Time Series Analysis of Starbucks Stock Price

Yahir Baltazar

yahirbaltazar09@gmail.com

University of California Santa Barbara

PSTAT 174 Time Series

March 21, 2025

ABSTRACT

This project applies time series analysis to model the daily stock prices of Starbucks (SBUX) using a Seasonal Autoregressive Integrated Moving Average (SARIMA) model. The goal of the project is to understand the underlying patterns in the stock price data using a SARIMA(1,2,1,1,1,0)[25] model. I selected these specific parameters after carefully considering and analyzing the autocorrelation function and partial autocorrelation function . Checking the diagnostics of the model, by checking the residuals, it showed that the model successfully captured the structure of the data. Additionally, a Unit Root Test (ADF) was applied indicating that the transformed series is in fact stationary. The final SARIMA model provides a good understanding of Starbucks' stock price trends, demonstrating the effectiveness of time series analysis in financial forecasting.

=

Introduction

Over the past year or so, the Boycott, Divestment and Sanctions (BDS) movement has become a major global political and social justice development. The BDS movement aims to influence the actions of companies by applying economic pressure. My project seeks to analyze the financial impact of these targeted companies by looking at the changes in stock price over time.

Starbucks has faced significant calls for boycotts that started on social media after having a legal dispute with the Starbucks Workers United (SWU) union. This dispute began in late October 2023, when the union publicly expressed solidarity with Palestine online. In response, Starbucks filed a lawsuit against the SWU union for speaking on behalf of the company on the basis of trademark infringement. After refusing to speak up about the ongoing genocide in Palestine, the pressure to boycott Starbucks has spread rapidly online.

My intention for this project is to better understand the financial impacts of the BDS movement by analyzing Starbucks' Stock Price over time. Through time series analysis using the SARIMA model, we aim to assess how the boycott and associated events have influenced the company's stock performance.

Data

This project will be using daily stock prices of Starbucks (SBUX) from October 01, 2023 to October 01, 2024. Since we are using daily frequency, this gives us 252 individual observations, since there are 251 days in a year (there is an extra day in total since we are considering October 01, 2024). These stock prices were downloaded directly from Investing.com with the following columns: Date, Price, Open, High, Low, Volume Change Percent. In the reference portion of this report, I have provided a link to the most recent SBUX stock data on Investing.com as well as a link to the tidied data I used for this project. For the purpose of this project, we will only consider the Date and Price category as variables for our time series. The number of observations is suitable for time series analysis, offering enough data for both short-term fluctuations as well as quarterly seasonal trends.

In tidying the data, I transformed the Date column into the appropriate MM/DD/YYYY value and arranged it in ascending order to ensure proper chronological sequence. Similarly, I dropped any missing values in order to prevent any future problems. Lastly I applied a log transformation to the stock price values to stabilize the variance and make the data more suitable for time series modeling.

The Methodology

Our goal with this dataset is to analyze and better understand any underlying patterns and seasonal structure in the SBUX stock price over time. Within the 'astsa' package in R Studio, I utilized SARIMA (Seasonal Autoregressive Integrated Moving Average) to account for both non-seasonality and seasonality in the data. Additionally, we apply a Unit Root Test to assess the stationarity of the data and guide our modeling decisions.

SARIMA Model

The SARIMA model is a tool used to model time series data while accounting for seasonal and non-seasonal patterns. The model is defined as,

SARIMA(p,d,q)×(P,D,Q)s

p - number of autoregressive terms (AR) in the non-seasonal part of the model
d - number of differences required to make the series stationary
q - The number of moving average terms (MA) in the non-seasonal part of the model.
P - The number of seasonal autoregressive terms (SAR).
D - The number of seasonal differences required to make the series stationary (i.e., seasonal differencing).
Q - The number of seasonal moving average terms (SMA).
S - The period of seasonality

Similarly, in order to meet the conditions to fit our SARIMA model, it's important to test whether our data is stationary. A stationary time series has constant mean and variance over time. To test for stationarity, we performed the **Augmented Dickey-Fuller (ADF) test**, which is a common unit root test. Based on the results, we will determine whether the data is suitable for fitting the SARIMA model.

Results

We first plot our daily Starbucks (SBUX) stock price time series. To better visualize and interpret the time series, I used the ggplot function (from the ggplot2 package in R) to plot the data showing each month. Looking at the plot, it's very evident that the data is not stationary whatsoever. There is significant variance, and the data lacks an obvious mean or trend. I applied the first and second differences to make the data stationary and compared the results..

Fig. 1 - First Differenced Starbucks Price

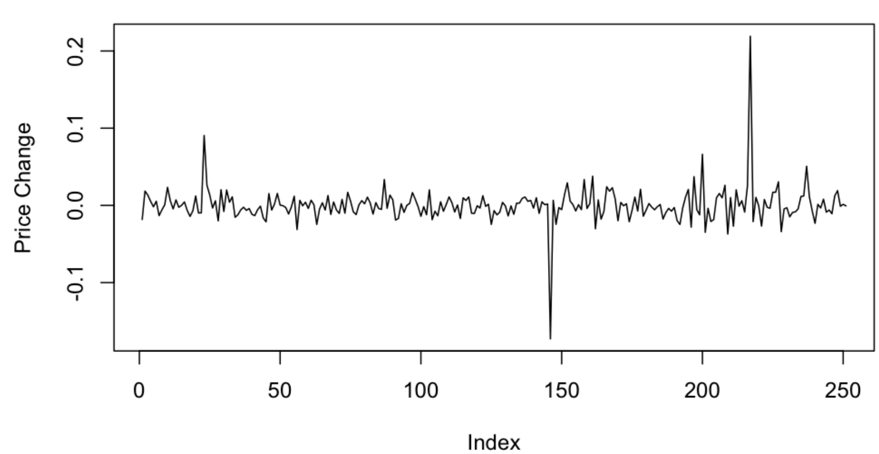
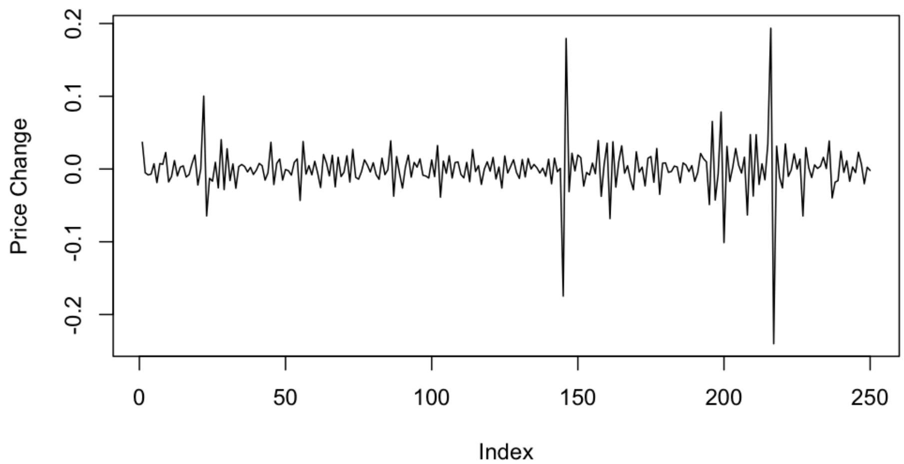


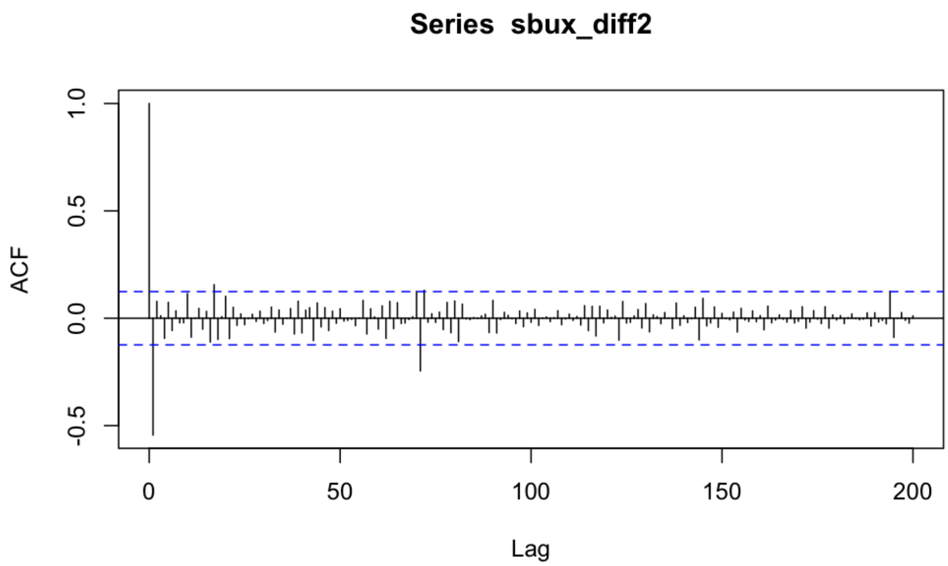
Fig. 2 - Second Differenced Starbucks Price

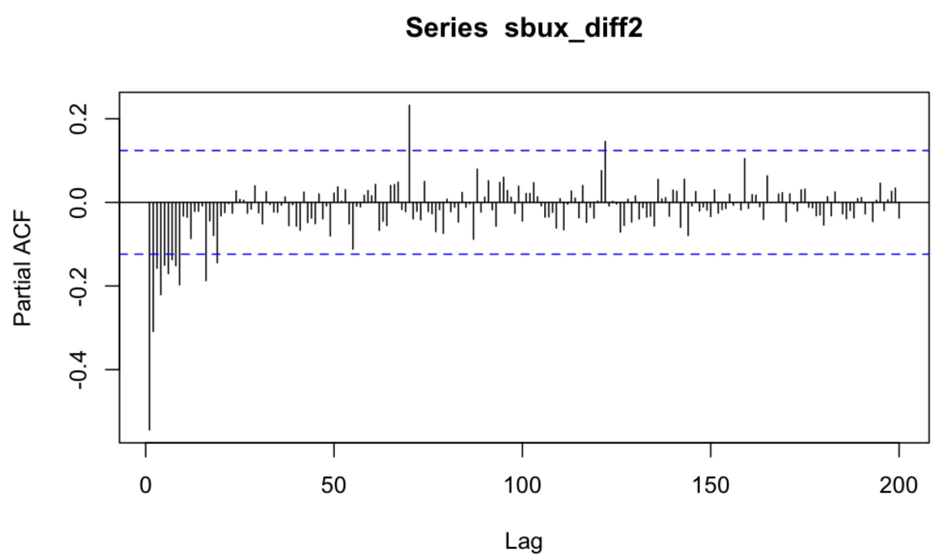


Looking at the first differencing (**Fig. 1**), there is some reduction of the trend, but the data still exhibits significant patterns that suggest it is not yet stationary. Therefore, I applied second differencing (**Fig. 2**), and the result shows a noticeable reduction in the trend, making the data more stationary. While **Fig.2** is not perfect, it is the most stationary I was able to get using differencing techniques.

I went ahead and performed a third differencing, however this plot seemed to only slightly affect the residual patterns without significantly improving stationarity. Likewise, I wanted to avoid over-differencing, as it can lead to a loss of valuable information. Thus, I decided that the second differencing was sufficient for the analysis, and proceeded with modeling using this transformed data.

We'll now look at Autocorrelation Function (**ACF**) and Partial Autocorrelation Function (**PACF**) to identify the appropriate parameters for the SARIMA model.





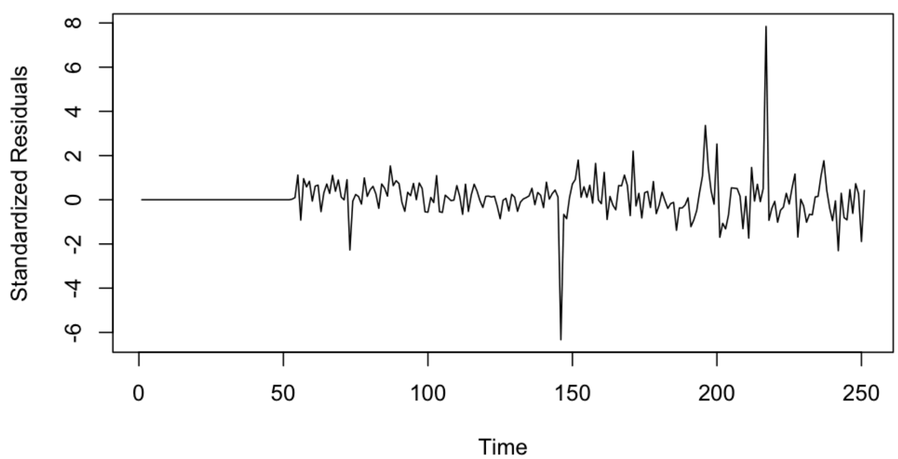
The **ACF** of the differenced data shows a **s**ignificant spike at lag 1 and a **gradual decay** after that. This suggests the presence of a moving average (MA) component in the model with q=1.

The PACF, on the other hand, shows several significant lags, specifically lags 1 through 8 and several outliers around 25, 75, 125. Hence we'll consider p=1 and P=1 for our seasonal component of the model. Since the lags happen in intervals of 25, I used S=25 as our period of analysis. This gives us our model, SARIMA(1,2,1,1,1,0)[25].

Diagnostics

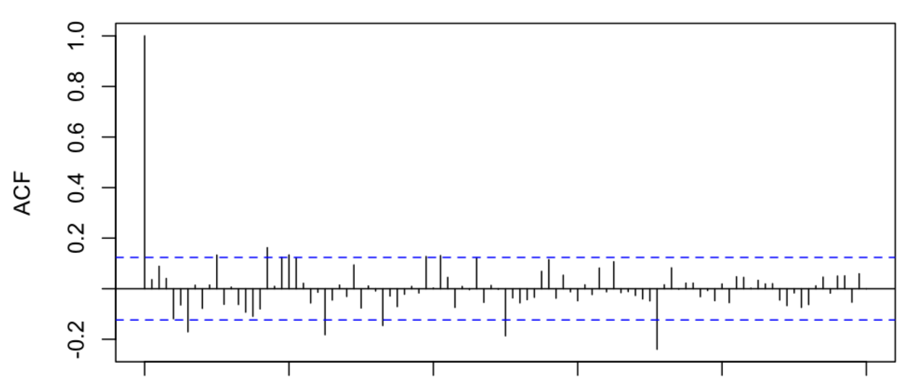
To see how our SARIMA(1,2,1,1,1,0)[25] did we'll consider and analyze the residuals of the model to assess whether they resemble white noise.

Fig. 3 -Standardized Residuals



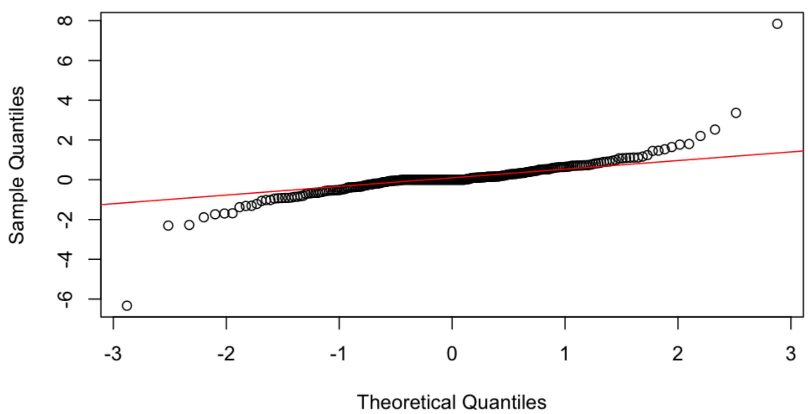
Unfortunately looking at Fig. 3 the standardized residuals don't appear to look like Gaussian White Noise. While the plot does look like white noise in some parts, mostly in the first half, there are significant spikes of variation toward the second half. We can really see this at approximately time 140 and time 225. Overall though, the mean and variance are somewhat constant, suggesting that our model may be somewhat stationary.

Fig. 4 - ACF of Residuals



Looking at the ACF of the residuals (**Fig.4**) we notice that there are a lot of significant spikes. This suggests that there are more underlying autocorrelations that the SARIMA model hasn't accounted for.

Fig. 5 -QQ Plot of Standardized Residuals



Lastly, we'll consider the QQ-Plot of our Standardized residuals. Since the assumption of normality is important for our SARIMA model, it's essential to verify whether the residuals conform to this assumption. Notice that most of the residuals fall on or near the red line. There are only a few outliers on either ends of the model that fall far from the best fit line. Therefore it seems that the residuals approximately follow a normal distribution. This is a good sign that the model has captured the underlying data trends well.

Finally after using the Augmented Dickey-Fuller (ADF) test with our model, we are given a value of -6.7272 with a p value less than 0.01. Our initial null hypothesis was that our data was not stationary and given the p value found from the ADF, we can confidently reject the null. This suggests that the model is in fact stationary.

Hence, our final model SARIMA(1, 2, 1, 1, 1, 0)[25] does a good job of capturing the underlying data patterns, ensuring that the residuals are stationary and uncorrelated. The stationarity of the residuals supports the assumption that the model has effectively accounted for any trends and seasonality.

Conclusion

In this time analysis project, we analyzed the financial impact of the Boycott, Divestment, and Sanctions movement on Starbucks daily stock price throughout an entire year. After finding the appropriate parameter by analyzing the ACF and PACF, we fit our SARIMA model with the appropriate parameters to model our data. Our final model was chosen after carefully considering the stationarity of the data, as well as the seasonal and non-seasonal components that influence Starbucks' stock price.

Looking at the ACF, PACF, and QQ plot of our standardized residuals we see that the model successfully captured the underlying trends, seasonality, and any potential autocorrelations in the data. While there were some discrepancies within the data, including a handful of outliers, we used the Augmented Dickey-Fuller (ADF) test to verify the stationarity of our model.

The final SARIMA model effectively captures the impact of external events, such as the BDS boycott, on Starbucks' stock price. This model provides valuable insights on the relationship between social and political movements and financial market behavior, offering a framework for future research on similar topics.

On a more inspirational note, this clearly demonstrates that we the people have power. It is very easy to become disheartened and feel hopeless in our modern society, as many times we are made to feel powerless. However, as shown through the BDS movement and its lasting impact on Starbucks' stock price, collective action and social movements can create change. In a world that often seems to prioritize profit over people, this study shows that social and political activism can have real, measurable consequences.

References

"Introduction to Time Series and Forecasting" by P.J. Brockwell and R. A. Davis

https://ggplot2.tidyverse.org/

https://www.bdsmovement.net/

https://www.investing.com/equities/starbucks-corp-historical-data

(updated data)

https://drive.google.com/file/d/1nKMBZQkwzUB5T8VoB2FOD-U92Cy8jQJ3/view?usp=sharing

https://stats.stackexchange.com/questions/84330/errors-in-optim-when-fitting-arima-model-in-r

174 Final Project

Yahir B

2025-03-19

174 Final Project R Code

```r
library(tidymodels)
```

```
## -- Attaching packages -------------------------------------- tidymodels 1.2.0 --

## v broom        1.0.6     v recipes      1.1.0
## v dials        1.3.0     v rsample      1.2.1
## v dplyr        1.1.4     v tibble       3.2.1
## v ggplot2      3.5.1     v tidyr        1.3.1
## v infer        1.0.7     v tune         1.2.1
## v modeldata    1.4.0     v workflows    1.1.4
## v parsnip      1.2.1     v workflowsets 1.1.0
## v purrr        1.0.2     v yardstick    1.3.1

## -- Conflicts ----------------------------------------- tidymodels_conflicts() --
## x purrr::discard() masks scales::discard()
## x dplyr::filter()  masks stats::filter()
## x dplyr::lag()     masks stats::lag()
## x recipes::step()  masks stats::step()
## * Search for functions across packages at https://www.tidymodels.org/find/
```

```r
library(tidyverse)
```

```
## -- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
## v forcats   1.0.0     v readr     2.1.5
## v lubridate 1.9.3     v stringr   1.5.1

## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x readr::col_factor() masks scales::col_factor()
## x purrr::discard()    masks scales::discard()
## x dplyr::filter()     masks stats::filter()
## x stringr::fixed()    masks recipes::fixed()
## x dplyr::lag()        masks stats::lag()
## x readr::spec()       masks yardstick::spec()
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become error
```

```r
library(ggplot2)
library(astsa)
```

```
##
## Attaching package: 'astsa'
##
## The following object is masked from 'package:parsnip':
##
##     bart
```

```r
library(tseries)
```

```
## Registered S3 method overwritten by 'quantmod':
##   method            from
##   as.zoo.data.frame zoo
```

```r
library(forecast)
```

```
##
## Attaching package: 'forecast'
##
## The following object is masked from 'package:astsa':
##
##     gas
##
## The following object is masked from 'package:yardstick':
##
##     accuracy
```

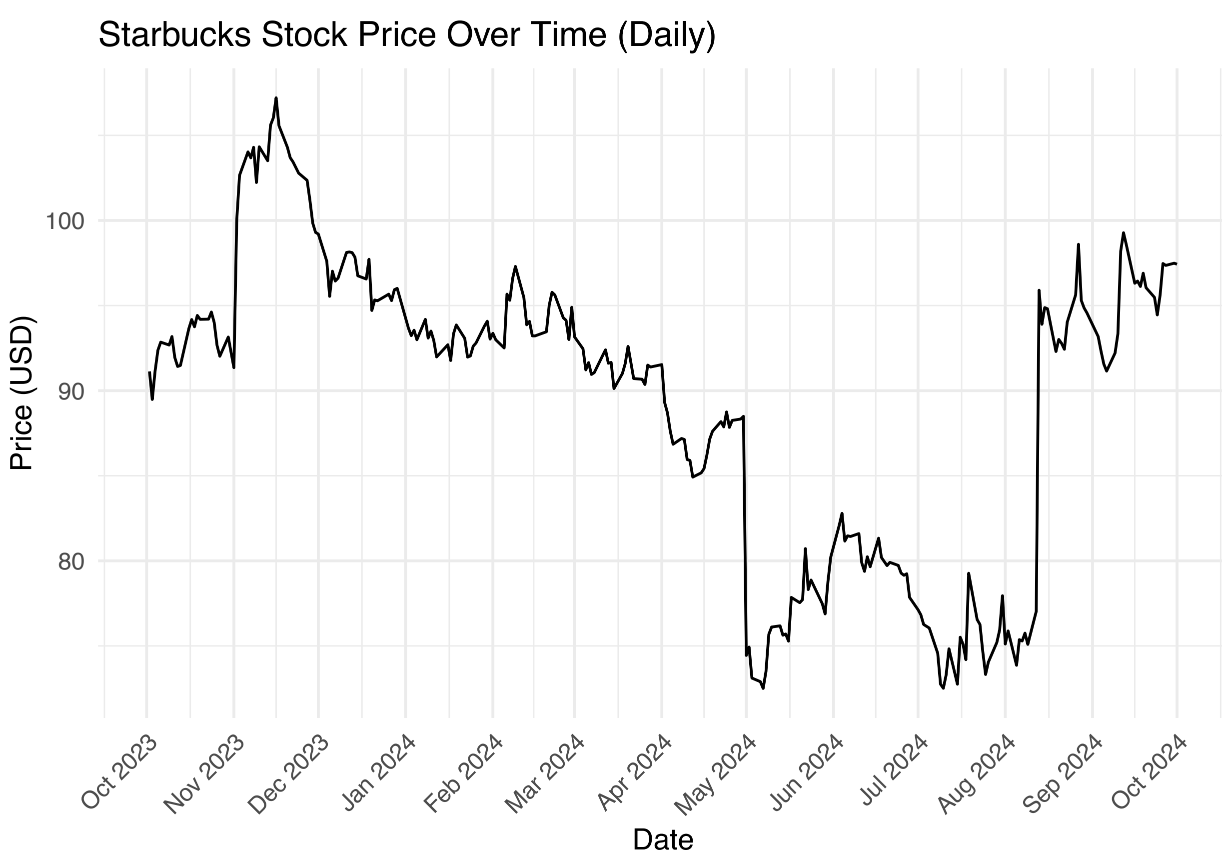
```r
sbux_raw <- read_csv('/Users/yahir/Starbucks_History(3) copy.csv')
```

```
## Rows: 252 Columns: 7
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr (3): Date, Vol., Change %
## dbl (4): Price, Open, High, Low
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```r
# Removing rows with NA values
sbux <- sbux_raw %>% drop_na()

#formating Date to MM/DD/YYY
sbux$Date <- as.Date(sbux$Date, format = "%m/%d/%y")

#Arranging Date from oldest to newest
sbux <- sbux %>% arrange(Date)
sbux
```

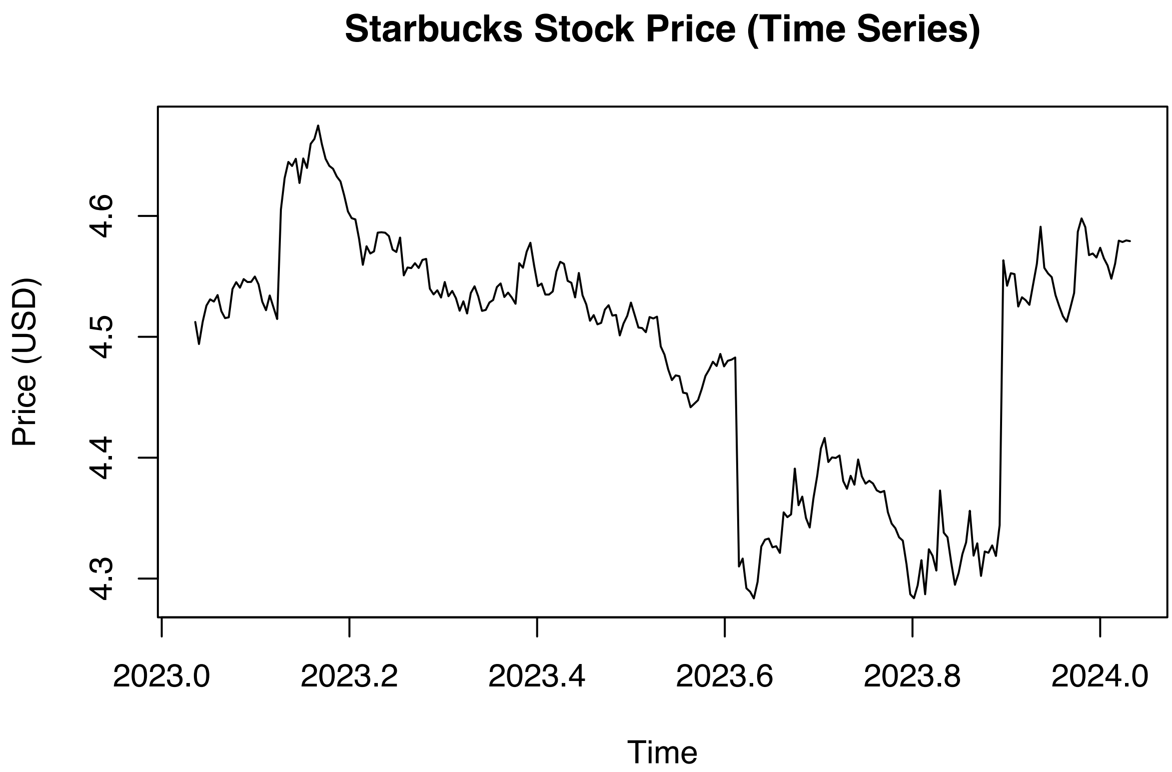
```
## # A tibble: 252 x 7
##    Date       Price  Open  High   Low Vol.  `Change %`
##    <date>     <dbl> <dbl> <dbl> <dbl> <chr> <chr>
##  1 2023-10-02  91.1  91.0  91.7  90.8 6.43M -0.15%
##  2 2023-10-03  89.5  90.5  91.1  89.2 6.67M -1.81%
##  3 2023-10-04  91.2  89.9  91.4  89.8 6.17M 1.87%
##  4 2023-10-05  92.4  91.1  93.0  91.1 7.55M 1.34%
##  5 2023-10-06  92.8  92    93.3  91.7 6.98M 0.52%
##  6 2023-10-09  92.7  92.4  92.8  91.4 4.12M -0.18%
##  7 2023-10-10  93.2  93.0  94.0  92.9 4.97M 0.54%
##  8 2023-10-11  92.0  93.3  93.6  91.8 4.50M -1.32%
##  9 2023-10-12  91.4  91.8  92.2  90.8 5.49M -0.58%
## 10 2023-10-13  91.5  91.3  92.1  91.0 4.78M 0.07%
## # i 242 more rows
```

```r
sbux_log <- log(sbux$Price) ###

#using ggplot for better interpretation using Months
ggplot(sbux, aes(x = Date, y = Price)) +
  geom_line() +
  ggtitle("Starbucks Stock Price Over Time (Daily)") +
  theme_minimal() +
  xlab("Date") +
  ylab("Price (USD)") +
  scale_x_date(labels = scales::date_format("%b %Y"), breaks = "1 month") +
  theme(axis.text.x = element_text(angle = 45, hjust = 1))
```

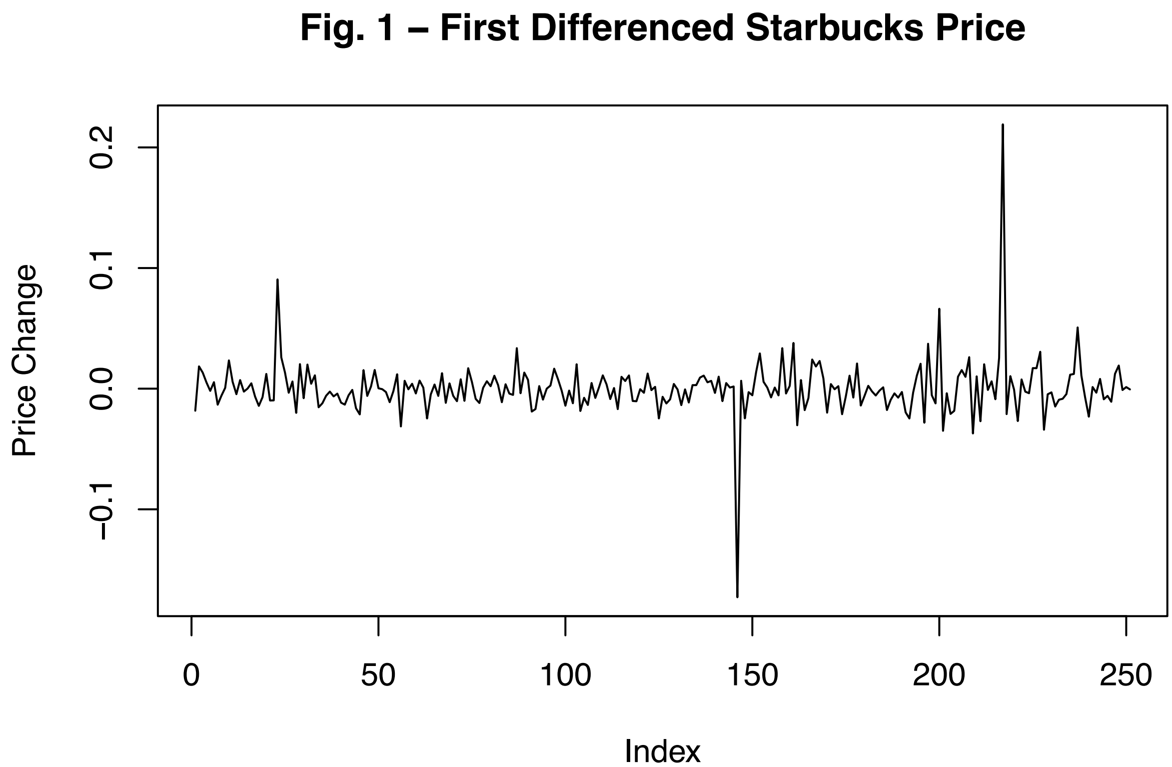


Here we created a time series object as we will be using this for future processing.

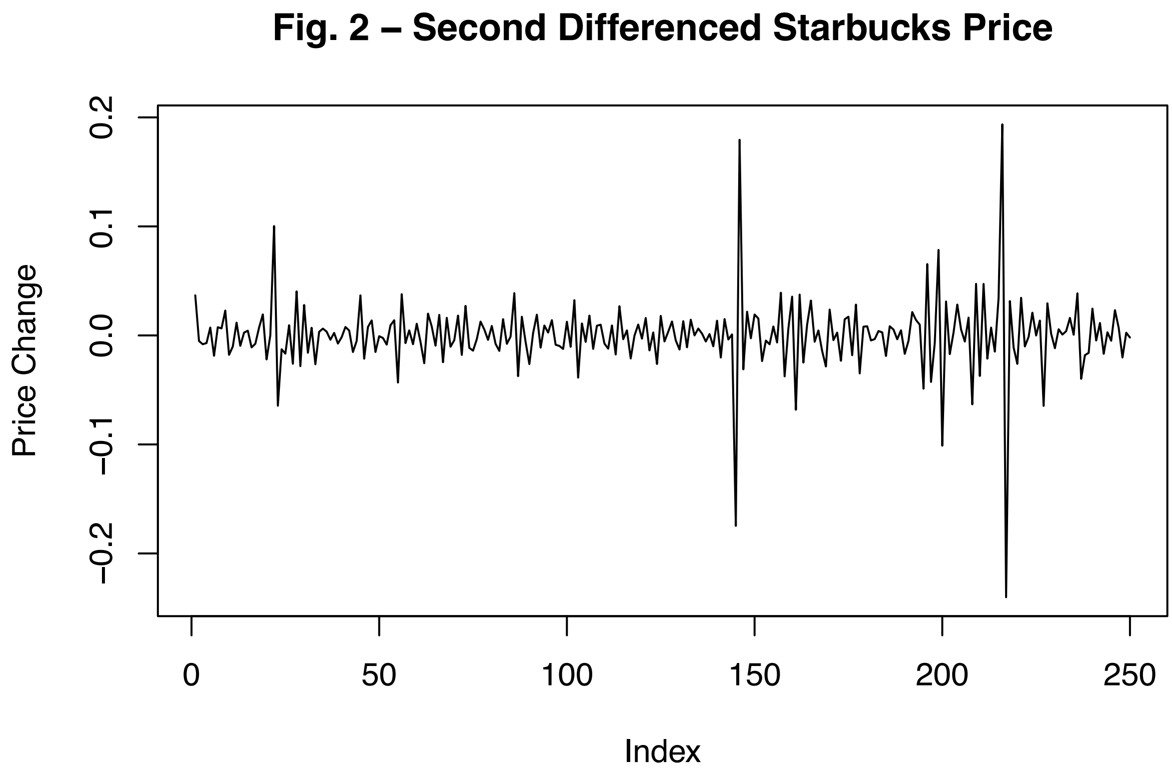
```
sbux_ts <- ts(sbux_log, start = c(year(min(sbux$Date)), month(min(sbux$Date))), frequency = 252)

plot(sbux_ts, main = "Starbucks Stock Price (Time Series)", ylab = "Price (USD)", xlab = "Time", type =
```




```
sbux_diff1 <- diff(sbux_log)
sbux_diff2 <- diff(sbux_diff1)


plot(sbux_diff1, type = "l", main = "Fig. 1 - First Differenced Starbucks Price", ylab = "Price Change")
```



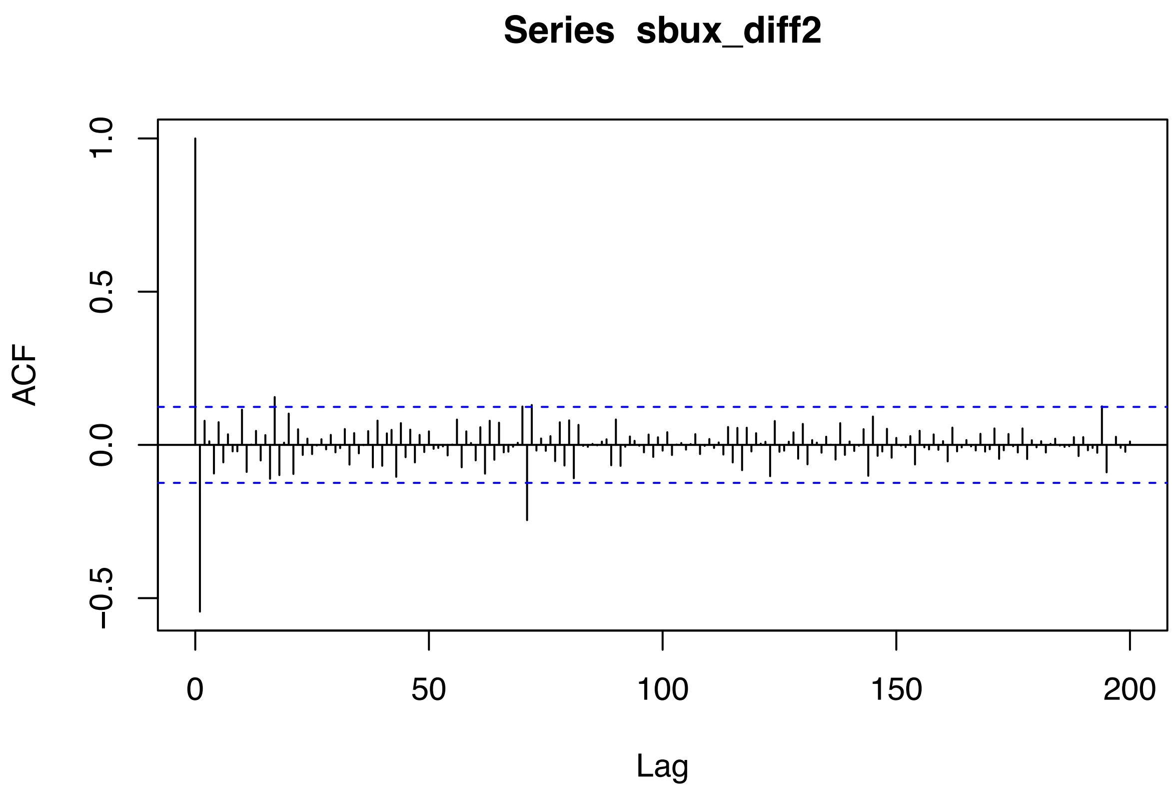

```
plot(sbux_diff2, type = "l", main = "Fig. 2 - Second Differenced Starbucks Price", ylab = "Price Change
```



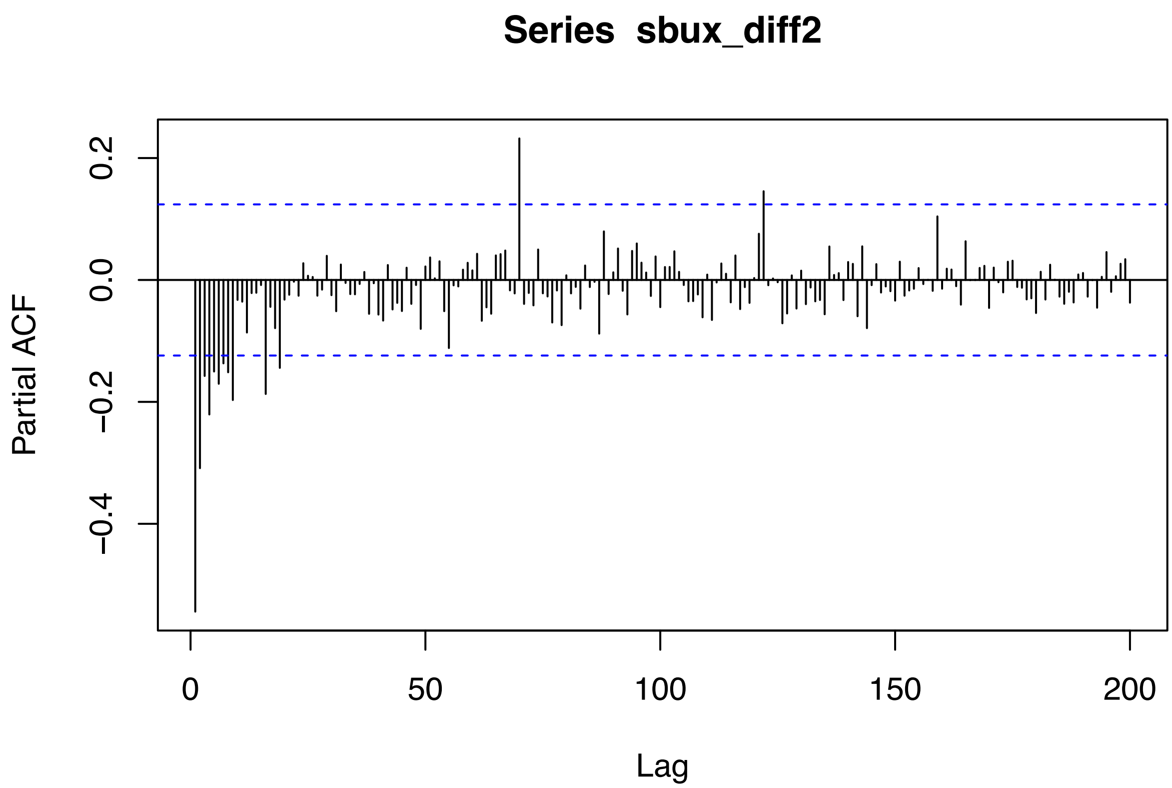
Looks a little better now that we have d=1.

Now lets find values for p and q by looking at the ACF and PACF of our newly differenced data.

```
acf(sbux_diff2, lag.max = 200)
```

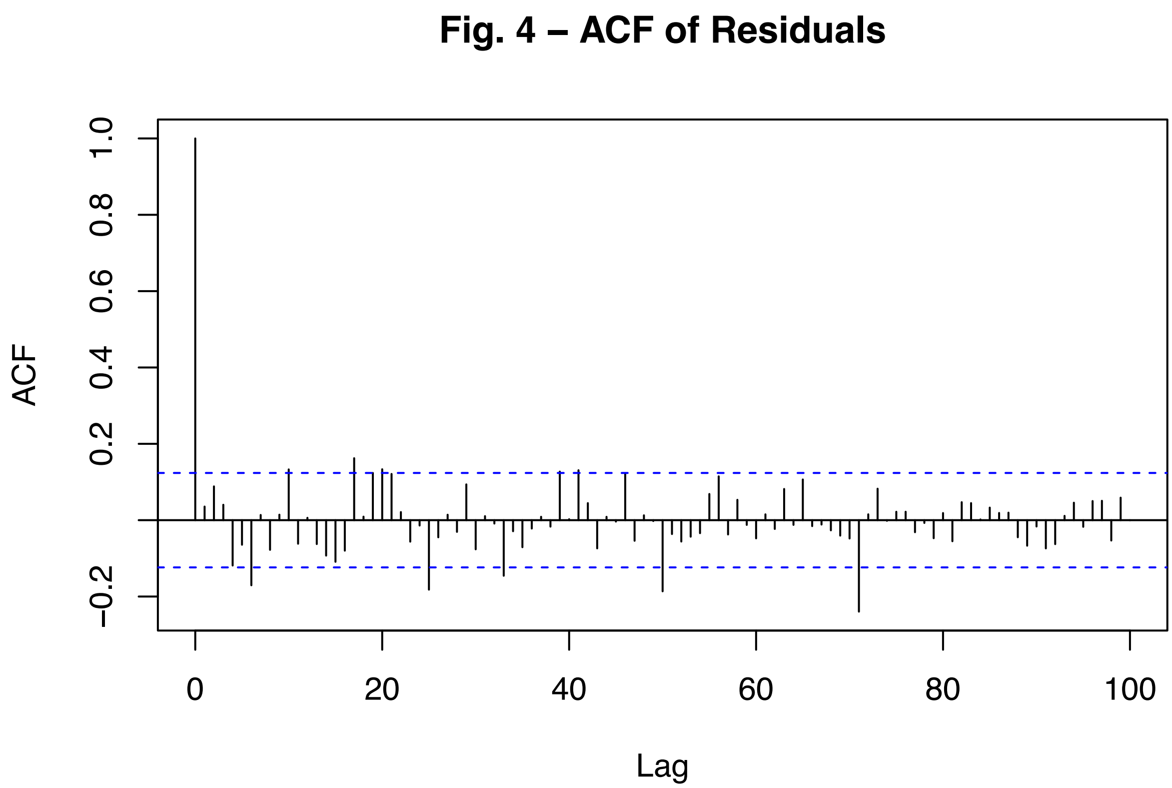



```
pacf(sbux_diff2,lag.max=200)
```

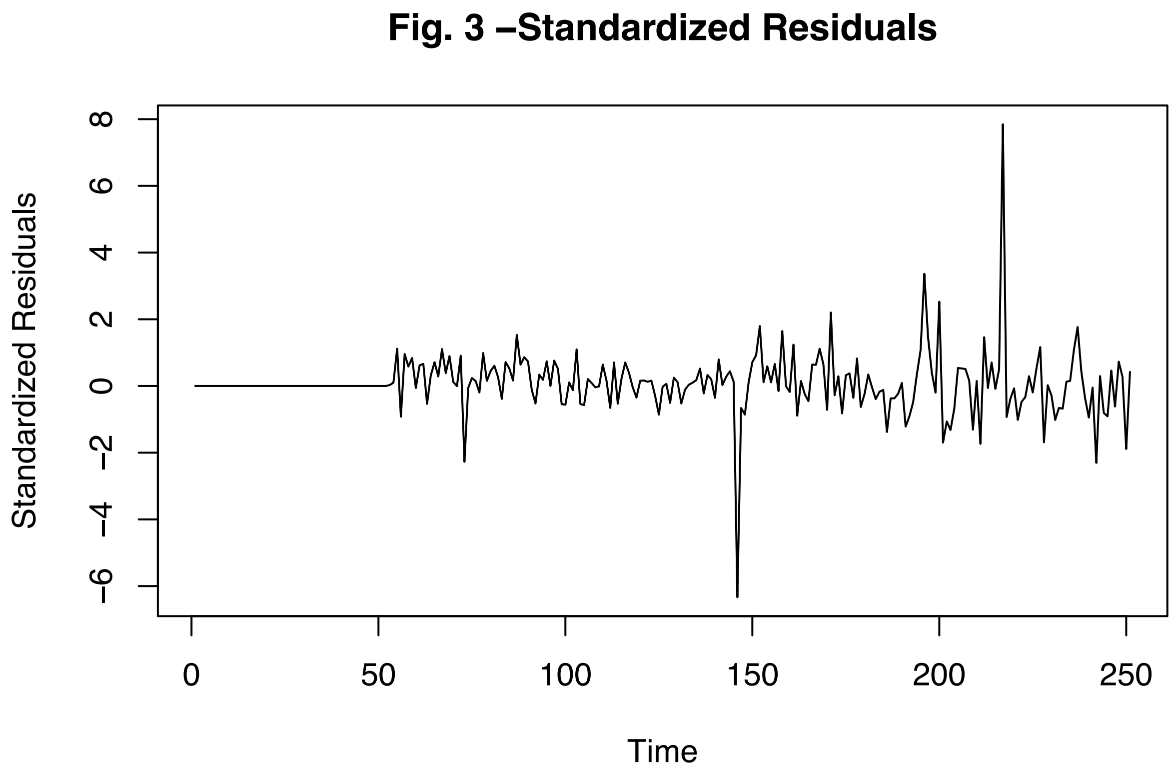



```
sbux_sarima1 = Arima(sbux_diff1, order=c(1,1,1), seasonal=list(order=c(1,1,0), period=25), method="CSS")
```

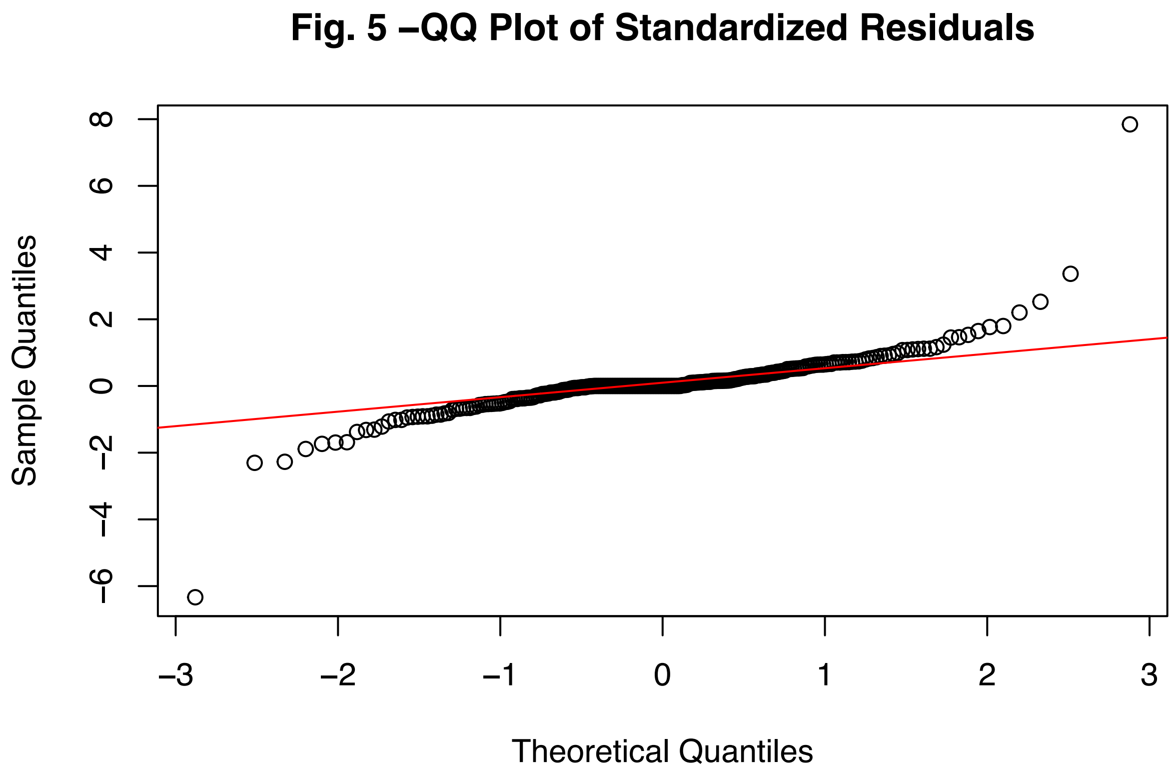
```
#calculating and plotting residuals
residuals <- residuals(sbux_sarima1)
sigma2<- 0.000745
std_residuals <- residuals / sqrt(sigma2)

#plotting residual acf
acf(residuals, main = "Fig. 4 - ACF of Residuals", lag.max = 100)
```




```
#plotting standardized residuals
plot(std_residuals, type = "l", main = "Fig. 3 -Standardized Residuals", xlab = "Time", ylab = "Standard
```




```r
#qqnorm of standardized residuals
qqnorm(std_residuals, main = "Fig. 5 -QQ Plot of Standardized Residuals")
qqline(std_residuals, col = "red")  # Add a reference line
```




```r
adf.test(residuals)
```

```
## Warning in adf.test(residuals): p-value smaller than printed p-value
```

```
##
##  Augmented Dickey-Fuller Test
##
## data:  residuals
## Dickey-Fuller = -6.7272, Lag order = 6, p-value = 0.01
## alternative hypothesis: stationary
```

According to our p-value of 0.715, our alternative hypothesis is very much in fact not significant. Hence we fail to reject our initial null hypothesis meaning our Starbucks Stock Price time series is not stationary, even after applying multiple SARIMA methods.

While we could proceed to