Probability and Statistics 174: Time Series

Dow Jones Data Analysis

Group 17410: Kristen Nipper, Shireen Mann, Sarah Salem, Josh Kurtz, and Jon Tsegaye

March 14, 2019

Table of Contents:

Abstract:

The purpose of our report is to analyze historical Dow Jones Industrial Average (DJIA) index data in order to successfully forecast future Dow Jones indices. The DJIA index is comprised of stock values of the largest 30 publicly traded companies based in the United States. That is, the daily DJIA value is the daily sum of each of the 30 stock prices, not a weighted average of the stock values. Our dataset was obtained from the Federal Reserve Economic Data website.

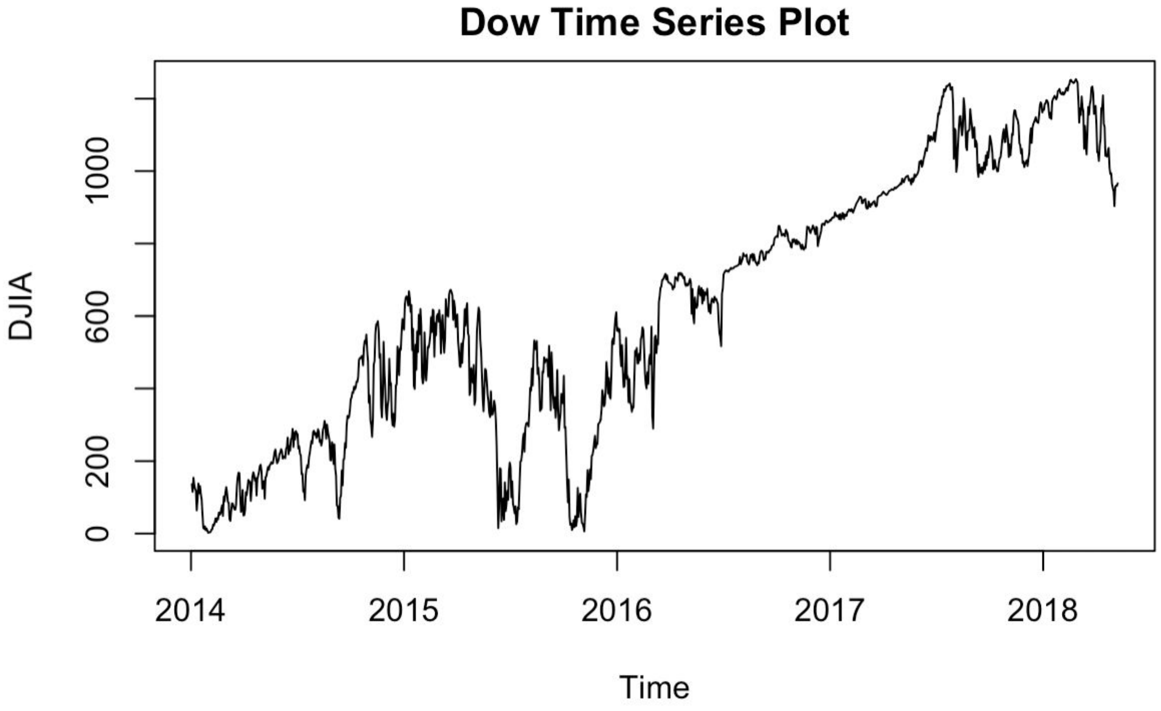
Introduction:

Our data set contains two variables: a Date variable, which is our predictor, and a daily Dow Jones Industrial Average index variable, which is our response. The data spans four years, from January 1st 2014 to December 31st 2018 . The frequency of the data is 260 as there are 260 business days during the year. In this report we will investigate the raw data, make transformations and stationarize our series to be able to predict the most accurate DJIA values.

The data displayed a strong seasonal component, with clear peaks at the beginning of each year. Trend was also very apparent, with a strong positive direction. Thus before choosing our model, we differenced at lag 365 and 12 to remove seasonality and lag 1 to remove trend. We tried various transformations to reduce variance, but the original data ended up having the best results. After this, we analyzed ACF and PACF plots to estimate models, and choose the best one based on AIC score. After our best models determined, we used Shapiro Wilk, Box-Ljung, and Box-Pierce tests to perform diagnostics on our assumptions, and selected the model that passed. Our final model was AR(12) after seasonality and trend were removed by differencing.

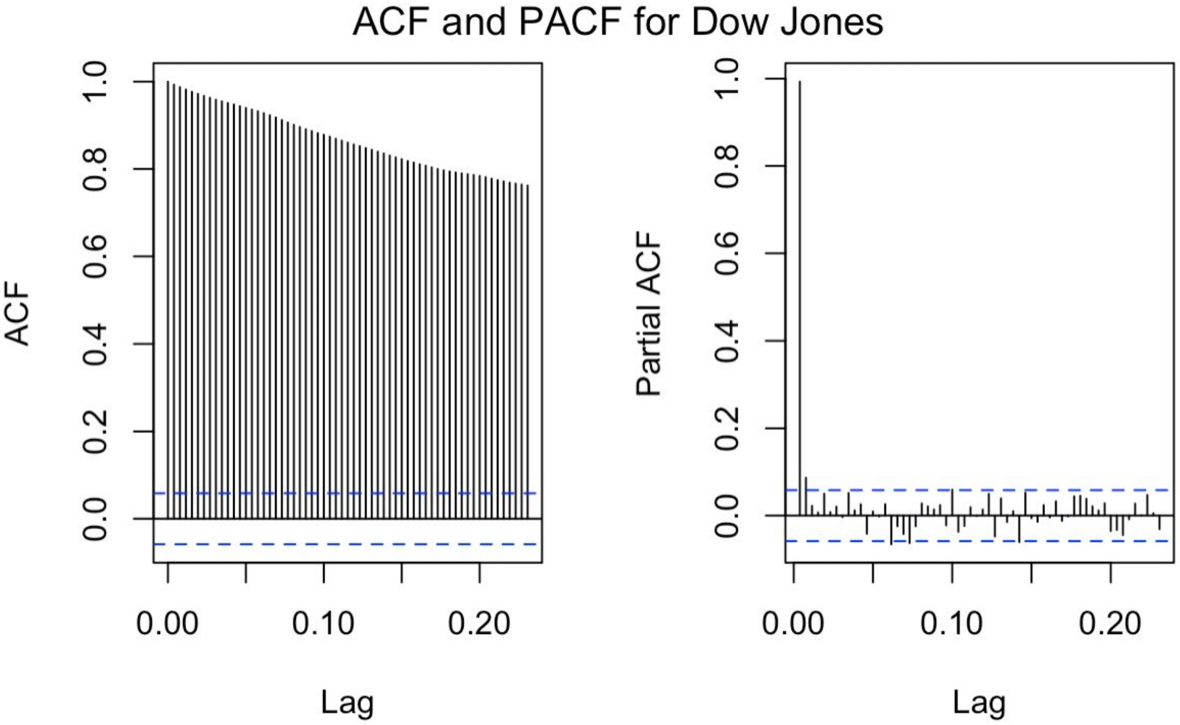
Initial Time Series Analysis:

To begin, we first split the original data into a training and test set so as to set aside some of the data for later analysis. After doing so, we plot the time series:

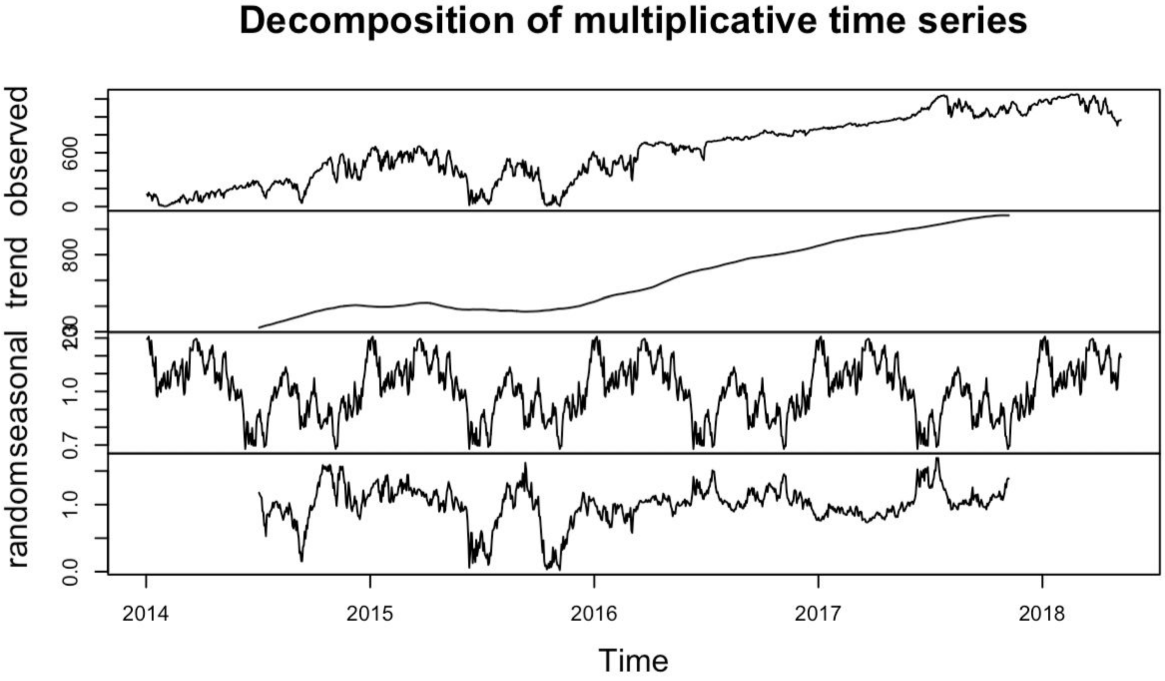


We find that the data has an increasing trend as time passes, and the spikes of data moving up and down show that it has seasonality. The upward increase in the data's values are indicative that it has non-constant variance.

The ACF and PACF, or Autocorrelation and Partial-autocorrelation functions, of our original data are as follows:

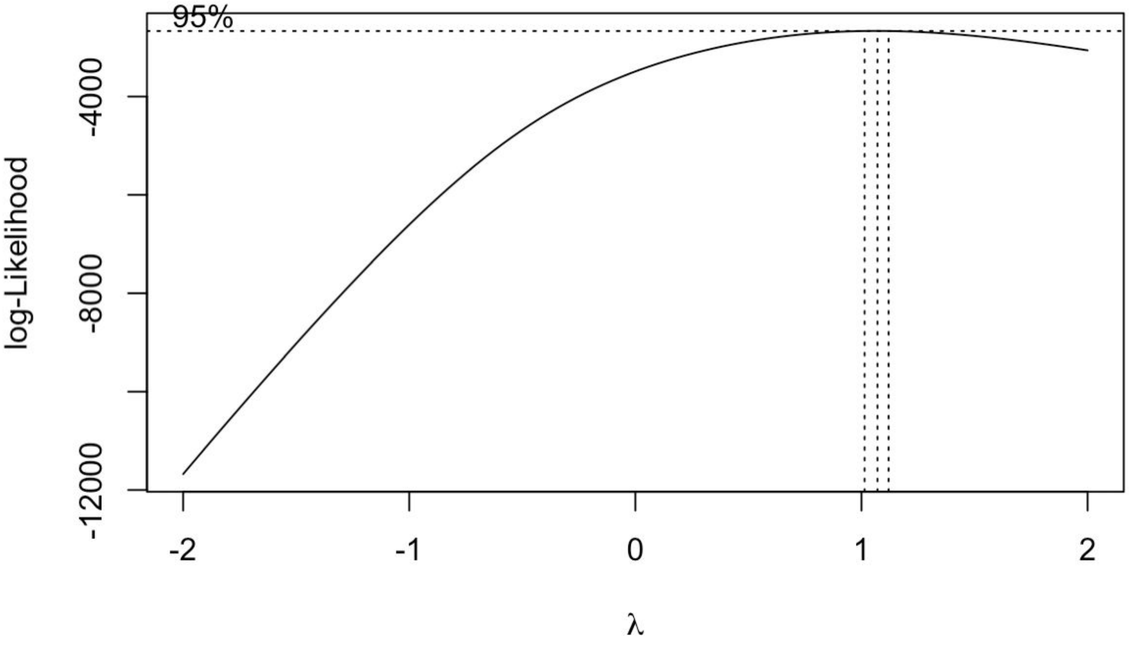


The ACF decreases very slowly, which illustrates that the data is not yet stationary. Additionally, the decomposition plot, shown below, of the Dow Jones data allows us to better visualize each component of the data: an increasing trend and prominent seasonal component contribute to the evidence that the data is not stationary.

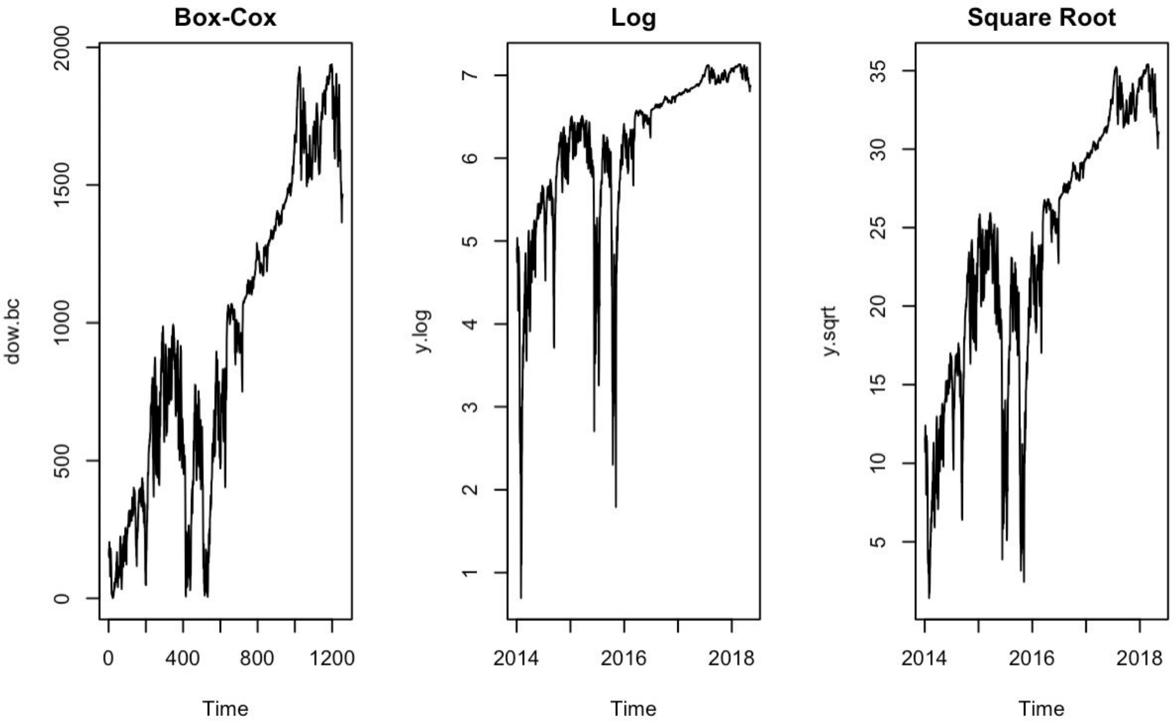


To confirm the observation regarding changing variance, we conducted a BoxCox test of the lambda values to determine whether a transformation of the data would be necessary.

Box-Cox Transformation:



The confidence interval given by the BoxCox plot includes a lambda value of 1, so we conclude no transformation of the data is necessary. This is because λ =1 is equivalent to using the original data. The 95% confidence interval given by the BoxCox plot of our data begins at a value only slightly larger than 1. The deviation from 1 is negligible, which tells us that we should use the original, non-transformed data, but we will explore the transformations to show that using the original data will produce the best model.

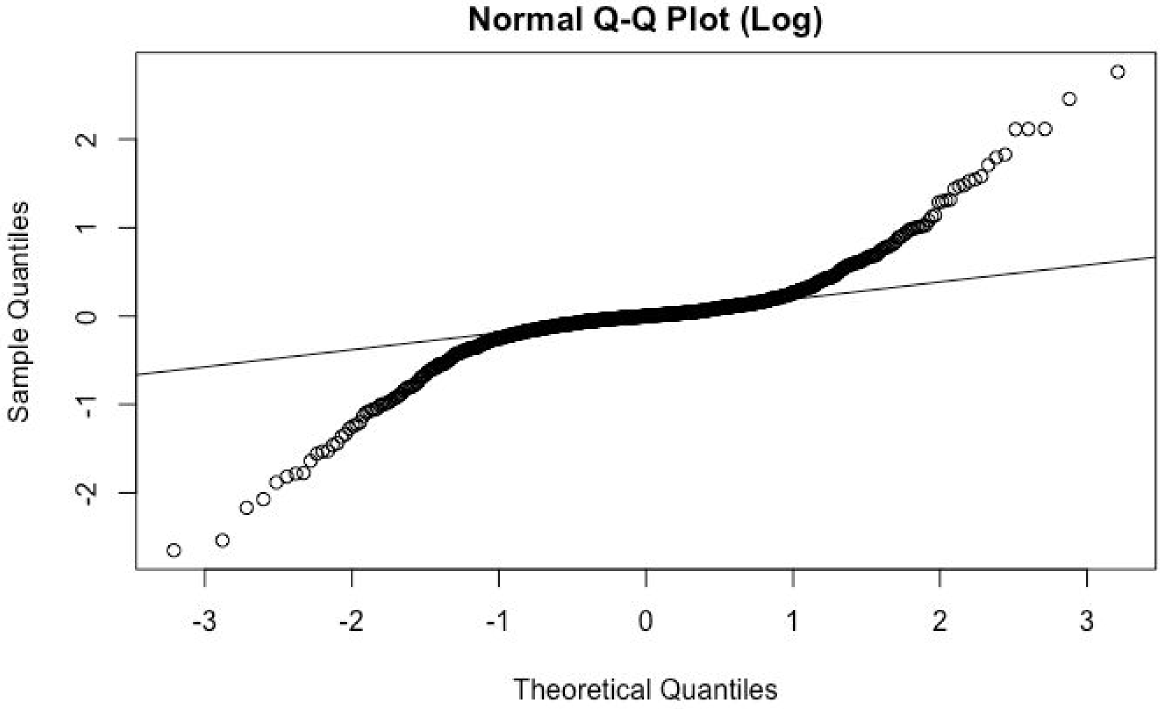


The variance of the original Dow Jones data is 130,612.4, and variances given by the BoxCox, Log, and Square Root transformations are 320,042.6, 0.9562617, and 69.37524, respectively. The log transformation of our data results in the smallest variance, so we will conduct analysis on the log-transformed data.

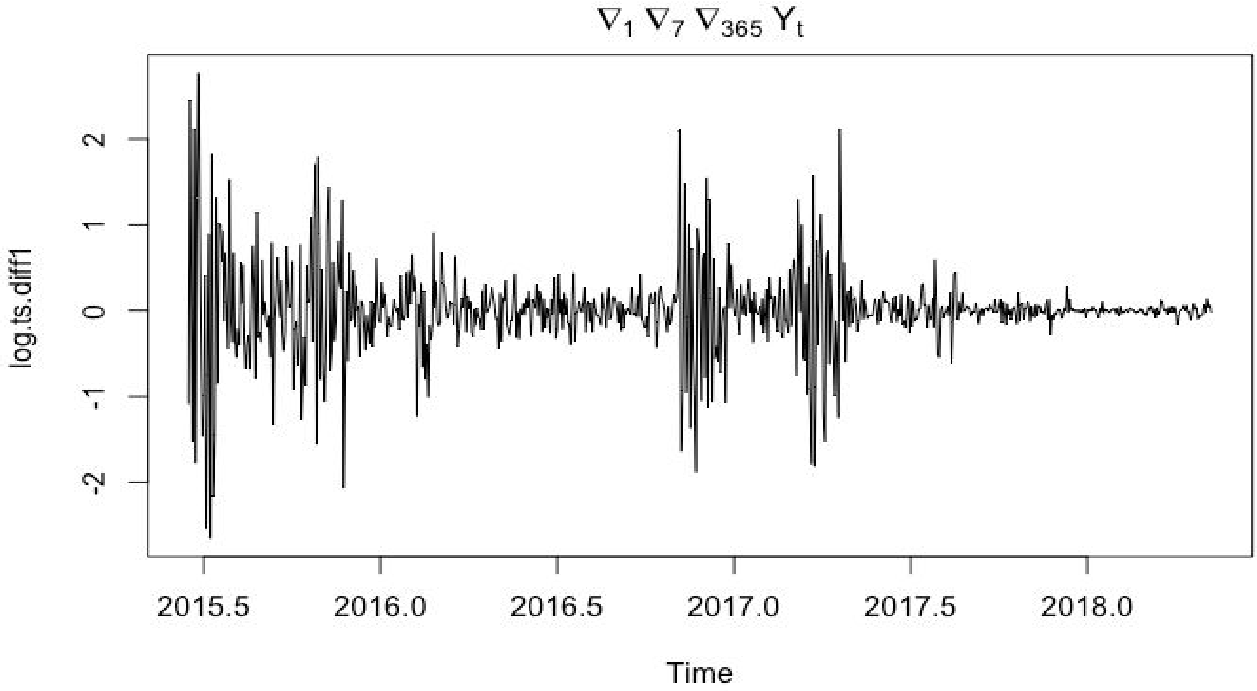
Removing Trend and Seasonality:

The data is a daily time series, so we will begin by differencing the data at lag 365 to remove seasonality of our data. We find that the variance decreases, so we can continue to difference the data again. At lags 91 and 7, the variance is not smaller than the variance preceding them at lag 365, so we cannot difference at those lag values. We hypothesized these would be important lags due to quarterly and weekly seasonality. However, at lags 12 (~biweekly) and 1, the variance does become smaller, so we can perform the final difference at lag 1 to remove trend.

An issue we encounter with the log-transformed data is that its QQ plot strays too far from the qq normal line.

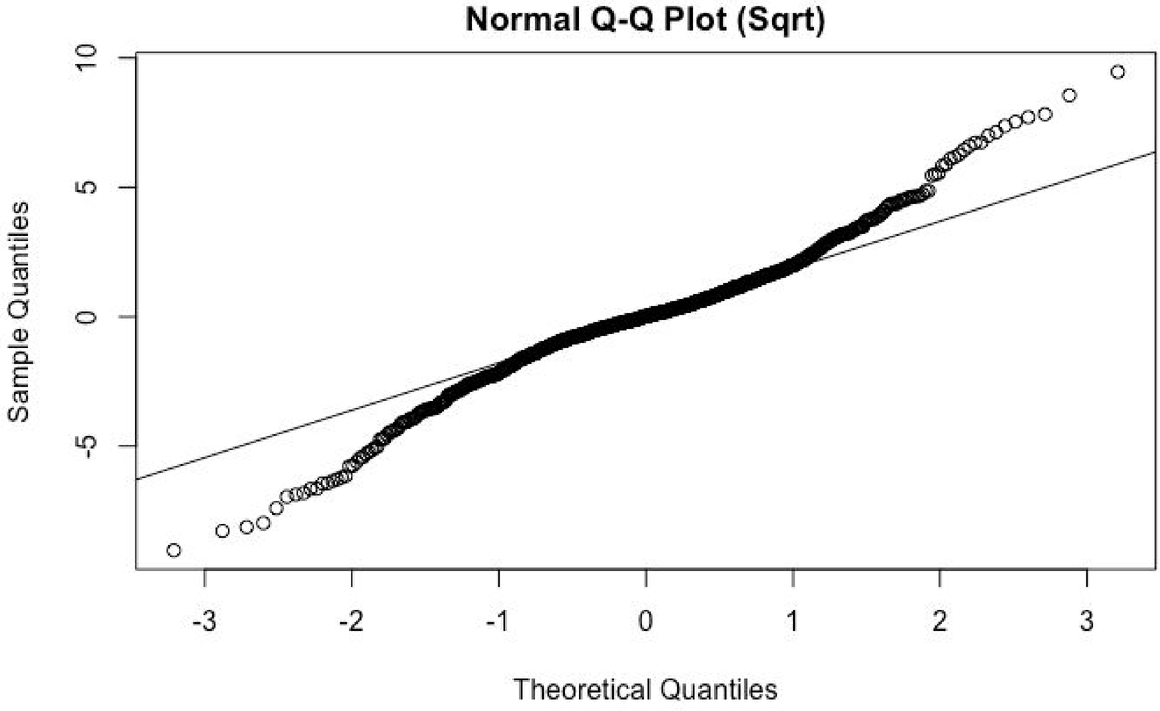


Despite the log transformation having lowest variance, its time series plot still shows signs of non constant variance with a "fanning" pattern present.

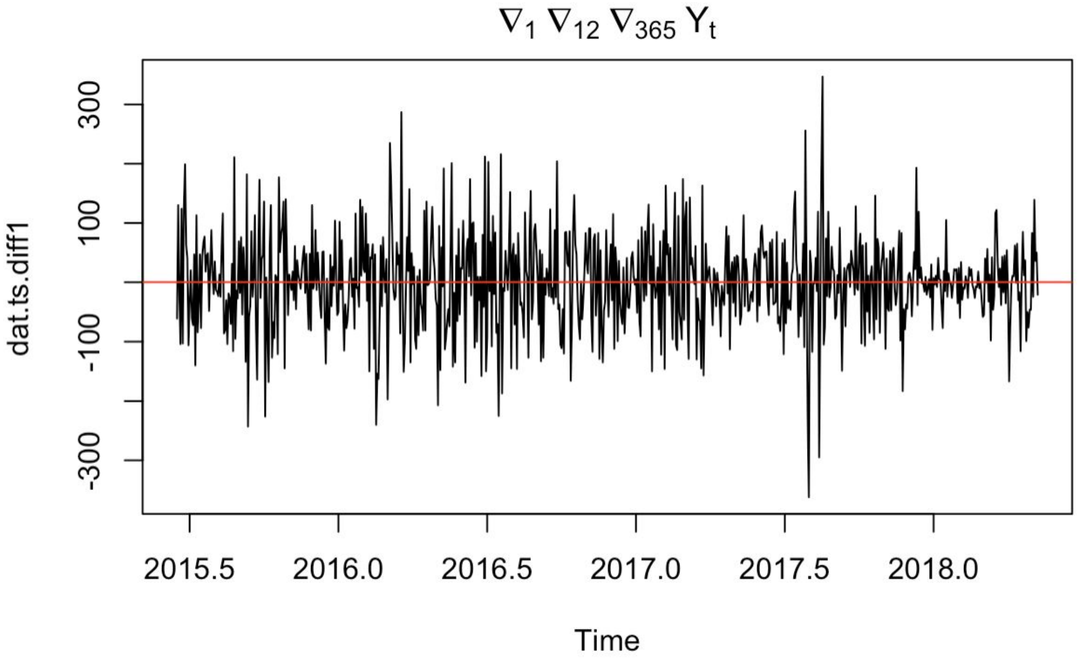


Therefore it can be said that the log-transformation will not produce the best model for our data. The next smallest variance came from the Square Root transformation. Differencing again, we find that we can successfully decrease variance by differencing at lags 365, 12, 7, and 1. We difference at lag 1 last to remove trend.

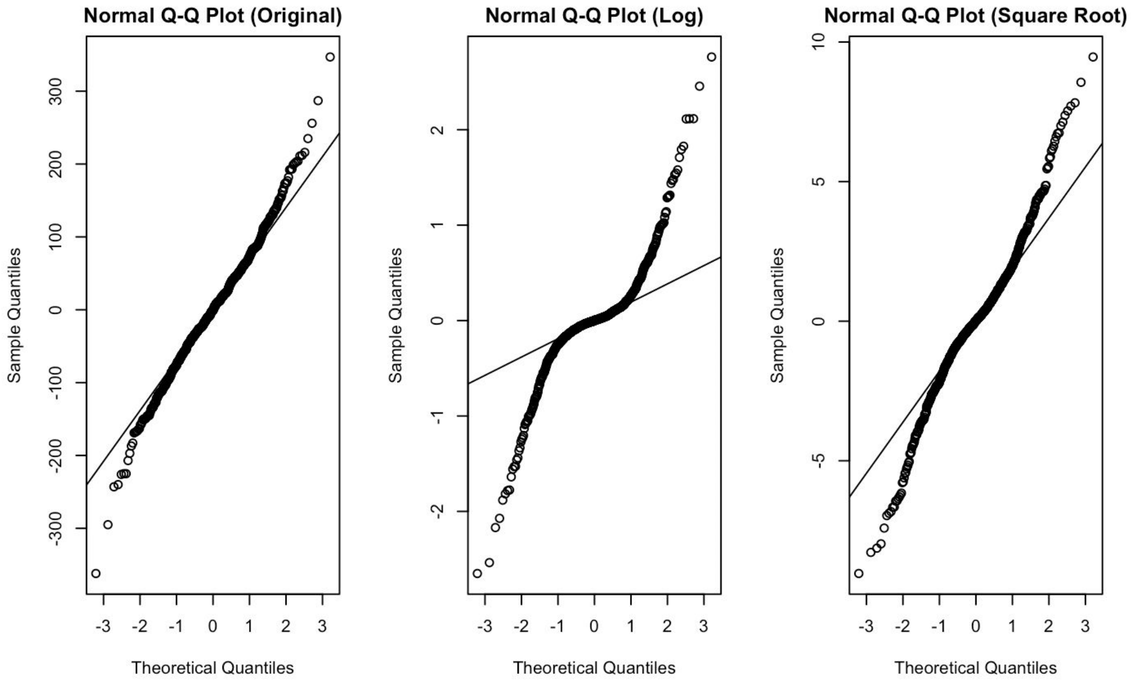
However, the QQ-plot still does not follow the normal line closely enough to keep normality assumptions:



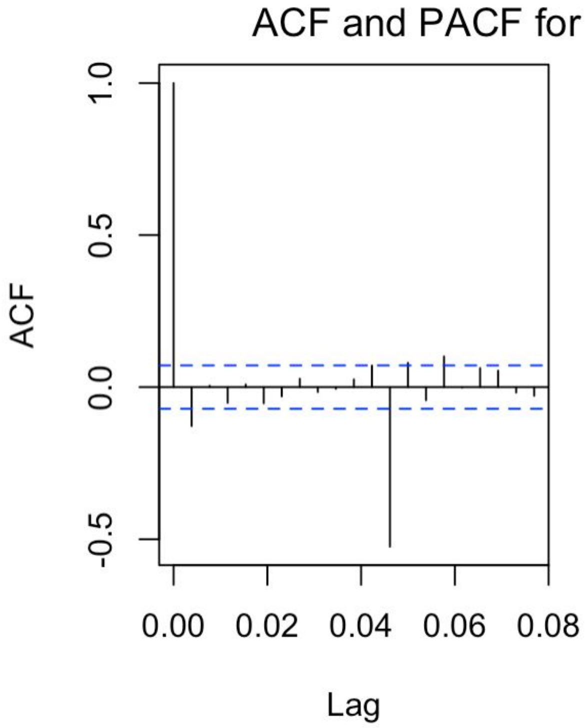
We then assert that it may be best to build a model from the original data. Again differencing to remove trend and seasonality in our data, we begin differencing at lag 365. We find that the variance decreases from 130,612.4 to 55,729.11, so we continue differencing. Differencing at lag 12, the variance decreases to 23,026.71. Differencing once more at lag 1 decreases variance to 6,592.17. The resulting plot of this data is shown below.



The red line is the mean of the data, at 0, which is indicative of stationarity. It is apparent that the trend and seasonality are no longer factors. Furthermore, in comparing the three Q-Q Plots, we find that the original non-transformed data follows the normal line more closely than the others, which supports our previous findings that the original data will produce the best model.



The ACF and PACF plots of the current differenced data, which is referred to as y.original.clean in the code, are as follows:

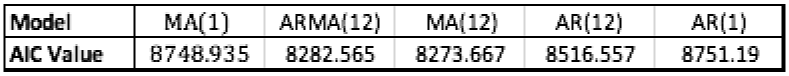


Each point on the lag axis of the ACF and PACF plots are 0.04 apart, and they are decimals due to being scaled to a year. To scale the values for daily interpretation, we multiply each lag value by 260, which is the frequency of data collection per year.

Models:

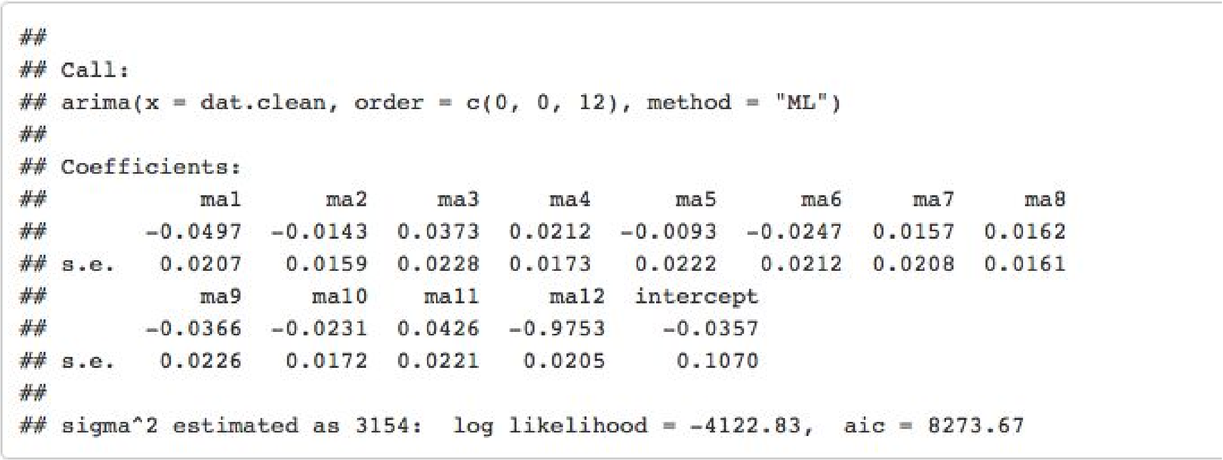
The possible models we find from the ACF and PACF plots are ARMA(0.048), MA(0.048), AR(0.048), AR(0.004), and MA(0.004). Scaling these parameters to days, we get: ARMA(12), MA(12), AR(12), AR(1), and MA(1). We will also apply an auto arima and compare the results to those we just found from the ACF and PACF.

After conducting the auto arima, we find that MA(12) has only one parameter and the lowest AIC score between all the possible models. However, although MA(12) has the lowest AIC score, it doesn't pass all the diagnostics checks. So we choose AR(12) as our best model since it is the model with both the lowest AIC score and it passes all the diagnostics checks in the section below.



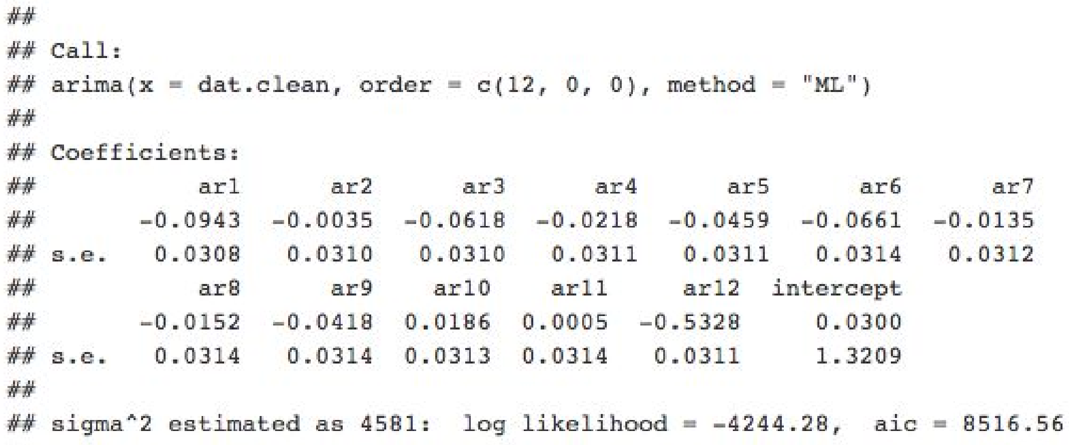
Model	MA(1)	ARMA(12)	MA(12)	AR(12)	AR(1)
AIC Value	8748.935	8282.565	8273.667	8516.557	8751.19

MA(12)



```
##
## Call:
## arima(x = dat.clean, order = c(0, 0, 12), method = "ML")
##
## Coefficients:
##           ma1      ma2     ma3     ma4      ma5      ma6     ma7     ma8
##       -0.0497  -0.0143  0.0373  0.0212  -0.0093  -0.0247  0.0157  0.0162
## s.e.   0.0207   0.0159  0.0228  0.0173   0.0222   0.0212  0.0208  0.0161
##           ma9     ma10    ma11     ma12  intercept
##       -0.0366  -0.0231  0.0426  -0.9753    -0.0357
## s.e.   0.0226   0.0172  0.0221   0.0205     0.1070
##
## sigma^2 estimated as 3154:  log likelihood = -4122.83,  aic = 8273.67
```

AR(12)



```
##
## Call:
## arima(x = dat.clean, order = c(12, 0, 0), method = "ML")
##
## Coefficients:
##           ar1      ar2      ar3      ar4      ar5      ar6      ar7
##       -0.0943  -0.0035  -0.0618  -0.0218  -0.0459  -0.0661  -0.0135
## s.e.   0.0308   0.0310   0.0310   0.0311   0.0311   0.0314   0.0312
##           ar8      ar9    ar10    ar11     ar12  intercept
##       -0.0152  -0.0418  0.0186  0.0005  -0.5328     0.0300
## s.e.   0.0314   0.0314  0.0313  0.0314   0.0311     1.3209
##
## sigma^2 estimated as 4581:  log likelihood = -4244.28,  aic = 8516.56
```

Diagnostic Checking:

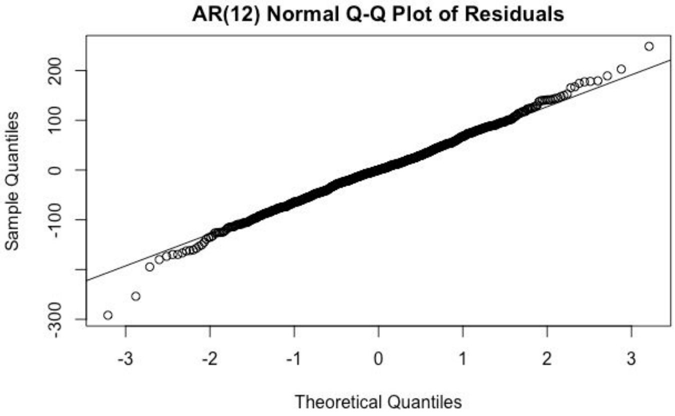
MA(12)

We perform diagnostics on MA(12) model to showcase that it does not pass all our validity checks. The model passes the Ljung-Box test and the Pierce-Box, but it fails the Shapiro-Wilk test.

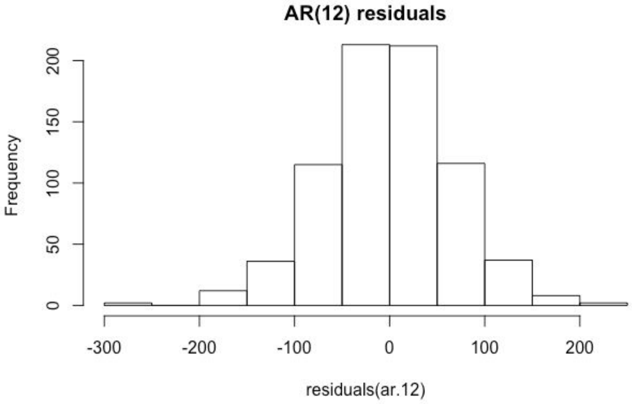
Box-Ljung test	Box-Pierce test	Shapiro-Wilk normality test
p-value = 0.06269	p-value = 0.5321	p-value = 2.019e-06

AR(12)

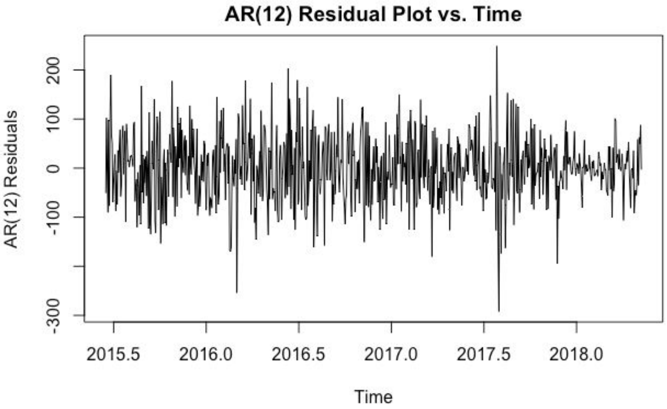
We perform diagnostics on our AR(12) model to ensure the validity of our assumptions. To check normality of errors, we observe the Normal QQ plot and conclude that errors are indeed normal as they do not deviate from the normal line.



Further, we can see from the histogram of the residuals that errors are normal as well.



From the Residual vs. Time (order) plot we can see that there is no serial correlation present.



We perform a Ljung-Box test to check the autocorrelation structure of the residuals, we fail to reject the null as our p-value is well above our **α** level 0.01. Meaning the model does not exhibit a lack of fit. Lastly we perform a Shapiro-Wilk test to verify normality of errors and again we fail to reject the null as our p-value is well above our **α** level 0.01.

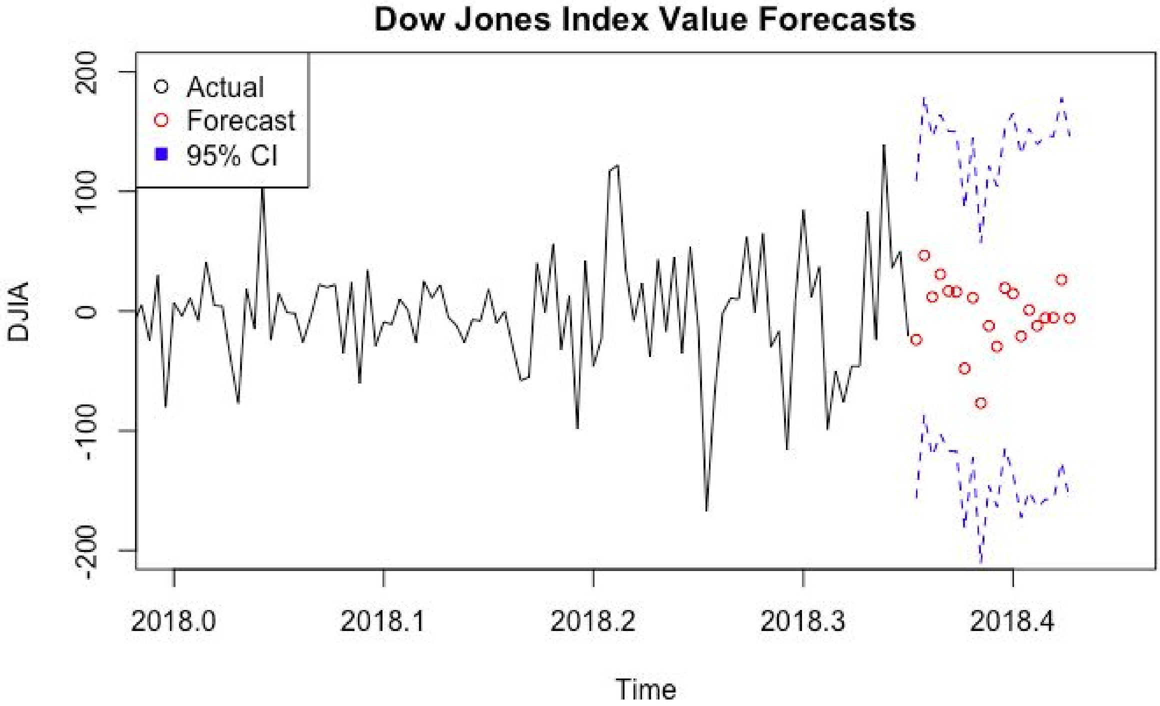
Box-Ljung test	Box-Pierce test	Shapiro-Wilk normality test
p-value = 0.4609	p-value = 0.9654	p-value = 0.02513

Analyzing Roots:

Due to having 12 coefficients, it was unreasonable to estimate the roots in our model. Thus we are assuming stationarity, since this is an AR model. Invertibility is given.

Forecasting:

We predict 20 future observations with a 95% confidence interval, where the red points are the predicted observations and the blue dashed lines represent the confidence interval depicted in the graph below.



Conclusion:

Luckily, our original data sufficed for time series analysis and was ready to analyze as is without any necessary transformations. Nonetheless, we included possible transformations to showcase how the original data is more suitable for analysis. Then, we performed exploratory data analysis by plotting the original data to observe any patterns, trends or stationarity, as well as plotting the ACF & PACF to discern a suitable model.

After observing trend and seasonality in the data, we differenced in order to obtain a simpler model, by removing the data's trend and seasonality. We differenced at varying lags to test which differenced lags reduced variance the most. We came to find that differencing by 365, 12, then 1 respectively removes trend and seasonal patterns. We plot ACF & PACF of the differenced data to determine the best possible models for our data. The ACF & PACF of the differenced data is now a lot simpler to read and we find that the best possible models are ARMA(12), MA(12), AR(12), AR(1), and MA(1).

To test which of these models best captures the behavior of our data we conduct several diagnostics tests to ensure the validity of each model. We verify normality of errors, lack of serial correlation, and equal variances by observing plots and verifying the visuals by running tests such as Ljung-Box test, Box-Pierce test and Shapiro-Wilk test. Initially we were inclined to choose MA(12) as the best model, since it yielded the lowest AIC score, however it didn't pass all the diagnostics test, so we settled for a model that both passes all the diagnostics tests and has the lowest AIC score and we finally conclude that AR(12) proves to be the best model for our data. We use the AR(12) model to forecast future DJIA index values with a 95% confidence interval.

References:

Dataset Source: Federal Reserve Economic Data (FRED)

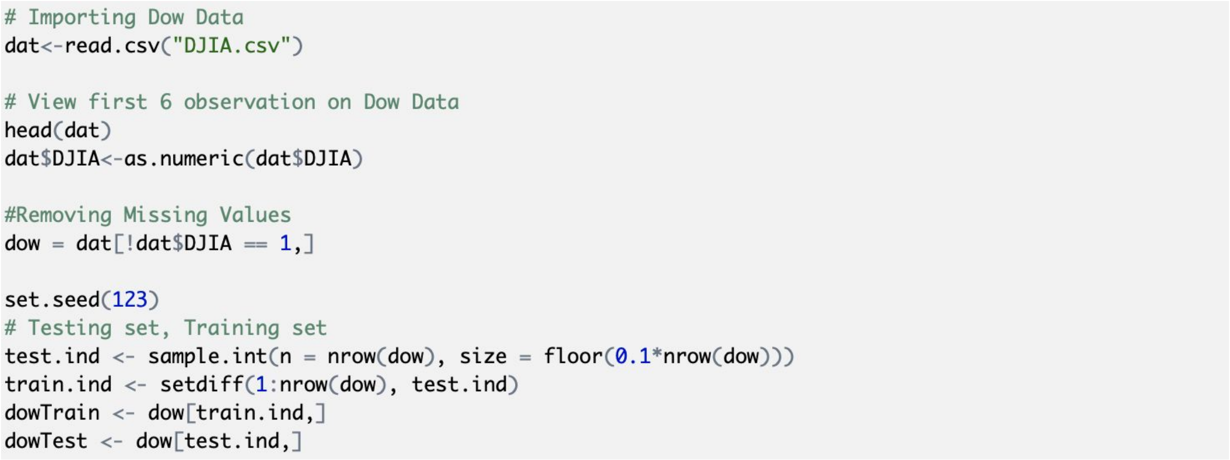
https://fred.stlouisfed.org/series/DJIA?fbclid=IwAR3xJ74uGL-F5KV8emaSfihYat12U1QTLTTxz14DODO8HhkCFn-kPG0uJIw

Box Cox information: Interpret the key results for Box-Cox Transformation

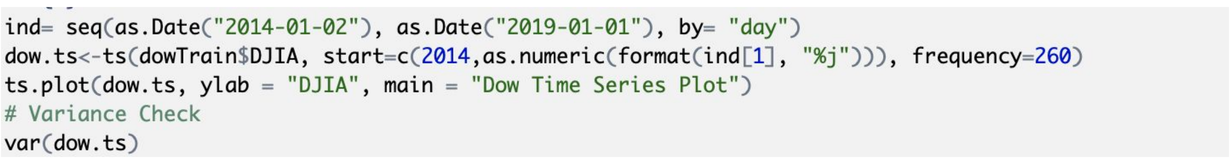
https://support.minitab.com/en-us/minitab/18/help-and-how-to/quality-and-process-improvement/control-charts/how-to/box-cox-transformation/interpret-the-results/key-results/

Appendix:

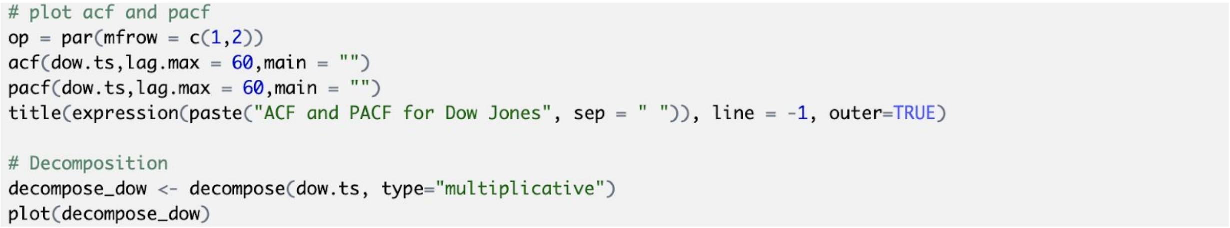
Initial Time Series Analysis:



```
# Importing Dow Data
dat<-read.csv("DJIA.csv")

# View first 6 observation on Dow Data
head(dat)
dat$DJIA<-as.numeric(dat$DJIA)

#Removing Missing Values
dow = dat[!dat$DJIA == 1,]

set.seed(123)
# Testing set, Training set
test.ind <- sample.int(n = nrow(dow), size = floor(0.1*nrow(dow)))
train.ind <- setdiff(1:nrow(dow), test.ind)
dowTrain <- dow[train.ind,]
dowTest <- dow[test.ind,]
```

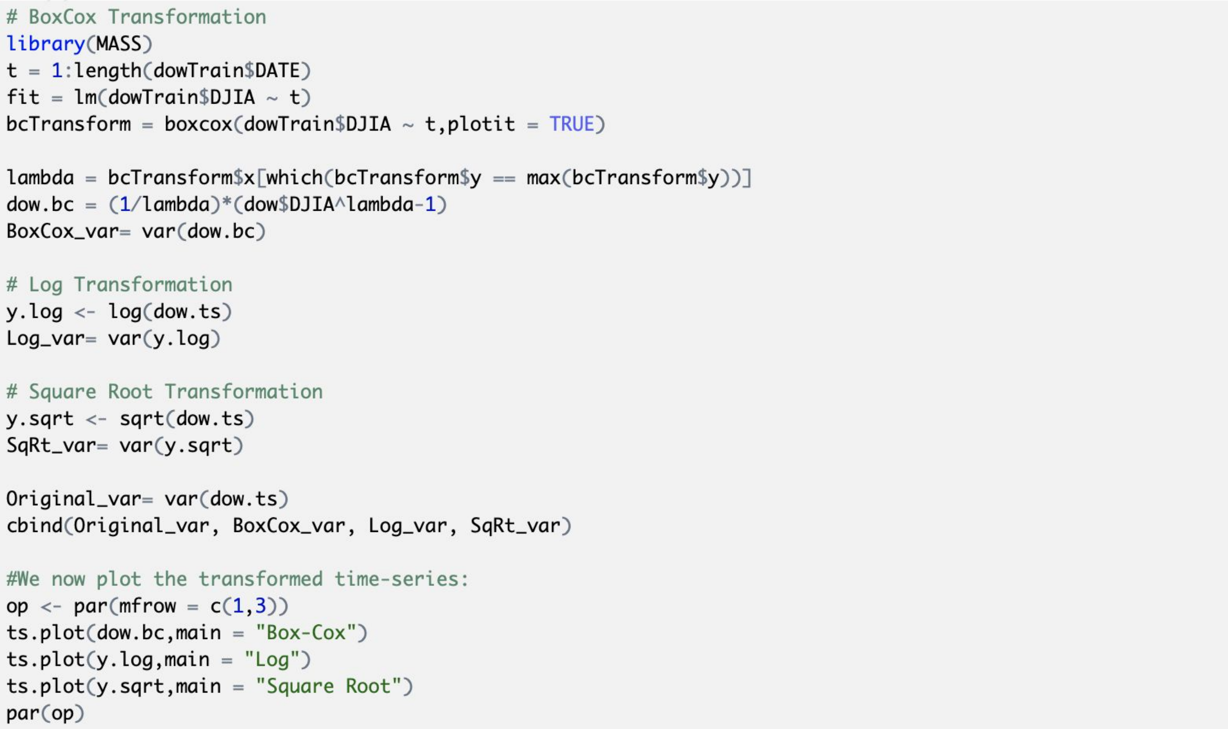


```
ind= seq(as.Date("2014-01-02"), as.Date("2019-01-01"), by= "day")
dow.ts<-ts(dowTrain$DJIA, start=c(2014,as.numeric(format(ind[1], "%j"))), frequency=260)
ts.plot(dow.ts, ylab = "DJIA", main = "Dow Time Series Plot")
# Variance Check
var(dow.ts)
```



```
# plot acf and pacf
op = par(mfrow = c(1,2))
acf(dow.ts,lag.max = 60,main = "")
pacf(dow.ts,lag.max = 60,main = "")
title(expression(paste("ACF and PACF for Dow Jones", sep = " ")), line = -1, outer=TRUE)

# Decomposition
decompose_dow <- decompose(dow.ts, type="multiplicative")
plot(decompose_dow)
```

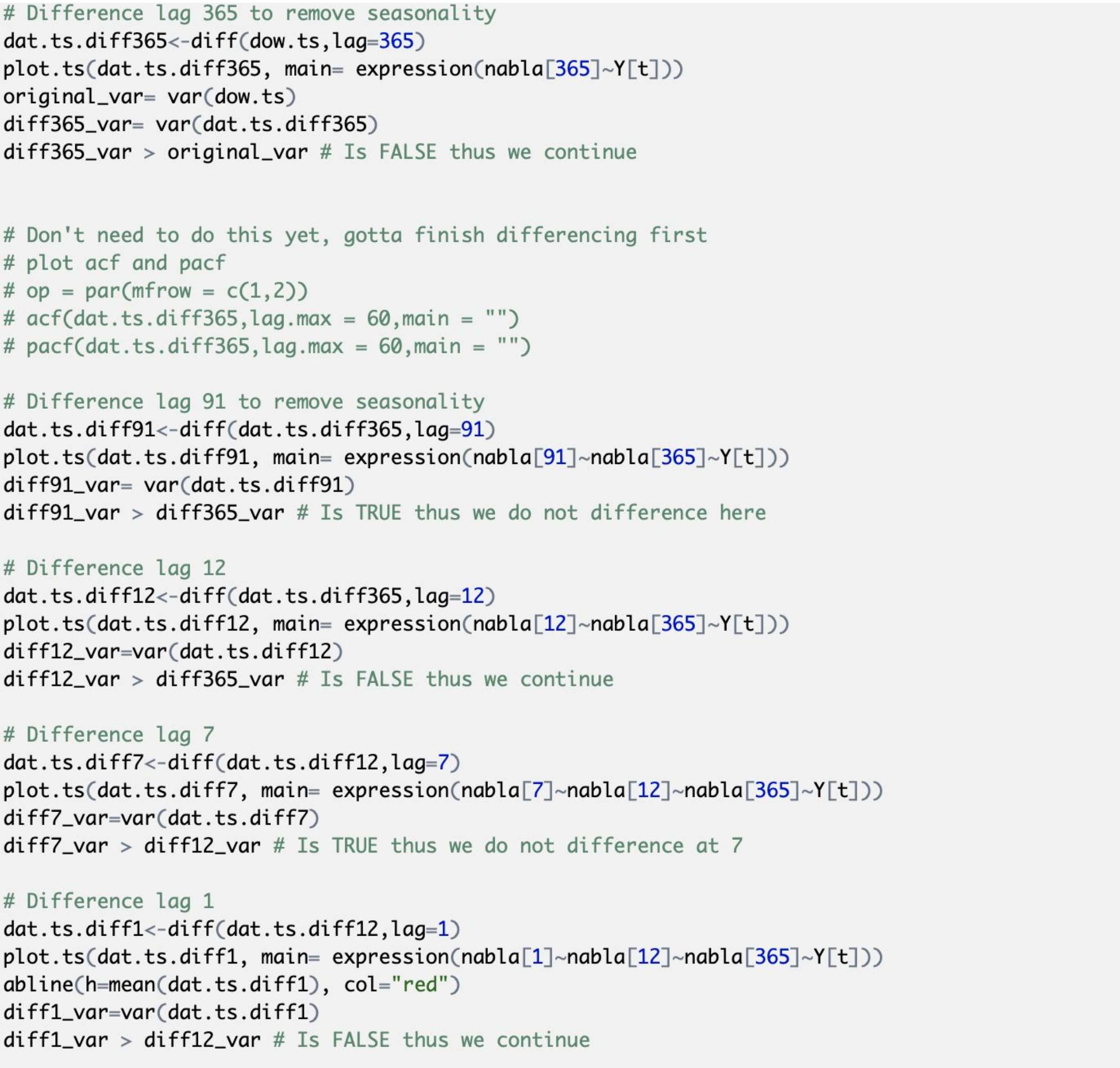
Box Cox Transformation:



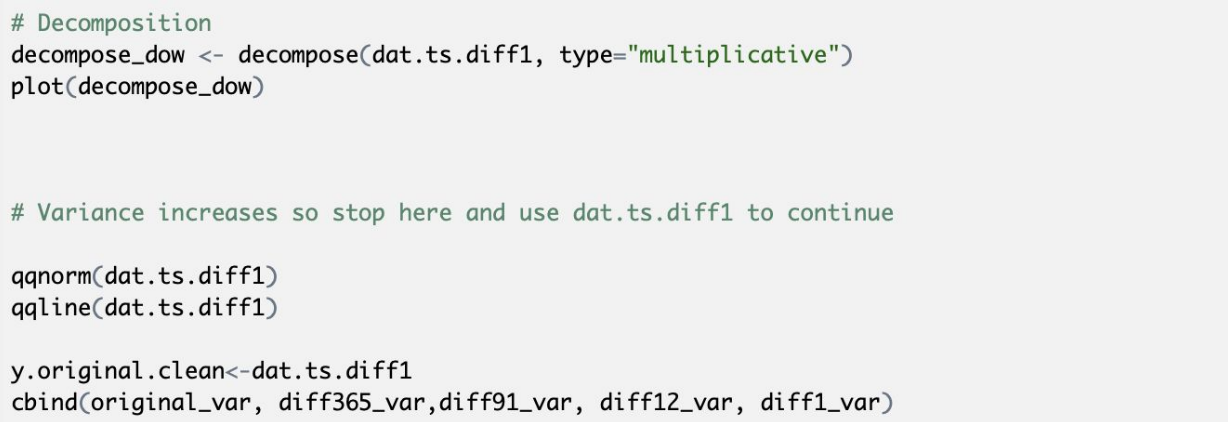
```
# BoxCox Transformation
library(MASS)
t = 1:length(dowTrain$DATE)
fit = lm(dowTrain$DJIA ~ t)
bcTransform = boxcox(dowTrain$DJIA ~ t,plotit = TRUE)

lambda = bcTransform$x[which(bcTransform$y == max(bcTransform$y))]
dow.bc = (1/lambda)*(dow$DJIA^lambda-1)
BoxCox_var= var(dow.bc)

# Log Transformation
y.log <- log(dow.ts)
Log_var= var(y.log)

# Square Root Transformation
y.sqrt <- sqrt(dow.ts)
SqRt_var= var(y.sqrt)

Original_var= var(dow.ts)
cbind(Original_var, BoxCox_var, Log_var, SqRt_var)

#We now plot the transformed time-series:
op <- par(mfrow = c(1,3))
ts.plot(dow.bc,main = "Box-Cox")
ts.plot(y.log,main = "Log")
ts.plot(y.sqrt,main = "Square Root")
par(op)
```

Removing Trend and Seasonality:

For initial time series data:

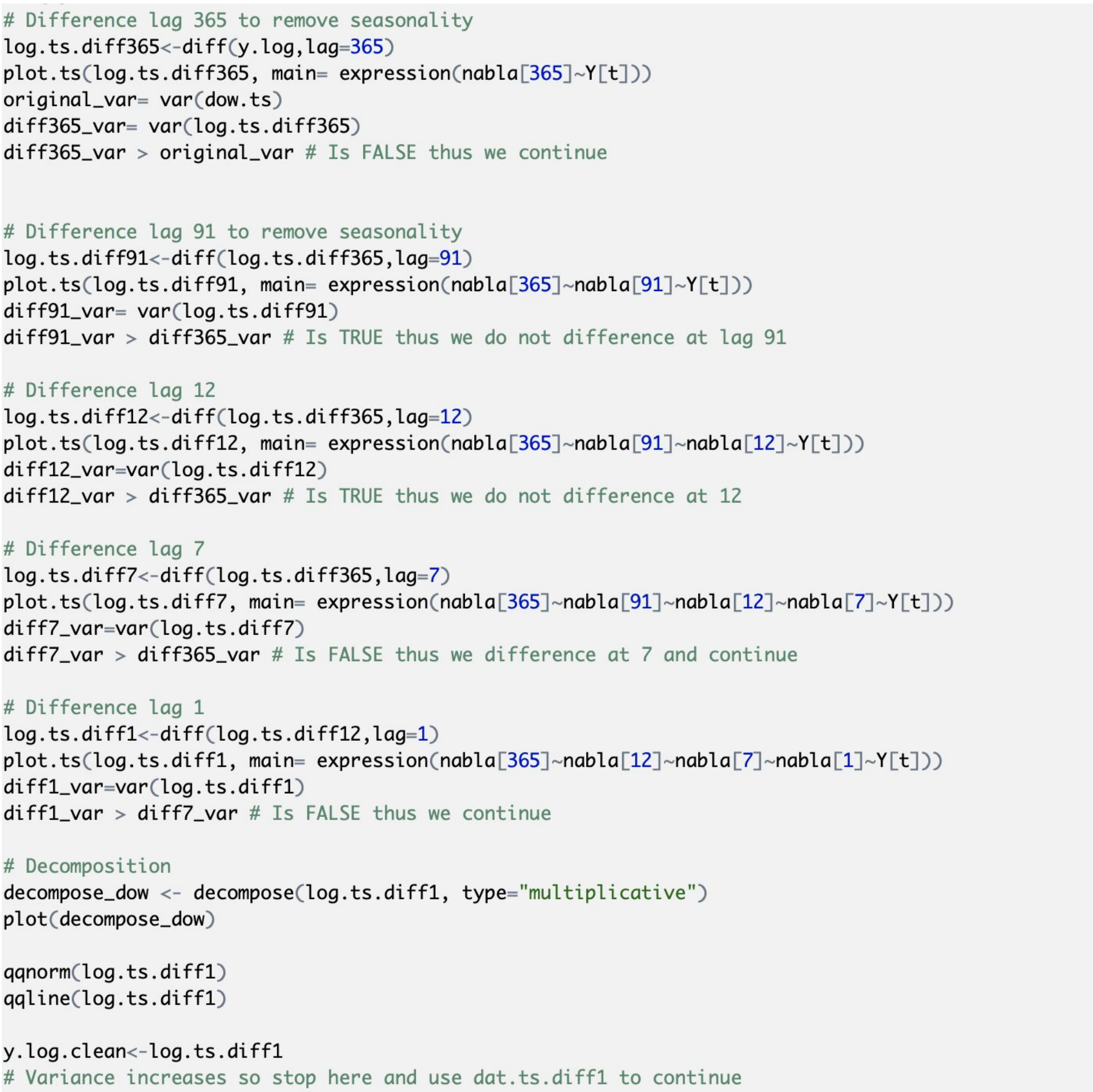


```
# Difference lag 365 to remove seasonality
dat.ts.diff365<-diff(dow.ts,lag=365)
plot.ts(dat.ts.diff365, main= expression(nabla[365]~Y[t]))
original_var= var(dow.ts)
diff365_var= var(dat.ts.diff365)
diff365_var > original_var # Is FALSE thus we continue


# Don't need to do this yet, gotta finish differencing first
# plot acf and pacf
# op = par(mfrow = c(1,2))
# acf(dat.ts.diff365,lag.max = 60,main = "")
# pacf(dat.ts.diff365,lag.max = 60,main = "")

# Difference lag 91 to remove seasonality
dat.ts.diff91<-diff(dat.ts.diff365,lag=91)
plot.ts(dat.ts.diff91, main= expression(nabla[91]~nabla[365]~Y[t]))
diff91_var= var(dat.ts.diff91)
diff91_var > diff365_var # Is TRUE thus we do not difference here

# Difference lag 12
dat.ts.diff12<-diff(dat.ts.diff365,lag=12)
plot.ts(dat.ts.diff12, main= expression(nabla[12]~nabla[365]~Y[t]))
diff12_var=var(dat.ts.diff12)
diff12_var > diff365_var # Is FALSE thus we continue

# Difference lag 7
dat.ts.diff7<-diff(dat.ts.diff12,lag=7)
plot.ts(dat.ts.diff7, main= expression(nabla[7]~nabla[12]~nabla[365]~Y[t]))
diff7_var=var(dat.ts.diff7)
diff7_var > diff12_var # Is TRUE thus we do not difference at 7

# Difference lag 1
dat.ts.diff1<-diff(dat.ts.diff12,lag=1)
plot.ts(dat.ts.diff1, main= expression(nabla[1]~nabla[12]~nabla[365]~Y[t]))
abline(h=mean(dat.ts.diff1), col="red")
diff1_var=var(dat.ts.diff1)
diff1_var > diff12_var # Is FALSE thus we continue
```

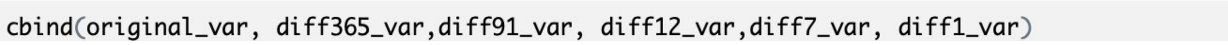


```
# Decomposition
decompose_dow <- decompose(dat.ts.diff1, type="multiplicative")
plot(decompose_dow)



# Variance increases so stop here and use dat.ts.diff1 to continue

qqnorm(dat.ts.diff1)
qqline(dat.ts.diff1)

y.original.clean<-dat.ts.diff1
cbind(original_var, diff365_var,diff91_var, diff12_var, diff1_var)
```

For log transformed data:

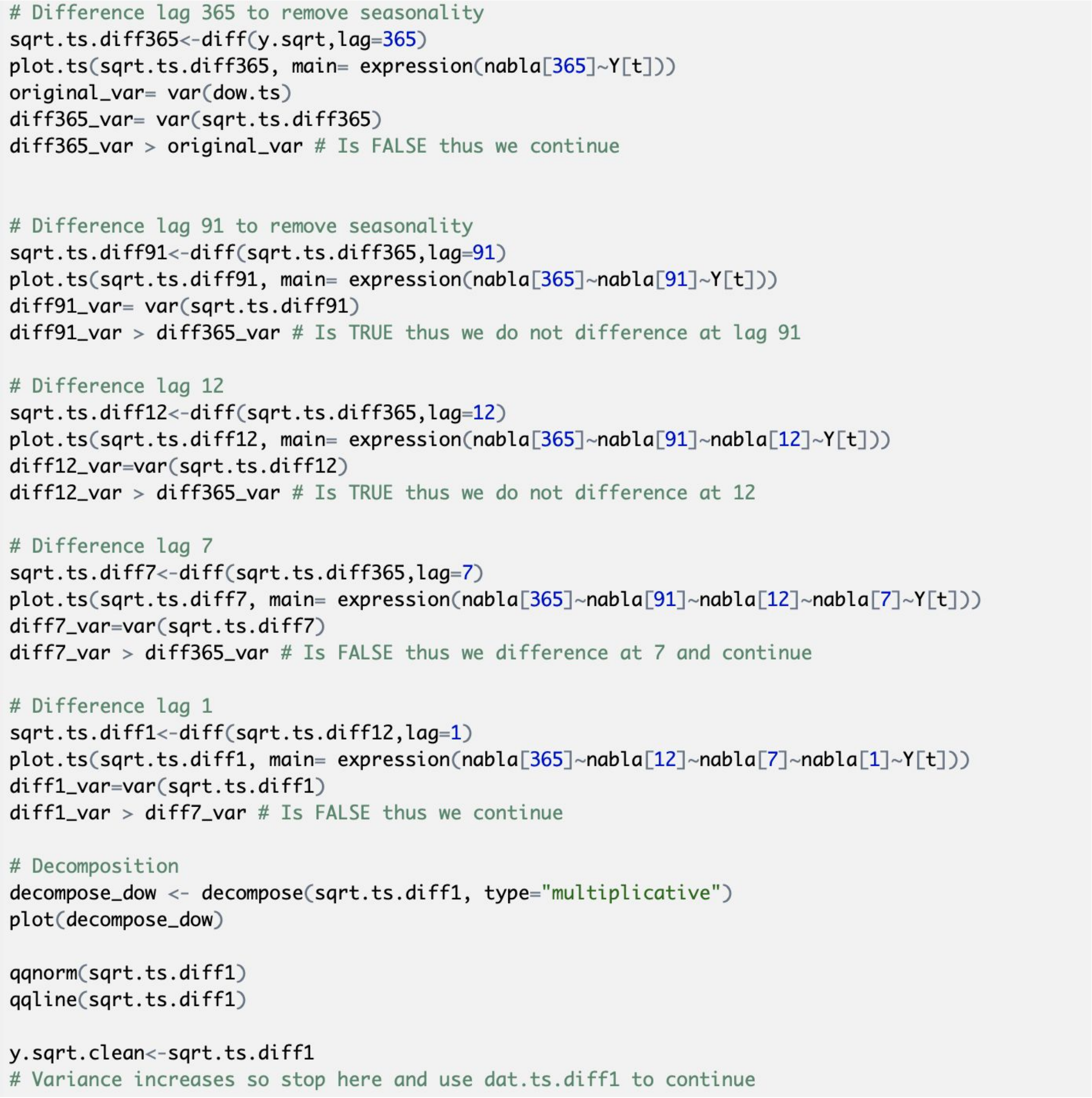


```
# Difference lag 365 to remove seasonality
log.ts.diff365<-diff(y.log,lag=365)
plot.ts(log.ts.diff365, main= expression(nabla[365]~Y[t]))
original_var= var(dow.ts)
diff365_var= var(log.ts.diff365)
diff365_var > original_var # Is FALSE thus we continue


# Difference lag 91 to remove seasonality
log.ts.diff91<-diff(log.ts.diff365,lag=91)
plot.ts(log.ts.diff91, main= expression(nabla[365]~nabla[91]~Y[t]))
diff91_var= var(log.ts.diff91)
diff91_var > diff365_var # Is TRUE thus we do not difference at lag 91

# Difference lag 12
log.ts.diff12<-diff(log.ts.diff365,lag=12)
plot.ts(log.ts.diff12, main= expression(nabla[365]~nabla[91]~nabla[12]~Y[t]))
diff12_var=var(log.ts.diff12)
diff12_var > diff365_var # Is TRUE thus we do not difference at 12

# Difference lag 7
log.ts.diff7<-diff(log.ts.diff365,lag=7)
plot.ts(log.ts.diff7, main= expression(nabla[365]~nabla[91]~nabla[12]~nabla[7]~Y[t]))
diff7_var=var(log.ts.diff7)
diff7_var > diff365_var # Is FALSE thus we difference at 7 and continue

# Difference lag 1
log.ts.diff1<-diff(log.ts.diff12,lag=1)
plot.ts(log.ts.diff1, main= expression(nabla[365]~nabla[12]~nabla[7]~nabla[1]~Y[t]))
diff1_var=var(log.ts.diff1)
diff1_var > diff7_var # Is FALSE thus we continue

# Decomposition
decompose_dow <- decompose(log.ts.diff1, type="multiplicative")
plot(decompose_dow)

qqnorm(log.ts.diff1)
qqline(log.ts.diff1)

y.log.clean<-log.ts.diff1
# Variance increases so stop here and use dat.ts.diff1 to continue
```

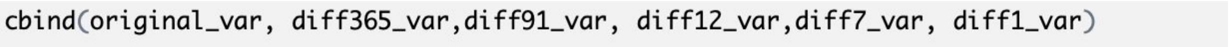


```
cbind(original_var, diff365_var,diff91_var, diff12_var,diff7_var, diff1_var)
```

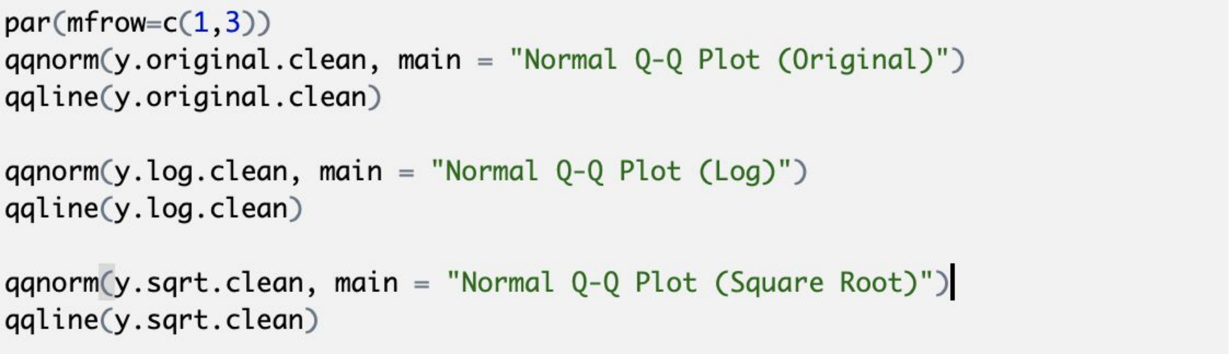
For square-root transformed data:



```
# Difference lag 365 to remove seasonality
sqrt.ts.diff365<-diff(y.sqrt,lag=365)
plot.ts(sqrt.ts.diff365, main= expression(nabla[365]~Y[t]))
original_var= var(dow.ts)
diff365_var= var(sqrt.ts.diff365)
diff365_var > original_var # Is FALSE thus we continue


# Difference lag 91 to remove seasonality
sqrt.ts.diff91<-diff(sqrt.ts.diff365,lag=91)
plot.ts(sqrt.ts.diff91, main= expression(nabla[365]~nabla[91]~Y[t]))
diff91_var= var(sqrt.ts.diff91)
diff91_var > diff365_var # Is TRUE thus we do not difference at lag 91

# Difference lag 12
sqrt.ts.diff12<-diff(sqrt.ts.diff365,lag=12)
plot.ts(sqrt.ts.diff12, main= expression(nabla[365]~nabla[91]~nabla[12]~Y[t]))
diff12_var=var(sqrt.ts.diff12)
diff12_var > diff365_var # Is TRUE thus we do not difference at 12

# Difference lag 7
sqrt.ts.diff7<-diff(sqrt.ts.diff365,lag=7)
plot.ts(sqrt.ts.diff7, main= expression(nabla[365]~nabla[91]~nabla[12]~nabla[7]~Y[t]))
diff7_var=var(sqrt.ts.diff7)
diff7_var > diff365_var # Is FALSE thus we difference at 7 and continue

# Difference lag 1
sqrt.ts.diff1<-diff(sqrt.ts.diff12,lag=1)
plot.ts(sqrt.ts.diff1, main= expression(nabla[365]~nabla[12]~nabla[7]~nabla[1]~Y[t]))
diff1_var=var(sqrt.ts.diff1)
diff1_var > diff7_var # Is FALSE thus we continue

# Decomposition
decompose_dow <- decompose(sqrt.ts.diff1, type="multiplicative")
plot(decompose_dow)

qqnorm(sqrt.ts.diff1)
qqline(sqrt.ts.diff1)

y.sqrt.clean<-sqrt.ts.diff1
# Variance increases so stop here and use dat.ts.diff1 to continue
```



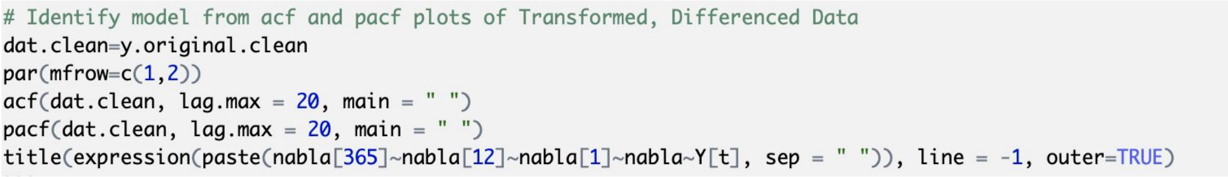
```
cbind(original_var, diff365_var,diff91_var, diff12_var,diff7_var, diff1_var)
```



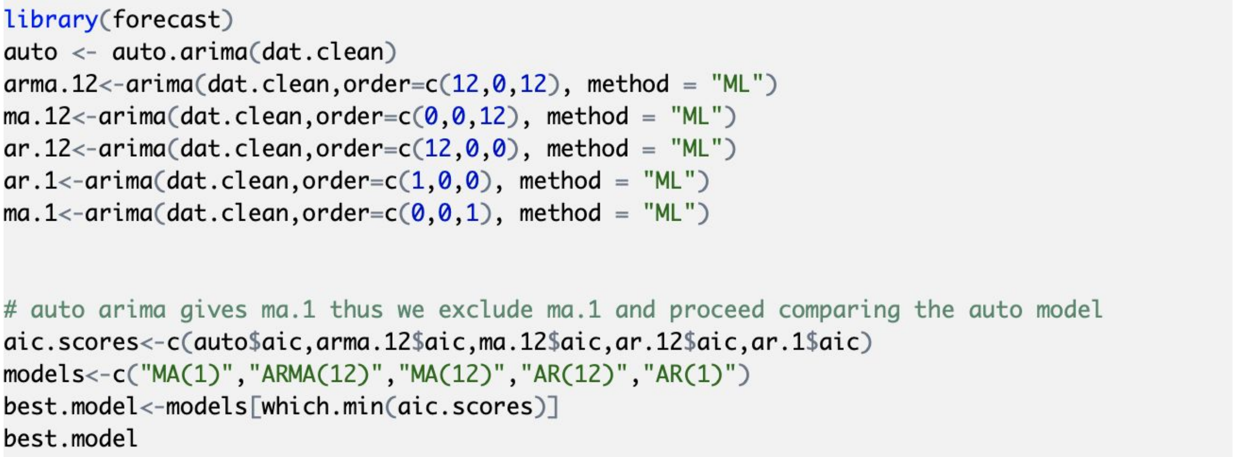
```r
par(mfrow=c(1,3))
qqnorm(y.original.clean, main = "Normal Q-Q Plot (Original)")
qqline(y.original.clean)

qqnorm(y.log.clean, main = "Normal Q-Q Plot (Log)")
qqline(y.log.clean)

qqnorm(y.sqrt.clean, main = "Normal Q-Q Plot (Square Root)")
qqline(y.sqrt.clean)
```

Models:

Model identification:

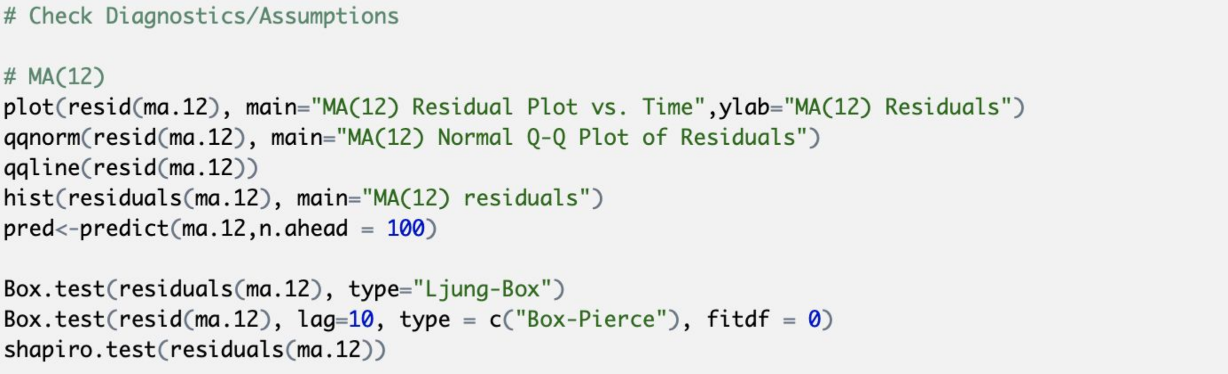


```r
# Identify model from acf and pacf plots of Transformed, Differenced Data
dat.clean=y.original.clean
par(mfrow=c(1,2))
acf(dat.clean, lag.max = 20, main = " ")
pacf(dat.clean, lag.max = 20, main = " ")
title(expression(paste(nabla[365]~nabla[12]~nabla[1]~nabla~Y[t], sep = " ")), line = -1, outer=TRUE)
```



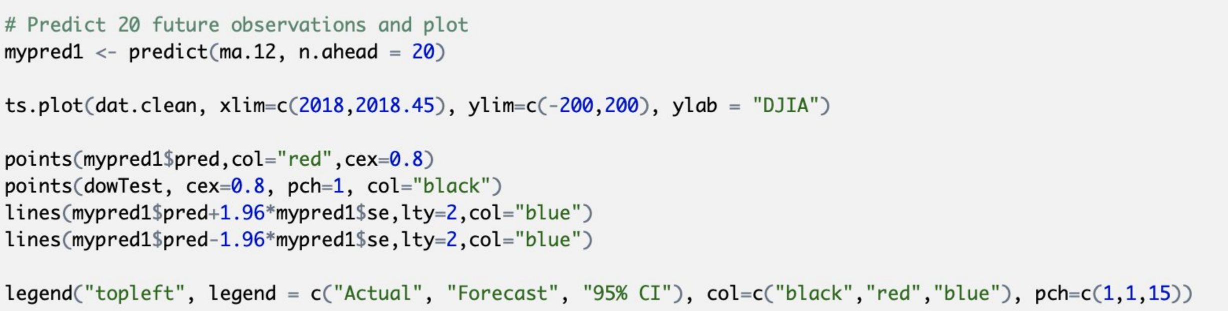
```r
library(forecast)
auto <- auto.arima(dat.clean)
arma.12<-arima(dat.clean,order=c(12,0,12), method = "ML")
ma.12<-arima(dat.clean,order=c(0,0,12), method = "ML")
ar.12<-arima(dat.clean,order=c(12,0,0), method = "ML")
ar.1<-arima(dat.clean,order=c(1,0,0), method = "ML")
ma.1<-arima(dat.clean,order=c(0,0,1), method = "ML")


# auto arima gives ma.1 thus we exclude ma.1 and proceed comparing the auto model
aic.scores<-c(auto$aic,arma.12$aic,ma.12$aic,ar.12$aic,ar.1$aic)
models<-c("MA(1)","ARMA(12)","MA(12)","AR(12)","AR(1)")
best.model<-models[which.min(aic.scores)]
best.model
```

Diagnostics Checking on Test Models:



```
# Check Diagnostics/Assumptions

# MA(12)
plot(resid(ma.12), main="MA(12) Residual Plot vs. Time",ylab="MA(12) Residuals")
qqnorm(resid(ma.12), main="MA(12) Normal Q-Q Plot of Residuals")
qqline(resid(ma.12))
hist(residuals(ma.12), main="MA(12) residuals")
pred<-predict(ma.12,n.ahead = 100)

Box.test(residuals(ma.12), type="Ljung-Box")
Box.test(resid(ma.12), lag=10, type = c("Box-Pierce"), fitdf = 0)
shapiro.test(residuals(ma.12))
```

Forecasting:



```
# Predict 20 future observations and plot
mypred1 <- predict(ma.12, n.ahead = 20)

ts.plot(dat.clean, xlim=c(2018,2018.45), ylim=c(-200,200), ylab = "DJIA")

points(mypred1$pred,col="red",cex=0.8)
points(dowTest, cex=0.8, pch=1, col="black")
lines(mypred1$pred+1.96*mypred1$se,lty=2,col="blue")
lines(mypred1$pred-1.96*mypred1$se,lty=2,col="blue")

legend("topleft", legend = c("Actual", "Forecast", "95% CI"), col=c("black","red","blue"), pch=c(1,1,15))
```