Stock Market Prediction Using Data Mining Techniques with R

Shounak Dutta¹, R.Rohit²
School of Electronics, KIIT University, India

Abstract:
The stock market is a place where shares of publicly listed companies are traded. The shares are bought and sold depending on the market indices. The price of stocks and assets are an important part of the economy. There are many factors that affect share prices. However there is no specific cause for the prices to rise or fall. This makes investment subject to various risks. The prices of the future stocks are affected by the preceding and current market indices. Hence stock market prediction techniques like ARIMA and ARMA are used for short term forecasting. This paper proposes a stock market prediction model based on the analysis of past data and ARIMA model. This model will assist investors to buy or sell stocks at the right time. The forecast results are visualized using R programming language.

Keywords: Stock Market, Data Mining, Prediction, ARIMA, Time Series Data, R

I. INTRODUCTION

The stock market system consists of 2 parts, the primary market and the secondary market. The primary market is the place where publicly listed companies offer their shares in an initial public offering (IPO) to raise resources to meet their requirements of investment. The secondary market refers to the market where stocks are traded after their initial offering to the public or after being listed on the Stock Exchange. It is loose network of economic transactions, not bound to any physical facility or entity. The prices of the stocks depend on market trends, investment tactics and other momentary inefficiencies. This randomness makes it difficult to model a framework to forecast stock prices with exactness. The basic assumption made while forecasting stock data is that future market trends are influenced by the stock information available publicly in the past. This means, the historical stock data provides an insight into its future behavior. According to the Random Walk theory for stock markets, “stock market prices evolve according to a random walk and thus cannot be predicted”. The theory is further divided into 2 separate parts. The first hypothesis states that successive price changes in an individual security are independent. The second hypothesis states the prices conform to a certain probability distribution. However, it is the probability distribution of data or the form of distribution that allows academicians and investors to forecast stock data. Recent studies have shown that Time Series data analysis techniques provide verifiable information for forecasting stock prices. Time series data is sequence of data collected over specified period of time. Time series data for stock market prediction can be collected on a daily, weekly, monthly or yearly basis. The analysis of the time series data extracts useful statistical information to understand characteristics of data. Time series forecasting techniques involve using models to predict future values based on past information. R is an open source programming language and software environment for statistical computing and graphics. It has numerous applications in the field of data analysis and widely used by statisticians and data miners. Along with a command line interface, it has several graphic front-ends. R is extensible through functions, extensions and packages, contributed by the global R community. As of 2016, 7801 additional packages are available for installation. This user created packages like forecast, stats, ggplot2 allows the user to perform specialized statistical and graphical procedures. RStudio is an open source integrated development environment (IDE) for R. The software is written in C++ programming and uses Qt framework for graphical user interface. It supports direct code execution as well as tools for statistical analysis, debugging and workspace management. There are 2 editions of RStudio, RStudio Desktop and RStudio Server. RStudio Desktop runs the program as a regular desktop application. Using the RStudio Server, RStudio running on a Linux server can be remotely accessed via a web browser. RStudio allows users to manage multiple working directories using projects. It also has extensive package development tools. The remainder of the paper covers the following topics. Literature review of pre published research papers related to forecasting stock market trends. System analysis covers the problem statement and the papers approach to forecast the market trends. Implementation describes in detail the methodology used to predict stock prices. Model simulation describes the R code used to develop the model. The section also consists of visualized outputs.

II. LITERATURE REVIEW

To forecast stock returns, scholars and researchers rely upon fundamental analysis and technical analysis. The author Suresh A.S. [1] describes fundamental analysis as the examination of underlying forces that affect the wellbeing of the economy. Fundamental analysis combines economic, industry and company analysis to derive a stocks fair value known as intrinsic value. According to fundamental analysis if the fair value is not equal to the current stock price, then the stock is either undervalued or overvalued. Fundamental analysis takes into account macroeconomic factors and individual specific factors. Fundamental analysis is believed to be effective predicting long term trends. The same paper describes technical analysis as a supplement for to fundamental analysis but more focused on predicting the price of a security. Technical analysis takes into account the change in demand and supply of securities as a function of time. Therefore it is preferred over fundamental analysis for short term and medium term forecasting.

Technical analysis is defined as the art and science of forecasting future prices based on the examination of past price movements by the author [C.Boobalan][2]. In addition to past stock prices, technical analysis also considers company fundamentals, broader economic factors, market psychology and prices them into the stock. There are different technical
factors that affect and set stock prices. These predictors can be used for forecasting a macroeconomic time series variable as done by authors [James H Stock][Mark W Watson] [3]. Indexes constructed by principal component analysis principal component analysis are used to combine these predicting factors. The authors developed an approximate dynamic factor model for estimation of indexes and construction of forecasts. The model constituted a set of 215 predictors that were simulated in real time for the period 1970-1988. The method can be used to construct 6, 12 or 24 month forecasts. It was observed that during the sample period, the given set of factors provided a forecast that outperformed univariate and small vector autoregressions. The forecasts outperformed leading indicator models as well. Authors [Wei Huang][Yoshiteru Nakamori][Shou-Yang Wang][4] forecast stock market movement direction with support vector machine, a machine learning technique that analyzes data for classification and regression analysis. The authors investigate the predictability of the SVM technique by forecasting the weekly movement of Nikkei 225 index. According to the paper, the key property of SVM is that training data in SVM model is equivalent to solving quadratic programming problem with linear limits. Therefore SVM always provides a solution that is unique and globally optimal. The authors also compare the SVM model with other forecasting methods like Linear Discriminant analysis, Quadratic Discriminant analysis and Neural Networks. The experimental results of the paper show that SVM has the highest forecasting accuracy as it minimizes structural risk. The integration of SVM with other methods improves forecasting performance. Author [Linhao Zhang][5] describes the effect of public sentiment on stock price by analyzing Twitter messages. The author examines the effects of tweets on stock prices and also determines which words in the tweet correlate to stock price change. The author uses machine learning techniques like Naïve Bayes classification, Maximum Entropy classification and Support Vector Machines to determine sentiment. Data is fetched from the Twitter’s Search API and classified into 2 different datasets for training. However, it was found that the classifiers were not much effective for negation statements. The author also correlates tweet sentiments and stock market prices on an intra-day scale. Yahoo’s finance API was used to gather data on 10 exchange traded funds or ETF’s. The paper uses the Pearson correlation coefficient to account for any time difference. The experimental results show that for the data to be truly effective a larger time granularity is required. To find deeper correlations, the system needs to be used over a longer period of time with more aggregate data. The author also describes the numerous challenges that analyzing Twitter sentiment poses. The very first challenge is searching for the right tweets without getting too arbitrary. Searching for keywords and interpreting slang jargon is another inherent challenge. For effective results, the system needs to be trained on much more data over a larger period of time. A relatively new method, Approximation and Prediction of Stock Time series data (APST) has been proposed by authors [Vishwanath R.H.], et al. [6]. The system generates a sequence of approximated values using multi-scale segment mean approach after preprocessing historical stock time series data. To identify the similar set of objects, the authors use Euclidian distance approach and find the nearest neighbors. Experimental results show that the average Mean Error Relative and average Mean Absolute Error for APST are 5.90% and 0.37%. This implies that the system shows a high level of accuracy. Authors [Ayodele A.], et al. [7] have used the auto regressive integrated moving average (ARIMA) model to predict stock prices. The experimental results showed that ARIMA models can predict prices on a short term basis. This would allow investors to make profitable investments. Authors [Jingtao Yao] [Hean-Lee Poh] have used artificial neural networks (ANN) to forecast indices of the Kuala Lumpur Stock Exchange (KLSE). Artificial neural networks have been effectively used to decode non-linear time series data. Artificial neural networks can recognize patterns and infer solutions from unknown data, thus making them extremely popular.

III. SYSTEM ANALYSIS

A. Problem Statement

The financial market or stock market is complex and evolutionary. It functions as a non-linear dynamic system. According to academic investigations movements in market prices are not random and depend upon numerous factors that correlate it with present and historical stock data. It is not possible for every investor to comprehend the various factors that cause the prices to change. Hence every investor desires a system to predict the future stock prices to help them take appropriate decisions.

B. Existing Systems

Numerous qualitative and qualitative analysis methods have been developed to estimate stock trends. There are various statistical models for forecasting stocks and decide the right time to sell or hold a stock. Depending upon the format of the data, a particular forecasting model can be used by the investor to predict trends.

C. Proposed Study

The paper proposes a model for predicting time series stock market data. The model based on technical analysis using ARIMA aims to automate the process of change of stock price indices. With the help of Data Mining techniques a prediction model is developed. R programming language in RStudio IDE is used for visualizing the experimental results.

IV. IMPLEMENTATION

Data mining is used to discover patterns in large data sets and has wide application in the field of statistics. Data mining techniques are devised to address forecasting problems by providing a reliable model with data mining features. We use the auto-regressive integrated moving average (ARIMA) model to predict the market trends. The complete architecture of the system is shown below.

Figure 1. Implementation
System architecture contains the information regarding the constituent elements of a system. It also describes the relationship between these elements. It is a model that provides information about the behavior of a system by breaking it into subordinate systems that perform the same functions. The ARIMA system includes seven major steps to implement the system and each step is explained below.

A. Understanding the Objective

The objective describes the essential requirements of the system. It helps in better understanding of the problem statement as well as the expected results. The objective this paper is to develop a system that can be used by investors to find the direction of the market trends and make right investment decisions. The experimental results are provided in a graphical format for better interpretation.

B. Data Collection

Understanding the objective also aids in collating the right datasets. Data collection involves gathering information relevant to the required variables and measuring them to evaluate outcomes. The paper uses R script to collect data from Google using the function getSymbols() available in the QuantMod package.

- QuantMod
  Quantmod refers to Quantitative Financial Modelling and Trading Framework for R. It is quantitative tool that helps traders in developing and testing trade based statistical models. The quantmod package makes modelling easier and faster by excluding repeated workflow. The package consists of comprehensive tools for data management and visualization. To extract and load the data from multiple sources we use a method called getSymbols(). As a source for obtaining the stock market data, most of the stock investors use Google finance or Yahoo finance. In our project the OHLC data is not directly downloaded from the Google finance (finance.google.com), or Yahoo Finance (finance.yahoo.com) instead a call to getSymbols() is used to fetch data. We didn’t specify the source here so the data is downloaded from default reference ie:-www.finance.yahoo.com.

C. Data Pre-processing:

Data collection is loosely controlled and more than often garbage values get added to the dataset. A high concentration of redundant information (noise) makes the data irrelevant and useless for further processing. Hence pre-processing of data is necessary to prepare the final dataset from given raw information. The method described in this paper converts the input data into a differentiated vector list. The function c(base) is used to address the combined vector list.

- Data Frames
  A data.frame() object in R has same dimensional properties as a matrix. But unlike matrices, data frames may contain both categorical and numeric data. It can be said that data frame is a list of variables with components as columns of a table. A list of variables with same number of rows and distinct row names of a class is defined as a data frame. The row names decide the number of rows, if no variables are involved. The behavior of the data.frame() object can be changed by writing methods according to its class.

D. Data Processing:

The first step in data processing is to train the data. The ARIMA(p, d, q) model is used to process data. Investors and analysts two methods to predict stocks namely auto regression and moving average. R provides auto.arima() method to forecast the time series data according to ARIMA (p, d, q). The ARIMA model is a tool for technical analysis. It focuses on repeated parameter estimation and forecasting to find the right approximation model.

- Auto Regression (AR)
  Auto regression technique estimates the future values based on the previous values. The function of an autoregressive model is denoted by AR(p), where p represents the order of the model. AR(0), the simplest process, involves no dependence between terms, preceding or current. For a first order autoregressive model AR(1), the preceding term and a percentage of error contribute to the output. AR(2) model takes into account 2 preceding values and noise to predict the output.

- Moving Average (MA)
  A moving average is a technique to model datasets that vary according to single factor. It finds the future trends based on the previous values that do not follow a definitive pattern. The two commonly used moving average techniques are exponential moving average (EMA) and the simple moving average (SMA).

- Order of ARIMA
  The order of an ARIMA model is generally represented as ARIMA(p,d,q), where-
p = order of the autoregressive part.
d = degree of first differencing involved.
q = order of the moving average part.

  Here if d=0, then the model becomes ARMA which is linear stationary model. The same stationary and in-variability conditions that are used for autoregressive and moving average models apply to this ARIMA (p,d,q) model. Selecting the appropriate values for p, d and q can be challenging. The auto.arima() function in R will do it automatically.

- Model Estimation for ARIMA
  Model estimation for ARIMA can be achieved based on the pre-processed historical data.

![Figure2. pre-processed historical data.](http://ijesc.org/)
In ARIMA model, the identification is to be accomplished using auto co-relation function and partial auto co-relation function in order to identify p, d and q standards. For any realistic time sequence generally p, d and q values vary between 0 and 2, but model estimation is executed for all probable combinations of p, d and q values. The pictorial representation of these steps is shown in Fig4.2

• ARIMA() Function in R
Predicting the right values for p, d and q for ARIMA model can be tough. The problem becomes more prominent when the given dataset is larger and contains data for a longer period of time. The auto.arima() function provided in the forecast package for R automates the process of finding the right combination of p, d and q. The value of d also has an effect on the prediction intervals i.e., the more complex the value of d, the more rapidly forecasting intervals surge in size. For d=0, the long-term prediction average deviance will go to the typical deviance of the historic data. Sometimes autocorrelation function (ACF) and partial autocorrelation function (PACF) are used to determine the number of or order of AR or MA terms required.

E. Forecasting Results
Forecasting allows us to predict future values based upon the knowledge of current and historical stock data. The model specified here uses the forecast package for R for predicting future stock values. The forecast package contains tools for analyzing univariate time series data using state space models and ARIMA modelling. The Arima () and auto. arima () functions used to model future stock prices are a part of the forecast package.

F. Plot Visualisation
Plot visualization involves representing the numerical data in graphical format. In the given methodology, line charts and histograms are used to represent the stock data. This is done using the plot function provided in R. The addBBands () function adds two additional lines that make data interpretation easier. The x-axis represents the time period in terms of year/months and days while the y axis shows stock price values.

V. MODEL SIMULATION
The step by step execution and code is provided below. We will start with the same basics of running basic checks on the data and then take a deeper dive in terms of modelling technique to use.

• Loading the quantmod library.

> library(quantmod)
Loading required package: xts
Loading required package: zoo

> library(tseries)
Loading required package: TTR
Version 0.4-0 included new data defaults. See tgetSymbols.

• Loading the OHLC data of MICROSOFT using getSymbols()

> getSymbols("MSFT")
Calculating the return of each day for the last 12 months.

> ret_MSFT<-100*diff(log(MSFT$MSFT.Adjusted))

Table.2. Intra-day returns values

<table>
<thead>
<tr>
<th>Date</th>
<th>Return (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>2015-10-06</td>
<td></td>
</tr>
<tr>
<td>2015-10-07</td>
<td>0.10680123</td>
</tr>
<tr>
<td>2015-10-08</td>
<td>1.57053703</td>
</tr>
<tr>
<td>2015-10-09</td>
<td>-0.71912506</td>
</tr>
<tr>
<td>2015-10-10</td>
<td>-0.59870000</td>
</tr>
<tr>
<td>2015-10-11</td>
<td>0.03431785</td>
</tr>
<tr>
<td>2015-10-12</td>
<td>-0.4486123</td>
</tr>
<tr>
<td>2015-10-13</td>
<td>0.70456008</td>
</tr>
<tr>
<td>2015-10-14</td>
<td>1.05766652</td>
</tr>
<tr>
<td>2015-10-15</td>
<td>0.33125357</td>
</tr>
<tr>
<td>2015-10-16</td>
<td>0.31452023</td>
</tr>
<tr>
<td>2015-10-17</td>
<td>-1.20030934</td>
</tr>
</tbody>
</table>

• Loading the forecast package

> library(forecast,quietly = T)

• Creating training and testing vectors

> MSFT_ret_train<-ret_MSFT[1:(0.9*length(ret_MSFT))]

Table.3. Values obtained after Training

<table>
<thead>
<tr>
<th>Date</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>2015-10-06</td>
<td>0.10659135</td>
</tr>
<tr>
<td>2015-10-07</td>
<td>1.37037432</td>
</tr>
<tr>
<td>2015-10-08</td>
<td>1.7012506</td>
</tr>
<tr>
<td>2015-10-09</td>
<td>-0.23577069</td>
</tr>
<tr>
<td>2015-10-10</td>
<td>-0.23431785</td>
</tr>
<tr>
<td>2015-10-11</td>
<td>0.04468123</td>
</tr>
<tr>
<td>2015-10-12</td>
<td>0.70456008</td>
</tr>
<tr>
<td>2015-10-13</td>
<td>1.05766652</td>
</tr>
<tr>
<td>2015-10-14</td>
<td>0.33125357</td>
</tr>
<tr>
<td>2015-10-15</td>
<td>0.31452023</td>
</tr>
<tr>
<td>2015-10-16</td>
<td>-1.20030934</td>
</tr>
</tbody>
</table>

> MSFT_ret_test<-ret_MSFT[(0.9*length(ret_MSFT)):length(ret_MSFT)]

Table.4. Values obtained after testing

<table>
<thead>
<tr>
<th>Date</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>2017-01-19</td>
<td>-0.20251502</td>
</tr>
<tr>
<td>2017-01-20</td>
<td>0.03781860</td>
</tr>
<tr>
<td>2017-01-21</td>
<td>0.55003547</td>
</tr>
<tr>
<td>2017-01-22</td>
<td>0.85525734</td>
</tr>
<tr>
<td>2017-01-23</td>
<td>0.25157270</td>
</tr>
<tr>
<td>2017-01-24</td>
<td>0.22202902</td>
</tr>
<tr>
<td>2017-01-25</td>
<td>0.60555281</td>
</tr>
<tr>
<td>2017-01-26</td>
<td>0.73907071</td>
</tr>
<tr>
<td>2017-01-27</td>
<td>-1.66071459</td>
</tr>
<tr>
<td>2017-01-28</td>
<td>-0.64009094</td>
</tr>
<tr>
<td>2017-01-29</td>
<td>0.00410723</td>
</tr>
</tbody>
</table>

• Applying ARIMA(),predict() and forecast() methods. In ARIMA(p,d,q) here we take
  p=2
d=0
q=2

> fit<-arima(MSFT_ret_train,order = c(2,0,2))
> preds<-predict(fit,n.ahead = (length(ret_MSFT) - (0.9*length(ret_MSFT)))//3)
> test_forecast<-forecast(fit,l=23)

• Plotting the ARIMA forecast

![ARIMA forecast for MICROSOFT returns](image)

Above are the results that we obtain with a simple ARIMA(2,0,2) model. The deeply shaded region provide us the 99% confidence level and the lightly shaded region provide the 95% confidence level for the forecast. An intrinsic shortcoming of the ARIMA models, which is evident from the plot above, is the assumption of the mean reversal of the series. What this means is that after some time in future the forecasts would tend to the mean of the time series’s historical values thus making it a poor model for long term predictions.

• Calculating the accuracy of the model.

The lower the value of RMSE (Root Mean Square Error) better is the accuracy of the model.

> accuracy(preds,MSFT_ret_test)

Here we find the RMSE value to be 0.7841242 which is quite low hence the model is found to be pretty accurate.

VI. CONCLUSION

In this paper an attempt was made to forecast the stock market prices of the MICROSOFT stock by developing a prediction model based on technical analysis of historical time series data and data mining techniques. This paper successfully predicted the stock price indices for short-term period using an ARIMA model. The potential of the ARIMA model in finding future stock price indices which will enable stock brokers/investors to make profitable investment is huge. The only drawback of this model as compared to its competitors is the tendency to compute the mean of the historical data as forecast when it comes to long term prediction. Thus it is not advisable to use this model for long-term forecasting of stock price indices.
VII. FUTURE SCOPE

The possibility of integrating this model with fundamental analysis can lead to better decision making when it comes to making decisions like buy/hold/sell a stock. Through a pertinent sentiment analysis performed by collecting social media data and combining it with the ARIMA forecast better profitable investment decisions could be made.

VIII. REFERENCES


