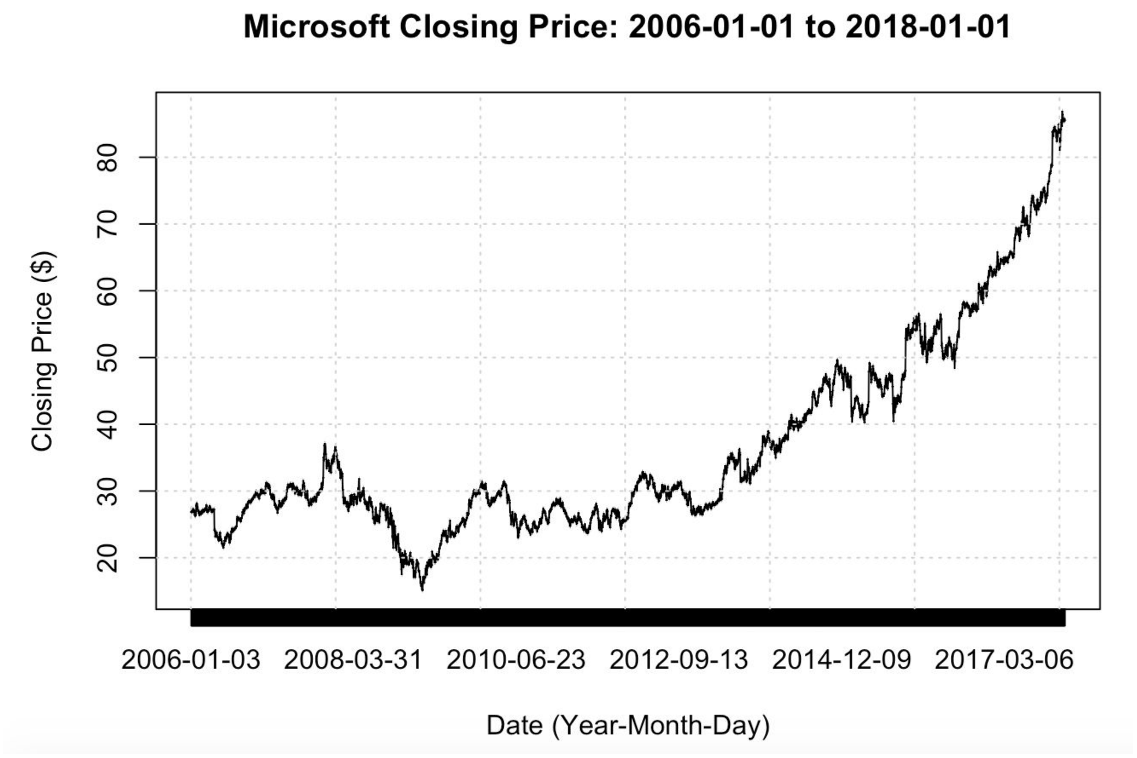
Predicting Microsoft Stock with ARIMA Models

University of Victoria
STATS457: Time Series Analysis
Department of Mathematics and Statistics
Eli Krag - V00819859
Pan Charoensuk - V00827791
Fei Di - V00835445

Table of Contents

Introduction to the Time Series Data

The Microsoft stock data set used in this study is from Kaggle, and showed the daily historical prices from January 1st, 2006 to January 1st, 2018. In order to analyze this data as a time series, we'll look at the Date and Closing price in (USD), as closing price is thought to be a better indicator of the trading days stock price than the open, high, or low prices. The plot below shows a visualization of the data.



We can observe a positive trend upwards after around 2011, as well as a strong dip around 2008 where the financial crisis occured. Right away, this is an indication that the series is not stationary, meaning we will need to look into possible transformations.

The Research Problem

Can we accurately predict the following 10 trading days closing prices ($) with an ARIMA model?

Our Approach:

- Using the historical Microsoft closing prices from January 1st, 2006 to January 1st, 2018 attempt to fit an ARIMA model and forecast the next 10 days
- Using the **actual** next 10 days of trading closing prices (downloaded from yahoo finance) we'll assess the accuracy of our predictions
- Accuracy metric: **Mean Squared Error (MSE) =** $\sum$(predicted - actual)^2

Fitting and ARIMA Model

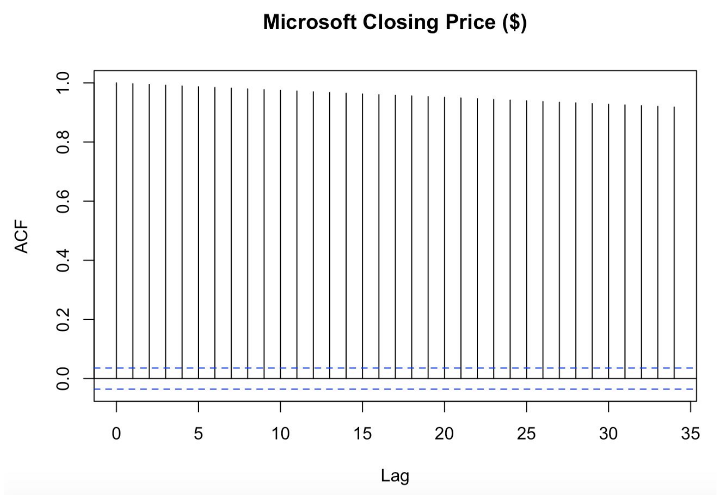
1. Time Series is clearly not stationary (overall strong positive upward trend)
    a. Explore possible transformations to achieve stationarity
2. With transformed series, visualize Autocorrelation and Partial Autocorrelation functions to identify 'p' and 'q' parameters for ARIMA(p,d,q) model

    a. Run model diagnostics to ensure model adequacy

3. With best model, forecast next 10 trading days

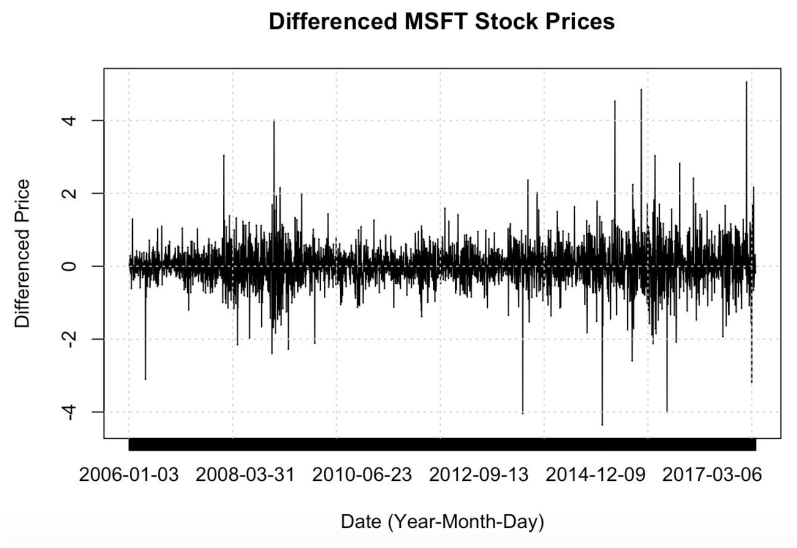
Visualizing the ACF (Figure 2), we can see if we need any transformations or differencing.

Figure 2



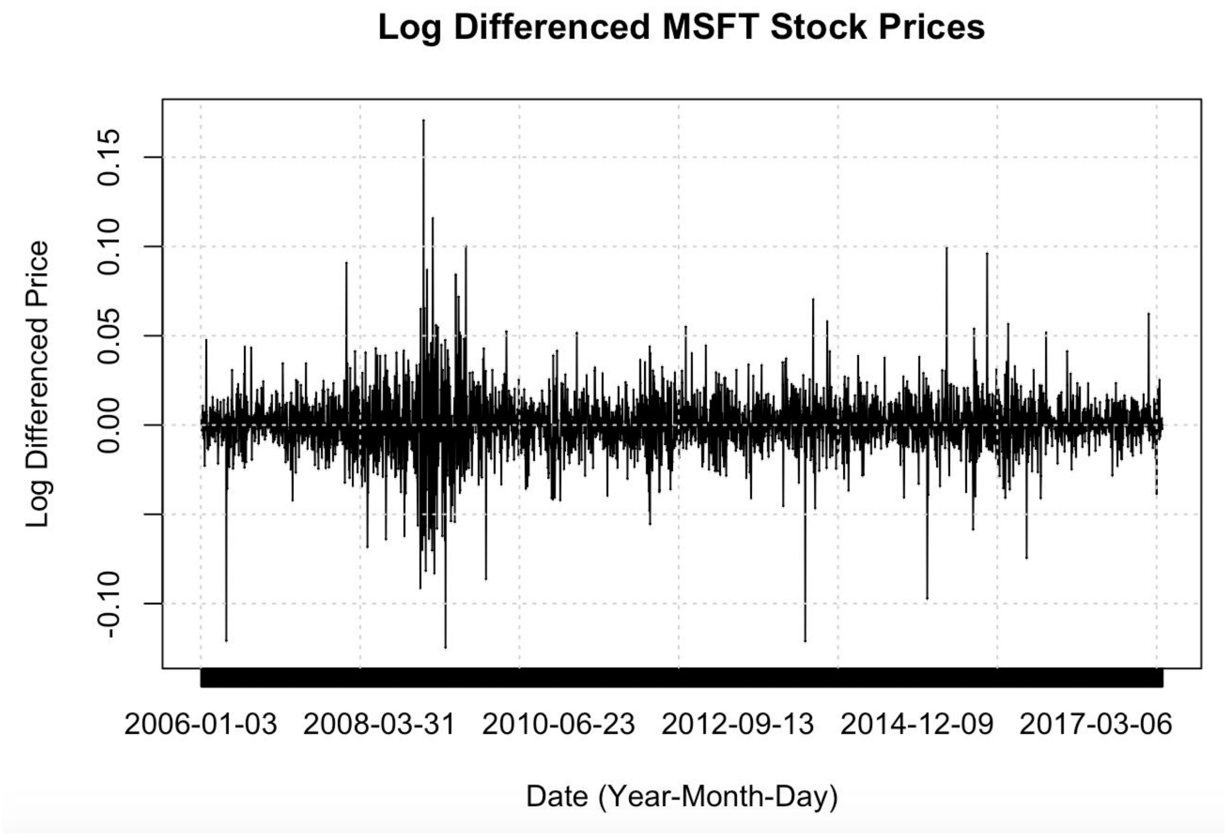
The Autocorrelation function has an extremely slow decay to zero indicating the need for a difference of the series.

First Transformation (Lag 1 Difference)



We can see the series is beginning to look more stationary, as we've removed the strong trend. However it appears we still have non-constant variance, since the tail end of the differenced series appear to change. In an attempt to remedy this, we'll take the log of the series before applying the lag 1 difference.

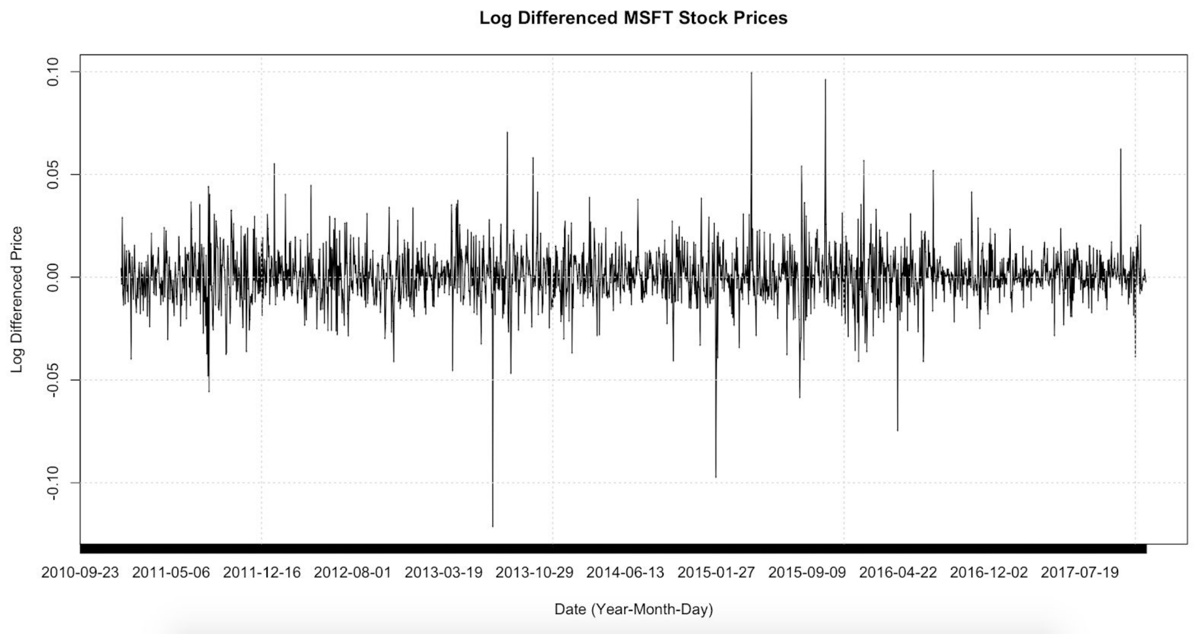
Second Transformation (Lag 1 Difference of the Log Series)



As we can see our variance looks fairly constant with the exception of the period around 2008, namely the 2008 financial crisis. This is an indication that the series is still not stationary, so we'll need to explore an alternative approach.

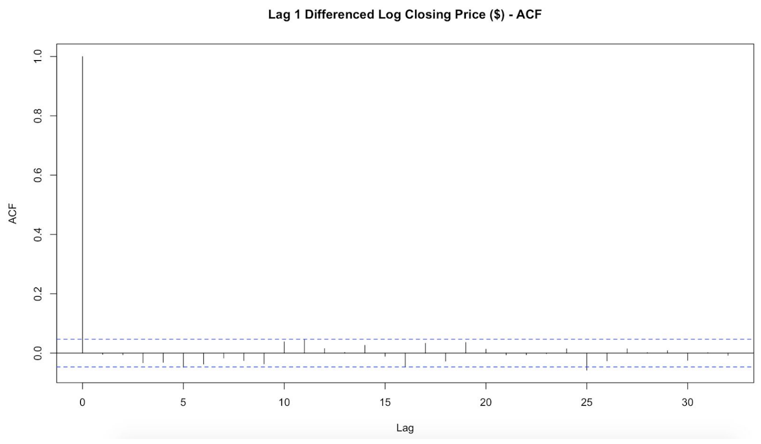
Adapting the Approach

Looking at the previous plot we can see that our problem seems to lie in the period around 2008, namely the financial crisis. Due to the studies time limitations, the data was cut off before January 1st, 2011. Given more time, a method that can account for the anomaly (financial crisis) should be explored, as ~42% of the data is being lost, and similar anomalies are likely to occur in the future.

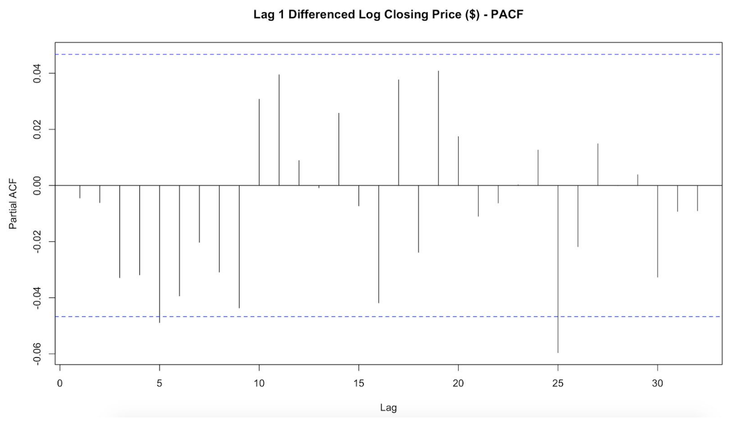


As we can see, the plot looks far more stationary now that we've cut off the financial crisis anomaly. Apart from a few small spikes there is relatively constant variance and no distinct trend.

Determining Parameter Values for ARIMA(p,d,q)



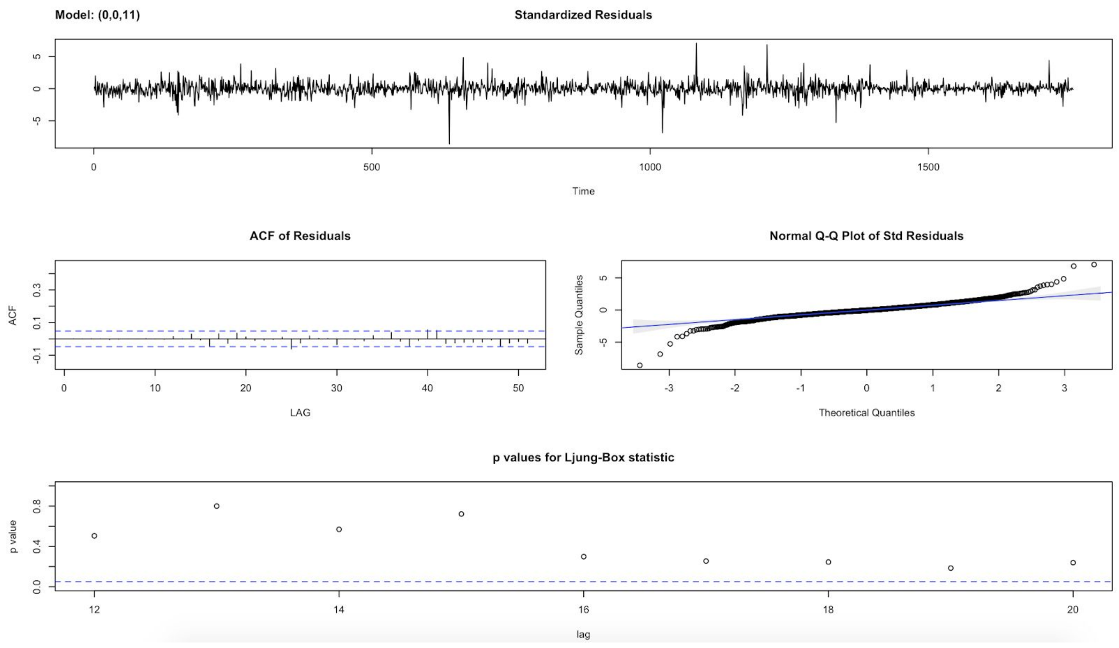
We can see the autocorrelation function instantly drops of to ~0 indicating the stationary nature of the series. Further, this indicates the lack of an autoregressive term in the ARIMA model, leaving us with a ARIMA(0,d,q) model.



The partial autocorrelation function appears to be zero with only one spike at lag 25, indicating that our series is stationary. The next step will be to find an appropriate ‘q’ term for the moving average aspect of our ARIMA model.

The Best Model

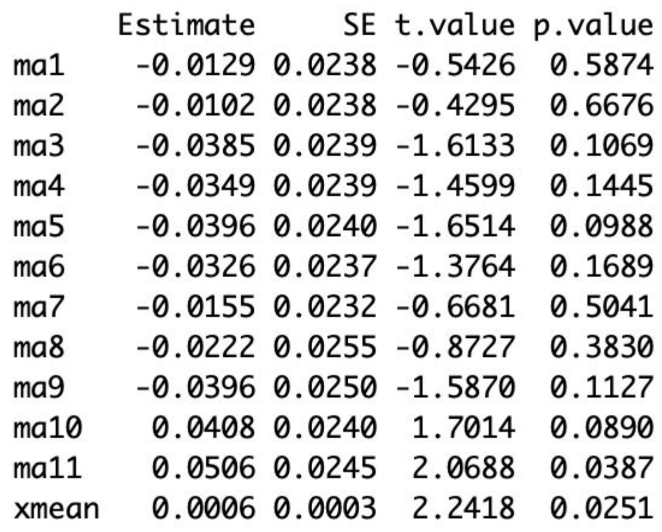
After trying many parameter combinations the **ARIMA(0,0,11)** appears to be the best.



- **Standardized Residuals:** resembles white noise well, no obvious pattern
- **ACF:** no strong correlations, values within dashed lines
- **Normal QQPlot:** residuals diverge from the theoretical quantiles near tails, but appear adequate enough
- **Ljung-Box P-Values:** very good, all the lags have high p-values

Model	**AIC**
ARIMA (0,0,5)	AIC=-7.523237
ARIMA (0,0,11)	AIC=-7.523187
ARIMA (0,0,25)	AIC=-7.518249

The MA(11) model was chosen over the MA(5) model for two main reasons. The first was that the AIC values are virtually identical. As AIC places a penalty on the number of parameters in a model, it was deduced that this was the reason the value for the MA(5) model being so close to the MA(11) model. The second reason was that the MA(11) model, as we will see in the accuracy assessment, produced a significantly lower MSE than the MA(5) model. As the performance was better and then AIC was virtually the same as the MA(5) model, we elected to go with the more complicated MA(11) model:



```
      Estimate     SE t.value p.value
ma1    -0.0129 0.0238 -0.5426  0.5874
ma2    -0.0102 0.0238 -0.4295  0.6676
ma3    -0.0385 0.0239 -1.6133  0.1069
ma4    -0.0349 0.0239 -1.4599  0.1445
ma5    -0.0396 0.0240 -1.6514  0.0988
ma6    -0.0326 0.0237 -1.3764  0.1689
ma7    -0.0155 0.0232 -0.6681  0.5041
ma8    -0.0222 0.0255 -0.8727  0.3830
ma9    -0.0396 0.0250 -1.5870  0.1127
ma10    0.0408 0.0240  1.7014  0.0890
ma11    0.0506 0.0245  2.0688  0.0387
xmean   0.0006 0.0003  2.2418  0.0251
```

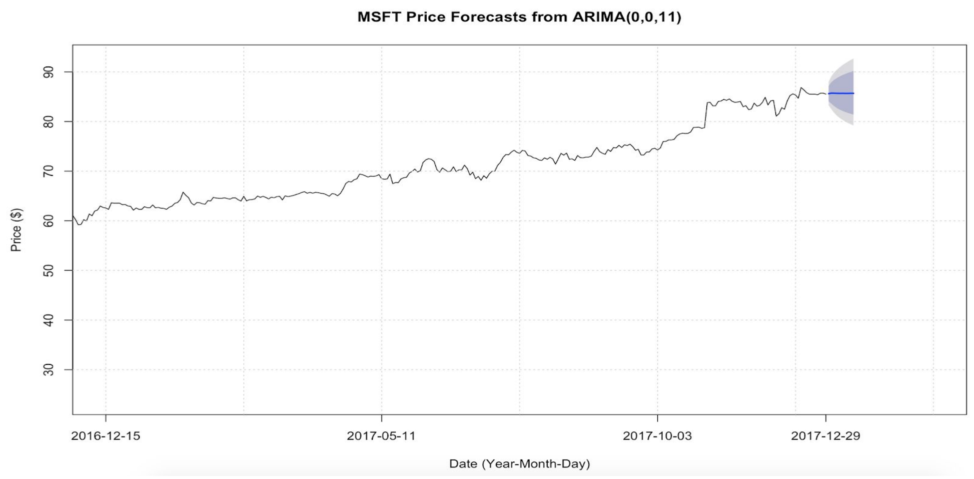
$$x_t = -0.0129w_{t-1} - 0.0102w_{t-2} - 0.0385w_{t-3} - 0.0349w_{t-4} - 0.0396w_{t-5} - 0.0326w_{t-6} - 0.0155w_{t-7}$$

$$- 0.222w_{t-8} - 0.0396w_{t-9} + 0.0408w_{t-10} + 0.0506w_{t-11} + w_t$$

Note that all of the terms are statistically insignificant, except for the 11th with p-value = 0.0387. This is an indication to explore removing terms in the model or to fit a seasonal model with 11 as the seasonal term. Due to time limitations of the study, the MA(11) model is adequate.

Forecasting and Assessing Accuracy

10 day forecast:



Comparing with the actual values produces **MSE = 54.67592 (**MSE = 0 is a perfect fit). The ARIMA(0,0,5) model produced MSE = 61.11532. We can see the comparison between the predicted ($USD) and actual stock prices ($USD) in the table below.

Date	ActualPrice	PredictedPrice
2018-01-02	85.95	85.61388
2018-01-03	86.35	85.73415
2018-01-04	87.11	85.72160
2018-01-05	88.19	85.68737
2018-01-08	88.28	85.67651
2018-01-09	88.22	85.68726
2018-01-10	87.82	85.67271
2018-01-11	88.08	85.67306
2018-01-12	89.60	85.68906
2018-01-16	88.35	85.68538

Conclusions

The only accurate predictions were from the first 2 trading days. Stock prices are notoriously difficult to predict. We should invest more time into building a model before considering its practical use for trading decisions. The financial crisis anomaly should be included in the model as it is likely that similar events will occur in the future and therefore should be incorporated into predictions. Possible this could be achieved using a Garch model. Further, we could explore the effectiveness of a seasonal model, as it’s possible there is a quarterly seasonal trend, because businesses release financial reports and other information from quarter to quarter, which may have a significant effect on price trends. The ARIMA model analyzes the mean, rather than the variation of the stock price. This results in a convergence after only a few days of forecasting. The only useful conclusions we can draw are on the confidence intervals around the predictions, i.e. we can say we are 95% certain that the price will not rise or fall below the interval limits on a given day. While there is some value in this, a better approach could be to model the variation, in order to receive better indicators of the stock rising up or sinking down in price. This could theoretically feed investment decisions better than inference on the mean can. Lastly, the convergence in the forecast is likely due to the fact that the model (like most financial models) becomes ressemblent of white noise.

References

Kaggle [Web page].(2018). Retrieved from
https://www.kaggle.com/szrlee/stock-time-series-20050101-to-20171231

Robert, H.S., & David, S.S.(2016). *Time Series Analysis and its Application: With R examples* (4th ed., pp. 137–147). New York, NY: Springer.

Stack Exchange [Web page].(2018). Retrieved from
https://stats.stackexchange.com/questions/222765/can-we-use-box-ljung-as-a-stationarity-test-for-time-series/222776

Appendix

Code

```r
setwd("~/Desktop/STAT457/Project")

msft = read.csv("stock-time-series/MSFT_2006-01-01_to_2018-01-01.csv")
msft = msft[ ,-c(2,3,4,6,7)]

visualizeStockData = function(date, close, title) {
  plot(date, close,
       xlab="Date (Year-Month-Day)",
       ylab="Closing Price ($)",
       main=title)
  lines(date, close); grid()
}

visualizeStockData(msft$Date, msft$Close, "Microsoft Closing Price: 2006-01-01 to 2018-01-01")
```

```r
#Determine ARIMA(p,d,q) model
time = msft$Date
price = msft$Close

acf(price, main="Microsoft Closing Price ($)")
```

```r
#Very slow ACF decay. Indicates we should attempt a difference.
price.diff = diff(price)
time.diff = time[-length(time)]
plot(time.diff, price.diff,
     xlab="Date (Year-Month-Day)", ylab="Differenced Price",
     main="Differenced MSFT Stock Prices")
lines(time.diff, price.diff); grid()
```

```r
#Trend removed, still have non-constant variance... take the log.
price.diffLog = diff(log(price))
plot(time.diff, price.diffLog,
     xlab="Date (Year-Month-Day)", ylab="Log Differenced Price",
     main="Log Differenced MSFT Stock Prices")
lines(time.diff, price.diffLog); grid()
```

```r
#Check for stationarity
acf(price.diffLog)
```

```r
pacf(price.diffLog)
```

```
#remove financial crisis
msft_sub = msft[c(1259:length(msft[,1])), ]
time2 = msft_sub$Date
price2 = msft_sub$Close
price.diffLog2 = diff(log(price2))
time.diff2 = time2[-length(time2)]
plot(time.diff2, price.diffLog2,
     xlab="Date (Year-Month-Day)", ylab="Log Differenced Price",
     main="Log Differenced MSFT Stock Prices",
     xlim=c(1259,length(msft[,1])))
lines(time.diff2, price.diffLog2); grid()
```

```
acf(price.diffLog2, main="Lag 1 Differenced Log Closing Price ($) - ACF")
```

```
pacf(price.diffLog2, main="Lag 1 Differenced Log Closing Price ($) - PACF")
```

```
#ACF no indication of AR term, p=0
#PACF decrease to zero slowly, q=11,12,13,14?
library(astsa)

model = sarima(price.diffLog2, 0, 0, 11)
  #standarized resids look good overall... resembles whitenoise closely enough
  #ACF of residuals good... no strong correlations
  #normal qq plot good enough...
  #p-values very good... majority above line

#model = sarima(price.diffLog2, 0, 0, 5)
#model = sarima(price.diffLog2, 0, 0, 25)

library(forecast)
```

```
#fit best model
#fit = Arima(price, order=c(0,1,11), lambda=0) #Arima() transforms diff(log) back
#fit = Arima(price2, order=c(0,1,25), lambda=0)
fit = Arima(price2, order=c(0,1,11), lambda=0)
fcast = forecast(fit, h=10) #predictions
fcast

#full plot with predictions
plot(fcast, xlab="Date (Year-Month-Day)", ylab="Price ($)",
     main="MSFT Price Forecasts from ARIMA(0,0,11)", xaxt="n")
axis(1, at=c(0, 500, 1000, 1500), labels=c("2011-01-03", "2012-12-27",
                                           "2014-12-22", "2016-12-15"))
grid()
```

```
#zoomed in plot with predictions
plot(fcast, xlab="Date (Year-Month-Day)", ylab="Price ($)",
     main="MSFT Price Forecasts from ARIMA(0,0,11)",
     xaxt="n", xlim=c(1500,1800))
axis(1, at=c(1500, 1600, 1700, 1761), labels=c("2016-12-15", "2017-05-11",
                                               "2017-10-03", "2017-12-29"))
grid()
```

```r
#get actual values from historical MSFT data downloaded from yahoo finance
msft_new = read.csv("MSFT.csv")
msft_new = msft_new[ ,-c(2,3,4,6,7)]
msft_new = msft_new[c(1:10), ]

colnames(msft_new)[2] = "ActualPrice"

preds = as.numeric(unlist(fcast[4]))
msft_new$PredictedPrice = preds

#Mean Squared Error for prediction performance
mse = sum((msft_new$ActualPrice - msft_new$PredictedPrice)^2)
mse #54.67592
```