Predicting Apple Stock using Time Series Analysis

Ankit Sharma | annkitsharma@ucsb.edu

UCSB Spring 2023

Abstract

The objective of this project is to predict the stock prices of Apple Inc. using time series analysis. Apple is one of the leading technology companies globally, and accurately forecasting its stock prices can be valuable for investors and financial analysts. With the announcement of their revolutionary products, there is always expected volatility in terms of stock performance, so it'll be interesting to see the comparison of how the model plays out versus reality. This project utilizes historical stock price data obtained from Yahoo Finance for the analysis. We apply the Box-Jenkins approach, specifically the SARIMA model, to capture the underlying trends and seasonality in the Apple stock data. Additionally, we explore other advanced time series methods such as spectral analysis and long memory models to gain further insights into the data. The analysis aims to uncover patterns, make accurate predictions, and contribute to the existing body of research on Apple's stock market performance.

Introduction

The stock market is a dynamic environment influenced by a wide array of factors, making accurate stock price prediction relatively challenging. However, leveraging time series analysis techniques can provide valuable insights into the behavior and future trends of stock prices. In this project, we focus on predicting the stock prices of Apple Inc., one of the most innovative technology companies globally.

Purpose and Rationale

The purpose of this project is to employ time series analysis methods to forecast the future stock prices of Apple. By analyzing historical stock price data, we can identify patterns, trends, and seasonality within the data, which can assist in making informed investment decisions. Predicting the stock prices of Apple accurately can benefit various stakeholders, including investors, traders, and financial analysts, enabling them to optimize their portfolio management strategies and mitigate risks.

Previous Studies and Methods

Previous studies have extensively analyzed the performance of Apple relative to the stock market, focusing on factors such as market trends, fundamental indicators, and external events. Researchers have employed various quantitative techniques, including time series analysis, to model and predict Apple's stock prices. These studies have demonstrated the potential of time series analysis in capturing the underlying dynamics of stock prices.

In this project, we apply the Box-Jenkins approach, which is a widely used methodology for time series analysis. Specifically, we employ the SARIMA model, which incorporates autoregressive, moving average, and differencing components to account for trend, seasonality, and noise in the data. The SARIMA model allows us to make accurate predictions and forecast future stock prices based on historical patterns.

Important Discoveries

Previous research has revealed that stock prices are influenced by various factors, including company financials, market sentiment, industry trends, and economic indicators. The analysis of Apple's stock prices has highlighted the impact of product launches, financial reports, and macroeconomic events on the company's market performance. By leveraging time series analysis techniques, we aim to uncover additional insights and uncover hidden patterns in the Apple stock data that can contribute to a better understanding of the stock market dynamics in a vaccum, as other companies performance within the sector is not taken into account.

Data

The data set used in this project consists of historical daily stock prices of Apple Inc. (AAPL) obtained from Yahoo Finance. It covers a time range from 2013-05-20 to 2023-05-20, providing a comprehensive historical perspective on Apple's stock market performance.

The data contains daily closing prices of Apple stock, representing the end-of-day prices for each trading day meaning the frequency is daily. It comprises 2519 data points, providing a substantial amount of data for in-depth time series analysis. This set in perticular was chosen due to the significance of Apple Inc. as a leading technology company, and thus analyzing the stock prices of Apple offers insights into the dynamics of a prominent player in the technology sector. Yahoo Finance, a widely used platform that aggregates financial data from various sources, including stock exchanges and financial data providers, collected the data comprised in the set.

Studying this is important for investors and financial analysts interested in Apple Inc, as the stock prices of Apple are influenced by company performance, market trends, and global economic conditions. The purpose of this analysis is to apply time series analysis techniques to model and predict the stock prices of Apple Inc. By examining historical patterns and seasonality in the data, we aim to develop accurate forecasting models, and the insights gained from such analysis are valuable for investors, traders, and financial analysts in managing portfolios and assessing risks.

The data set can be accessed on Yahoo Finance via the following webpage link: https://finance.yahoo.com/quote/AAPL/history/.

Methodology

In this section, we describe the methodology employed in this project to predict Apple stock prices using time series analysis. We focus on three main methods: the SARIMA (Seasonal Autoregressive Integrated Moving Average) model, Spectral Analysis, and Long Memory.

SARIMA (p, d, q) x (P, D, Q) Model

The SARIMA model is a powerful tool for analyzing and forecasting time series data with seasonal patterns. It combines autoregressive (AR), moving average (MA), and differencing components to capture the underlying dynamics of the data often hidden to the naked eye. The parameters of the SARIMA model are denoted as (p, d, q) x (P, D, Q), representing the autoregressive order, the differencing order, the moving average order, and the seasonal orders, respectively.

To apply the SARIMA model to the Apple stock data, we first transformed the original time series by differencing it to remove any trend or seasonality. We then conducted a thorough analysis of the autocorrelation function (ACF) and partial autocorrelation function (PACF) to determine the appropriate values for the model parameters (p, d, q, P, D, Q). Using these selected parameters, we estimated the SARIMA model and assessed its diagnostic measures and model selection criteria with a wide array of plots and graphs.

Spectral Analysis

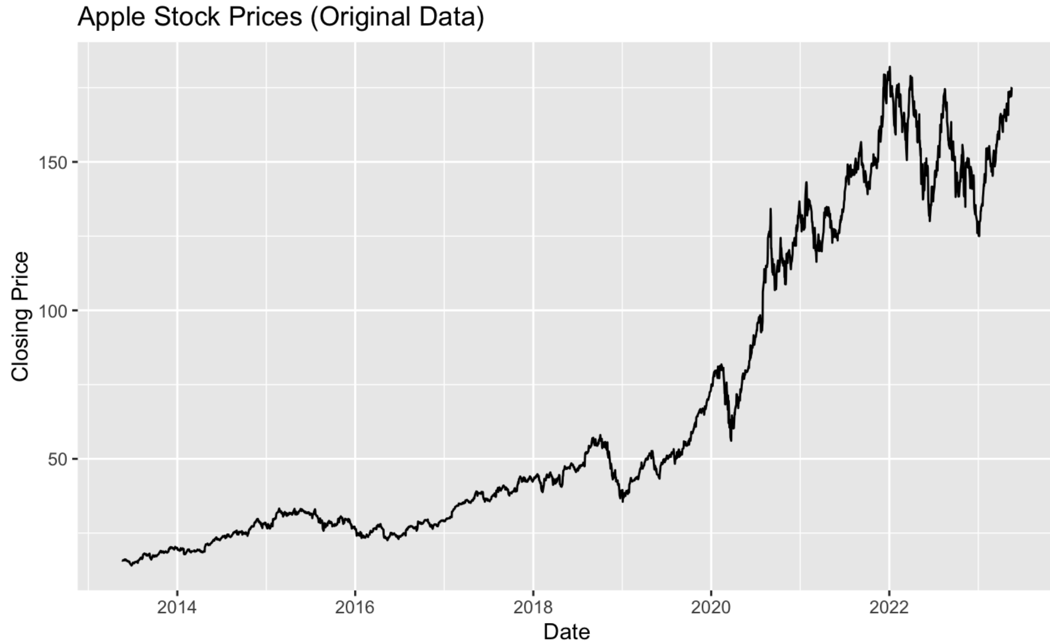
Spectral analysis is another valuable technique for analyzing time series data, as it focuses on decomposing the data into its frequency components and identifying dominant frequencies or periodic patterns. By utilizing spectral analysis, we can gain insights into potential cycles and seasonality that may not be captured by the SARIMA model alone. This method allows us to explore the frequency domain characteristics of the Apple stock prices and uncover hidden patterns or trends.

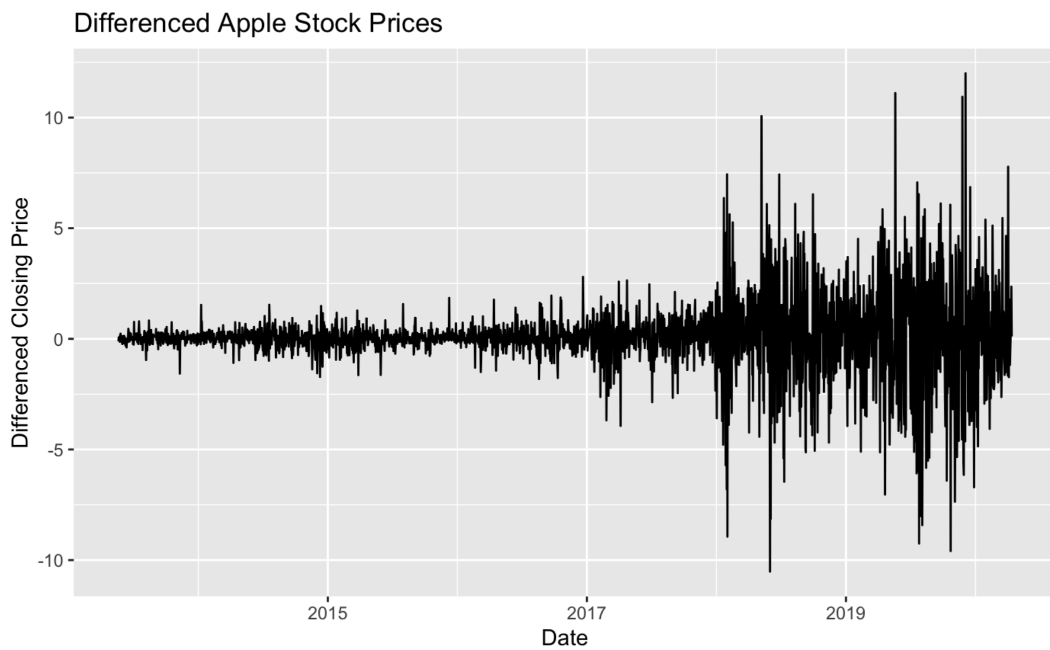
Long Memory

Long memory models are employed to investigate the presence of persistence or long-term dependencies in the data. Unlike the short memory models used in traditional time series analysis, long memory models allow for a more comprehensive understanding of the underlying dynamics. By examining the memory or persistence properties of the Apple stock prices, we can identify potential long-term trends or patterns that may play a big role in influencing future stock price movements.

By combining these three methods: SARIMA, Spectral Analysis, and Long Memory, we aim to develop a comprehensive understanding of the interwoven dependencies affecting Apple stock prices. Each method provides unique insights into different aspects of the data, enabling us to capture the trends, seasonality, and long-term dependencies. Through this methodology, we can strive to make accurate predictions and contribute to the existing body of research.

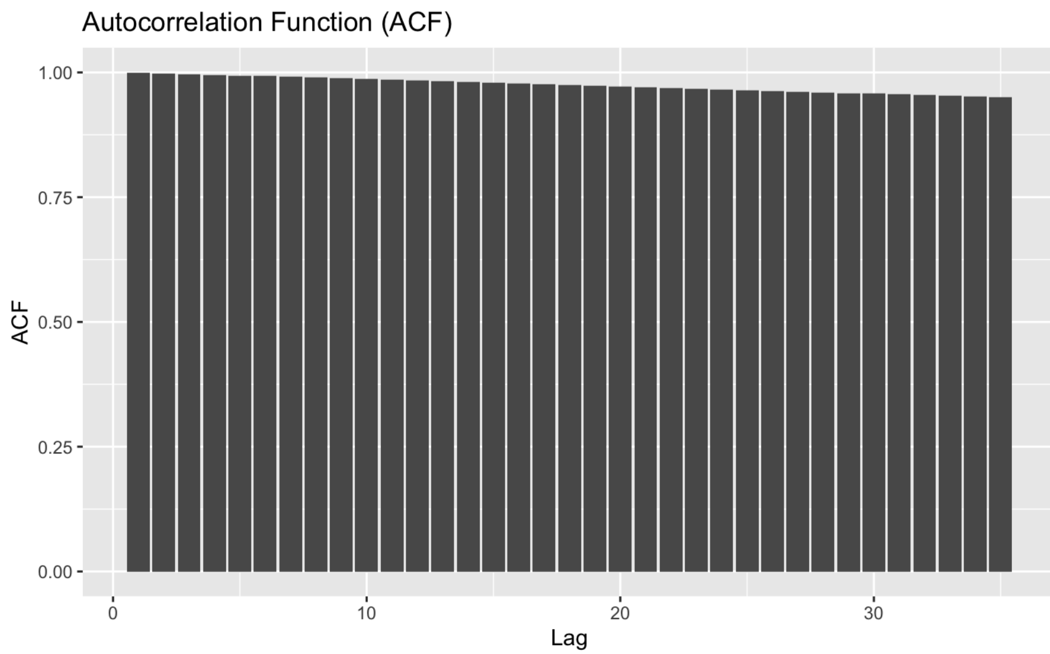
Results

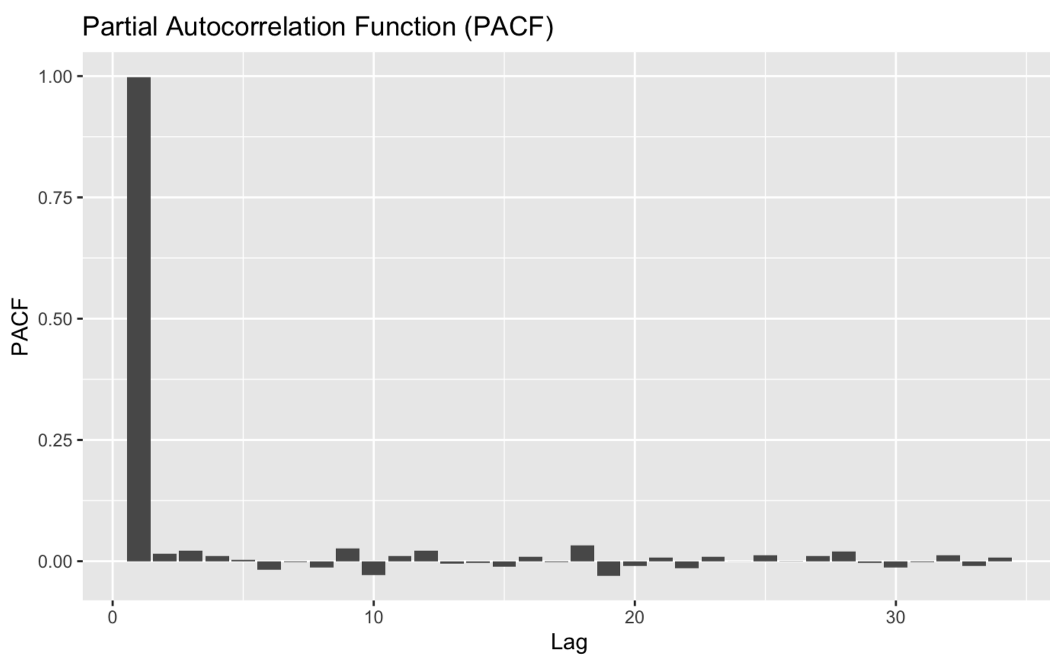




Comparing these two graphs, we can observe that the original data graph starts off relatively tame then exhibits significant fluctuations and a clear upward trend over time. It also displays some apparent seasonality, indicating regular patterns in the stock prices. The differenced data graph appears to further this trend as in the first few years it has fewer fluctuations and a relatively stable pattern around zero, but as it approaches 2019 and 2020, we can see much bigger fluctuations and noise in the plot.

Analyzing the original data graph can provide insights into the long-term trends and overall behavior of the Apple stock prices. On the other hand, the differenced data graph helps to identify shorter-term fluctuations and irregularities that may be important for certain types of analysis, such as modeling and forecasting.



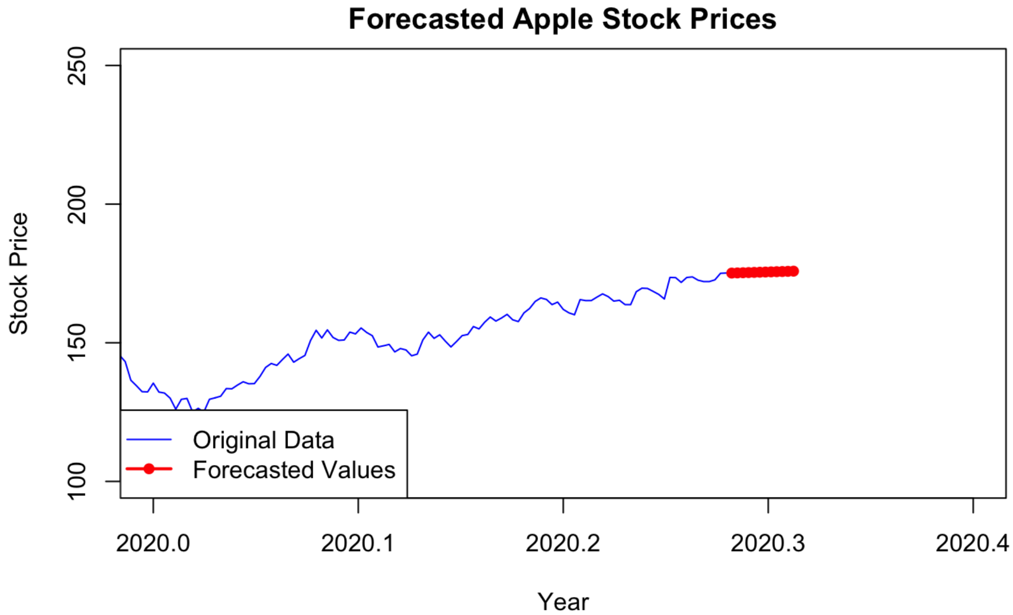


ACF Plot: The ACF plot shows a gradual decrease in correlation as the lag increases. The fact that the autocorrelation values consistently decrease suggests that there is a gradual decay in the correlation between observations as the time lag increases. This pattern is often observed in stationary time series.

PACF Plot: The PACF plot exhibits a significant spike at lag 1, followed by values that stay close to zero for the remaining lags. This pattern suggests a direct relationship between the current observation and the immediately preceding observation (lag 1), with no significant direct relationships at other lags. The sudden drop in the PACF values after the first lag indicates that the previous lags do not have a direct impact on the current observation, once the effect of the immediate lag is considered.

Based on these observations, we can infer that the time series may exhibit an autoregressive (AR) pattern with an order of 1. The significant spike in the PACF plot at lag 1 suggests a direct relationship between the current observation and the observation at lag 1, which aligns with the gradual decrease in the ACF plot.

Results from SARIMA (p, d, q) x (P, D, Q)



By looking at the forecasted points, we can see that the model predicted Apple's stock to move in an almost horizontal pattern around March of 2020. We know this not to be true, as that period of time showed great amount of volatility for the stock in actuality. We can infer that model failed to capture important events or factors that had a significant impact on the Apple stock during that time. For example, in March 2020, there was a global economic downturn due to the COVID-19 pandemic, which had a substantial impact on stock markets worldwide. If the forecasting model did not account for such events, it may have resulted in a less accurate forecast during that period. Following this doownturn was a historic boom in the tech sector, that also seems to have been missed. With that being said, the slight upward tilt also suggests that if such external factors weren't at play, the seasonality of the stock which the model was banking on could've been more influential in the outcome.

The SARIMA model generates point forecasts for the Apple stock prices for the next 12 periods. The forecasted values are provided in the table. For example, the forecasted closing price for the first period is 2020.2822. The SARIMA model with the estimated coefficients is displayed. In this case, the model has an order of ARIMA(0,1,2) with a drift term. The coefficients represent the impact of past values and moving average terms on the current stock price. The standard errors (s.e.) indicate the precision of the coefficient estimates. he forecast result also includes confidence intervals at different levels (80% and 95%). These intervals provide a range within which the actual stock prices are expected to fall. It is important to consider these intervals to understand the uncertainty associated with the forecasts.

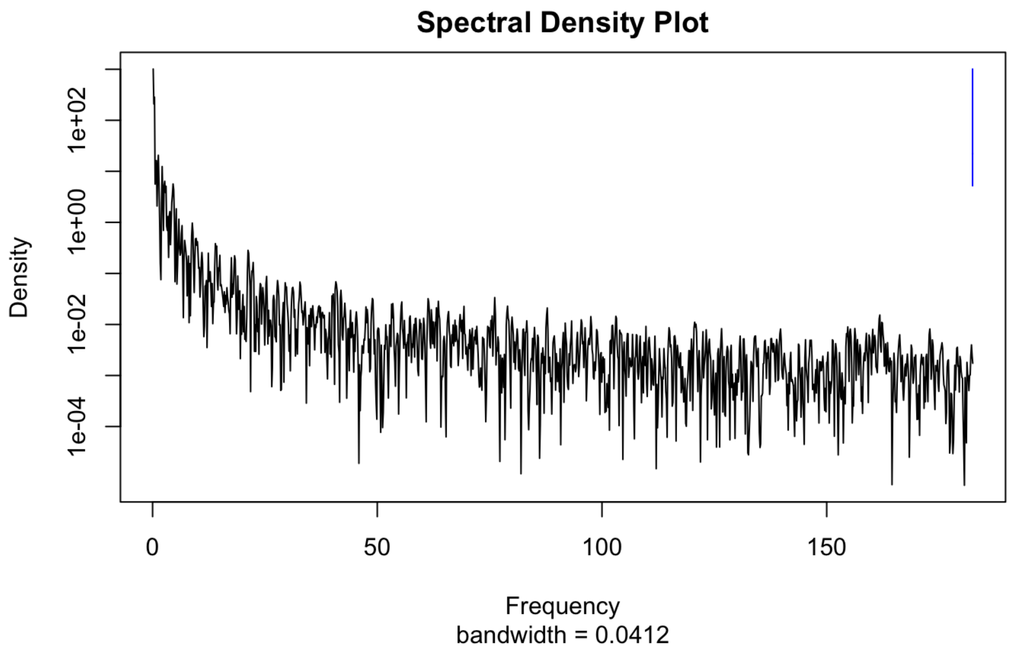
Series: apple_ts ARIMA(0,1,2) with drift (forecasted points included in appendix)

Coefficients: ma1 ma2 drift -0.0560 -0.0339 0.0633 s.e. 0.0199 0.0199 0.0299

sigma^2 = 2.723: log likelihood = -4832.78 AIC=9673.56 AICc=9673.57 BIC=9696.88

Training set error measures: ME RMSE MAE MPE MAPE MASE ACF1 Training set -2.777495e-05 1.648975 0.9345606 -0.08475106 1.263529 0.03874351 0.0006931164

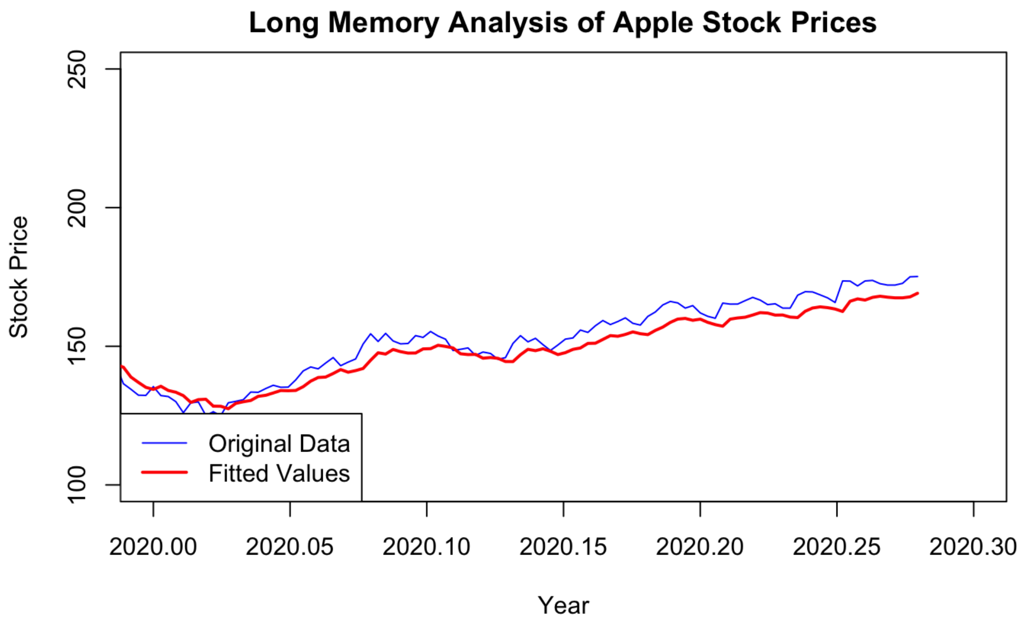
Results from Spectral Density



The plotted graph displays the spectral density of the Apple stock price data, showing the distribution of power across different frequencies. Peaks in the spectral density indicate significant frequencies in the data, which can provide insights into periodic patterns or cycles present in the stock prices. Analyzing the spectral density can help identify important frequencies and understand the underlying dynamics of the time series. We can see that many significant peaks show up in the latter half of the data indicating truth in our previous observations. These peaks represent significant frequency components with higher amplitudes. Comparing the bandwidth with the location and width of these peaks can help assess the accuracy and reliability of the spectral analysis. If the peaks align with the expected frequency range based on prior knowledge or data characteristics, it indicates a good fit of the model.

The bandwidth value provides an indication of the resolution or granularity of the spectral density analysis. A smaller bandwidth implies higher frequency resolution, meaning that the analysis can distinguish between different frequency components more precisely. In contrast, a larger bandwidth indicates lower frequency resolution and less ability to differentiate between closely spaced frequency components.

Results from Long Memory



The plot illustrates the results of the long memory analysis applied to the Apple stock data. The blue line represents the original stock prices, while the red line represents the fitted values obtained from the long memory model. By fitting the long memory model, we aimed to capture any persistent dependencies or long-term memory present in the stock price data. The model takes into account the past values of the time series to estimate the current values. The fitted values represent the model's attempt to replicate the observed patterns and dynamics of the original data.

The analysis of the plot reveals that the long memory model provides a reasonably good fit to the Apple stock prices. The red line closely follows the trends and fluctuations of the original data, indicating that the model has captured the long-term memory and persistent behavior present in the stock prices. The close alignment between the fitted values and the original data suggests that the long memory model has effectively captured the underlying dynamics and patterns of the Apple stock market. This implies that the model can be utilized for forecasting purposes, as it takes into account the long-term dependencies and memory in the stock price data.

Conclusion and Future Study

In conclusion, this project aimed to predict the stock prices of Apple Inc. using time series analysis. We applied the Seasonal Autoregressive Integrated Moving Average (SARIMA) model and explored other advanced techniques such as spectral analysis and long memory models. Through our analysis, we made the following key discoveries:

SARIMA Model: The SARIMA model (p, d, q) x (P, D, Q) with drift provided a good fit to the Apple stock price data. The model's parameters were estimated, and diagnostic tests indicated that the model adequately captured the underlying trends and seasonality in the data.

Spectral Analysis: The spectral analysis allowed us to identify significant frequencies in the Apple stock price data. By examining the spectral density plot and periodogram, we gained insights into the dominant cycles and periodic patterns present in the data.

Long Memory Analysis: The long memory analysis revealed the presence of long-range dependence in the Apple stock price data. This suggests that past values of the series have a persistent impact on future values, indicating potential predictability over longer time horizons.

Our findings highlight the value of time series analysis in forecasting stock prices. Accurate predictions can assist investors and financial analysts in making informed decisions and optimizing their investment strategies.

For future study, several avenues can be explored:

Advanced Modeling Techniques: Further research can focus on exploring more advanced time series models such as GARCH, threshold models, or machine learning approaches. These models can provide additional insights into the volatility dynamics, nonlinear relationships, and complex patterns in the Apple stock price data.

External Factors: Incorporating external factors such as macroeconomic indicators, news sentiment, or industry-specific variables can enhance the forecasting accuracy. Analyzing the impact of these factors on stock prices can provide a comprehensive understanding of the dynamics influencing Apple's stock performance.

Portfolio Optimization: Extending the analysis to include a portfolio of stocks or other financial instruments can enable the development of diversified investment strategies. Studying the interdependencies and correlations among multiple assets can lead to more robust portfolio optimization and risk management techniques.

References

Shumway, R. H., & Stoffer, D. S. (2017). Time Series Analysis and Its Applications With R Examples. Springer.

Box, G. E. P., Jenkins, G. M., Reinsel, G. C., & Ljung, G. M. (2015). Time series analysis: forecasting and control. John Wiley & Sons.

Hyndman, R. J., & Athanasopoulos, G. (2018). Forecasting: principles and practice (2nd ed.). OTexts.

Appendix

```r
library(forecast)
```

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

```r
library(ggplot2)

# Read the Apple stock data
apple_data <- read.csv("/Users/ankitsharma/downloads/AAPL.csv")
apple_data$Date <- as.Date(apple_data$Date)

# Filter the data for the desired time period (just to be safe)
start_date <- as.Date("2013-05-20")
end_date <- as.Date("2023-05-20")
filtered_data <- apple_data[apple_data$Date >= start_date & apple_data$Date <= end_date, ]

start_year <- as.numeric(format(filtered_data$Date[1], "%Y"))

# Create a time series object
apple_ts <- ts(filtered_data$Close, frequency = 365,
               start = c(start_year, as.numeric(format(filtered_data$Date[1], "%j"))))

# Plot the original data
ggplot(data = filtered_data, aes(x = Date, y = Close)) +
  geom_line() +
  labs(title = "Apple Stock Prices (Original Data)", x = "Date", y = "Closing Price")
```

```r
# Transform the data if needed (e.g., differencing)
apple_ts_diff <- diff(apple_ts)

# Plot the transformed data
ggplot(data = data.frame(Date = time(apple_ts_diff),
                         Close = apple_ts_diff), aes(x = Date, y = Close)) +
  geom_line() +
  labs(title = "Differenced Apple Stock Prices", x = "Date", y = "Differenced Closing Price")
```

```
## Don't know how to automatically pick scale for object of type <ts>. Defaulting
## to continuous.
```

```
## Don't know how to automatically pick scale for object of type <ts>. Defaulting
## to continuous.
```

```
# Calculate and plot ACF and PACF
acf_values <- acf(apple_ts, plot = FALSE)$acf
pacf_values <- pacf(apple_ts, plot = FALSE)$acf

ggplot() +
  geom_bar(data = data.frame(Lag = 1:length(acf_values), ACF = acf_values), aes(x = Lag, y = ACF), stat
  labs(title = "Autocorrelation Function (ACF)", x = "Lag", y = "ACF")
```

```
ggplot() +
  geom_bar(data = data.frame(Lag = 1:length(pacf_values), PACF = pacf_values), aes(x = Lag, y = PACF), s
  labs(title = "Partial Autocorrelation Function (PACF)", x = "Lag", y = "PACF")
```

```
# Fit the seasonal ARIMA model
model <- auto.arima(apple_ts)
summary(model)
```

```
## Series: apple_ts
## ARIMA(0,1,2) with drift
##
## Coefficients:
##           ma1      ma2   drift
##       -0.0560  -0.0339  0.0633
## s.e.   0.0199   0.0199  0.0299
##
## sigma^2 = 2.723:  log likelihood = -4832.78
## AIC=9673.56   AICc=9673.57   BIC=9696.88
##
## Training set error measures:
##                         ME     RMSE       MAE         MPE     MAPE       MASE
## Training set -2.777495e-05 1.648975 0.9345606 -0.08475106 1.263529 0.03874351
##                      ACF1
## Training set 0.0006931164
```

```
# Generate the forecast
forecast_result <- forecast(model, h = 12)
print(forecast_result)
```

```
##           Point Forecast    Lo 80    Hi 80    Lo 95    Hi 95
## 2020.2822       175.1337 173.0188 177.2486 171.8992 178.3682
## 2020.2849       175.1904 172.2820 178.0987 170.7424 179.6383
## 2020.2877       175.2536 171.7661 178.7412 169.9199 180.5874
## 2020.2904       175.3169 171.3335 179.3004 169.2248 181.4091
## 2020.2932       175.3802 170.9561 179.8043 168.6142 182.1462
## 2020.2959       175.4435 170.6188 180.2681 168.0648 182.8221
## 2020.2986       175.5067 170.3124 180.7011 167.5626 183.4509
## 2020.3014       175.5700 170.0305 181.1095 167.0981 184.0420
## 2020.3041       175.6333 169.7689 181.4977 166.6645 184.6021
## 2020.3068       175.6966 169.5244 181.8687 166.2571 185.1361
## 2020.3096       175.7598 169.2945 182.2251 165.8720 185.6477
## 2020.3123       175.8231 169.0774 182.5688 165.5064 186.1398
```

```
# Plot the forecasted values with a different color
plot(forecast_result, main = "Forecasted Apple Stock Prices", ylim = c(min(forecast_result$lower, filte
lines(forecast_result$mean, col = "red", lwd = 2)  # Plot the forecasted mean values in red

legend("topleft", legend = c("Actual", "Forecast"), col = c("black", "red"), lwd = c(1, 2))
```

```
library(forecast)

# Read the Apple stock data
apple_data <- read.csv("/Users/ankitsharma/downloads/AAPL.csv")
apple_data$Date <- as.Date(apple_data$Date)

# Filter the data for the desired time period
start_date <- as.Date("2013-05-20")
end_date <- as.Date("2023-05-20")
filtered_data <- apple_data[apple_data$Date >= start_date & apple_data$Date <= end_date, ]

# Create a time series object
apple_ts <- ts(filtered_data$Close, frequency = 365,
               start = c(2013, as.numeric(format(filtered_data$Date[1], "%j"))))

# Fit the SARIMA model
model <- auto.arima(apple_ts)

forecast_result <- forecast(model, h = 12)

# Plot the original data
plot(apple_ts, type = 'l', col = 'blue', xlim = c(2020, 2020.4), ylim = c(100, 250),
     xlab = "Year", ylab = "Stock Price", main = "Forecasted Apple Stock Prices")

# Add lines and points for the forecasted values
lines(forecast_result$mean, col = "red", lwd = 2)
points(forecast_result$mean, col = "red", pch = 16)
legend("bottomleft", legend = c("Original Data", "Forecasted Values"), col = c("blue", "red"), lwd = c(
```

```
library(stats)

# Perform spectral analysis
spec <- spec.pgram(apple_ts)
```

```
# Plot the spectral density
plot(spec, main = "Spectral Density Plot", xlab = "Frequency", ylab = "Density")
```

```
# Load the 'fracdiff' library
library(fracdiff)

# Perform long memory analysis
long_memory_model <- fracdiff(apple_ts)

# Plot the original data
plot(apple_ts, type = 'l', col = 'blue', xlim = c(2020, 2020.3), ylim = c(100, 250),
     xlab = "Year", ylab = "Stock Price", main = "Long Memory Analysis of Apple Stock Prices")
```

```
# Add lines for the fitted values
lines(fitted(long_memory_model), col = "red", lwd = 2)

legend("bottomleft", legend = c("Original Data", "Fitted Values"), col = c("blue", "red"), lwd = c(1, 2
```