**Ramaiah Institute of Technology**

(An Autonomous Institute, Affiliated to VTU)

MSR Nagar, MSRIT post, Bangalore-54

A Dissertation Report on

Stock Market Prediction and Portfolio Optimization

Submitted by

Name of the student: **SULOCHANA ROY** USN number:**1MS15CS150**

Bachelor of Engineering in Computer Science & Engineering

Under the guidance of

Name of the guide : Mrs. Soumya BJ

Designation: Assistant Professor

Dept name: Computer Science Engineering, RIT

**DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING**

**M.S.RAMAIAH INSTITUTE OF TECHNOLOGY**

**(Autonomous Institute, Affiliated to VTU)**

**BANGALORE-560054**

www.msrit.edu, **2017**

**Introduction**

Stock market prediction is to try determining the future value of a company stock (or other financial instruments) traded on an exchange. While there is risk associated with every stock, optimizing a portfolio of stocks might help in reduced risk and increased returns.

Recently, a lot of interesting work has been done in the area of applying different Algorithms for analysing price patterns and predicting stock prices and index changes. Most stock traders nowadays depend on Intelligent Trading Systems which help them in predicting prices based on various situations and conditions, thereby helping them in making instantaneous investment decisions. Stock Prices are considered to be very dynamic and susceptible to quick changes because of the underlying nature of the financial domain and in part because of the mix of known parameters (Previous Days Closing Price, P/E Ratio etc.) and unknown factors (like Election Results, Rumors etc.) An intelligent trader would predict the stock price and buy a stock before the price rises, or sell it before its value declines. Though it is very hard to replace the expertise that an experienced trader has gained, an accurate prediction algorithm can directly result into high profits for investment firms, indicating a direct relationship between the accuracy of the prediction algorithm and the profit made from using the algorithm.

Also, portfolio optimisation focuses on minimizing the risks involved in investing in one stock and choosing various stocks and optimizing the investments to get maximum returns. This project focuses on how individual stock values in the market can be predicted and finding optimum solutions which increases the accuracy of the predictions. It also involves optimization of a combination of stocks to get maximum returns over a fixed period of time.

**State-of-the-art**

In practice, there are 2 Stock Prediction Methodologies:

*Fundamental Analysis:* Performed by the Fundamental Analysts, this method is concernedmore with the company rather than the actual stock. The analysts make their decisions based on the past performance of the company, the earnings forecast etc.

*Technical Analysis:* Performed by the Technical Analysts, this method deals with thedetermination of the stock price based on the past patterns of the stock (using time-series analysis.) When applying Machine Learning to Stock Data, we are more interested in doing a Technical Analysis to see if our algorithm can accurately learn the underlying patterns in the stock time series. This said, Machine Learning can also play a major role in evaluating and

forecasting the performance of the company and other similar parameters helpful in Fundamental Analysis. In fact, the most successful automated stock prediction and recommendation systems use some sort of a hybrid analysis model involving both Fundamental and Technical Analysis.

Indian stock broking firms are now moving towards big data analytics to predict movement of stock portfolios by analyzing huge volumes of numbers and data. This trend is gathering momentum as trading is getting extensively algorithm driven, and time sensitivity is more critical than ever before.

Here are few of the Indian stock broking companies who are offering a bouquet of alternative tools and platforms that help their customers analyze the stock market better.

**Reliance Securities**

Reliance Securities has implemented an integrated online trading platform—Tick, which uses analytics to provides robotic insights to investors. It scans and captures vast amounts of gathered data, processes it using advanced algorithms, and presents real time analyses. It helps the firm get a macro view of stocks, and provide analytics and optimal risk strategies.

**Kotak Institutional Equities**

Kotak Institutional Equities uses a web-based platform named Consumer Querimetrix that explains and predicts short term behavior of Indian investors by analyzing vast amounts of data. It uses machine learning techniques and merges big data analytics to provide consumer insights and capture inflection points.

**Angel Broking**

Angel Broking has also incorporated big data analytics in its day-to-day operations to automate processes, speed up activities and enhance customer experience. The firm uses analytics to predict margin-limit multiplier, e-mails and calls classifier, and analyzing customer sentiments, queries and complaints. Currently, about 30 percent of Angel Broking’s trades happen online and nearly 80 percent of its new clients demand online access.

**HDFC Securities**

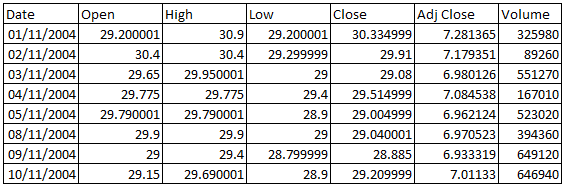
HDFC Securities is using a mobile app that helps its clients to trade stocks, track market movements, manage portfolios and analyze industry trends, using big data analytics. The company has also implemented Oracle SuperCluster that supports its increasing customer base and daily transaction load. The platform has increased online trading speed by up to 60 percent and enabled HDFC to produce reports 67 percent faster while reducing risk and cutting data center costs.

* **Data Set description**
* **Source of Dataset**

**-**Yahoo Finance

* **Attributes Description**
* *Open:* The Opening value of the stock
* *High:* Highest value of the stock on a given day
* *Low:* Lowest value of the stock on a given day
* *Close:* Closing value of the stock
* *Volume:* Average volume of the stock traded.
* *Adjusted:* Adjusted closing value of the stock.

Sample Data:



The fields chosen for prediction here:

* Date
* Open
* **Data Set size in terms of Bytes and Number of tuples**

. The data set comprises of 3216 rows with 7 columns.

* **Inference:**

**We mainly focus on two grounds here......Stock market prediction & Portfolio Optimization.**

The two attributes here chosen here for analysis are date and open values of the stocks . We analyse the trend of these portfolio's over a span of more than 13 years, till the current date and from the various plots obtained after applying the Holt Winters algorithm and Auto Arima Model, alongwith the help of linear programming for predicting the return values, we are able to suggest to the mass, how to optimally invest their money in certain stocks that are chosen here, to get the best returns and suffer less loss comparitively, which can later be implemented even on a larger scale, including more no of stocks .

* **Algorithm Description**

**Algorithms used for Stock Market prediction and Portfolio Optimization**

In general, the following algorithms are used for Stock market prediction:

* Time Series(ARIMA)
* Holt-Winters
* Genetic Algorithm
* Various Machine Learning Algorithms
* Neural Networks

The following methods can be considered for Portfolio Optimization:

* Linear Programming
* Quadratic Programming
* Monte-Carlo

**Method:**

The algorithms used for prediction in this project are:

* Auto ARIMA
* Holt-Winters

The Method used for portfolio optimization is:

* Linear Programming

**Decompose into more times series**

Time series decomposition is a mathematical procedure which transforms a time series into multiple different time series. The original time series is often computed (decompose) into 3 sub-time series:

***Seasonal:*** Patterns that repeat with fixed period of time. For eg. A website might receive more visits during weekends. This is a seasonality of 7 days.

***Trend:*** The underlying trend of the metrics. For eg. A website that gains in popularity should have a general trend that go up.

***Random:*** (also call “noise”, “Irregular” or “Remainder”) is the residuals of the time series after allocation into the seasonal and trends time series.

**AUTOCORRELATION FUNCTION (ACF) & PARTIAL AUTOCORRELATION FUNCTION (PACF):**

Autocorrelation and partial autocorrelation are measures of association between current and past series values and indicate which past series values are most useful in predicting future values. With this knowledge, you can determine the order of processes in an ARIMA model. More specifically,

• Autocorrelation function (ACF). At lag *k*, this is the correlation between series values that are *k* intervals apart.

• Partial autocorrelation function (PACF). At lag *k*, this is the correlation between series values that are *k* intervals apart, accounting for the values of the intervals between.

Best ACF and PACF values are directly identified by Auto.Arima models while running the algorithm.

 In ARIMA (p,d,q) output, p refers to the number autoregressive terms considered in building the model, while q refers to the number of Moving Averages considered while building the model. d is the number of level of differencing applied by the auto.arima model in identifying the best possible model that forecasts the stock prices.

p - autoregressive terms - is identified by the auto.arima model from PACF graph

q - Moving averges - is identified by the auto.arima model from ACF graph

compute.sens=1 is used so that model gives you only positive results including non-zero value.

**ARIMA**

Forecasting involves predicting values for a variable using its historical data points or it can also involve predicting the change in one variable given the change in the value of another variable. Forecasting approaches are primarily categorized into qualitative forecasting and quantitative forecasting. Time series forecasting falls under the category of quantitative forecasting wherein statistical principals and concepts are applied to a given historical data of a variable to forecast the future values of the same variable. Some time series forecasting techniques used include:

* Autoregressive Models (AR)
* Moving Average Models (MA)
* Seasonal Regression Models
* Distributed Lags Models

What is Autoregressive Integrated Moving Average (ARIMA)?

ARIMA stands for Autoregressive Integrated Moving Average. ARIMA is also known as Box-Jenkins approach. Box and Jenkins claimed that non-stationary data can be made stationary by differencing the series, Yt. The general model for Yt is written as,

**Yt =ϕ1Yt−1 +ϕ2Yt−2…ϕpYt−p +ϵt + θ1ϵt−1+ θ2ϵt−2 +…θqϵt−q**

Where, Yt is the differenced time series value, ϕ and θ are unknown parameters and ϵ are independent identically distributed error terms with zero mean. Here, Yt is expressed in terms of its past values and the current and past values of error terms.

The ARIMA model combines three basic methods:

* AutoRegression (AR) – In auto-regression the values of a given time series data are regressed on their own lagged values, which is indicated by the “p” value in the model.
* Differencing (I-for Integrated) – This involves differencing the time series data to remove the trend and convert a non-stationary time series to a stationary one. This is indicated by the “d” value in the model. If d = 1, it looks at the difference between two time series entries, if d = 2 it looks at the differences of the differences obtained at d =1, and so forth.
* Moving Average (MA) – The moving average nature of the model is represented by the “q” value which is the number of lagged values of the error term.

This model is called Autoregressive Integrated Moving Average or ARIMA(p,d,q) of Yt. We will follow the steps enumerated below to build our model.

**Step 1: Testing and Ensuring Stationarity**

To model a time series with the Box-Jenkins approach, the series has to be stationary. A stationary time series means a time series without trend, one having a constant mean and variance over time, which makes it easy for predicting values.

**Testing for stationarity –** We test for stationarity using the Augmented Dickey-Fuller unitroot test. The p-value resulting from the ADF test has to be less than 0.05 or 5% for a time series to be stationary. If the p-value is greater than 0.05 or 5%, you conclude that the time series has a unit root which means that it is a non-stationary process.

**Differencing –** To convert a non-stationary process to a stationary process, we apply thedifferencing method. Differencing a time series means finding the differences between consecutive values of a time series data. The differenced values form a new time series dataset which can be tested to uncover new correlations or other interesting statistical properties.

We can apply the differencing method consecutively more than once, giving rise to the “first differences”, “second order differences”, etc.

We apply the appropriate differencing order (d) to make a time series stationary before we can proceed to the next step.

**Step 2: Identification of p and q**

In this step, we identify the appropriate order of Autoregressive (AR) and Moving average (MA) processes by using the Autocorrelation function (ACF) and Partial Autocorrelation function (PACF).

**Identifying the p order of AR model**

For AR models, the ACF will dampen exponentially and the PACF will be used to identify the order (p) of the AR model. If we have one significant spike at lag 1 on the PACF, then we have an AR model of the order 1, i.e. AR(1). If we have significant spikes at lag 1, 2, and 3 on the PACF, then we have an AR model of the order 3, i.e. AR(3).

**Identifying the q order of MA model**

For MA models, the PACF will dampen exponentially and the ACF plot will be used to identify the order of the MA process. If we have one significant spike at lag 1 on the ACF, then we have an MA model of the order 1, i.e. MA(1). If we have significant spikes at lag 1, 2, and 3 on the ACF, then we have an MA model of the order 3, i.e. MA(3).

**Step 3: Estimation and Forecasting**

Once we have determined the parameters (p,d,q) we estimate the accuracy of the ARIMA model on a training data set and then use the fitted model to forecast the values of the test data set using a forecasting function. In the end, we cross check whether our forecasted values are in line with the actual values.

**How does auto.arima() work ?**

The auto.arima() function in R uses a variation of the [Hyndman and Khandakar](http://robjhyndman.com/papers/automatic-forecasting/) [algorithm](http://robjhyndman.com/papers/automatic-forecasting/) which combines unit root tests, minimization of the AICc and MLE to obtain an ARIMA model. The algorithm follows these steps.

1. The number of differences d is determined using repeated KPSS tests.
2. The values of p and q are then chosen by minimizing the AICc after differencing the data d times. Rather than considering every possible combination of p and q, the algorithm uses a stepwise search to traverse the model space.

a. The best model (with smallest AICc) is selected from the following four:

ARIMA(2,d,2),

ARIMA(0,d,0),

ARIMA(1,d,0),

ARIMA(0,d,1).

If d=0, then the constant c is included; if d≥1 then the constant c is set to zero. This is called the "current model".

b. Variations on the current model are considered

1. vary p and/or q from the current model by ±1; o include/exclude c from the current model.

The best model considered so far (either the current model, or one of these variations) becomes the new current model.

c. Repeat Step 2(b) until no lower AICc can be found.

**Holt-Winters seasonal method:**

Holt (1957) and Winters (1960) extended Holt’s method to capture seasonality. The Holt-Winters seasonal method comprises the forecast equation and three smoothing equations— one for the level ℓtℓt, one for trend bt, and one for the seasonal component denoted by st, with smoothing parameters α, β∗ and γ. We use mm to denote the period of the seasonality, i.e., the number of seasons in a year. For example, for quarterly data m=4m=4, and for monthly data m=12m=12.

There are two variations to this method that differ in the nature of the seasonal component. The additive method is preferred when the seasonal variations are roughly constant through the series, while the multiplicative method is preferred when the seasonal variations are changing proportional to the level of the series. With the additive method, the seasonal component is expressed in absolute terms in the scale of the observed series, and in the level equation the series is seasonally adjusted by subtracting the seasonal component. Within each year the seasonal component will add up to approximately zero. With the multiplicative method, the seasonal component is expressed in relative terms (percentages) and the series is seasonally adjusted by dividing through by the seasonal component. Within each year, the seasonal component will sum up to approximately mm.

**Holt-Winters additive method**

The component form for the additive method is:

y^t + h|t = ℓt + hbt + st − m + hm +

ℓt = α(yt − st − m) + ( − α)(ℓt − + bt − ) bt = β ∗ (ℓt − ℓt − ) + ( − β ∗)bt − st = γ(yt − ℓt − − bt − ) + ( − γ)st − m,

here hm +=⌊(h−1) , which ensures that the estimates of the seasonal indices used for forecasting come from the final year of the sample. (The notation ⌊u⌋ means the largest integer not greater than u.) The level equation shows a weighted average between the seasonally adjusted observation (yt−st−m) and the non-seasonal forecast (ℓt−1+bt−1)for time t. The trend equation is identical to Holt’s linear method. The seasonal equation shows a weighted average between the current seasonal index, (yt−ℓt−1−bt−1), and the seasonal index of the same season last year (i.e., m time periods ago).

The equation for the seasonal component is often expressed as

St = γ ∗ (yt − ℓt) + (1 − γ ∗) st − m

If we substitute ℓt from the smoothing equation for the level of the component form above, we get

t = γ ∗ (1 − α)(yt − ℓt − 1 − bt − 1) + [1 − γ ∗ (1 − α)]st − m

Which is identical to the smoothing equation for the seasonal component we specify here with γ=γ∗ (1−α). The usual parameter restriction is 0≤γ∗≤1, which translates to 0≤γ≤1−α. The error correction form of the smoothing equations is:

ℓ = ℓ − 1 + − 1 +

= − 1 + ∗

= − + ..

Where

et=yt−(ℓt−1+bt−1+st−m)=yt−y^t|t−1are the one-step training forecast errors.

**Holt-Winters multiplicative method**

The component form for the multiplicative method is:

y^t + h|t = (ℓt + hbt)st − m + h + m.

|  |  |  |  |
| --- | --- | --- | --- |
| yt | + (1 − α)(ℓt − 1 | + bt − 1) |  |
| ℓt = α st − m |  |

bt = β ∗ (ℓt − ℓt − 1) + (1 − β ∗)bt − 1

γyt

st = (ℓt − 1 + bt − 1) + (1 − γ)st − m

**Linear Programming:**

Linear programming is a simple technique where we **depict** complex relationships through linear functions and then find the optimum points. The important word in previous sentence is **depict**. The real relationships might be much more complex – but we can simplify them to linear relationships.

**Common terminologies used in Linear Programming**

* **Decision Variables:** The decision variables are the variables which will decide myoutput. They represent my ultimate solution. To solve any problem, we first need to identify the decision variables.
* **Objective Function:** It is defined as the objective of making decisions. **Constraints:** Theconstraints are the restrictions or limitations on the decision variables. They usually limit the value of the decision variables.
* **Non-negativity restriction:** For all linear programs, the decision variables shouldalways take non-negative values. Which means the values for decision variables should be greater than or equal to 0
* **Snapshot of the code**

1. **Holt-Winter’s Model Building:**

|  |
| --- |
| HoltWinters\_Plot **<-** **function(**Stock\_DF**){** par**(**mfrow **=** c**(**1,1**))**  StockTS **<-** ts**(**Stock\_DF**[**,1**]**, frequency **=** 365**)**  StockTS\_train **=** ts**(**StockTS**[**1**:**4741**]**, frequency **=** 365**)**  StockTS\_test **=** ts**(**StockTS**[**4742**:**4748**]**, frequency **=** 365**)**  ## Holt Winters model  Stock\_HW **<-** HoltWinters**(**StockTS\_train**)**  #Forecasting  Stock\_forecast\_HW **<-**  forecast.HoltWinters**(**Stock\_HW,h**=** 7**)**  #Plotting the forecast  plot.forecast**(**Stock\_forecast\_HW,shadecols**=**"oldstyle"**)**  **}** |

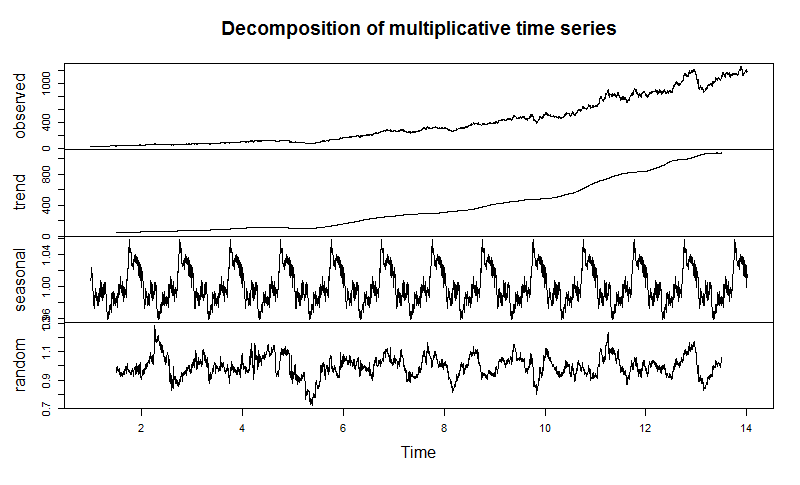
**2.Auto ARIMA Model Building:**

|  |
| --- |
| AutoArima\_Plot **<-** **function(**Stock\_DF**){**  par**(**mfrow **=** c**(**1,1**))**  StockTS **<-** ts**(**Stock\_DF**[**,1**]**, frequency **=** 365**)**  StockTS\_train **=** ts**(**StockTS**[**1**:**4741**]**, frequency **=** 365**)**  StockTS\_test **=** ts**(**StockTS**[**4742**:**4748**]**, frequency **=** 365**)**  ## AutoArima model  LogARIMA **<-** auto.arima**(**log10**(**StockTS\_train**))**  #Forecasting  Stock\_forecast\_Arima **<-**  forecast.Arima**(**LogARIMA,h**=** 7**)**  #Plotting the forecast  plot.forecast**(**Stock\_forecast\_Arima,shadecols**=**"oldstyle"**)}** |

* **Result Snapshot and its description**

**ASIAN PAINTS PLOTS**

**Decomposition:**



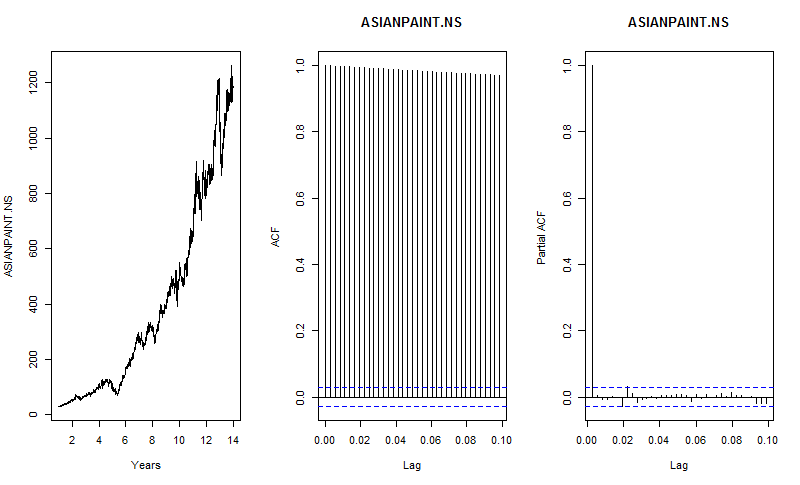
The observed plot and trend suggests that the stock value of Asian Paints started to rise from the year 2009. Some of the key assumptions causing this behavior are:

UPA retaining its power in 2009 to form Union Government with a very high positive note hoping to increase the infrastructure projects might have helped the Asian Paint’s stock to rise.

Also, the scope of building constructions increase in India happened in second half of first decade of the millennium (From 2006), which could have helped increasing the stock value of Asian Paints Limited.

There is a significant spike in 2014 and also major drop in 2016, but the stcok value is again picked at the later stages of 2017

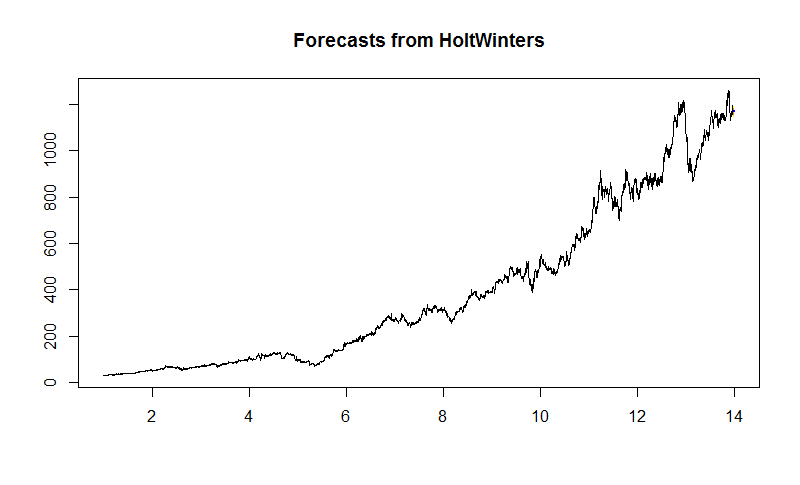
**ACF and PACF Plots:**

****

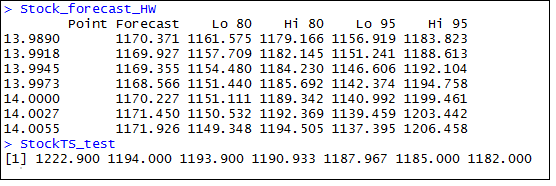
The ACF plot indicates that the data is highly correlated with one lag, but definitely decreasing at a very slow pace which indicates that data is non-stationary.

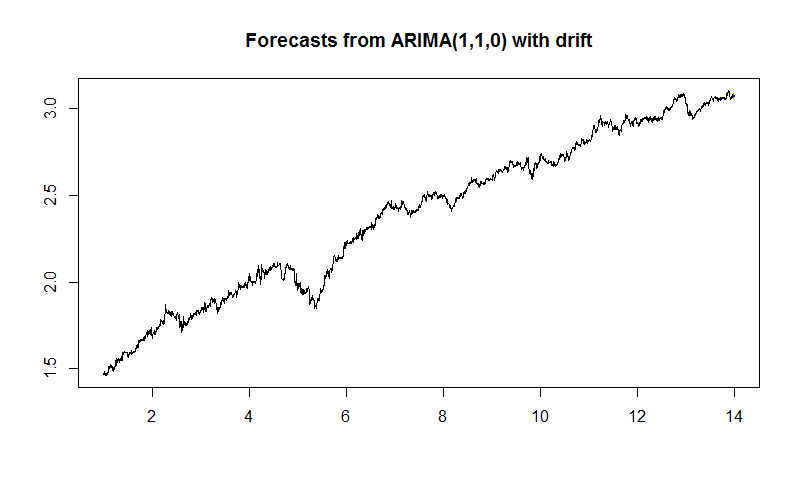
In PACF plot, Note that the first lag value is statistically significant, whereas partial autocorrelations for all other lags are not statistically significant.  This suggests a possible AR(1) model for these data.

**Holt-Winters and ARIMA Forecasting**:

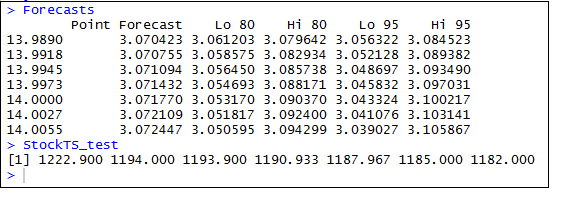
****

**FORECASTED VALUES vs ACTUAL VALUES (HOLT WINTERS):**



****

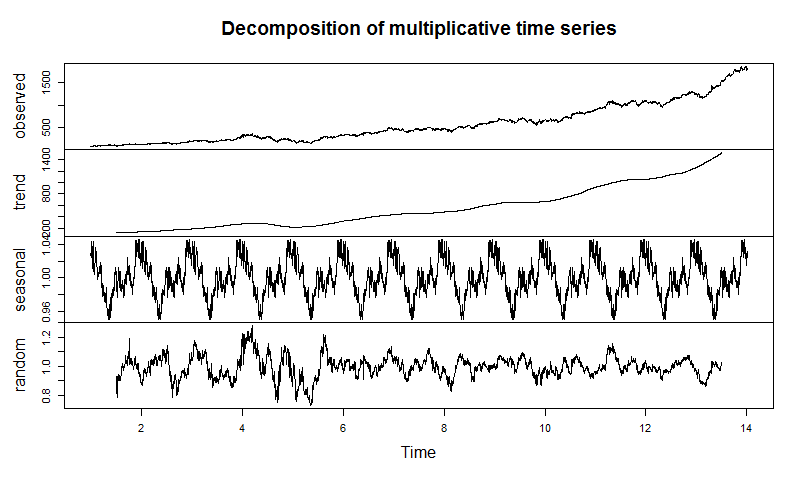
**FORECASTED VALUES vs ACTUAL VALUES (AUTO ARIMA):**

****

The forecasted values in Auto.Arima model has to be converted using 10^forecasted values, as log10 was applied to the original data while applying the model on the data set.

**HDFC BANK PLOTS**

**Decomposition:**

****

The observed and trend plots suggest that the HDFC bank stock prices significantly started to rise post year 2005.

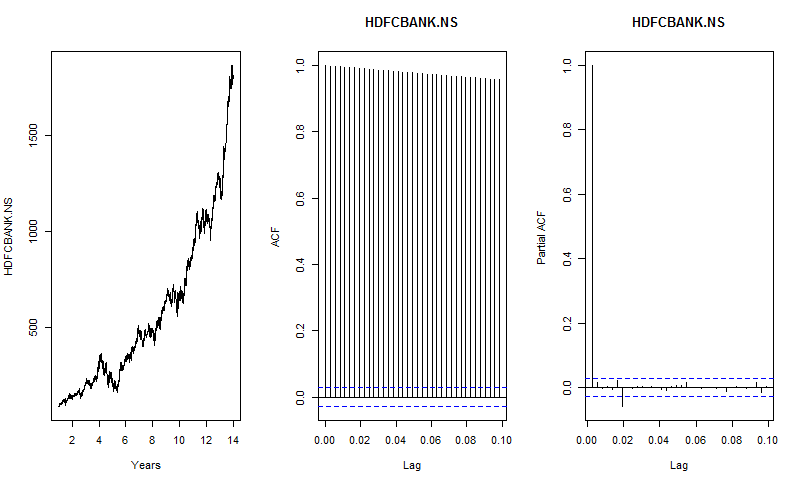
Some of the key assumptions are:

Plots suggest that year 2008 had increased drastically in the year 2008 as HDFC bank acquired Centurion bank in the same year which is considered as one of the largest mergers in the financial sectors of India.

Increase in the private sectors of India and HDFC Bank being a larger player in supporting those ventures for banking operations which has in turn has increased the customer base of the HDFC bank. This could possibly be one of the reasons why HDFC bank stocks have raised consistently in the last 7-9 years.

Easy allocation of credits to their customer has generated large returns to HDFC bank, which might have also increased their profits base.

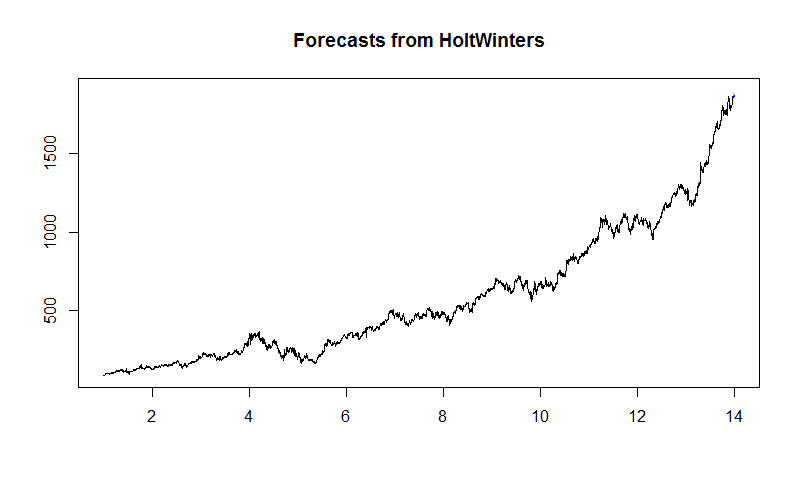
**ACF and PACF Plots:**



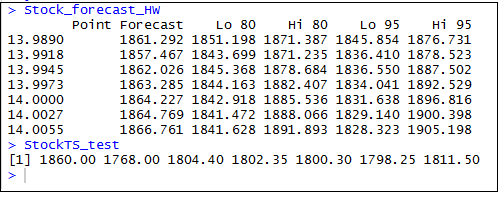
The ACF plot indicates that the data is highly correlated with one lag, but definitely decreasing at a very slow pace which indicates that data is non-stationary.

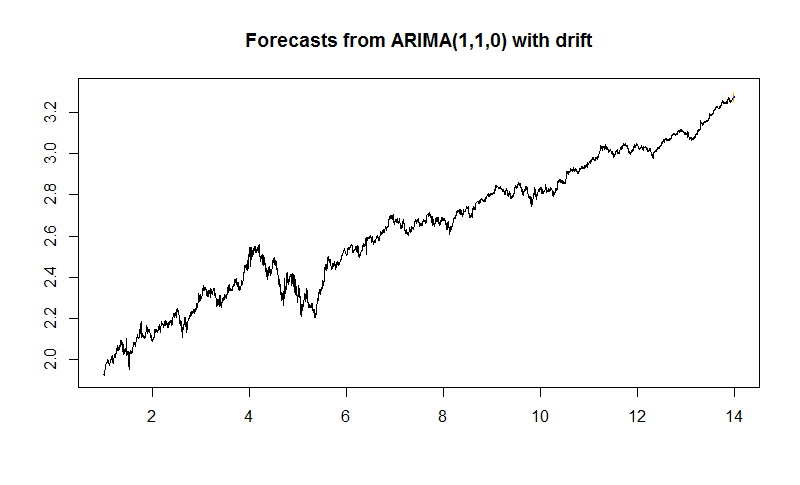
In PACF plot, Note that the some lag values are statistically significant, whereas partial autocorrelations for all other lags are not statistically significant. Since there is one negative and positive correlation, this suggests a possible AR (0) model for these data.

**Holt-Winters and ARIMA Forecasting**:

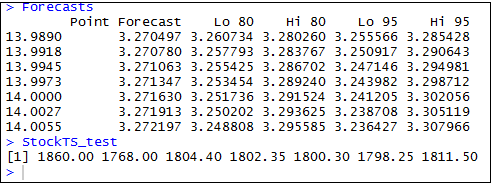


**FORECASTED VALUES vs ACTUAL VALUES (HOLT WINTERS):**

****

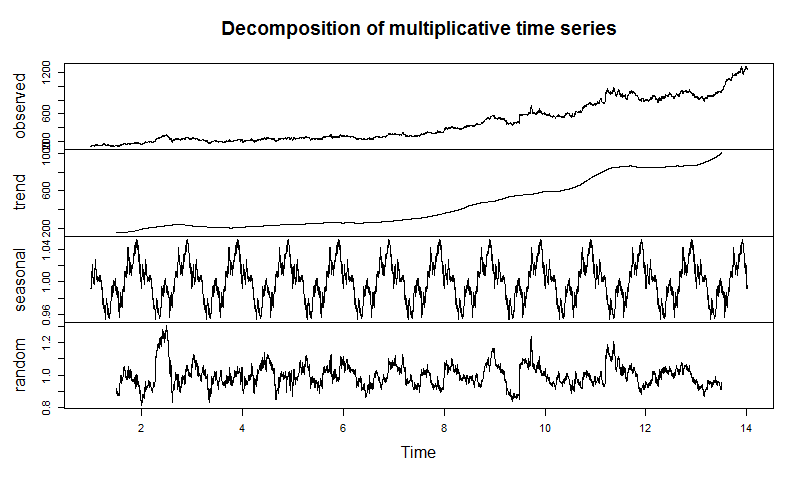


**FORECASTED VALUES vs ACTUAL VALUES (AUTO ARIMA):**



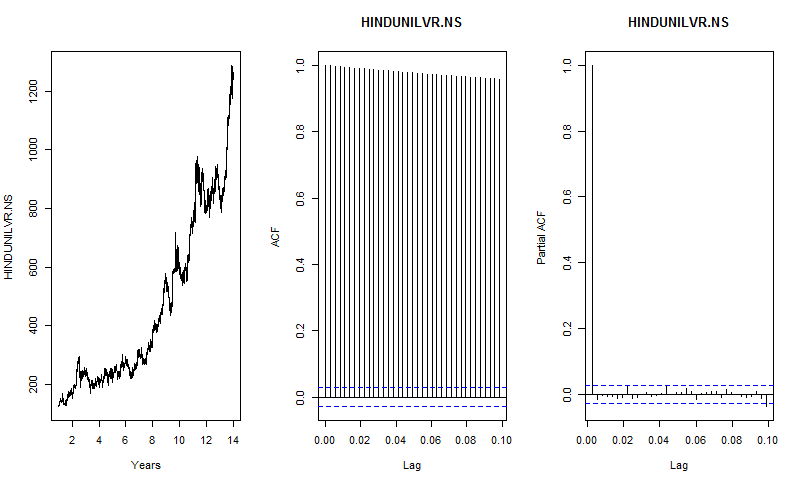
**HINDUSTAN UNILEVER PLOTS**

**Decomposition:**



The Observed and Trend plots suggest that the Hindustan Unilever Limited stock prices actually performed pretty consistently without any drastic rise or fall down between the years 2004 – 2011, except for the year 2007, when the company was renamed to “Hindustan Unilever Limited” from “Hindustan Lever Limited”.

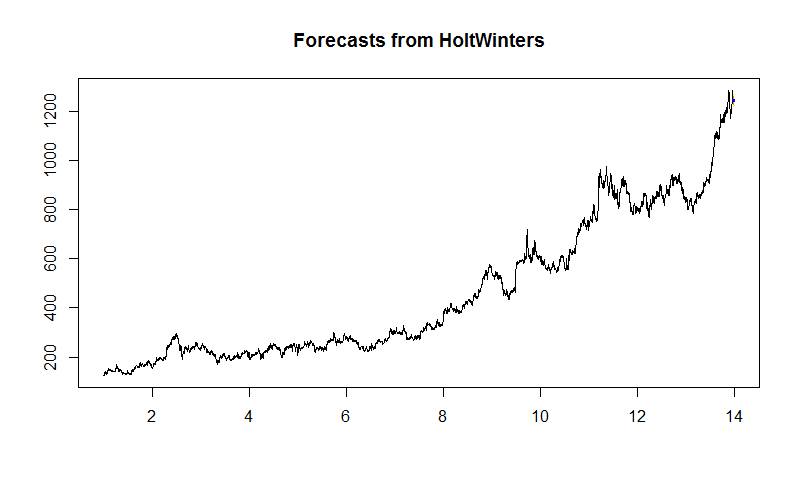
**ACF and PACF Plots:**



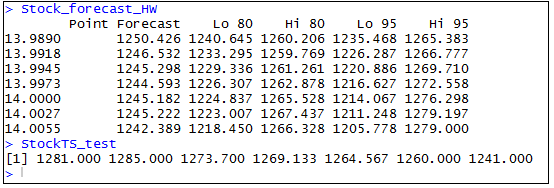
The ACF plot indicates that the data is highly correlated with one lag, but definitely decreasing at a very slow pace which indicates that data is non-stationary.

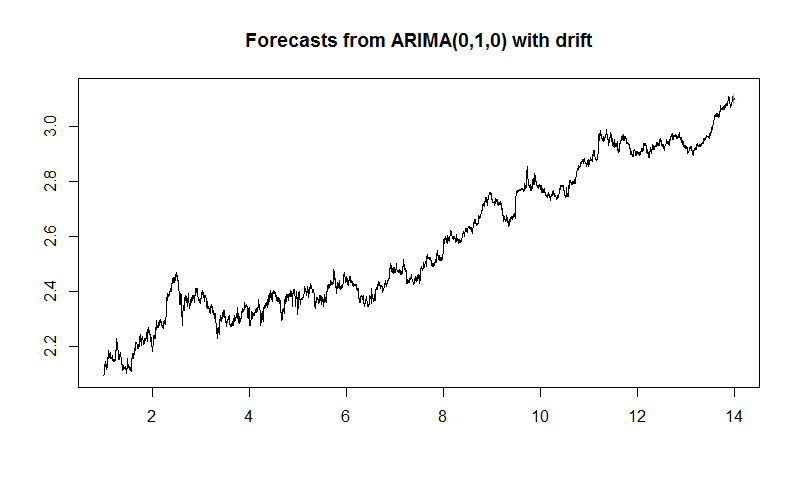
In PACF plot, Note that the some lag values are statistically significant, whereas partial autocorrelations for all other lags are not statistically significant. Since there is one negative and positive correlation, this suggests a possible AR (0) model for these data.

**Holt-Winters and ARIMA Forecasting:**

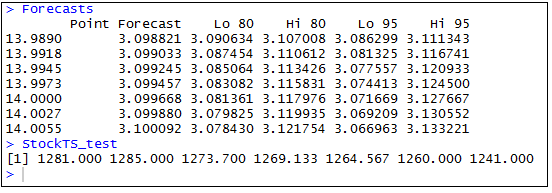
****

**FORECASTED VALUES vs ACTUAL VALUES (HOLT WINTERS):**

****

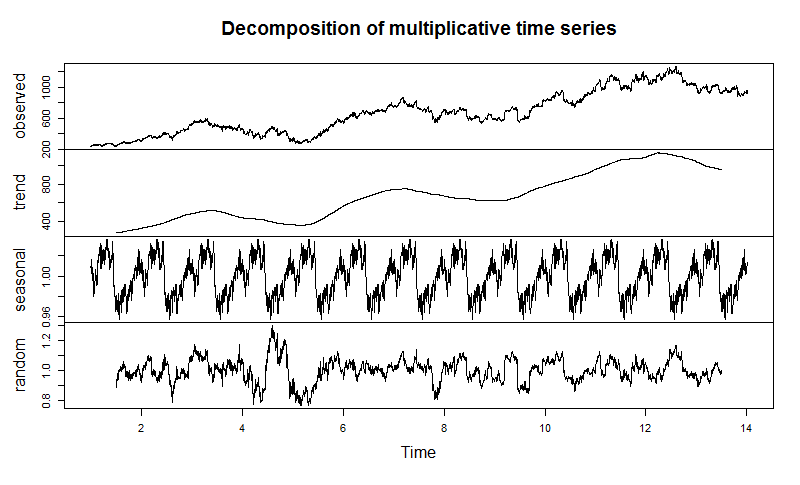
****

**FORECASTED VALUES vs ACTUAL VALUES (AUTO ARIMA):**

****

**INFOSYS PLOT:**

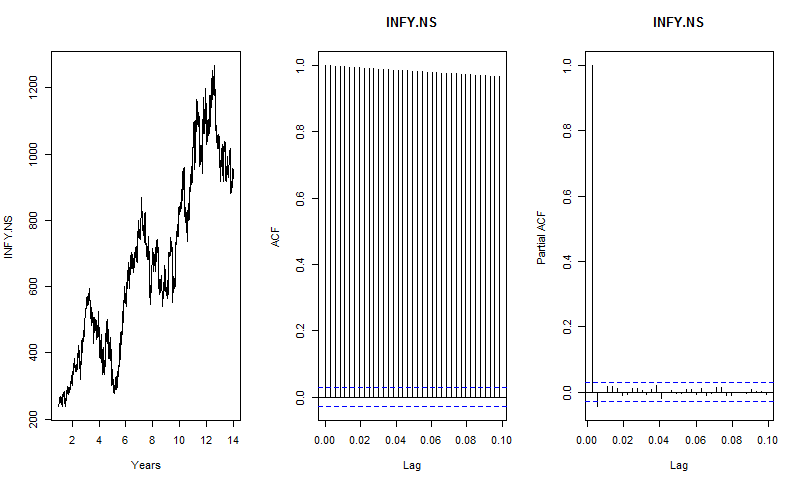
**Decomposition:**

****

The Observed and Trend plots suggest the following things:

It looks like the recession period from second half of 2007 to 2009 had larger effect on stock prices of Infosys and the stock prices were very low.It looks like the change in the leadership in the year 2014, where in for the first time a non-founder member of Infosys was made to lead the office has great impact in the increase of stock prices.

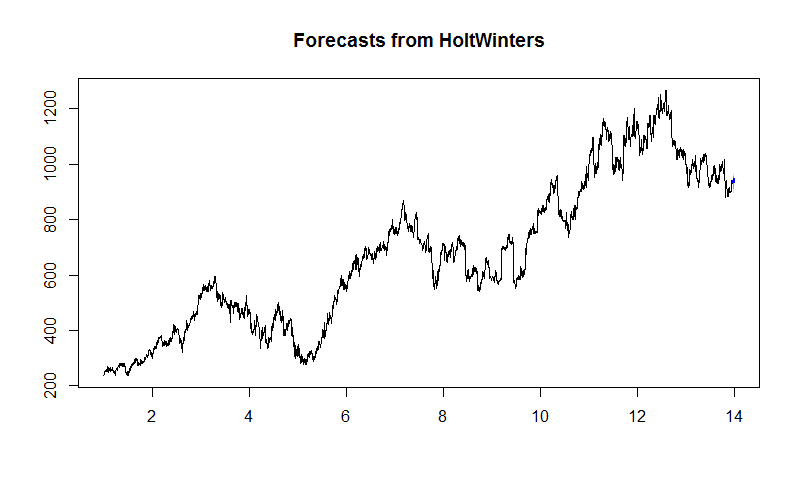
**ACF and PACF Plot:**



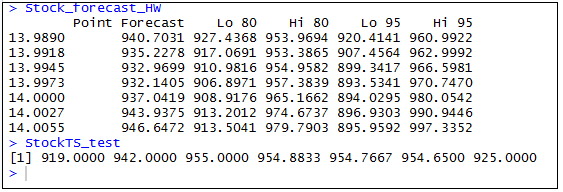
The ACF plot indicates that the data is highly correlated with one lag, but definitely decreasing at a very slow pace which indicates that data is non-stationary.

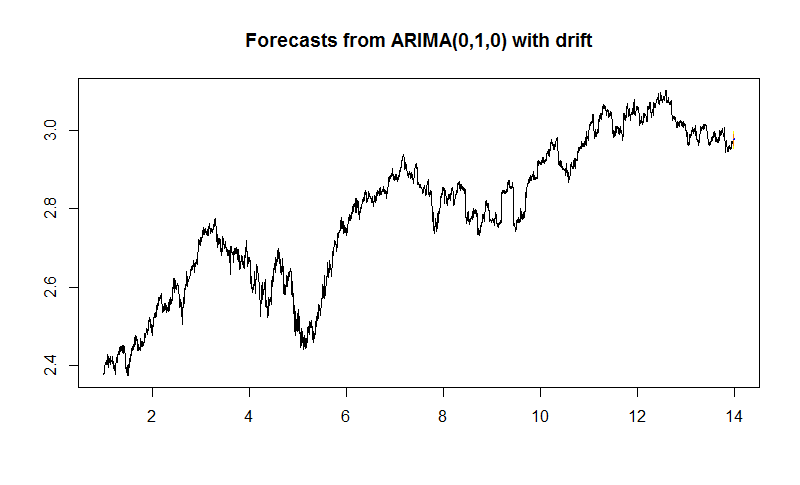
In PACF plot, Note that the first and second lag values are statistically significant, whereas partial autocorrelations for all other lags are not statistically significant. Since there is one negative and positive correlation, this suggests a possible AR (0) model for these data.

**Holt-Winters and ARIMA Forecasting:**

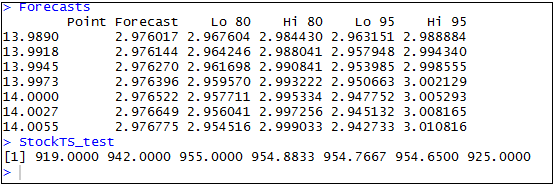


**FORECASTED VALUES vs ACTUAL VALUES (HOLT WINTERS):**



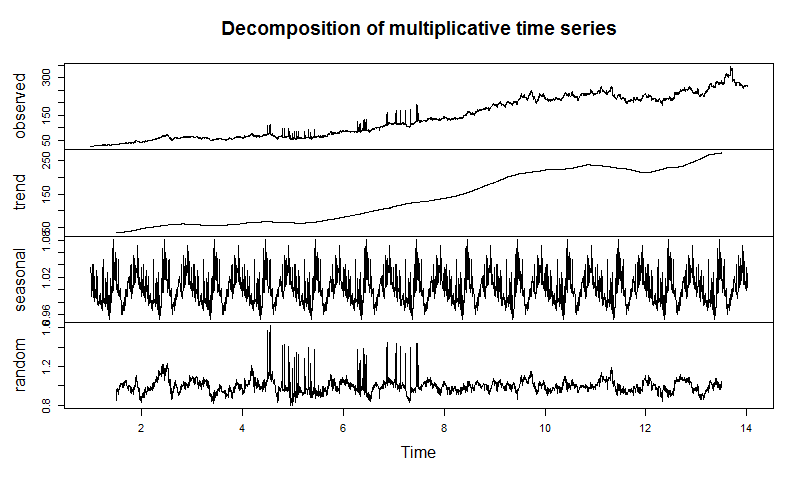


**FORECASTED VALUES vs ACTUAL VALUES (AUTO ARIMA):**

****

**ITC Plots**

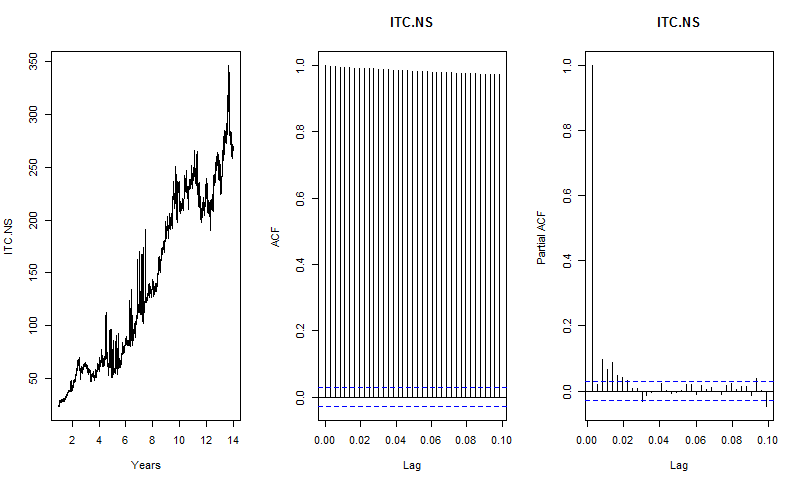
**Decomposition:**



The Observed and Trend plots suggest that the stock prices of ITC started to increase from the year 2008 and have a slight fall down in the year 2014 which was not too long before it started to increase.

One more assumption is that it had a lot of seasons between the years 2007 – 2011.

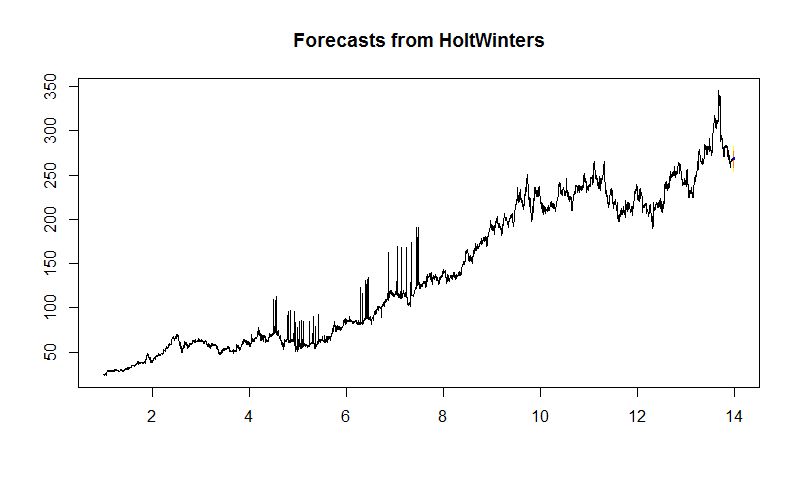
**ACF and PACF Plots:**



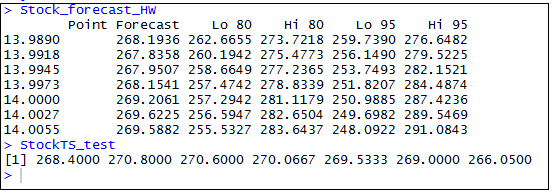
The ACF plot indicates that the data is highly correlated with one lag, but definitely decreasing at a very slow pace which indicates that data is non-stationary.

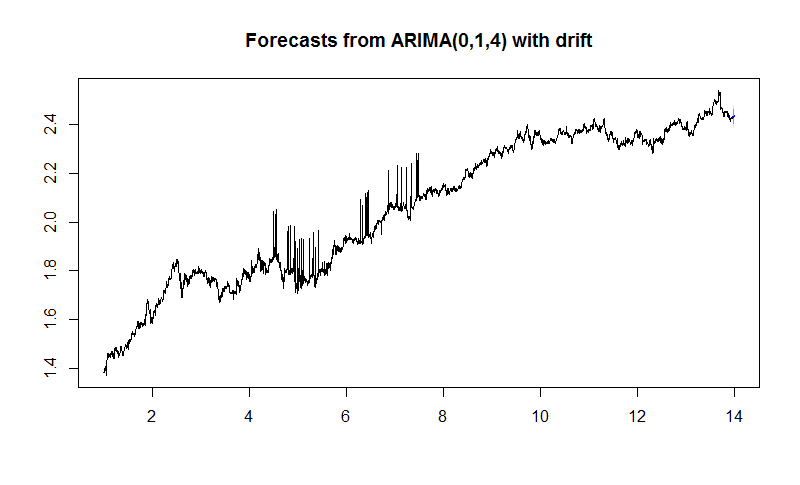
In PACF plot, Note that the many lag values are statistically significant, whereas partial autocorrelations for all other lags are not statistically significant. There is negative correlation for one lag.  This suggests a possible AR (4) model for these data.

**Holt-Winters and ARIMA Forecasts:**

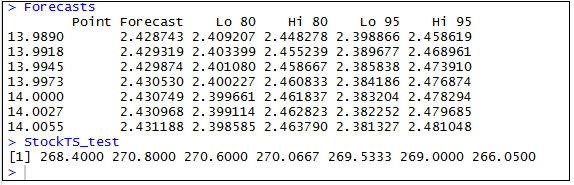


**FORECASTED VALUES vs ACTUAL VALUES (HOLT WINTERS):**

****

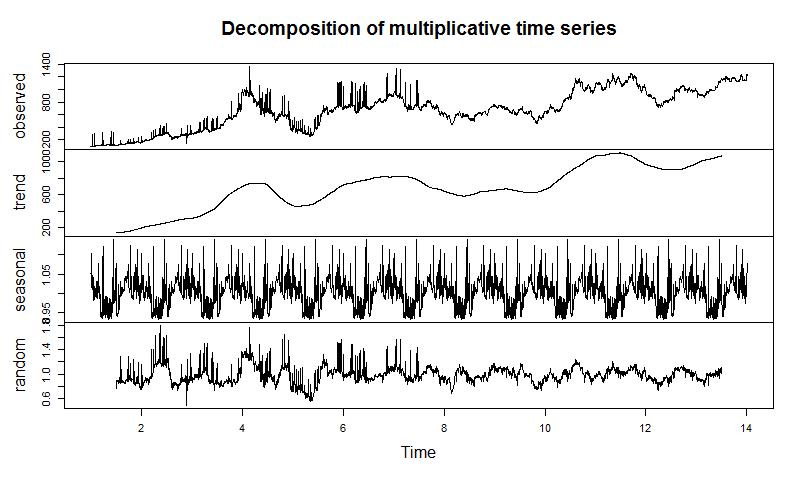


**FORECASTED VALUES vs ACTUAL VALUES (AUTO ARIMA):**

****

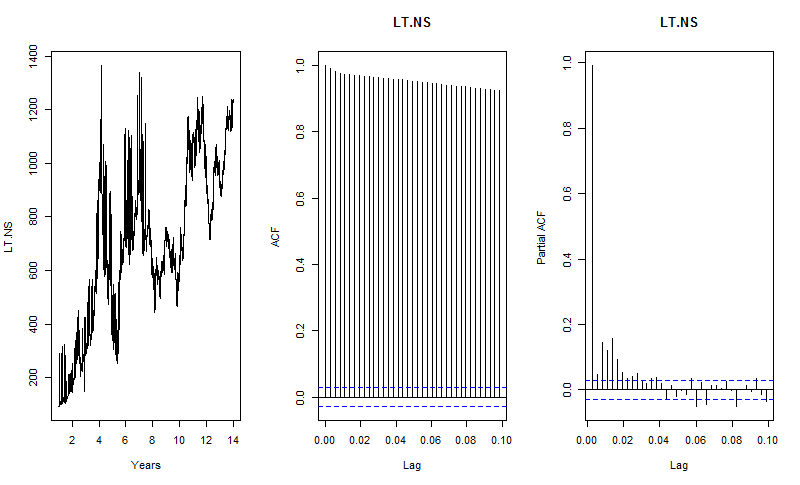
**L&T Plots:**

**Decomposition:**



The Observed & Trend plots suggest that L&T stock prices are very inconsistent in nature. As we can observe, there were too many seasons during the years 2004 to end of year 2010.

**ACF and PACF Plots:**



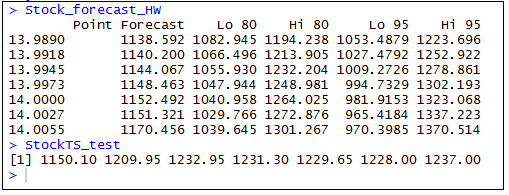
The ACF plot indicates that the data is highly correlated with one lag, but definitely decreasing at a very slow pace which indicates that data is non-stationary.

In PACF plot, Note that the many lag values are statistically significant, whereas partial autocorrelations for all other lags are not statistically significant. There are negative correlation as well for some of the lags..  This suggests a possible AR (4) model for these data.

**Holt-Winter’s and ARIMA Forecasting:**

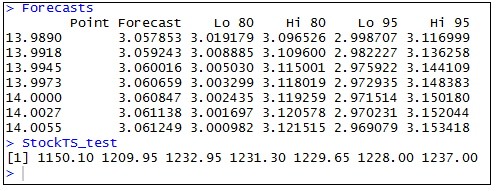
****

**FORECASTED VALUES vs ACTUAL VALUES (HOLT WINTERS):**

****

****

**FORECASTED VALUES vs ACTUAL VALUES (AUTO ARIMA):**

****

**MARUTI SUZUKI Plots:**

**Decomposition:**



The Observed and Trend plots suggest a very high consistent behavior of Maruti Suzuki stock prices from the year 2004 to 2012. But post 2012, there is a huge spike in its stock prices.

There is also a random fall down of stock prices in the year 2008, which might be due to the selling of shares of Union Government of India in Maruti Suzuki.

**ACF and PACF Plots:**



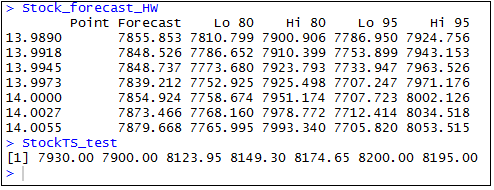
The ACF plot indicates that the data is highly correlated with one lag, but definitely decreasing at a very slow pace which indicates that data is non-stationary.

In PACF plot, Note that the many lag values are almost statistically significant, whereas partial autocorrelations for all other lags are not statistically significant. There is negative correlation for one lag.  This suggests a possible AR (2) model for these data.

**Holt-Winter’s and ARIMA forecasting:**

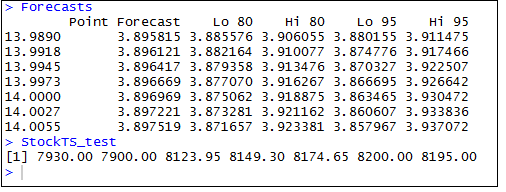


**FORECASTED VALUES vs ACTUAL VALUES (HOLT WINTERS):**





**FORECASTED VALUES vs ACTUAL VALUES (AUTO ARIMA):**



**RELIANCE Plots:**

**Decomposition:**



The Observed and Trend plots suggest that Reliance stocks were very consistent during the years 2011 to 2016. There was a significant increase in the year 2005 and slight fall down in the year 2008 before performing consistently from then.

There were lots of seasons during the years 2008 to 2011.

**ACF and PACF Plots:**



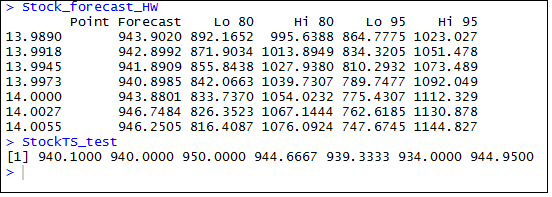
The ACF plot indicates that the data is highly correlated with one lag, but definitely decreasing at a very slow pace which indicates that data is non-stationary.

In PACF plot, Note that the many lag values are statistically significant, whereas partial autocorrelations for all other lags are not statistically significant. There are negative correlations as well for some of the lags.  This suggests a possible AR (4) model for these data.

**Holt-Winter’s and ARIMA Forecasts:**

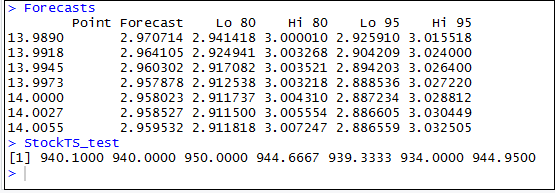
****

**FORECASTED VALUES vs ACTUAL VALUES (HOLT WINTERS):**

****

****

**FORECASTED VALUES vs ACTUAL VALUES (AUTO ARIMA):**

****

**TCS Plots:**

**Decomposition:**



The observed and trend plots suggest that TCS stock prices are very consistent in nature. Till 2008 the stock prices were very low, but post that the trend looks only increasing in nature.

There was a random drop down in the year 2008, but might be due to the recession in the Indian Job market during that period.

It is possible that due to attacks on Taj hotels Mumbai, the Tata group companies might have faced a loss in their stock prices.

**ACF and PACF Plots:**



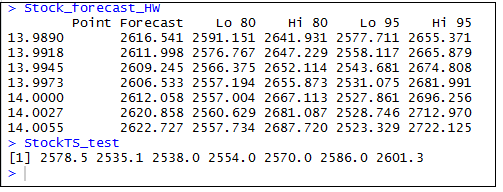
The ACF plot indicates that the data is highly correlated with one lag, but definitely decreasing at a very slow pace which indicates that data is non-stationary.

In PACF plot, Note that the some values in the centre are statistically significant, whereas partial autocorrelations for all other lags are not statistically significant. This suggests a possible AR (2) model for these data.

**Holt-Winter’s and ARIMA Forecast:**

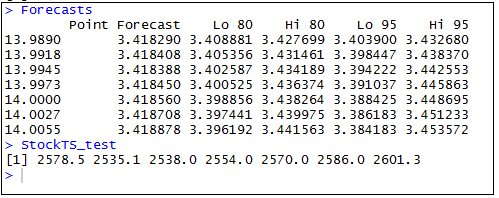


**FORECASTED VALUES vs ACTUAL VALUES (HOLT WINTERS):**





**FORECASTED VALUES vs ACTUAL VALUES (AUTO ARIMA):**

****

**KOTAK MAHINDRA BANK Plots:**

**Decomposition:**



The observed and trend plots suggest that the stocks of Kotak Mahindra bank is very consistent in nature. The stock prices have relatively increased from the year 2015. This might be due to the acquisition on ING Vysya bank by Kotak Mahindra Bank in the same year.

It looks like there was a random increase in the stock prices during the year 2007 and random decrease in the year 2008.

**ACF and PACF Plots:**



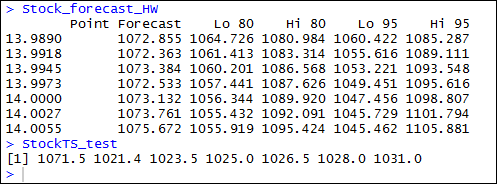
The ACF plot indicates that the data is highly correlated with one lag, but definitely decreasing at a very slow pace which indicates that data is non-stationary.

In PACF plot, Note that the first and second lag values are statistically significant, whereas partial autocorrelations for all other lags are not statistically significant. Since there is one negative and positive correlation, this suggests a possible AR (0) model for these data.

**Holt-Winter’s and ARIMA Forecast:**

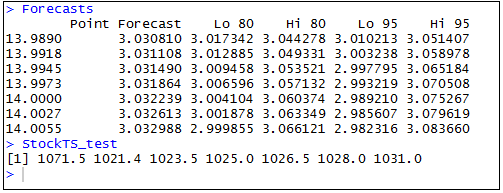


**FORECASTED VALUES vs ACTUAL VALUES (HOLT WINTERS):**

****



**FORECASTED VALUES vs ACTUAL VALUES (AUTO ARIMA):**

****

**Analysis:**

**Stock Market Prediction:**

The error metric used to calculate the error of the forecast here is MAPE (Mean Absolute Percentage Error), which is defined as:

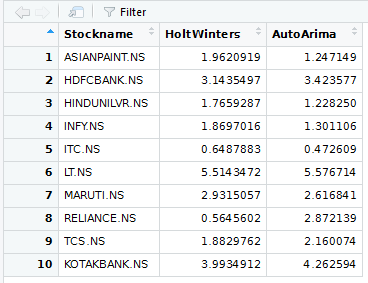
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| = |  |  | ∑ | Y − Y^| | ∗ 100 |  |
|  |  |  |  |
|  |  | Y |  |

Where y is the actual value and y^ is the forecasted value.

MAPE of all the 10 stocks for both Holt-Winter’s and ARIMA is given below:

**MAPE TABLE:**

The below MAPE table displays the error percentage of each of the stock’s forecasting values using HoltWinters and Auto Arima techniques. We can infer that the models have performed well considering the fact that all the error percentages are very minimal in nature:

******

The error percentages of the each of the stock market prices are relatively less in nature. This could have been reduced to certain extent if we could have trained our model on recent data rather than very old data, as we could have ignored unnecessary seasons, trends or randomness which could have had no significance as of today.

**PORTFOLIO OPTIMIZATION:**

Calculating the yearly returns:

#Creating a table to store yearly returns colClasses **=** c**(**"character", "double"**)** col.names **=** c**(**"Stockname", "Returns"**)**

Returns\_Table **<-** read.table**(**text **=** "", colClasses **=** colClasses, col.names **=** col.names**)**

#List of all the stocks

StockList <- list(ASIANPAINT.NS = ASIANPAINT.NS, HDFCBANK.NS = HDFCBANK.NS, HINDUNILVR.NS = HINDUNILVR.NS, INFY.NS = INFY.NS, ITC.NS = ITC.NS,LT.NS = LT.NS, MARUTI.NS = MARUTI.NS,RELIANCE.NS = RELIANCE.NS,TCS.NS = TCS.NS,KOTAKBANK.NS = KOTAKBANK.NS)

# Calculating the yearly stock returns

**for (**i **in** 1**:**length**(**StockList**)){**

#Storing the name of the stock in the 1st column Returns\_Table **[**i,1**] <-** names**(**StockList**)[**i**]**

* Converting the Stock to xts format and extracting Open Value **Open <- xts(ModelDataList[[i]][,1])**

#Calculating Mean Yearly Return

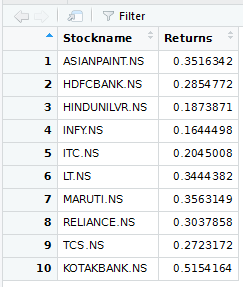
Return **=** mean**(**yearlyReturn**(**Open**))**

#Storing the results in the table

Returns\_Table **[**i,2**]<-** Return

**}**

Yearly Returns calculated:



A snapshot of the code:-

library**(**"lpSolve"**)**

obj = c(0.5154,0.3563,0.3516,0.3444,0.3037)

con = rbind(c(1,0,0,0,0),

c(0,1,0,0,0),

c(0,0,1,0,0),

c(0,0,0,1,0),

c(0,0,0,0,1),

c(1,1,1,1,1))

dir = c(">=", ">=", ">=", ">=", ">=", "=")

rhs = c(1000, 1000, 1000, 1000, 1000, 6000)

res = lp("max", obj, con, dir, rhs, compute.sens=1)

res$solution

The Solution achieved through Linear Programming to invest say available 6000 INR into the best performing stocks is to invest 2000 INR in Kotak Mahindra Bank and rest 4000 to be invested equally in the remaining 4 best performing stocks chosen here.

**How to implant it??**

Continuously training the model using the latest market data, so that the model gets trained on the latest market behavior as well. It can also be deployed for more no of portfolio's in future, using the current data set and then the trend can be analysed for the same to predict better investment plans.

**Social Impact**

"Helping people optimally invest in the stocks , depending upon the present market analysis, which would help them fetch maximum return and suffer comparitively lesser loss!"