TOPIC 6

STOCK

Name: Akash Mandal

Roll No: 16MA20007

Problem Statement

For the given data from stock exchange predict the stock value in the month 1/10/2017.

**Professor's Instruction-**

Use time series regression to forecast data required. Choose any one of the values among open, high low and close for prediction.

Assumption - We choose to predict close values on 1-10-2017. We will use ARIMA test for the above.

Processing and Preparation

For time series modelling and plotting, we install packages zoo, ggplot2, tseries and forecast.

We convert Date attribute to type Date, and use zoo() and ts() to get a time series for our close values.

We clean out time series from outliers with tsclean(), plotting the series before and after cleaning.

library(zoo)

library(ggplot2)

library(tseries)

library(forecast)

STOCKS$Date <- as.Date(STOCKS$Date)

ZOO <- zoo(STOCKS$Close, order.by = STOCKS$Date)

time\_series <- ts(ZOO)

ggplot(STOCKS, aes(Date, Close)) + geom\_line() + scale\_x\_date('month') + ylab("Close values") +

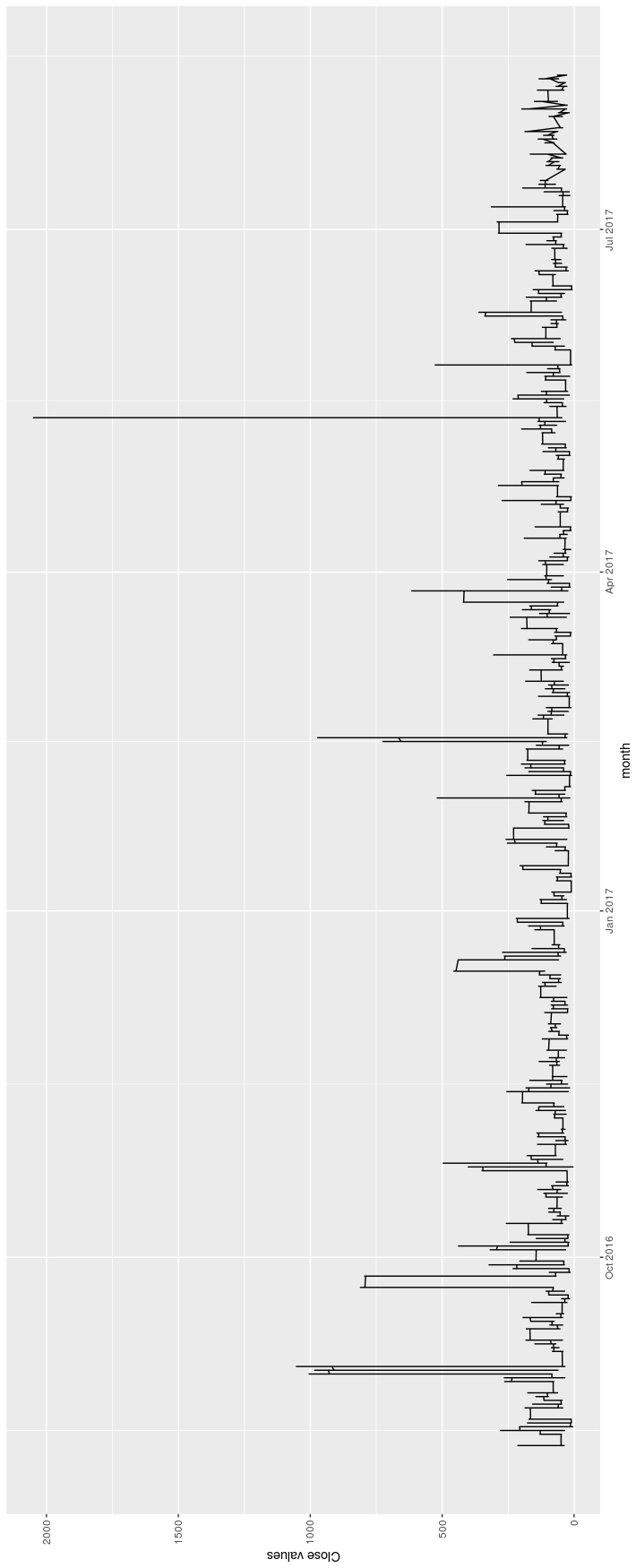
xlab("")

STOCKS$clean\_ts = tsclean(time\_series)

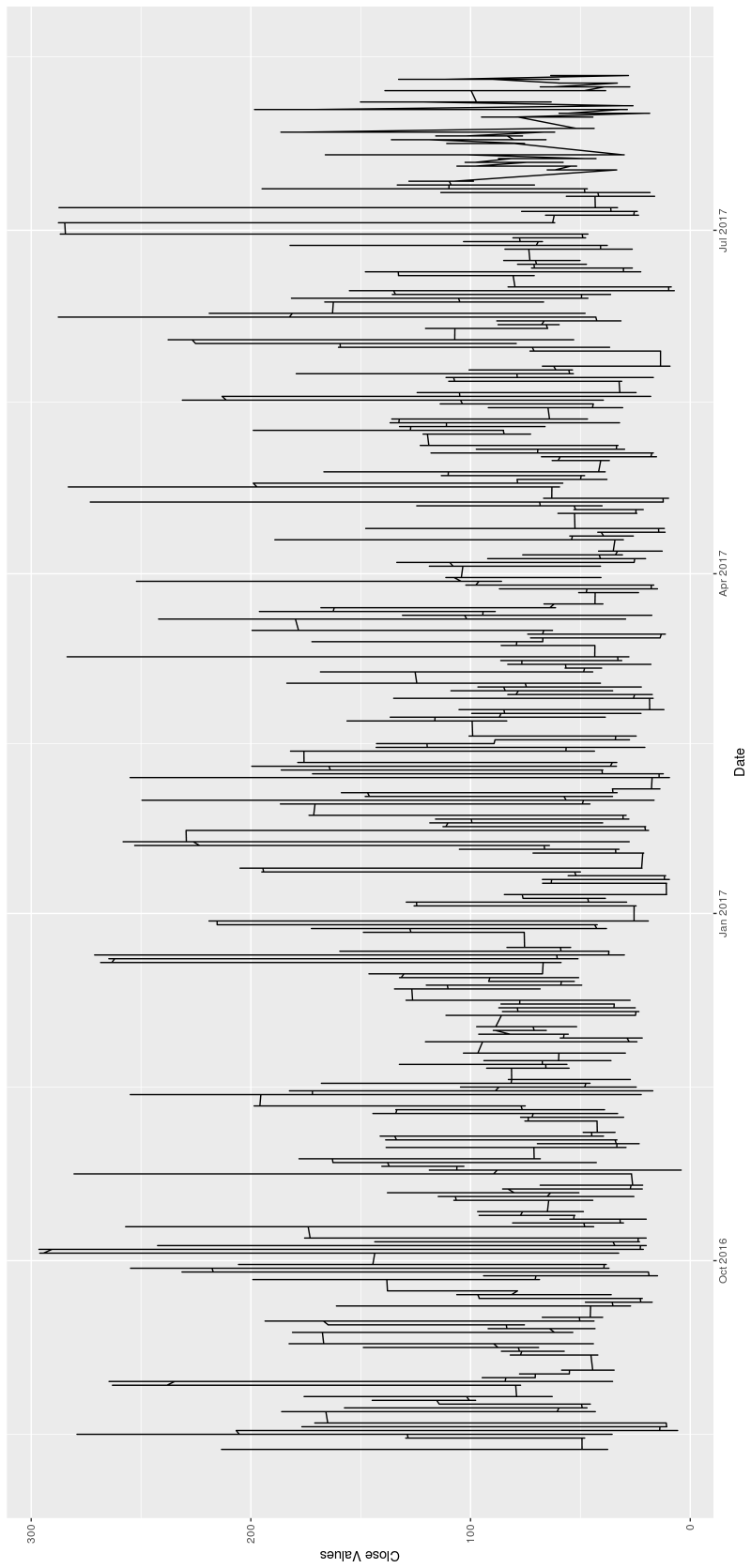
clean\_ts = STOCKS$clean\_ts

ggplot() +

geom\_line(data = STOCKS, aes(x = Date, y = clean\_ts)) + ylab('Close Values')

With Outliers:

Without outliers:



ADF Testing

Since our auto regression model requires a stationary series, we check if our series is stationary or not using an ADF test( Augmented Dickey-Fuller Test) We get a p value < 0.01, which allows us to reject the null hypothesis that the series was not stationary.

> adf.test(clean\_ts, alternative = "stationary")

Augmented Dickey-Fuller Test

data: clean\_ts

Dickey-Fuller = -17.605, Lag order = 50, p-value = 0.01

alternative hypothesis: stationary

Moving Average

To smoothen out the random fluctuations in data, we take a monthly moving average

> close\_ma30 <- ma(clean\_ts, order = 30)

ARIMA Model

We choose the auto.arima() to model our time series.

> fit <- auto.arima(close\_ma30, seasonal = FALSE)

> print(fit)

Series: close\_ma30

ARIMA(5,1,0)

Coefficients:

ar1 ar2 ar3 ar4 ar5

1.7908 -1.3910 0.9820 -0.5781 0.1785

s.e. 0.0028 0.0055 0.0062 0.0055 0.0028

We get ARIMA(5, 1, 0) with the coefficients as above. Thus we model our time series as follows

Yt = 1.791Yt-1 - 1.391Yt-2 + 0.982Yt-3 - 0.578Yt-4 + 0.178Yt-5 + E

Forecast

We now observe that we need to forecast for 51 days ahead of the last entry, ie 2017-08-11, to get the value at 2017-10-1

Sonce we chose a moving average of order 30, we choose to predict 2 periods ahead (51 / 30 rounded up)

This gives us Forecast of 61.94, with a 95% confidence interval of [61.24, 62.64]

> fcast <- forecast(fit, h = 2)

> print(fcast)

Point Forecast Lo 80 Hi 80 Lo 95 Hi 95

4207.233 62.04494 61.88975 62.20013 61.80760 62.28228

**4207.267 61.94177 61.48170 62.40184 61.23815 62.64539**

However we note that the question required prediction to one single day, hence we should use the daily timeseries instead of the moving average. Taking that into consideration. we obtain with a forecast of 51 periods(days)

> fit <- auto.arima(clean\_ts, seasonal = FALSE)

> print(fit)

Series: clean\_ts

ARIMA(0,1,2)

Coefficients:

ma1 ma2

0.0072 0.0105

s.e. 0.0028 0.0028

sigma^2 estimated as 22.33: log likelihood=-375114.7

AIC=750235.4 AICc=750235.4 BIC=750264.7

> fcast <- forecast(fit, h = 51)

> print(fcast)

Point Forecast Lo 80 Hi 80 Lo 95 Hi 95

\*\*\*\*\*

\*\*\*\*\*

\*\*\*\*\*

126256 59.71874 21.25568 98.18180 0.8945691 118.54291

126257 59.71874 20.76500 98.67248 0.1441383 119.29334

126258 59.71874 20.28043 99.15705 -0.5969566 120.03444

126259 59.71874 19.80173 99.63575 -1.3290556 120.76654

126260 59.71874 19.32871 100.10877 -2.0524785 121.48996

126261 59.71874 18.86117 100.57631 -2.7675266 122.20501

126262 59.71874 18.39891 101.03857 -3.4744844 122.91196

126263 59.71874 17.94177 101.49571 -4.1736202 123.61110

126264 59.71874 17.48958 101.94790 -4.8651882 124.30267

126265 59.71874 17.04218 102.39530 -5.5494289 124.98691

126266 59.71874 16.59942 102.83806 -6.2265703 125.66405

126267 59.71874 16.16116 103.27632 -6.8968291 126.33431

**126268 59.71874 15.72727 103.71021 -7.5604108 126.99789**

Using this model we forecast a closing value of 59.72, with a 80% confidence interval of [15.73, 103.71]

Conclusion

Here we have implemented a few of the steps required for a good time series model.

But we have not decomposed the data to look for seasonal variation, and assumed there isn't any. While we have used adf test, we have not differenced the series based on ACF(Auto Correlation Function) and PACF(Partial Auto Correlation Function) plots.

Also after fitting an ARIMA model, we have not evaluated how well it fits using AIC(Aikaike Information Criteria) or BIC(Bayesian Information Criteria).

However, we have accomplised a simple analysis given data, and forecast our close value with some measure of certainty.The rest of the rigorous analysis may be attempted in further projects, but as of now it is beyond the scope of this Assignment.