pricing changing rate is positive and price will go high in the future k days. Based on trading

rule, buy at low and sell at high, then buying signal will be made.

If T is negative and less than some criteria value, selling signal will be made.

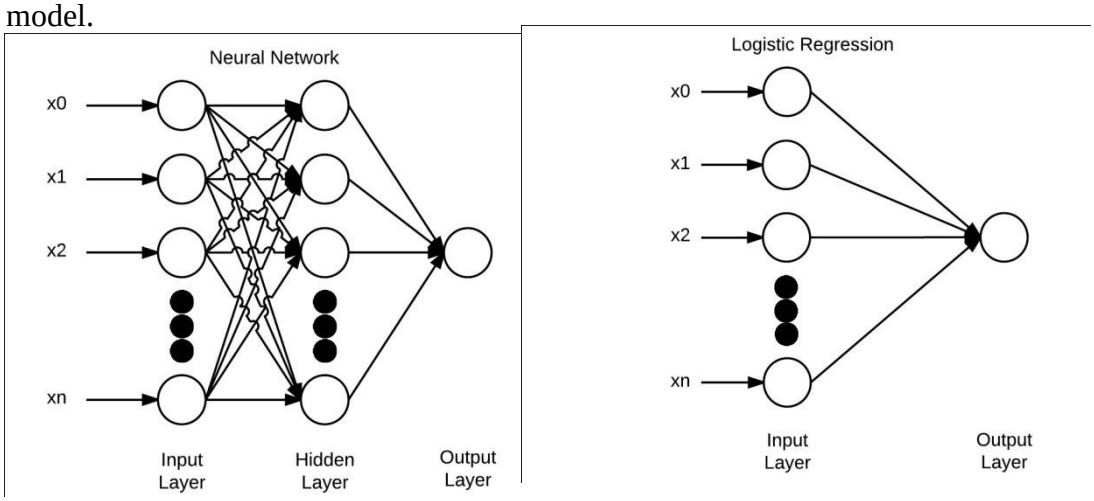
Otherwise, holding signal will be made.

Above can be given:

$$signal = \begin{cases} sell & \text{if } T < -0.1 \\ hold & \text{if } -0.1 \le T \le 0.1 \\ buy & \text{if } T > 0.1 \end{cases}$$

Model Description

Logistic regression and neural network algorithm are being applied to create the model.



Above figure shows the architecture of the two learning models. The most left layer in both model is input layer which receives the feature data. And the output is one among buy, sell and hold.

Moving Window

For predication of an economic system, such as stock market, the trends are changing continuously, thus the training and test sample should be ordered by time. This also means some technics such as cross validation should not be applied to our model.

We applied a method called moving window. In this system, the training and test data would be moving by a gap once current predication is done.

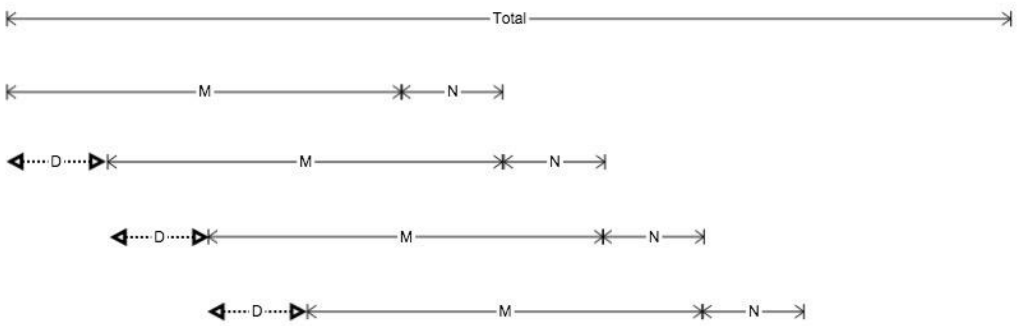


Figure 2

This moving window method is described in figure 2. The top line of the figure indicates the entire data time period. For each predication process, only data for M+N time long are being used. Then the the data just slides D month for next predication process.

Evaluation

We will evaluate the result from two aspect:

Accuracy

$$\text{accuracy} = \frac{\text{number of true positives} + \text{number of true negatives}}{\text{number of true positives} + \text{false positives} + \text{false negatives} + \text{true negatives}}$$

Overall Earning Yields

The goal of trading stock is to make a profit, thus earning yields is a good way to evaluating any trading system.

At time t, asset is a(t), after a period dt, asset is a(t+dt). During the same period time, the index changing rate is r%, thus the overall earning yields is given by:

$$\frac{a(t+\Delta t) - a(t)}{a(t)} - r\%$$

		Predictions		
		Sell	Hold	Buy
1	Sell	n(s,s)	n(s,s)	n(s,s)
	Hold	n(h,s)	n(h,h)	n(h,b)

		Predictions		
		Sell	Hold	Buy
	Buy	n(b,s)	n(b,h)	n(b,b)

Result and Discussion

we will do the testing in two ways: first we just roughly split the period into two parts: the training datasets are the first 80% days, the testing datasets are the sub sequential 20% trading days second, we would use a validation method specified for time series data,which is called "moving window" In addition, we are using two metrics to evaluate this predicting system : confusion window and the (adjusted) return rate. Confusion matrix is as same as we did in the traditional machine learning evolution. The adjusted return rate is as follows: 1: The return rate means the profit rate we could get from which we follow the signal to buy/sell stocks. 2: The corresponding index move indicates the SPX index performance in corresponding period 3: The adjusted rate means the adjusted rate towards the SPX index performance, if we get a positive value; it means we beat the market, otherwise we fail the market.

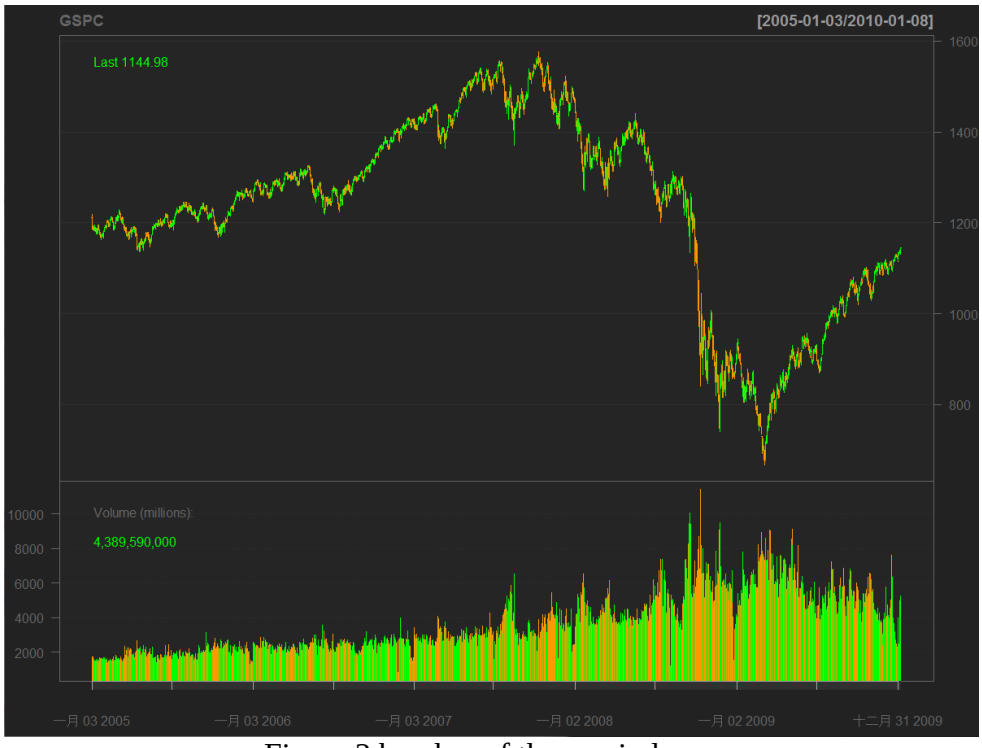


Figure 2 barchar of the spx index

In the following part, we are using SPX data from with start date="2005-01-02" and end date="2010-01-10" which was randomly selected.

First, we are doing method 1st: simple overall training and testing
the size of the training date is 2723 trading days, while the testing data is the subsequent 686 trading days

	Buy	HOLD	Sell
Buy	30	102	13
Hold	26	360	19
sell	21	104	11

Table 1 confusion matrix generalized by method 1 using neuron network
Accuracy=58.45%

	Buy	HOLD	Sell
Buy	14	55	8
Hold	38	472	27
sell	25	39	8

Table 2 confusion matrix generalized by method 1 using logistic regression
Accuracy=72.01%

We can see through the comparison, the two methods both show a relatively good performance. The performance of the neuron network is a bit lower than that of logistical regression. But I don't think it means that we need to think this algorithm is not good since the critical failure(predicting sell as buy or predicting buy as sell) is relatively in the same level. Using neuron network, we get 13+21=34 critical miss classification. In logistic regression, we get 25+8=33 critical miss classification.

```
 predicting using neroun netowrk

the return rate is
 0.3537267
the corresponding index move is
 0.4967307
the adjusted retrun rate is
 -0.143004
 predicting using logistic regression

the return rate is
 0.1580611
the corresponding index move is
 0.4967307
the adjusted retrun rate is
 -0.3386696
```

Table 3 return rate using two algorithm
We can see that neuron network is doing really good than logistic regression in this part. Although they are not able to beat the market.

Second, we are doing method 2nd: moving window training and testing
The size of the "window" is 1000 trading days, and the size of the step length of the window moving is 500 trading days.

	Buy	HOLD	Sell
Buy	29	74	9
Hold	47	405	107
sell	14	96	23

Table 4 cumulative confusion matrix generalized by method 2 using neuron network
Accuracy=56.09%

	Buy	HOLD	Sell
Buy	27	23	16
Hold	48	431	88
sell	15	121	35

Table 5 cumulative confusion matrix generalized by method 2 using logistic regression
Accuracy=61.31%

We can see through the comparison, the two methods both show a relatively good performance. The performance of the neuron network is a bit lower than that of logistical regression. But I don't think it means that we need to think this algorithm is not good since the critical failure(predicting sell as buy or predicting buy as sell) is relatively in the same level. Using neuron network, we get 23 critical miss classifications. In logistic regression, we get 31 critical miss classifications. In general, the accuracy of the 2nd method is not as good as the first "batch" training and testing. It may because the overall trends are monotone and each moving window is kind of starving of data. However, we must notice that the critical miss classifications in the 2nd method are significantly small than that of the first method, so we may expect they may be a good system in the real world.

```
 predicting using neroun netowrk

the return rate is
 0.01392717
 predicting using logistic regression

the return rate is
 -0.005932302
```

Table 6 in window 1, return rate using two algorithms

```
 predicting using neroun netowrk

the return rate is
 0.03637174

the return rate is
 0.02956359
```

Table 7 in window 2, return rate using two algorithms

```
 predicting using neroun netowrk

the return rate is
 0.05184375
 predicting using logistic regression

the return rate is
 0.00215508
```

Table 8 in window 3, return rate using two algorithms

```
 predicting using neroun netowrk

the return rate is
 -0.0009573273
 predicting using logistic regression

the return rate is
 0.04865752
```

Table 9 in window 4, return rate using two algorithms

From the above data, we could calculate that the overall return rate of neuron network is 9.37%, while the overall return rate of logistic regression is 3.379%.both return rates are not as good as we did in the previous 1st method. The corresponding return rate of the 1st method is 35% and 15%

Conclusion and Future Works

The series of experiments we did shows that it is possible to use neuron network and logistic regression as the key algorithm to build up a robust stock buy/sell predicting system effectively. We are creatively using different complex technical features together with financial features to support our research. The performance and accuracy is good with very small critical miss classification(mark buy as sell, and mark sell as buy). Although the prediction system is unable to beat the market, it still makes some profit and shows a good potential to improve.

Future work would be using more data in different situation to train the system. For instance, we would use different source data like NASDAQ or CRUDE OIL Future to test the performance. Moreover, since there exist some situations that some feature index would fail, we would study

how to select effective variables to improve the performance and accuracy of this predicting system.

Reference

Gong, Jibing, and Shengtao Sun. "A New Approach of Stock Price Trend Prediction Based on Logistic Regression Model." *A New Approach of Stock Price Trend Prediction Based on Logistic Regression Model*(2009): n. pag. *International Conference on New Trends in Information and Service Science*. Web.

Kimoto, Takashi, and Kazuo Asakawa. "Stock Market Prediction System with Modular Neural Networks." (1991): n. pag. *IEEE Explore*. Web.

"Quantmod: Quantitative Financial Modelling Framework." *Quantmod: Quantitative Financial Modelling Framework*. Quantmod Lab, 20 May 2008. Web. 20 Nov. 2014.