Apple Stock Price

Group 1- Charaf Lachouri, Alaba Olanipekun, Oscar Pacheco, Chandler Westmoreland

2022-12-03

1. Introduction

There are few components as important as the stock market, in a free market system. In a nutshell, a stock market is a tool that allows companies to raise money by offering small parts of the company, aka shares, which then gives shareholders chances to make a profit off the success of a company.
Maneuvering the stock market can be a 4-year degree. Knowing when to buy and sell a stock can cost someone a fortune. The best traders can see things in a line graph that ordinary people cannot see. Often, their ability to correctly decipher a trend in the movement of stocks gives them an automatic leg up, in one of the most competitive sectors of the global economy. That's where a general understanding of time series analysis comes in. Now there are tools available for traders that allow one to access trends with the press of a button, giving regular individuals more information to make educated assessments of a stock potential values. In a field as potentially volatile as the stock market, an educated risk can be the difference between financial life or death.
In this analysis, an attempt to find an appropriate model of the movement of Apple stock (AAPL) from the years 1980 to 2022. Once an appropriate model to use for comparison is found, forecasting of future values in the next 12 months will be attempted, in the hopes of potentially finding an advantageous time to purchase apple stock.

2. Exploratory Data Analysis

This data was collected via yahoo Finance, following Apple Stock (AAPL) prices from Dec of 1980 to June 2022. Outside of the year 1980, it includes the data for opening and closing price, high and low on a particular day, adjusted closing which takes in company expenses and the volume of times shares were traded during a day; for about 250 days in each year.

This data set was found on Kaggle.com. It came in the form of a CSV and was not augmented outside or tampered with outside of RStudio. The link is attached below.

https://www.kaggle.com/datasets/meetnagadia/apple-stock-price-from-19802021

Data Description:

The Apple Stock Price from 1980-2022 data ser contains 500 observations (rows) and 7 variables (columns). There are few components as important as the stock market, in a free market system. In a nutshell, a stock market is a tool that allows companies to raise money by offering small parts of the company, aka shares, which then gives shareholders chances to make a profit off the success of a company. Maneuvering the stock market can be a 4-year degree. Knowing when to buy and sell a stock can cost someone a fortune. The best traders can see things in a line graph that ordinary people cannot see. Often, their ability to correctly decipher a trend in the movement of stocks gives them an automatic leg up, in one of the most competitive sectors of the global economy. That's where a general understanding of time series analysis comes in. Now there are tools available for traders that allow one to access trends with the press of a button, giving regular individuals more information to make educated assessments of a stock potential values. In a field as potentially volatile as the stock market, an educated risk can be the difference between financial life or death.

In this analysis, an attempt to find an appropriate model of the movement of Apple stock (AAPL) from the years 1980 to 2022. Once an appropriate model to use for comparison is found, forecasting of future values in the next 12 months will be attempted, in the hopes of potentially finding an advantageous time to purchase apple stock.

Variables:

Ordinal Categorical Variable:

Our one ordinal categorical variable "Date" contains the name of the month and its corresponding year from 1980 – 2022.

Numerical or Quanititative Variable:

Our Numerical variables "Open", "High", "Low", "Close", "Adj.Close" and "Volume", contain the average values of monthly trading prices and values. By looking at the distribution statistics provided by summary (), we can observe that all our numerical variables, do not have normal distributed data, which makes sense for the changes over the time of this very profitable and successful stock.

3. Statistical Summary

```r
# Descriptions
summary(grpdata)
```

```
##      Date                Open               High                Low
##  Length:500         Min.   :  0.0575   Min.   :  0.0579   Min.   :  0.05732
##  Class :character   1st Qu.:  0.2860   1st Qu.:  0.2922   1st Qu.:  0.27920
##  Mode  :character   Median :  0.4632   Median :  0.4733   Median :  0.45703
##                     Mean   : 14.8392   Mean   : 15.0037   Mean   : 14.67413
##                     3rd Qu.: 15.0648   3rd Qu.: 15.2470   3rd Qu.: 14.87696
##                     Max.   :173.1555   Max.   :175.6323   Max.   :171.25273
##      Close              Adj.Close            Volume
##  Min.   :  0.05732   Min.   :  0.04474   Min.   :1.693e+07
##  1st Qu.:  0.28566   1st Qu.:  0.23661   1st Qu.:1.457e+08
##  Median :  0.46608   Median :  0.37634   Median :2.317e+08
##  Mean   : 14.84402   Mean   : 14.21225   Mean   :3.307e+08
##  3rd Qu.: 15.04315   3rd Qu.: 13.02727   3rd Qu.:4.256e+08
##  Max.   :173.55273   Max.   :173.07760   Max.   :1.801e+09
```

```r
# Checking the structure of the dataset
str(grpdata)
```

```
## 'data.frame':    500 obs. of  7 variables:
##  $ Date     : chr  "Dec-80" "Jan-81" "Feb-81" "Mar-81" ...
##  $ Open     : num  0.136 0.141 0.118 0.111 0.122 ...
##  $ High     : num  0.136 0.141 0.119 0.112 0.122 ...
##  $ Low      : num  0.136 0.141 0.118 0.111 0.122 ...
##  $ Close    : num  0.136 0.141 0.118 0.111 0.122 ...
##  $ Adj.Close: num  0.1061 0.1098 0.092 0.0865 0.0949 ...
##  $ Volume   : num  1.03e+08 2.94e+07 1.69e+07 3.19e+07 2.56e+07 ...
```

```r
cl = grpdata$Close
summary(cl)
```

```
##      Min.   1st Qu.    Median      Mean   3rd Qu.      Max.
##   0.05732   0.28566   0.46608  14.84402  15.04315 173.55273
```

```r
mode(cl)
```

```
## [1] "numeric"
```

```r
#standard deviation
sd(cl)
```

```
## [1] 32.11468
```

```r
#variance
var(cl)
```

```
## [1] 1031.353
```

4. Data cleaning (remove noise and inconsistent data)

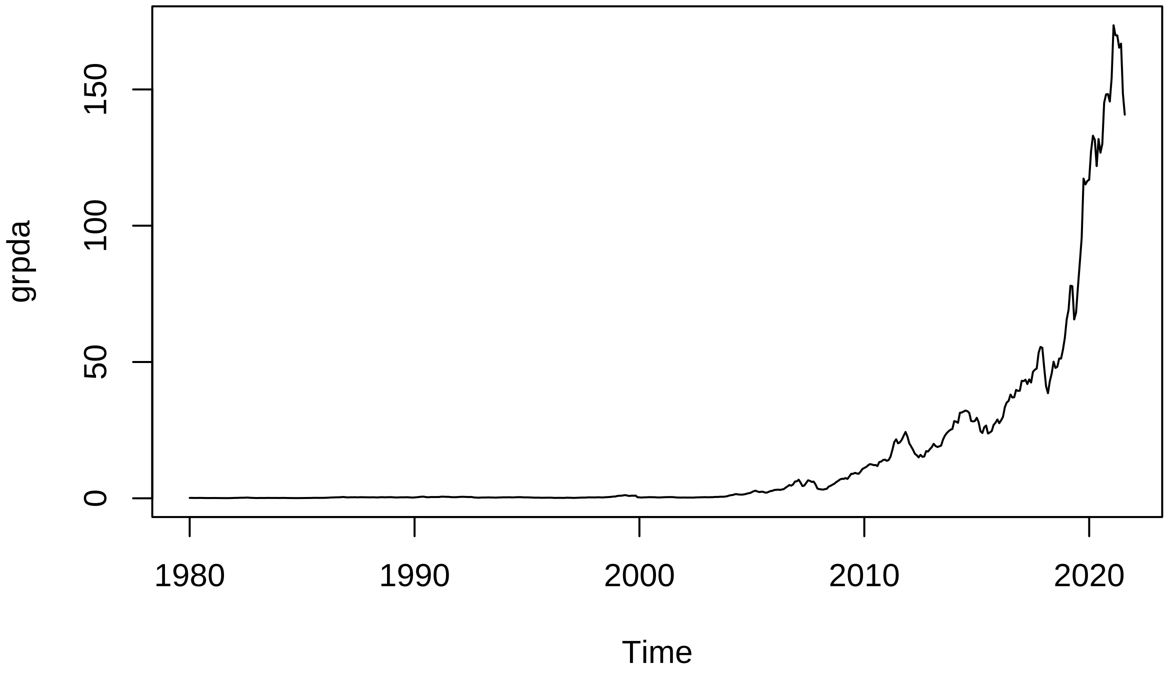
```r
# Missing values ?
sum(is.na(grpdata))
```

```
## [1] 0
```

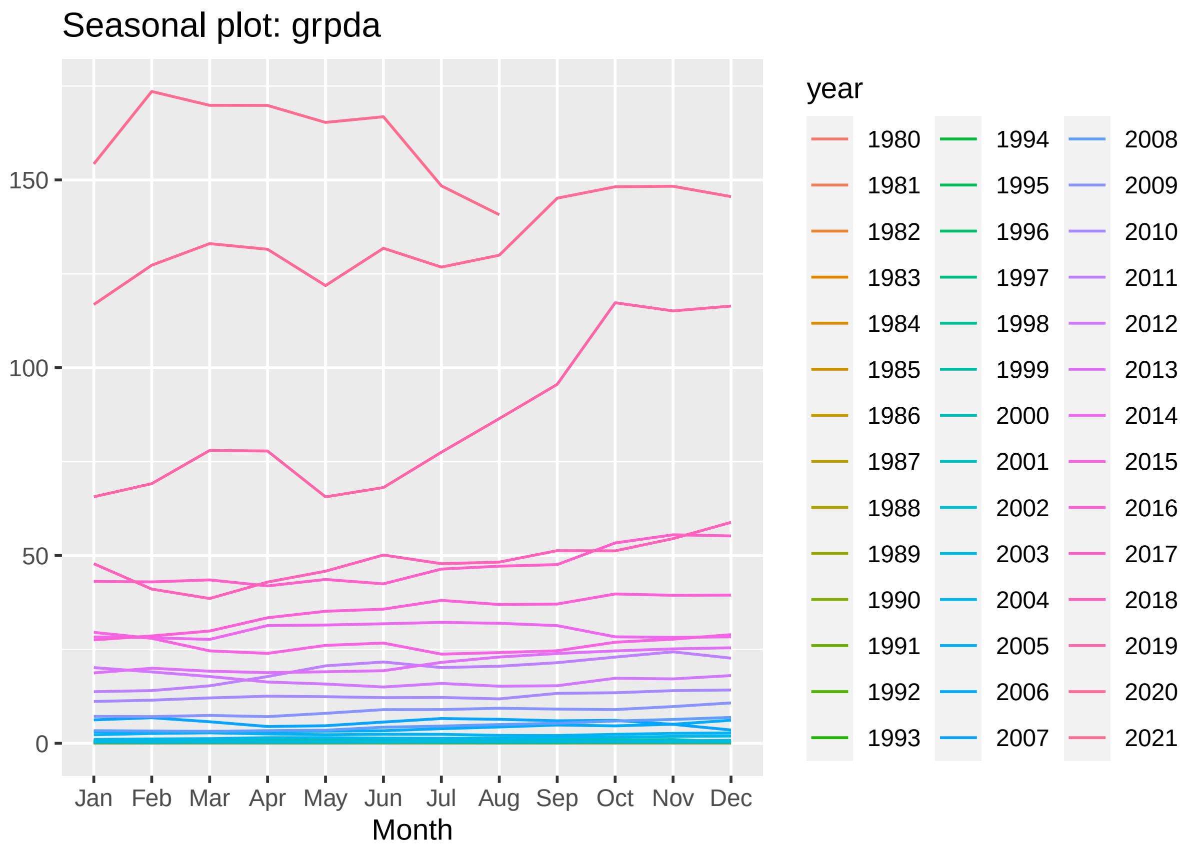
No missing values found.

5. Data Visualization

```r
grpda <- ts(grpdata$Close, start = 1980, frequency = 12)
plot.ts(grpda)
```



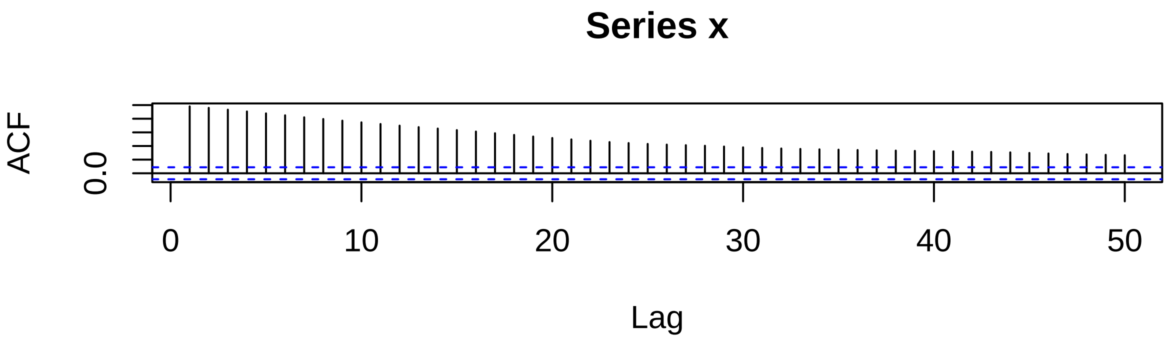
```
ggseasonplot(grpda)
```

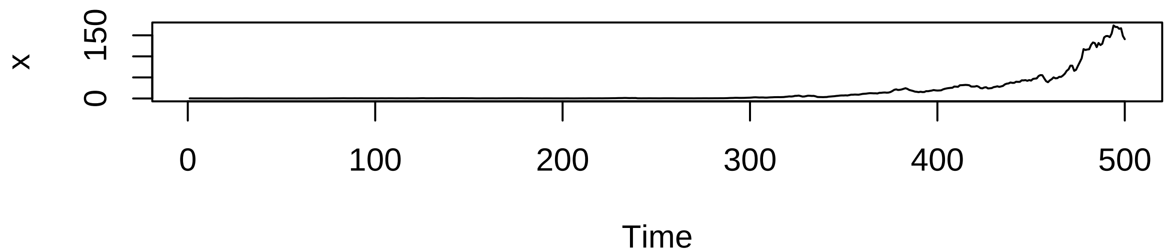


From the season plot above, you can make out some of the volatility in term of by month. The slight incline from beginning to end present in most years does show that some seasonality should be expected.

6. Stationarity Check out

First an the plot for x, our time series closing price data, will accessed for stationarity with its acf.

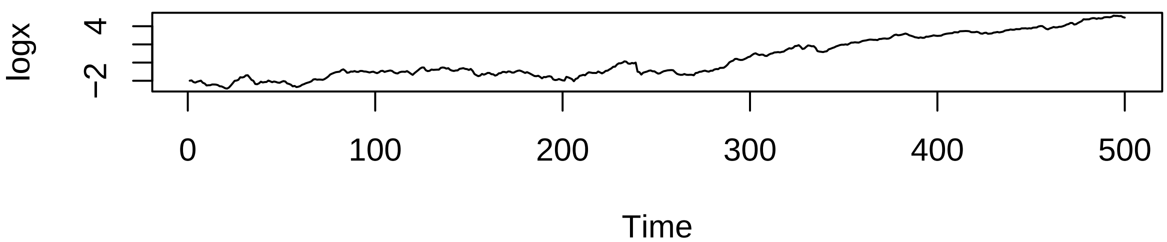


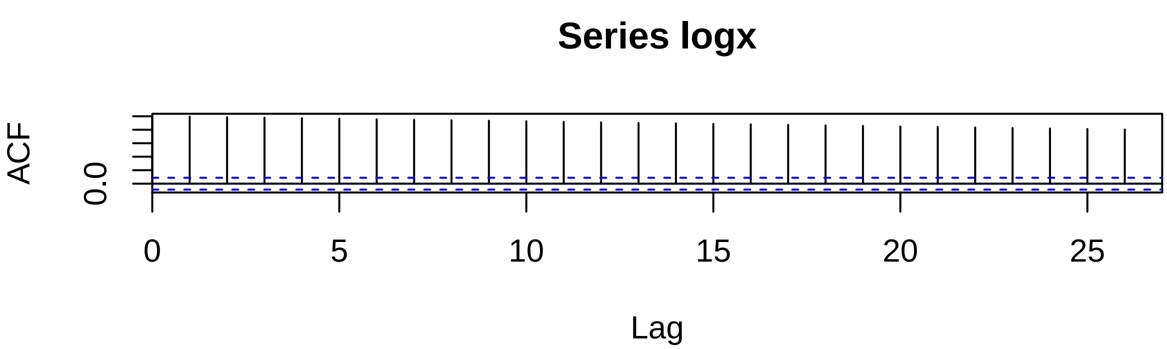


Clearly upper trend and slowly decay ACF, no stationarity so far, let's apply some data transformation on our dataset in order to stabilize the variance.

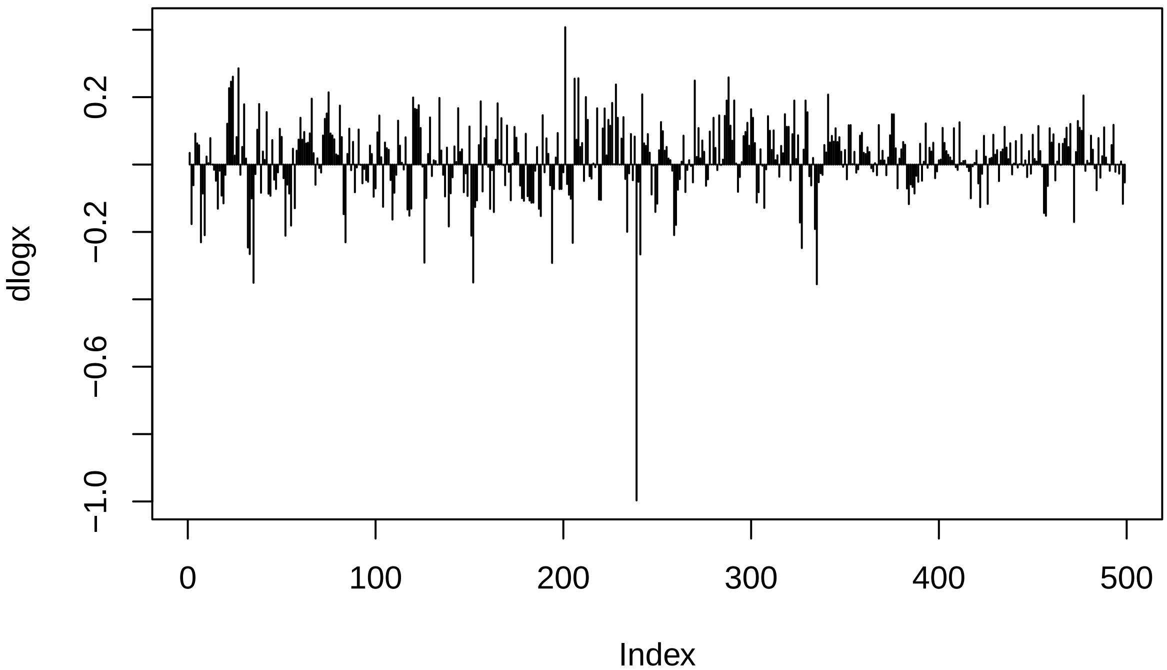
7. Data Transformation

First a log transformation was performed.

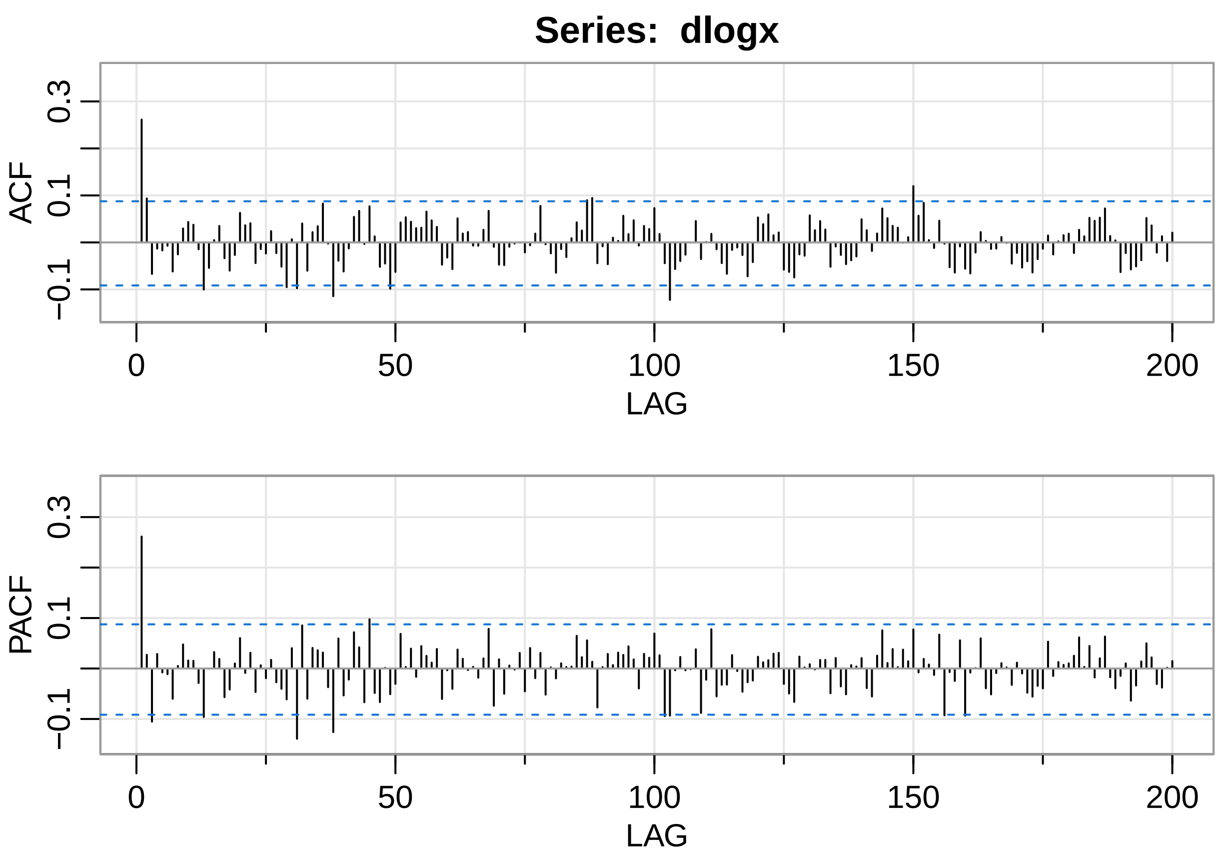




Still no stationarity, so differencing was applied.

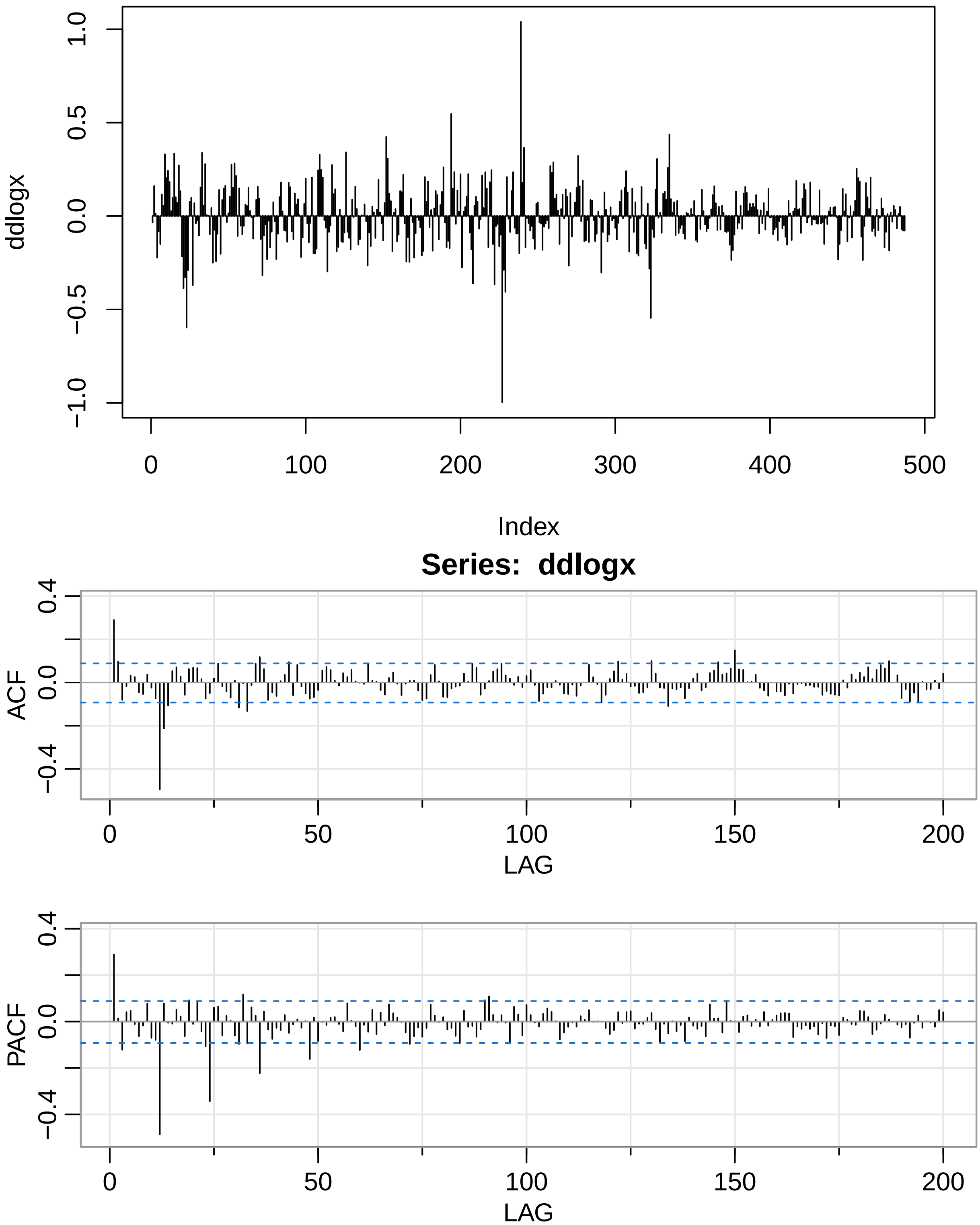


Despite couple spikes, it Looks much better.



The acf drops quickly which is what you want in stationary data.

Let's apply one more differencing to see if it will look a bit better.



The seasonal ACF cuts off after H = 1 and its PACF tails off which looks like we're having SMA(1) model here. Non-seasonal ACF and its PACF are tailing off which means we're having ARMA model here. In this case we need to check with the Extended ACF.

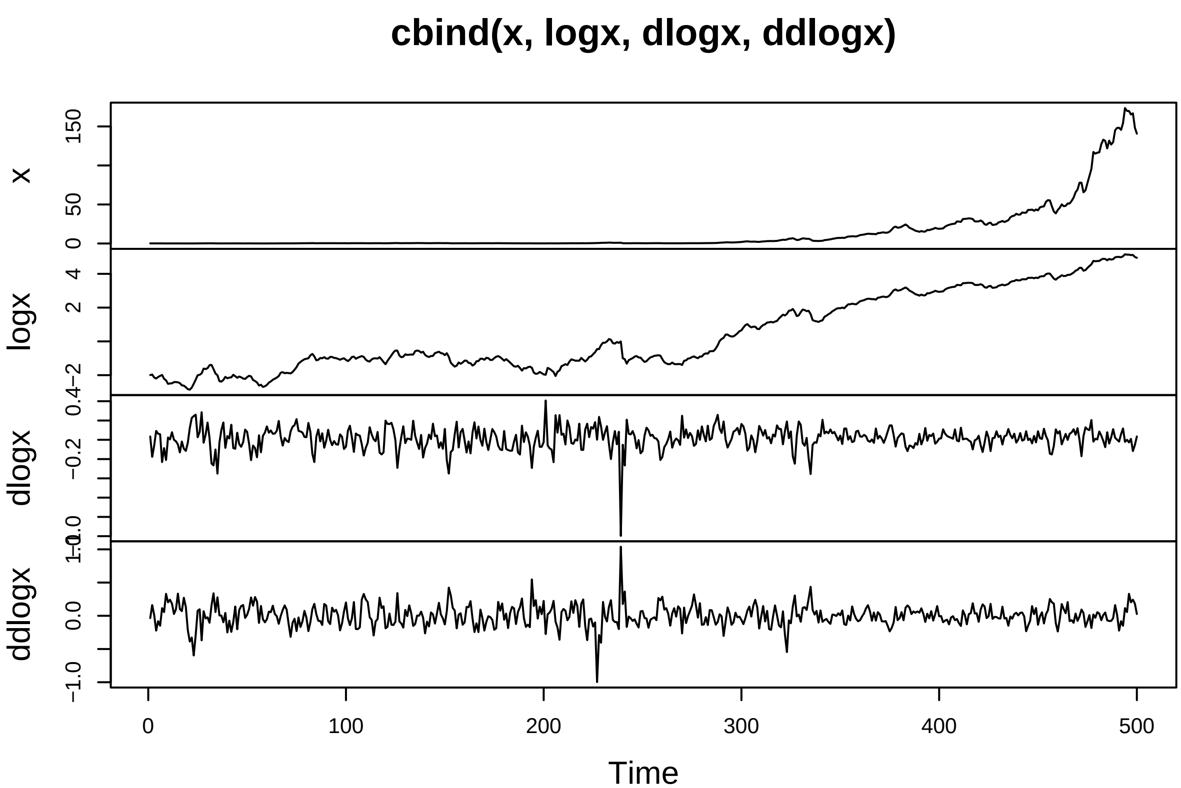
8. Model Selection

Using EACF, the appropriate terms for non-seasonal ARMA models will be determined.

```
## AR/MA
##   0 1 2 3 4 5 6 7 8 9 10 11 12 13
## 0 x x o o o o o o o o o  x  x  x
## 1 o x x o o o o o o o o  x  o  x
## 2 x x o o o o o o o o o  x  o  x
## 3 x x x o o o o o o o o  x  o  x
## 4 x x x o o o o o o o o  x  x  o
## 5 x x o o o o o o o o o  x  x  x
## 6 x x o o x o o o o o o  x  x  x
## 7 x x o o o o o o o o o  x  o  o
```

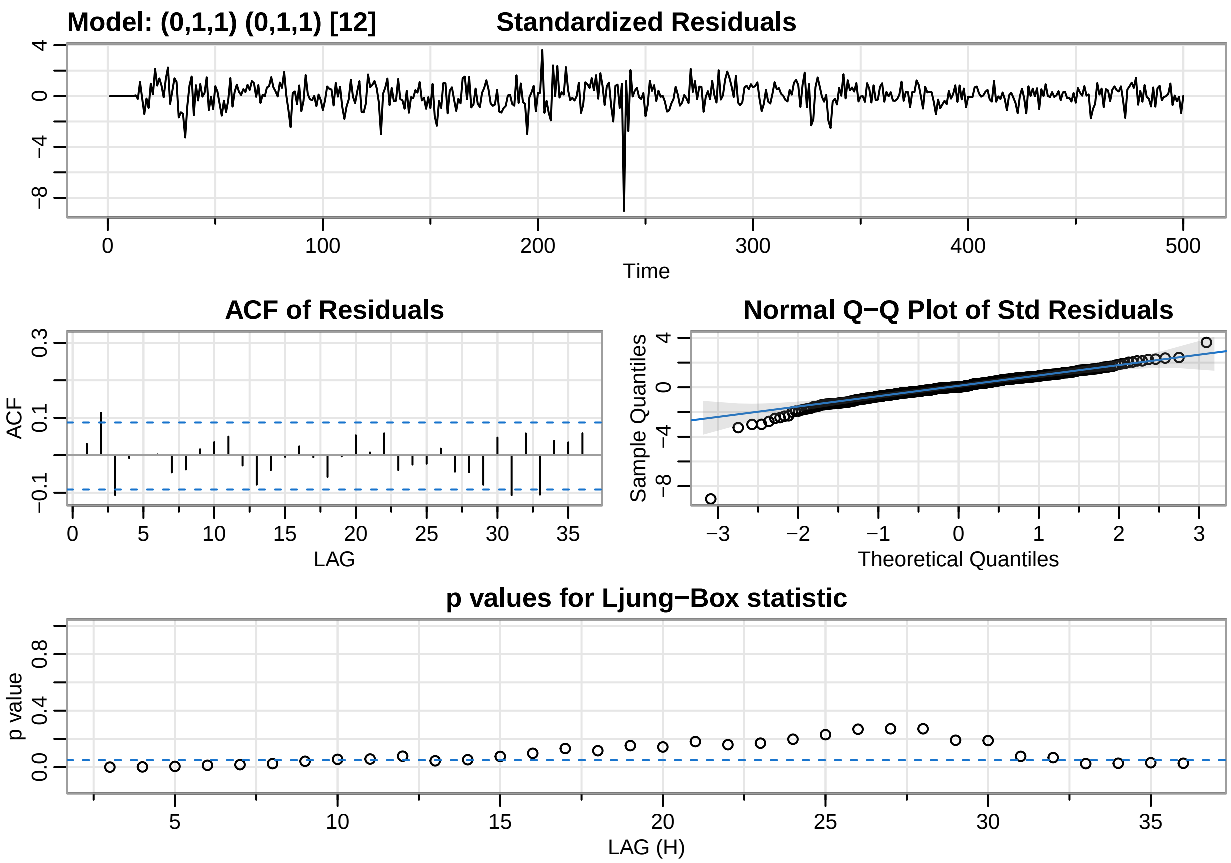
#(p,q) = (0,1),(1,1),(1,2),(2,2). Let's try all of them.

```
#plots
plot.ts(cbind(x,logx,dlogx,ddlogx ))
```



From here you can see the minimal effect 2nd order differencing does. Also, the 1st order differenced acf has the steep drop seen in stationary data, so models will only take into account first order differencing. The idea here is to not overfit.

```
sarima(logx,0,1,1, 0,1,1, 12) #model 1
```

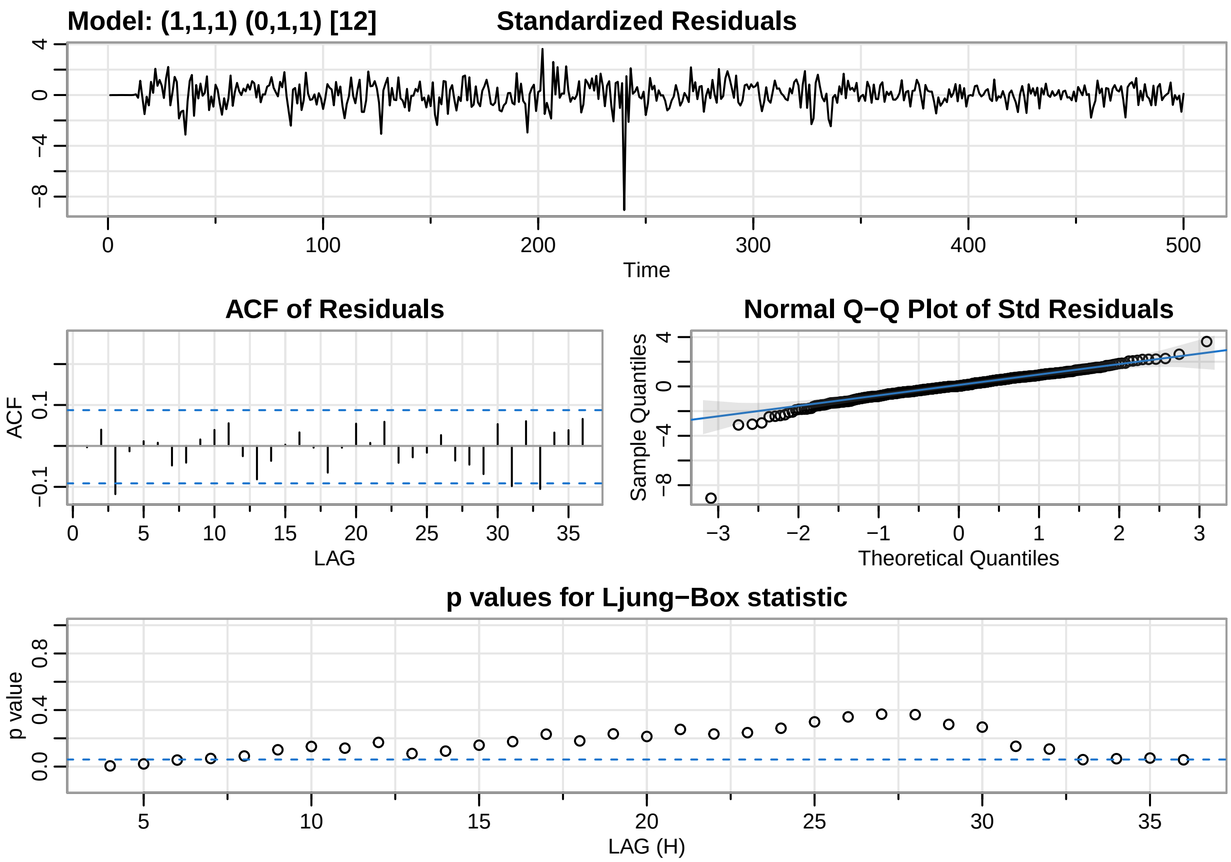



```
## $fit
##
## Call:
## arima(x = xdata, order = c(p, d, q), seasonal = list(order = c(P, D, Q), period = S),
##     include.mean = !no.constant, transform.pars = trans, fixed = fixed, optim.control = list(trace =
##         REPORT = 1, reltol = tol))
##
## Coefficients:
##          ma1     sma1
##       0.2368  -0.9690
## s.e.  0.0385   0.0391
##
## sigma^2 estimated as 0.01205:  log likelihood = 368.61,  aic = -731.23
##
## $degrees_of_freedom
## [1] 485
##
## $ttable
##      Estimate     SE  t.value p.value
## ma1    0.2368 0.0385   6.1574       0
## sma1  -0.9690 0.0391 -24.7860       0
##
## $AIC
## [1] -1.501493
##
## $AICc
## [1] -1.501442
```

```
##
## $BIC
## [1] -1.475693
```

Good residuals variation, good normal distribution, all the data points in the confidence bound except one data point (It could be outlier), Most of the P-values are under the 0.05 bar. Let's try the next model ARMA(1,1).

```
sarima(logx, 1,1,1, 0,1,1, 12) # model2
```

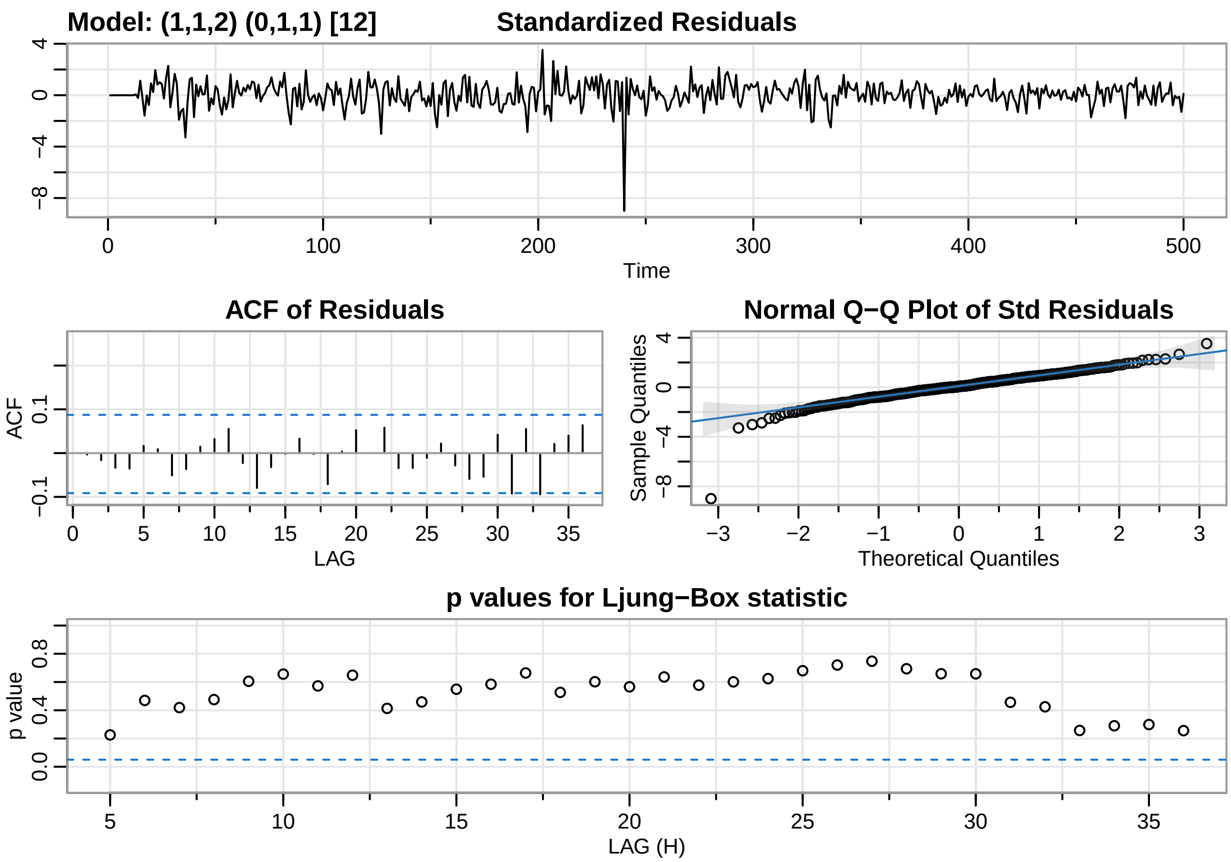



```
## $fit
##
## Call:
## arima(x = xdata, order = c(p, d, q), seasonal = list(order = c(P, D, Q), period = S),
##     include.mean = !no.constant, transform.pars = trans, fixed = fixed, optim.control = list(trace =
##         REPORT = 1, reltol = tol))
##
## Coefficients:
##          ar1      ma1     sma1
##       0.3262  -0.0503  -0.9749
## s.e.  0.1175   0.1205   0.0468
##
## sigma^2 estimated as 0.01185:  log likelihood = 371.65,  aic = -735.3
##
## $degrees_of_freedom
## [1] 484
##
## $ttable
```

```
##      Estimate     SE  t.value p.value
## ar1     0.3262 0.1175   2.7755  0.0057
## ma1    -0.0503 0.1205  -0.4175  0.6765
## sma1   -0.9749 0.0468 -20.8451  0.0000
##
## $AIC
## [1] -1.509861
##
## $AICc
## [1] -1.509759
##
## $BIC
## [1] -1.475461
```

Better than model one but still is not good enough. Let's try the next model ARMA (1,2).

```
sarima(logx, 1,1,2, 0,1,1, 12) #model3
```

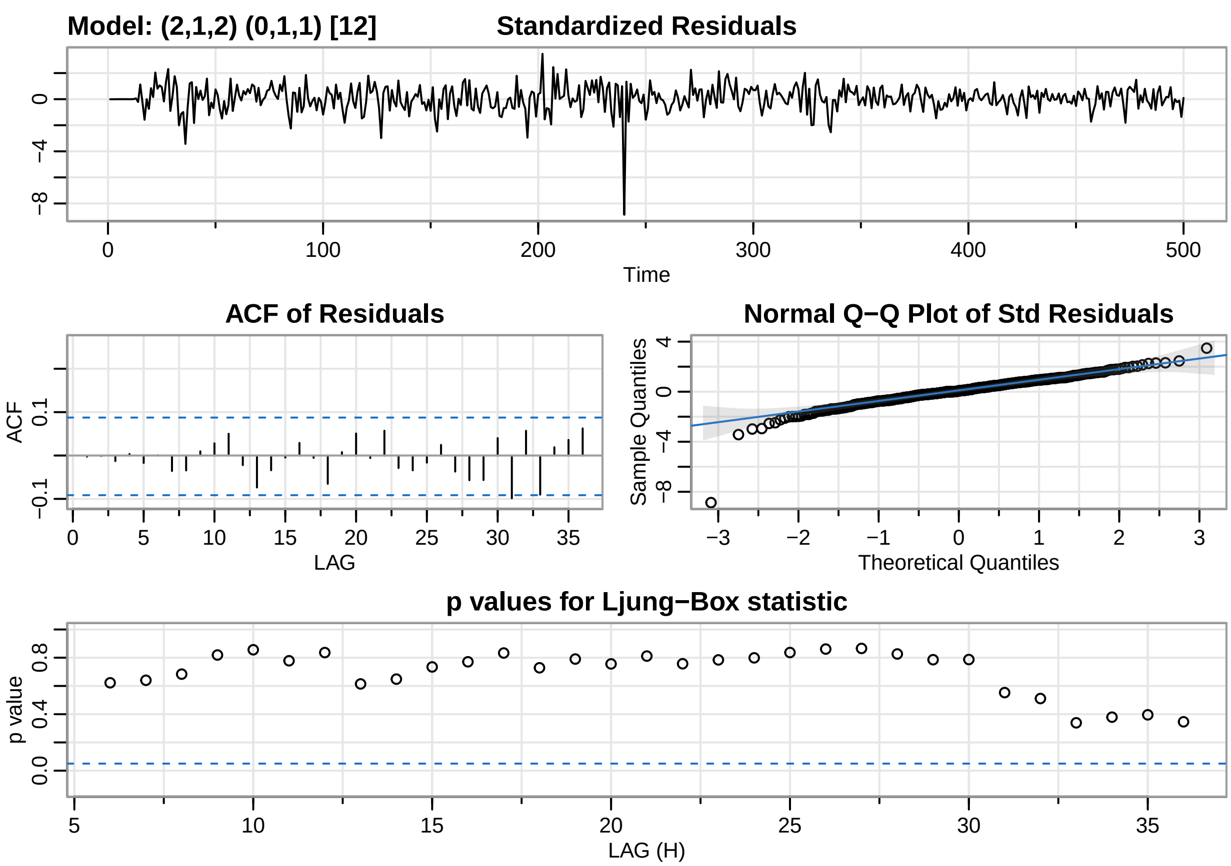



```
## $fit
##
## Call:
## arima(x = xdata, order = c(p, d, q), seasonal = list(order = c(P, D, Q), period = S),
##     include.mean = !no.constant, transform.pars = trans, fixed = fixed, optim.control = list(trace =
##         REPORT = 1, reltol = tol))
##
## Coefficients:
##           ar1     ma1     ma2     sma1
##       -0.2600  0.5410  0.2232  -0.9792
```

```
## s.e.   0.2286  0.2218  0.0614   0.0547
##
## sigma^2 estimated as 0.01166:  log likelihood = 374.83,  aic = -739.65
##
## $degrees_of_freedom
## [1] 483
##
## $ttable
##      Estimate     SE  t.value p.value
## ar1   -0.2600 0.2286  -1.1375  0.2559
## ma1    0.5410 0.2218   2.4397  0.0151
## ma2    0.2232 0.0614   3.6357  0.0003
## sma1  -0.9792 0.0547 -17.9048  0.0000
##
## $AIC
## [1] -1.518799
##
## $AICc
## [1] -1.518628
##
## $BIC
## [1] -1.475798
```

It looks way better than model2. Let's try the last model ARMA(2,2).

```
sarima(logx, 2,1,2, 0,1,1, 12) #model4
```




```
## $fit
```

```
##
## Call:
## arima(x = xdata, order = c(p, d, q), seasonal = list(order = c(P, D, Q), period = S),
##     include.mean = !no.constant, transform.pars = trans, fixed = fixed, optim.control = list(trace =
##         REPORT = 1, reltol = tol))
##
## Coefficients:
##          ar1      ar2     ma1     ma2     sma1
##       0.0640  -0.2637  0.2148  0.3815  -0.9737
## s.e.  0.3658   0.2083  0.3625  0.1350   0.0448
##
## sigma^2 estimated as 0.01168:  log likelihood = 375.4,  aic = -738.81
##
## $degrees_of_freedom
## [1] 482
##
## $ttable
##      Estimate     SE  t.value p.value
## ar1    0.0640 0.3658   0.1749  0.8612
## ar2   -0.2637 0.2083  -1.2655  0.2063
## ma1    0.2148 0.3625   0.5926  0.5537
## ma2    0.3815 0.1350   2.8258  0.0049
## sma1  -0.9737 0.0448 -21.7242  0.0000
##
## $AIC
## [1] -1.517055
##
## $AICc
## [1] -1.516799
##
## $BIC
## [1] -1.465454
```

It looks perfect. Let's compare model 3 and 4.

9. Models Comparison

```
##   Test    model3    model4
## 1  AIC -1.518799 -1.517055
## 2 AICc -1.518628 -1.516799
## 3  BIC -1.475798 -1.465454
```

All AIC, AICc and BIC of model 3 are smaller than model 4's. Our winner is model 3. Let's see what auto.sarima suggests!

```
auto.arima(logx)
```

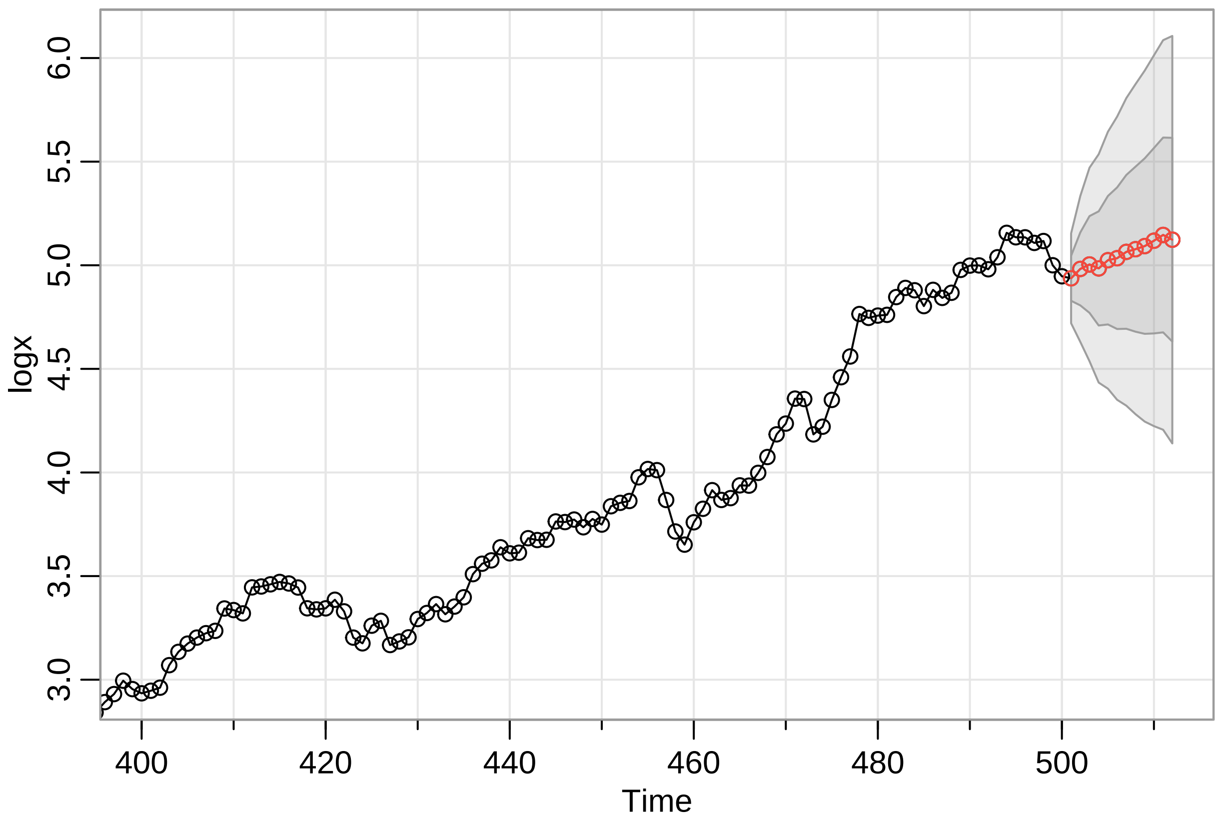
```
## Series: logx
## ARIMA(2,1,2) with drift
##
## Coefficients:
##          ar1      ar2     ma1     ma2   drift
##       0.0148  -0.2423  0.2446  0.3577  0.0140
## s.e.  0.3489   0.2234  0.3461  0.1613  0.0063
##
```

```
## sigma^2 = 0.01183:  log likelihood = 401.5
## AIC=-790.99   AICc=-790.82   BIC=-765.72
```

Even though Auto.arima suggests ARIMA(2,1,2) which is our model 4. We decided to stick with model 3 because its AIC, AICc and BIC are better than model 4's.

10. Forcasting

```
#predicting the next 12 months of Apple stock price
pred_mod = sarima.for(logx,12,2,1,2,0,1,1,12)
```




```
## Time Series:
## Start = 501
## End = 512
## Frequency = 1
##     Month Predicted Price Possible High Possible Low
## 501     J          139.36        141.54       137.17
## 502     F          145.82        148.15       143.48
## 503     M          148.98        151.45        146.5
## 504     A          146.17        148.75       143.59
## 505     M          152.08        154.75        149.4
## 506     J          153.54         156.3       150.79
## 507     J          158.32        161.16       155.48
## 508     A           160.4        163.32       157.48
## 509     S           162.8         165.8       159.81
## 510     O          167.06        170.13          164
## 511     N          171.79        174.93       168.66
## 512     D          167.91        171.11       164.71
```

11. Conclusion

From the prediction tables above it appears that November appears to be the best possible time to sell. It predicted value is highest of all months, even when considering the +- effects of the standard error. This could be due to the fact that its the start of the holiday season, when sales are probably at the all time highest for this particular company. The effects of being on this side of Christmas probably brings in a lot of revenue for the company during this time period.

If buying was the plan, it appears this January is the best time to get an opportunity at a low price. Its potential low price is smaller than any value predicted. On the flip side of the best time for selling, being on the other side of Christmas probably have a negative effect on sales, which can affect the stock of a company for the short term.

There appears to be an unexplained anomaly during the month of April. Our guess is that this is because Feburary sees a spike due to tax refunds, in which people might decide to spend a refund on an apple product, and then the regular steady increases until holiday season, where people again start to spend money.

12. References

Hayes, Adam. "What Is a Time Series and How Is It Used to Analyze Data?" Investopedia, Investopedia, 8 Oct. 2022, https://www.investopedia.com/terms/t/timeseries.asp#:~:text=In%20investing%2C%20a%20time%20series,points%

Shumway, Robert H., and David S. Stoffer. Time Series Analysis and Its Applications: With R Examples. Springer International Publishing, 2017.

emrearslan123. "Apple Inc.. Stock Price Analysis and Forecasting." Kaggle, Kaggle, 7 Sept. 2022, https://www.kaggle.com/code/emrearslan123/apple-inc-stock-price-analysis-and-forecasting?scriptVersionId=105163729.

Aurellia, Christine. "Time Series and Stock Analysis." RPubs, https://www.rpubs.com/AurelliaChristie/time-series-and-stock-analysis.