Comparative Study of Time Series Models on Stock Data

Gourav Patidar (150258)
Akash Gupta (150069)
Sunny Kant (150740)

1. Abstract:

The idea of this project is to analyze the historical stock data for DJIA 30 companies using basic and advanced time series models such as ARIMA, VAR, Holt Winters exponential smoothing, gradient boosting models and other non-linear models. The main stress is given upon a new method of gradient boosted ARIMA model. The broader outline is to go through a comparative study through various models and to decide on the better one only after checking on the underlying assumptions of each model.

2. Theory:

● Time Series Modelling :-

The main aim of time series modeling is to carefully collect and rigorously study the past observations of a time series to develop an appropriate model which describes the inherent structure of the series. This model is then used to generate future values for the series, i.e. to make forecasts. Time series forecasting thus can be termed as the act of predicting the future by understanding the past.

● Time Series :-

A time series is a sequential set of data points, measured typically over successive times. It is mathematically defined as a set of vectors x(t),t = 0,1,2,... where t represents the time elapsed. The variable x(t) is treated as a random variable. The measurements taken during an event in a time series are arranged in a proper chronological order.

- ## Decomposition of Time Series :-

Additive Model :- $Y(t) = T(t) + S(t) + C(t) + I(t)$
Here Y(t) is the observation and T(t), S(t), C(t) and I(t) are respectively the trend, seasonal, cyclical and irregular variation at time .

- ## Autoregressive Integrated Moving Average (ARIMA) Models

Given a time series of data X_t where t is an integer index and the X_t are real numbers, and ARMA(p,q) model is given by :-

$$X_t - \alpha_1 X_{t-1} - \cdots - \alpha_{p'} X_{t-p'} = \varepsilon_t + \theta_1 \varepsilon_{t-1} + \cdots + \theta_q \varepsilon_{t-q}$$

or equivalently by

$$\left(1 - \sum_{i=1}^{p'} \alpha_i L^i\right) X_t = \left(1 + \sum_{i=1}^{q} \theta_i L^i\right) \varepsilon_t$$

An ARIMA(p,d,q) process expresses this polynomial factorisation property with $p=p'-d$, and is given by:

$$\left(1 - \sum_{i=1}^{p} \phi_i L^i\right) (1 - L)^d X_t = \left(1 + \sum_{i=1}^{q} \theta_i L^i\right) \varepsilon_t$$

- ## Holt's Winter Model :-

The Holt-Winters seasonal method comprises the forecast equation and three smoothing equations — one for the level ℓ(t), one for the trend b(t), and one for the seasonal component s(t), with corresponding smoothing parameters α, β∗ and γ. We use m to denote the frequency of the seasonality, i.e., the number of seasons in a year.

$$\begin{aligned}
\hat{y}_{t+h|t} &= \ell_t + hb_t + s_{t+h-m(k+1)} \\
\ell_t &= \alpha(y_t - s_{t-m}) + (1-\alpha)(\ell_{t-1} + b_{t-1}) \\
b_t &= \beta^*(\ell_t - \ell_{t-1}) + (1-\beta^*)b_{t-1} \\
s_t &= \gamma(y_t - \ell_{t-1} - b_{t-1}) + (1-\gamma)s_{t-m},
\end{aligned}$$

- Gradient Boosted ARIMA

Here we tried to mix the traditional approach of ARIMA with extreme gradient boosting approach. We tried on implementing a gradient boosted approach for ARIMA where we tried to derive the importance for each lag using boosting method and did the forecast We used on an R package forecastxgb which will adapt this approach to extreme gradient boosting, popularly implemented by the astonishingly fast and effective xgboost algorithm.

3. Dataset:

Stock market data can be interesting to analyze and as a further incentive, strong predictive models can have large financial payoff. There is huge amount of financial data on the web. Here we have a dataset with historical stock prices (last 12 years) for 29 of 30 DJIA companies. Input file contains 13 years of stock data (all_stocks_2006-01-01_to_2018-01-01.csv) and have the following columns: Date, Open (price of the stock at market open, in USD), High (Highest price reached in the day), Low Close (Lowest price reached in the day), Volume (Number of shares traded) and Name (the stock's ticker name)

Date	Open	High	Low	Close	Volume	Name
1/3/2006	77.76	79.35	77.24	79.11	3117200	MMM
1/4/2006	79.49	79.49	78.25	78.71	2558000	MMM
1/5/2006	78.41	78.65	77.56	77.99	2529500	MMM
1/6/2006	78.64	78.9	77.64	78.63	2479500	MMM
1/9/2006	78.5	79.83	78.46	79.02	1845600	MMM
1/10/2006	79	79.01	78.08	78.53	1919900	MMM
1/11/2006	78.44	78.66	77.84	78.37	1911900	MMM

Fig. : Data for the company 3M with ticker symbol MMM

Fig. : Graphical Representation of a few stocks from the data

4. Model fitting and Predictions :-

- Stationarity :-

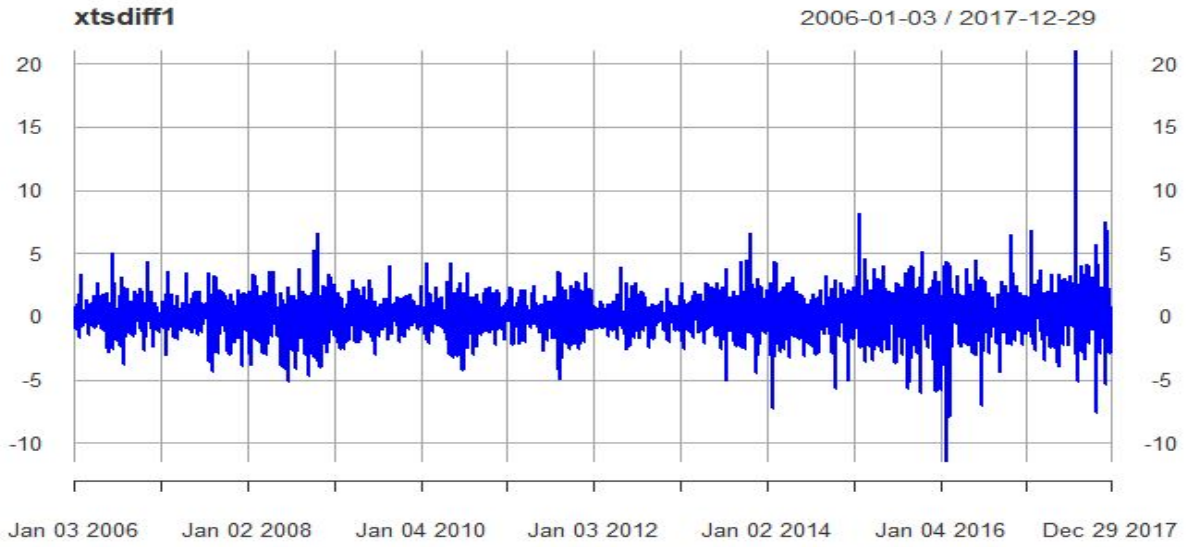
A stochastic process is called *stationary* if the mean is constant and variance does not depend on time(i.e., their joint distribution does not change over time). we will utilize the **Augmented Dickey-Fuller Test** for stationarity. The null hypothesis states that large *p values* indicate non-stationarity and smaller *p* values indicate stationarity. (We will be using 0.05 as our alpha value. We saw our *p* value for the ADF test is relatively high. For that reason, we did difference our time series for stationarity.

```
p-value greater than printed p-value
        Augmented Dickey-Fuller Test

data:  xts
Dickey-Fuller = 1.7367, Lag order = 0, p-value = 0.99
alternative hypothesis: stationary
```

As the p-value for the ADF Test is too high the series is clearly non-stationary, so we tried differencing approach for getting a stationary series. We have considered stock data of Boeing Airways below.

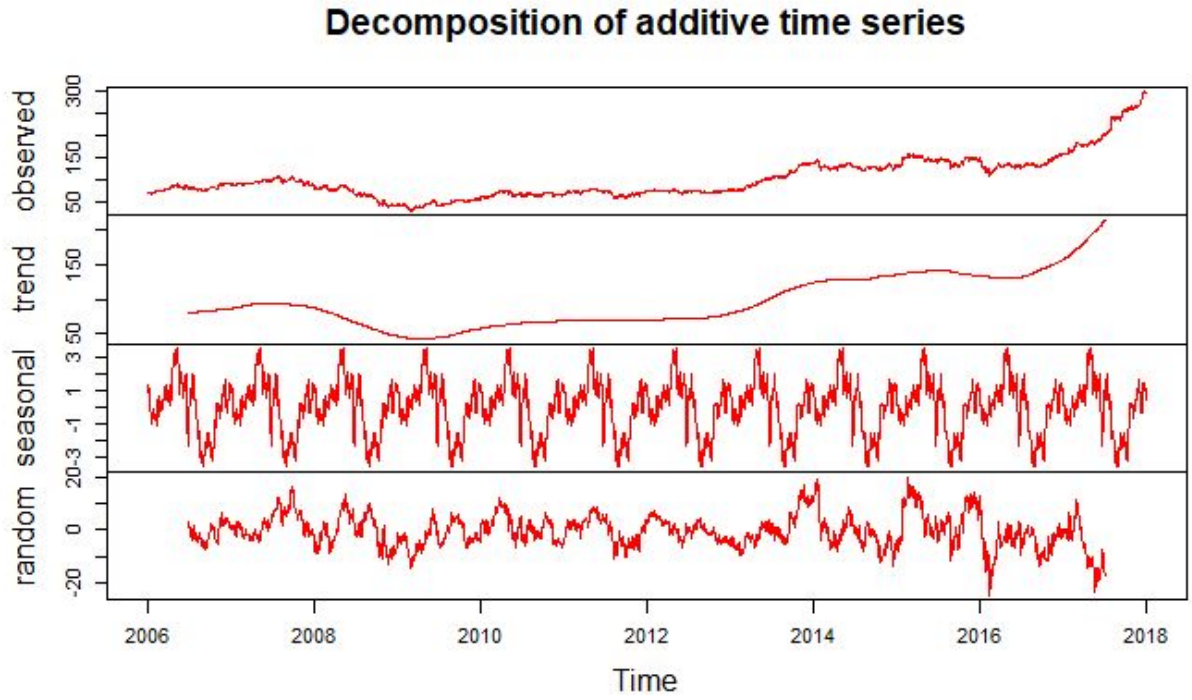
```
p-value smaller than printed p-value
        Augmented Dickey-Fuller Test

data:  tsdiff1
Dickey-Fuller = -52.941, Lag order = 0, p-value = 0.01
alternative hypothesis: stationary
```

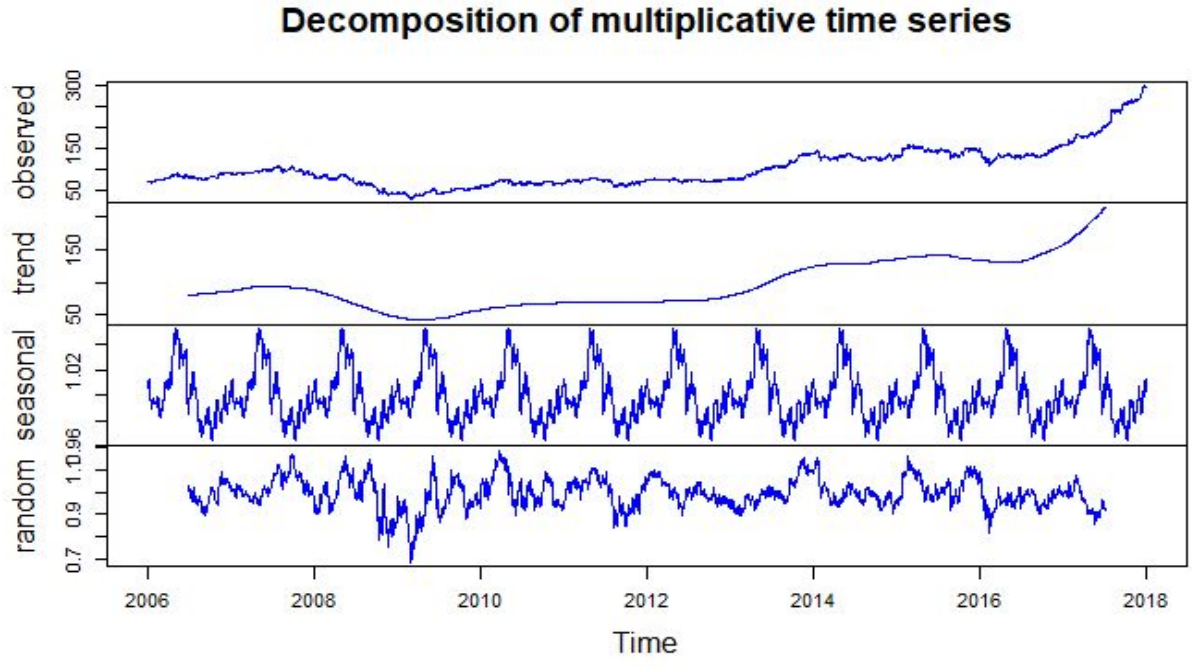
Now as the p-value is 0.01 which is less than 0.05 the differenced series is stationary.



- Decomposition :-

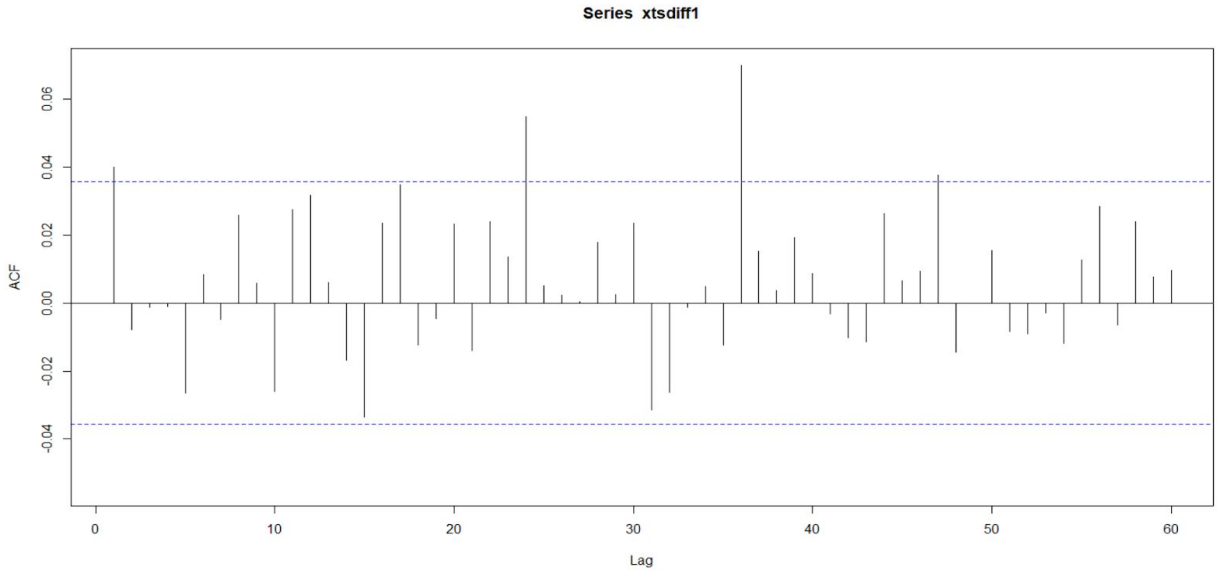
Beyond understanding the *trend* of your time series, you want to further understand the anatomy of your data. For this reason, we will break down our time series into its *seasonal component*, *trend*, and *residuals*.

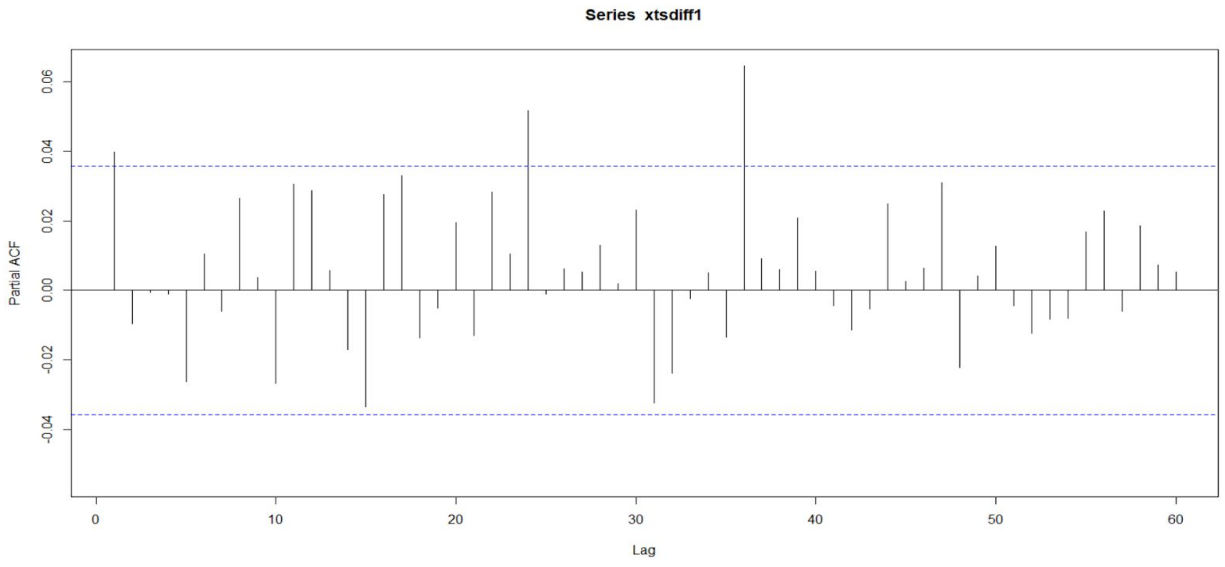




- ACF and PACF plots :-

ACF stands for "autocorrelation function" and *PACF* stands for "partial autocorrelation function." The *ACF* and *PACF* diagnosis is employed over a time-series to determine the order in which we are going to create our model using *ARIMA* modeling. Loosely speaking, a time series is *stationary* when its mean, variance, and *autocorrelation* remain constant over time.





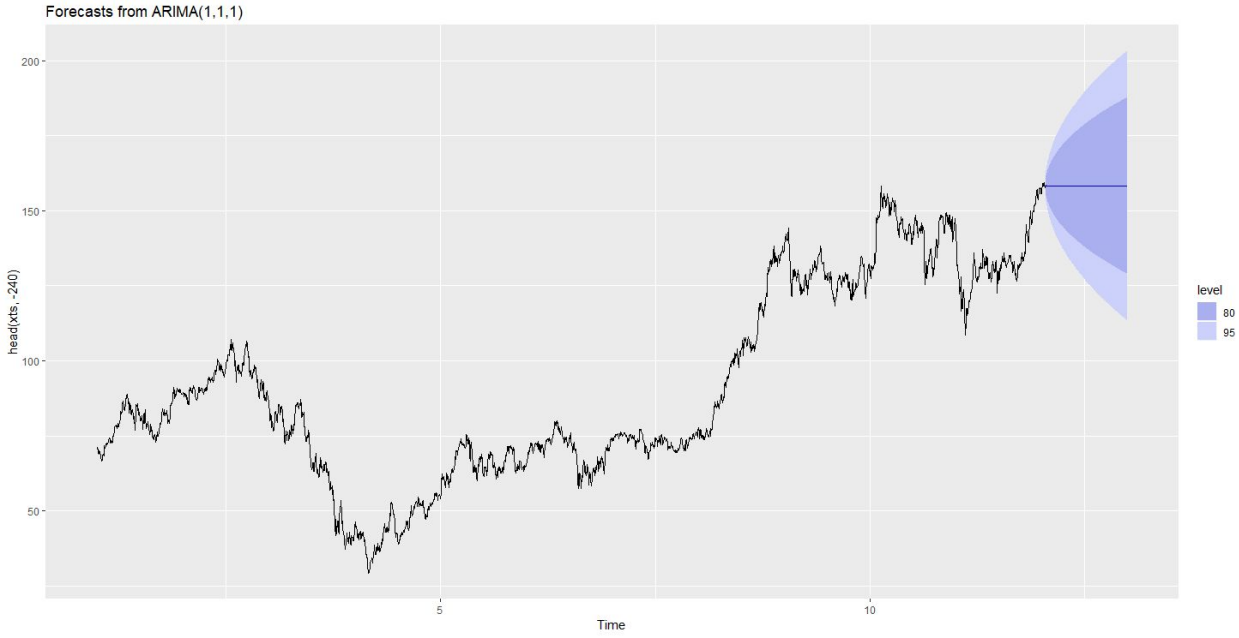
- ARIMA :-

The next step is to select an appropriate ARIMA model, which means finding the most appropriate values of p and q for an ARIMA(p,d,q) model. We automatically generated a set of optimal *(p, d, q)* using auto.arima(). This function searches through combinations of order parameters and picks the set that optimizes model fit criteria.

Observations (when number of data points left for forecasting is) :-

1) 240

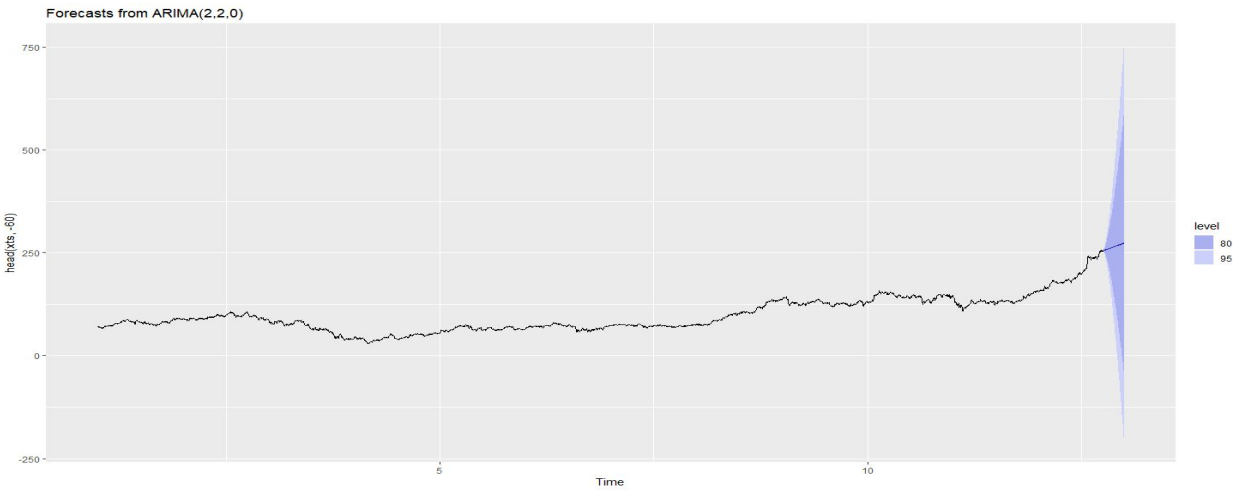
```
Series: head(xts, -240)
ARIMA(1,1,1)

Coefficients:
          ar1     ma1
      -0.5465  0.5916
s.e.   0.2698  0.2600

sigma^2 estimated as 2.057:  log likelihood=-4944.72
AIC=9895.44   AICc=9895.45   BIC=9913.23
```




```
> accuracy(tsforecasts240, head(tail(xts, 240), 240))
                     ME      RMSE       MAE          MPE      MAPE      MASE          ACF1
Training set  0.03075828  1.433613  1.045386  0.01284046  1.258562 0.05560615 -0.004253667
Test set     59.48226230 71.932986 59.486741 24.80622482 24.809062 3.16421828           NA
```

2) 60

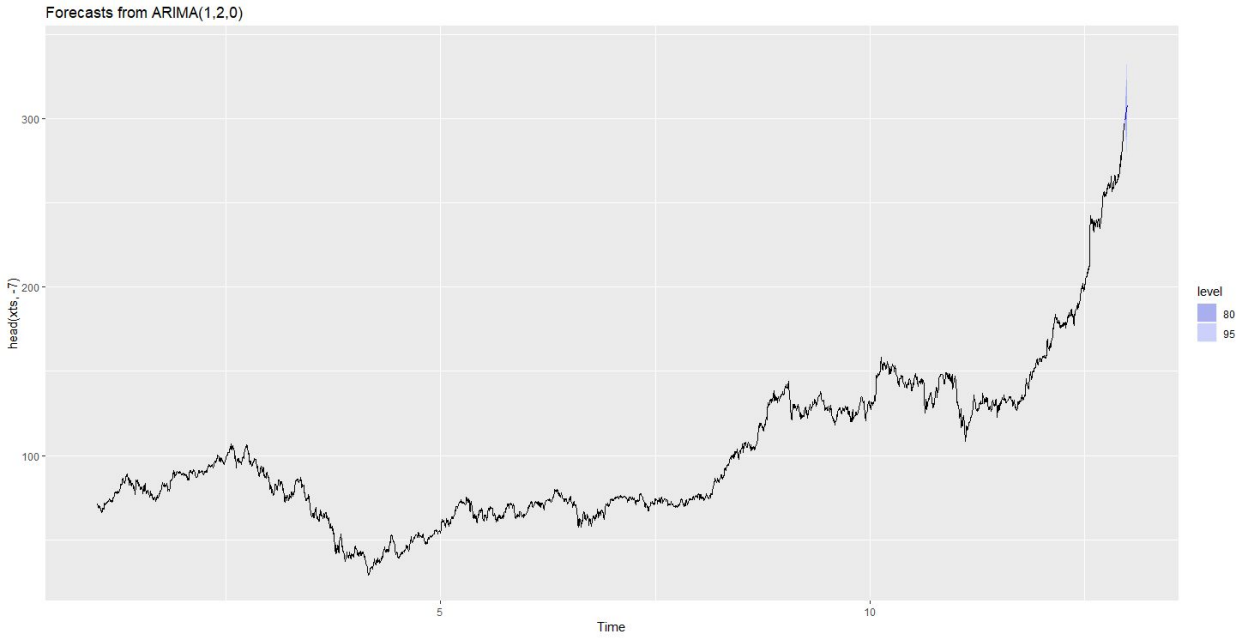
```
Series: head(xts, -60)
ARIMA(2,2,0)

Coefficients:
          ar1      ar2
      -0.5989  -0.3462
s.e.   0.0173   0.0173

sigma^2 estimated as 3.033:  log likelihood=-5837.38
AIC=11680.76   AICc=11680.77   BIC=11698.74
```




```
> accuracy(tsforecasts60, head(tail(xts, 60), 60))
                      ME      RMSE      MAE           MPE     MAPE      MASE         ACF1
Training set -0.0001142228  1.740313 1.260180 4.865541e-05 1.432327 0.05652773 -0.07950449
Test set      6.1406544545 11.273391 7.551879 2.103076e+00 2.644243 0.33875360           NA
```

3) 7

```
Series: head(xts, -7)
ARIMA(1,2,0)

Coefficients:
          ar1
      -0.4752
s.e.   0.0160

sigma^2 estimated as 3.584:  log likelihood=-6193.78
AIC=12391.57   AICc=12391.57   BIC=12403.59
```



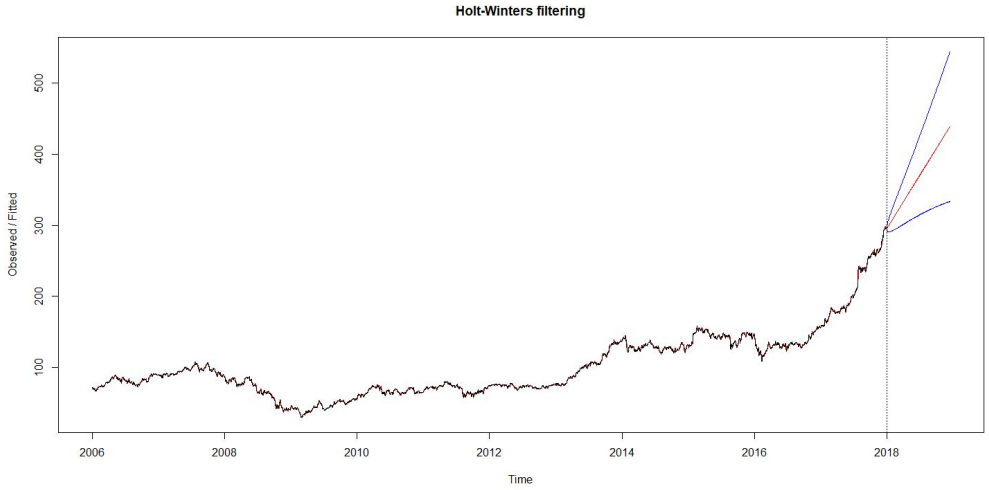
	ME	RMSE	MAE	MPE	MAPE	MASE	ACF1
Training set	0.0004001267	1.892199	1.368114	-9.685294e-05	1.523037	0.05645894	-0.1561261
Test set	-7.4598594240	8.215804	7.459859	-2.524818e+00	2.524818	0.30785131	NA

- Holt's Winter Model

If we have a time series that can be described using an additive model with increasing or decreasing trend and seasonality, we can use Holt-Winters exponential smoothing to make short-term forecasts. Holt-Winters exponential smoothing estimates the level, slope and seasonal component at the current time point.

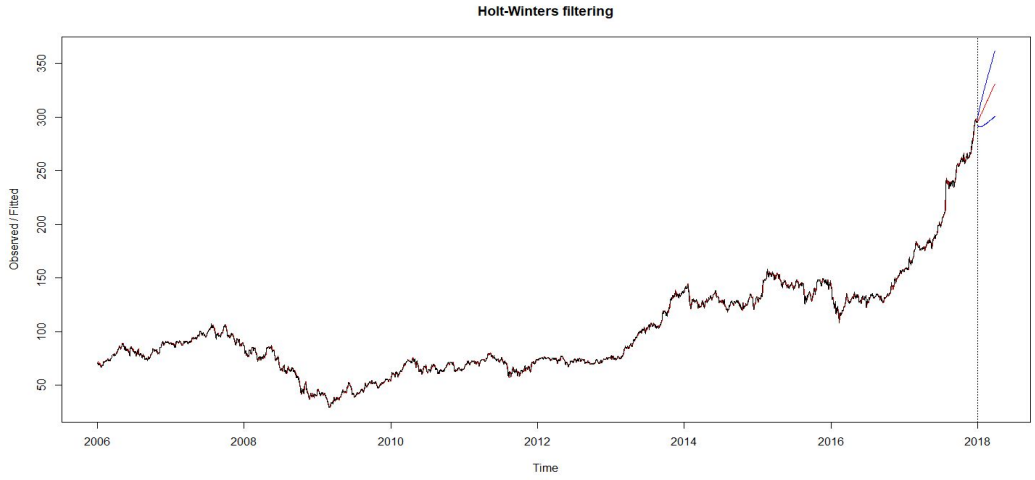
Observations (when number of data points left for forecasting is) :-

1) 240



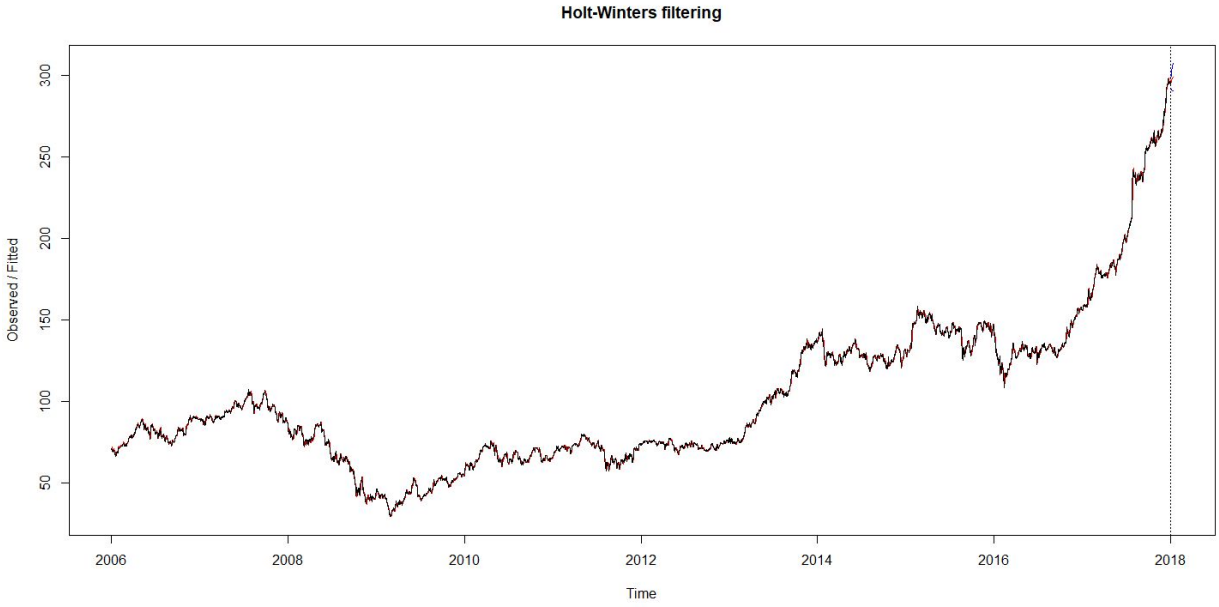
RMSE for 240 observations :- 150.1506

2) 60



RMSE for 60 observations: 42.90603

3) 7



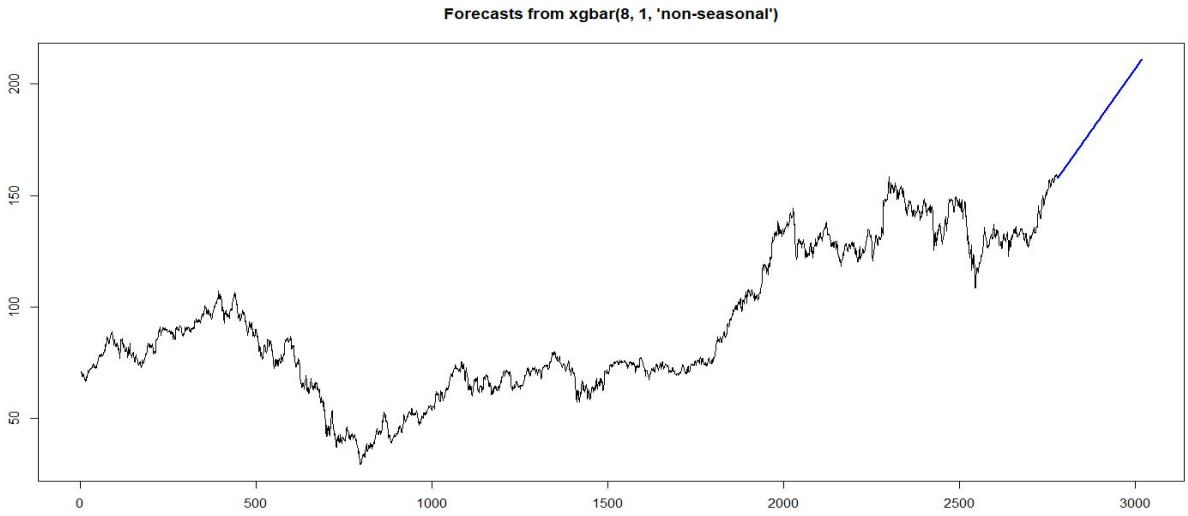
RMSE for 7 observations: 2.277894

- Gradient Boosted ARIMA :-

Here we tried to mix the traditional approach of ARIMA with extreme gradient boosting approach. We tried on implementing a gradient boosted approach for ARIMA where we tried to derive the importance for each lag using boosting method and did the forecast We used on an R package forecastxgb which will adapt this approach to extreme gradient boosting, popularly implemented by the astonishingly fast and effective xgboost algorithm.

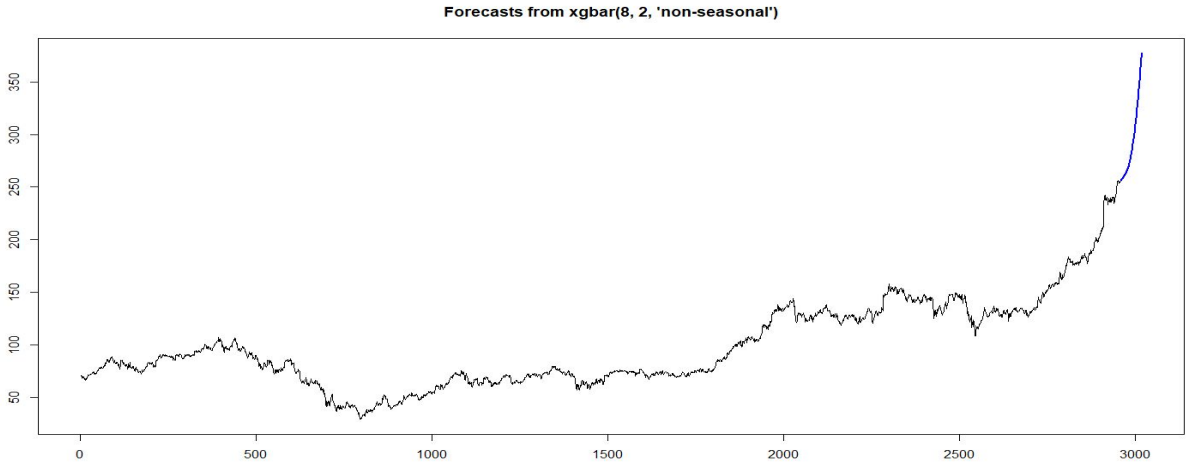
Observations (when number of data points left, is) :-

1) 240



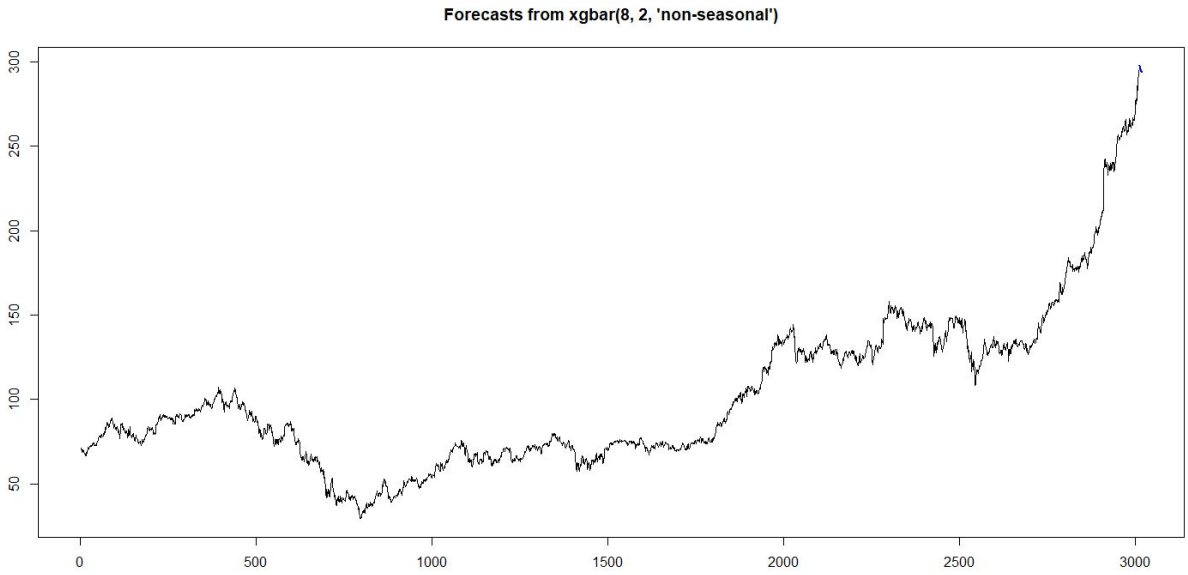
RMSE for 240 observation: 41.6936

2) 60



RMSE for 60 observation: 33.01421

3) 7



RMSE for 7 observations: 1.240714

5. Comparison

We will try to do a comparative study based on obtained RMSE values.

Observation left / Model	ARIMA	Holt's Winter	ARIMAxgb
240	71.932	150.150	41.693
60	11.273	42.906	33.014
7	8.215	2.277	1.240

Since RMSE value for forecasting for future values of particular stock is smaller if we are predicting on short interval in case of boosted method. So boosted method gives us more accurate forecasting on short interval. For forecasting on interval of large range, ARIMA and boosted method both gives good forecasting but we are not able to draw conclusions as observed from the above table of RMSE values. We have to consider different datasets and then we can draw conclusion about larger interval on the basis of average forecasting of both methods.

6. Conclusion:

Among the considered models i.e. ARIMA, Holt's Winter and xgboost method, xgboost method gives more accurate predictions on short interval.

Different Boosting methods can be applied on various time series models like MA models, ARIMA models to improve the accuracy of forecasting. Forecasting from ARIMA model fitted on dataset can be taken as underlying as naive step of Adaboost algorithm to improve forecasting and make RMSE small by resembling the other models with ARIMA for forecasting.