Amazon COVID Effect

Mootaz Abdel-Dayem

1/6/2021

Overview

COVID-19 was the biggest disruption in the history of the world, affecting the entire world's population in a way that was never seen before. Many industries were affected as a result of COVID-19 and the enforcement of the shutdown. One company has experienced the opposite during the crisis of COVID-19, is Amazon. The company has been very busy selling and shipping goods around the globe from medical masks, gloves to food and regular grocery products. As the company did well during COVID, the stock price must have been affected as well to mirror the economic state of the company.

The goal of this project is to study the effects of COVID-19 on Amazon's stock price and to predict if another wave of COVID-19 happens again, what to expect to the Amazon stock price. A machine learning algorithms are used to create 3 different models to study and analyze the effects of COVID on the Amazon stock price during and after the global pandemic.

Introduction

In this project, a Forecasting model is used to to predict the future Amazon daily stock price based on current and historic data. The historical data are extracted and prepared to predict the effect of COVID-19 on the stock price values for the Amazon dataset. To properly examine the dataset, we need to create two sub-sets: 1) Amazon daily stock prices before COVID-19 2) Amazon daily stock prices During/After COVID-19

The $Before$ dataset end date is before the COVID-19 crisis. It shows us the stock prices before COVID became a global pandemic. The $After$ dataset end date extends to after the COVID pandemic to show up the effects of COVID on the stock price. Hence, the $Before$ dataset is shorter in time length than the $After$ dataset because the After dataset extends to mid 2020.

Three machine learning models are used to study this effect. First, the Arima model is used to predict the stock prices during and after COVID crisis. Second, the KNN Regression Time Series Forecasting Model is used to predict and study the effects of the pandemic on the Amazon daily stock price. Finally, the Neural Network Model is used to predict the stock values during and after the COVID-19 crisis. Once we have the results of all the models, we compare them and conclude which model is more accurate and more efficient in the prediction.

Loading Required Libraries

```r
if(!require(quantmod)) install.packages("quantmod", repos = "http://cran.us.r-project.org")
```

```
## Loading required package: quantmod
```

```
## Loading required package: xts
```

```
## Loading required package: zoo
```

```
##
## Attaching package: 'zoo'
```

```
## The following objects are masked from 'package:base':
##
##     as.Date, as.Date.numeric

## Loading required package: TTR

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

```r
if(!require(forecast)) install.packages("forecast", repos = "http://cran.us.r-project.org")
```

```
## Loading required package: forecast
```

```r
if(!require(tseries)) install.packages("tseries", repos = "http://cran.us.r-project.org")
```

```
## Loading required package: tseries
```

```r
if(!require(timeSeries)) install.packages("timeSeries", repos = "http://cran.us.r-project.org")
```

```
## Loading required package: timeSeries

## Loading required package: timeDate

##
## Attaching package: 'timeSeries'

## The following object is masked from 'package:zoo':
##
##     time<-
```

```r
if(!require(dplyr)) install.packages("dplyr", repos = "http://cran.us.r-project.org")
```

```
## Loading required package: dplyr

##
## Attaching package: 'dplyr'

## The following objects are masked from 'package:timeSeries':
##
##     filter, lag

## The following objects are masked from 'package:xts':
##
##     first, last

## The following objects are masked from 'package:stats':
##
##     filter, lag

## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```r
if(!require(readxl)) install.packages("readxl", repos = "http://cran.us.r-project.org")
```

```
## Loading required package: readxl
```

```r
if(!require(kableExtra)) install.packages("kableExtra", repos = "http://cran.us.r-project.org")
```

```
## Loading required package: kableExtra

##
## Attaching package: 'kableExtra'
```

```
## The following object is masked from 'package:dplyr':
##
##     group_rows
```

```r
if(!require(data.table)) install.packages("data.table", repos = "http://cran.us.r-project.org")
```

```
## Loading required package: data.table

##
## Attaching package: 'data.table'

## The following objects are masked from 'package:dplyr':
##
##     between, first, last

## The following objects are masked from 'package:xts':
##
##     first, last
```

```r
if(!require(DT)) install.packages("DT", repos = "http://cran.us.r-project.org")
```

```
## Loading required package: DT
```

```r
if(!require(tsfknn)) install.packages("tsfknn", repos = "http://cran.us.r-project.org")
```

```
## Loading required package: tsfknn
```

```r
if(!require(ggplot2)) install.packages("ggplot2", repos = "http://cran.us.r-project.org")
```

```
## Loading required package: ggplot2
```

```r
library(quantmod)
library(forecast)
library(tseries)
library(timeSeries)
library(dplyr)
library(readxl)
library(kableExtra)
library(data.table)
library(DT)
library(tsfknn)
library(ggplot2)
```

Data Preparation

Importing the data

Using the quantmod package, we can obtain the Amazon stock prices from 2010-01-01 for our analysis. In addition, two sets of data are needed: 1) Amazon daily stock prices before COVID-19 2) Amazon daily stock prices During/After COVID-19

To make sure we are picking the right *Before* prices, we will limit our *Before* list to Feb 2019.

For the *After* dataset, our list starts from 2017 until now (Early January 2021)

```r
# Set 1: Before COVID-19 Crisis
getSymbols("AMZN", src = "yahoo", from = "2010-01-01", to = "2019-02-28")
```

```
## 'getSymbols' currently uses auto.assign=TRUE by default, but will
## use auto.assign=FALSE in 0.5-0. You will still be able to use
```

```
## 'loadSymbols' to automatically load data. getOption("getSymbols.env")
## and getOption("getSymbols.auto.assign") will still be checked for
## alternate defaults.
##
## This message is shown once per session and may be disabled by setting
## options("getSymbols.warning4.0"=FALSE). See ?getSymbols for details.
```

```
## [1] "AMZN"
```

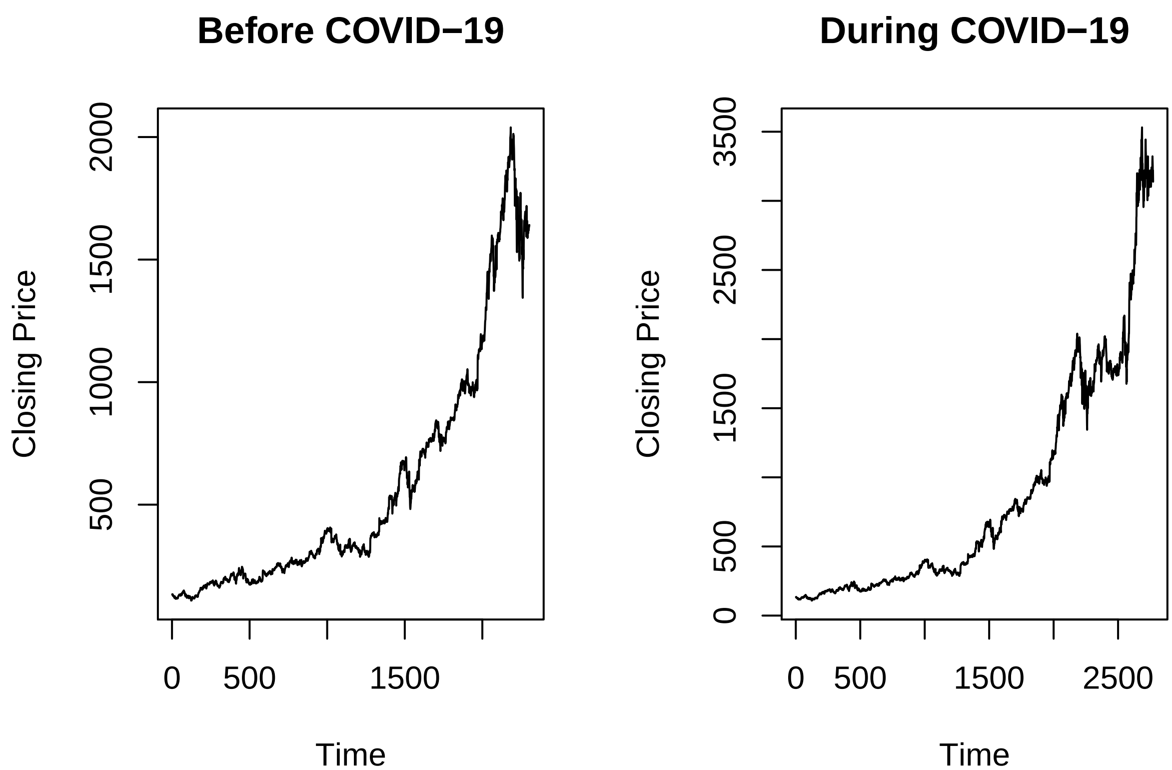
```
AMZN_data_before_covid <- as.data.frame(AMZN)
tsData_before_covid <- ts(AMZN_data_before_covid$AMZN.Close)

# Set 2: During COVID-19 Crisis
getSymbols("AMZN", src = "yahoo", from = "2010-01-01")
```

```
## [1] "AMZN"
```

```
AMZN_data_after_covid <- as.data.frame(AMZN)
tsData_after_covid <- ts(AMZN_data_after_covid$AMZN.Close)
```

Visualizing the data:

```
par(mfrow = c(1,2))
plot.ts(tsData_before_covid, ylab = "Closing Price", main = "Before COVID-19")
plot.ts(tsData_after_covid, ylab = "Closing Price", main = "During COVID-19")
```



Dataset Preview

The *Before* dataset is from "2010-01-01" to "2019-12-31"

```
summary(AMZN_data_before_covid)
```

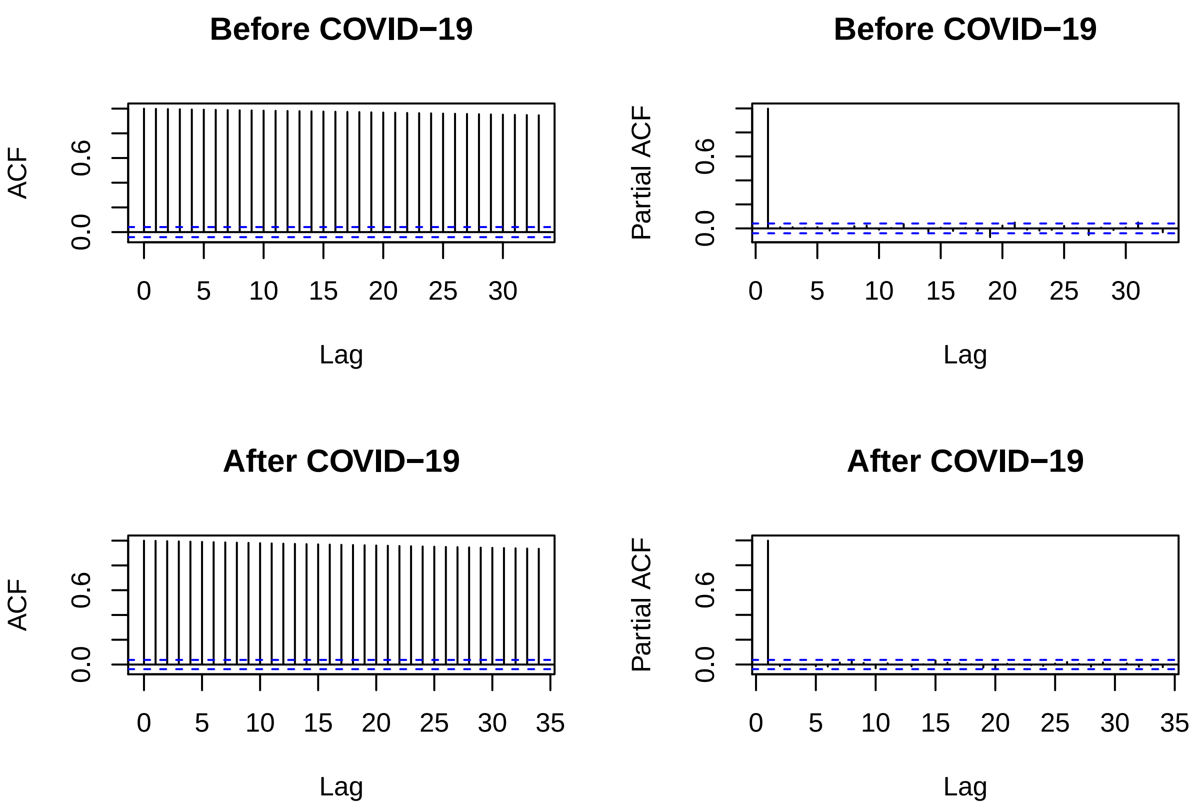
```
##    AMZN.Open        AMZN.High         AMZN.Low        AMZN.Close
##  Min.   : 105.9   Min.   : 111.3   Min.   : 105.8   Min.   : 108.6
```

```
##  1st Qu.: 219.1   1st Qu.: 221.5   1st Qu.: 215.3   1st Qu.: 218.4
##  Median : 340.0   Median : 342.5   Median : 334.7   Median : 338.6
##  Mean   : 570.8   Mean   : 576.6   Mean   : 564.1   Mean   : 570.6
##  3rd Qu.: 779.7   3rd Qu.: 783.6   3rd Qu.: 771.7   3rd Qu.: 780.2
##  Max.   :2038.1   Max.   :2050.5   Max.   :2013.0   Max.   :2039.5
##   AMZN.Volume        AMZN.Adjusted
##  Min.   :  984400   Min.   : 108.6
##  1st Qu.: 2800050   1st Qu.: 218.4
##  Median : 3890900   Median : 338.6
##  Mean   : 4616681   Mean   : 570.6
##  3rd Qu.: 5442000   3rd Qu.: 780.2
##  Max.   :42421100   Max.   :2039.5
```

Models Building

1) ARIMA Model

```r
par(mfrow = c(2,2))
acf(tsData_before_covid, main = "Before COVID-19")
pacf(tsData_before_covid, main = "Before COVID-19")

acf(tsData_after_covid, main = "After COVID-19")
pacf(tsData_after_covid, main = "After COVID-19")
```



Model Fitting

The *auto.arima* function is used to determine the time series model for each of the datasets

```r
modelfit_before_covid <- auto.arima(tsData_before_covid, lambda = "auto")
summary(modelfit_before_covid)
```

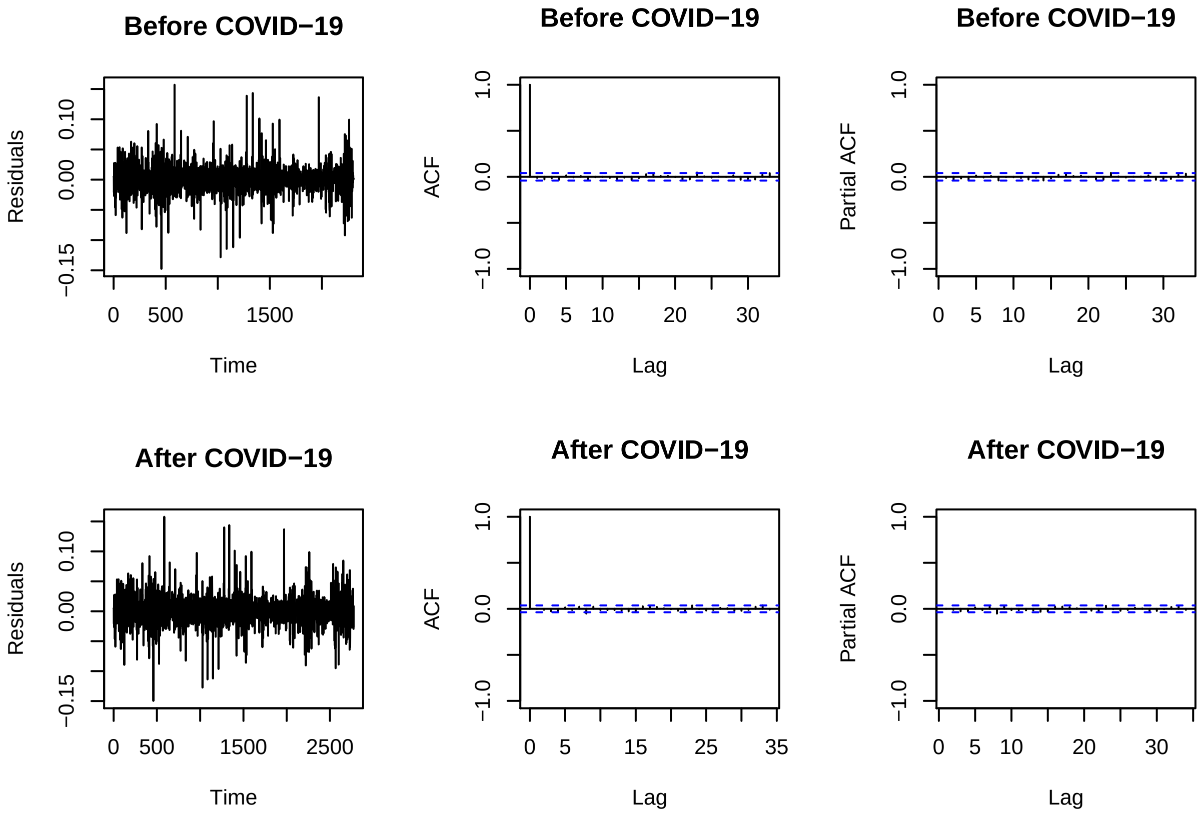
```
## Series: tsData_before_covid
## ARIMA(0,1,0) with drift
## Box Cox transformation: lambda= 0.01480029
##
## Coefficients:
##        drift
##       0.0012
## s.e.  0.0005
##
## sigma^2 estimated as 0.0004751:  log likelihood=5541.58
## AIC=-11079.17   AICc=-11079.16   BIC=-11067.68
##
## Training set error measures:
##                      ME     RMSE      MAE         MPE     MAPE      MASE
## Training set 0.03851497 14.98337 7.589411 -0.01988417 1.374156 0.9951071
##                     ACF1
## Training set -0.02993025
```

```r
modelfit_after_covid <- auto.arima(tsData_after_covid, lambda = "auto")
summary(modelfit_after_covid)
```

```
## Series: tsData_after_covid
## ARIMA(0,1,1) with drift
## Box Cox transformation: lambda= 0.01538506
##
## Coefficients:
##           ma1   drift
##       -0.0293  0.0013
## s.e.   0.0192  0.0004
##
## sigma^2 estimated as 0.0004815:  log likelihood=6652.44
## AIC=-13298.87   AICc=-13298.86   BIC=-13281.09
##
## Training set error measures:
##                     ME     RMSE      MAE      MPE     MAPE      MASE
## Training set 0.1230294 24.42145 11.98898 -0.01995 1.379886 0.9947883
##                     ACF1
## Training set -0.03493255
```

Let's check the residual diagnostics for each of the fitted models

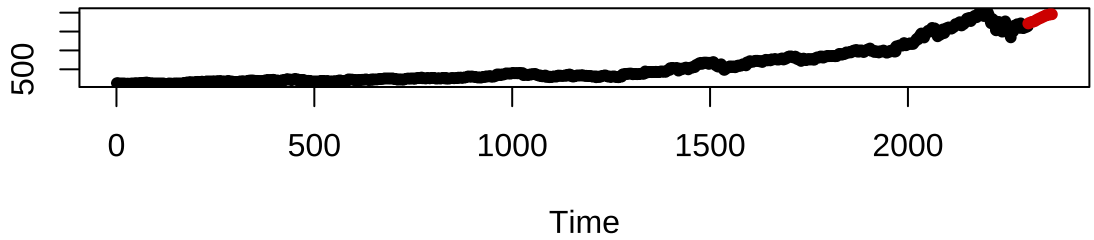
```r
par(mfrow = c(2,3))

plot(modelfit_before_covid$residuals, ylab = 'Residuals', main = "Before COVID-19")
acf(modelfit_before_covid$residuals,ylim = c(-1,1), main = "Before COVID-19")
pacf(modelfit_before_covid$residuals,ylim = c(-1,1), main = "Before COVID-19")

plot(modelfit_after_covid$residuals, ylab = 'Residuals', main = "After COVID-19")
acf(modelfit_after_covid$residuals,ylim = c(-1,1), main = "After COVID-19")
pacf(modelfit_after_covid$residuals,ylim = c(-1,1), main = "After COVID-19")
```

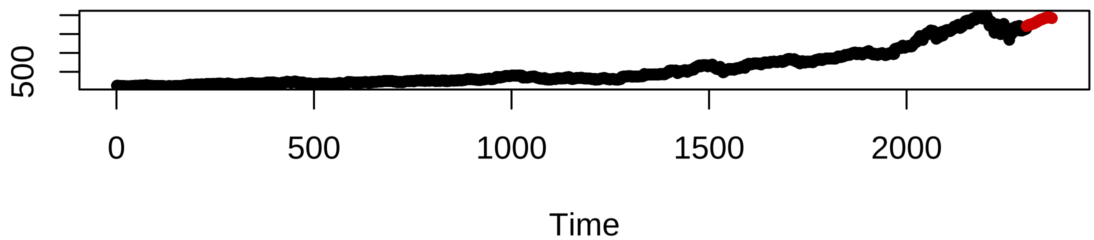


2) KNN Regression Time Series Forecasting Model

KNN model can be used for both classification and regression problems. The most popular application is to use it for classification problems.

```
par(mfrow = c(2,1))
predknn_before_covid <- knn_forecasting(AMZN_data_before_covid$AMZN.Close,
                                        h = 61, lags = 1:30, k = 32, msas = "MIMO")
predknn_after_covid <- knn_forecasting(AMZN_data_before_covid$AMZN.Close,
                                       h = 65, lags = 1:30, k = 36, msas = "MIMO")

plot(predknn_before_covid, main = "Before COVID-19")
plot(predknn_after_covid, main = "After COVID-19")
```





KNN model evaluation to forecast the time series

```
knn_ro_before_covid <- rolling_origin(predknn_before_covid)
knn_ro_after_covid <- rolling_origin(predknn_after_covid)
```

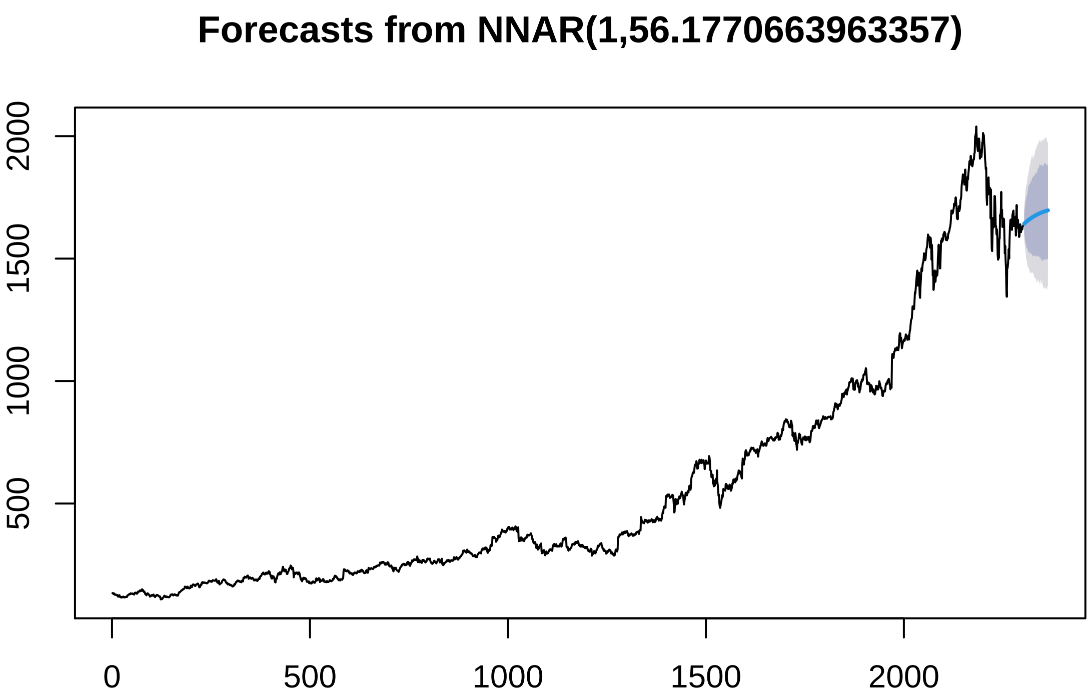
3. Feed Forward Neural Network Model

This model is a forecasting model with neural networks. The function model approach is to use lagged values of the time series as input data, reaching to a non-linear auto-regressive model.

First, we need to determine the number of hidden layers in the neural network

```
#Creating Hidden layers
alpha <- 1.5^(-10)
hn_before_covid <- length(AMZN_data_before_covid$AMZN.Close)/
              (alpha*(length(AMZN_data_before_covid$AMZN.Close) + 61))
hn_after_covid <- length(AMZN_data_after_covid$AMZN.Close)/
              (alpha*(length(AMZN_data_after_covid$AMZN.Close) + 65))

#Fitting nnetar
lambda_before_covid <- BoxCox.lambda(AMZN_data_before_covid$AMZN.Close)
lambda_after_covid <- BoxCox.lambda(AMZN_data_after_covid$AMZN.Close)
dnn_pred_before_covid <- nnetar(AMZN_data_before_covid$AMZN.Close,
                                size = hn_before_covid, lambda = lambda_before_covid)
dnn_pred_after_covid <- nnetar(AMZN_data_after_covid$AMZN.Close,
                               size = hn_after_covid, lambda = lambda_after_covid)

# Forecasting with nnetar
dnn_forecast_before_covid <- forecast(dnn_pred_before_covid, h = 61, PI = TRUE)
dnn_forecast_after_covid <- forecast(dnn_pred_after_covid, h = 65, PI = TRUE)

plot(dnn_forecast_before_covid, title = "Before COVID-19")
```

```
## Warning in plot.window(xlim, ylim, log, ...): "title" is not a graphical
## parameter

## Warning in title(main = main, xlab = xlab, ylab = ylab, ...): "title" is not a
## graphical parameter

## Warning in axis(1, ...): "title" is not a graphical parameter

## Warning in axis(2, ...): "title" is not a graphical parameter

## Warning in box(...): "title" is not a graphical parameter
```



The performance of the neural network model using the following parameters:

```
accuracy(dnn_forecast_before_covid)
```

```
##                     ME     RMSE      MAE         MPE     MAPE      MASE
## Training set 0.1091264 14.90727 7.595229 -0.01992887 1.374276 0.9958699
##                     ACF1
## Training set -0.02696231
```

```
accuracy(dnn_forecast_after_covid)
```

```
##                     ME     RMSE      MAE         MPE     MAPE     MASE
## Training set 0.1685796 24.41102 11.98748 -0.02022897 1.380511 0.994664
##                     ACF1
## Training set -0.06428384
```

All Models Comparaison

Analysis of all the three models with parameters including RMSE (Root Mean Square Error), MAE (Mean Absolute Error) and MAPE (Mean Absolute Percentage Error):

```
summary_table_before_covid <- data.frame(Model =
                              character(), RMSE = numeric(), MAE = numeric(),
                              MAPE = numeric(), stringsAsFactors = FALSE)
```

Table 1: Summary of Models for data before COVID-19

Model	RMSE	MAE	MAPE
ARIMA	13.08	8.81	1.02
KNN	44.04	33.78	3.17
Neural Network	13.01	8.77	1.02

Table 2: Summary of Models for data after COVID-19

Model	RMSE	MAE	MAPE
ARIMA	16.64	10.44	1.09
KNN	45.97	35.78	3.36
Neural Network	14.71	9.82	1.03

```
summary_table_after_covid <- data.frame(Model =
                            character(), RMSE = numeric(), MAE = numeric(),
                            MAPE = numeric(), stringsAsFactors = FALSE)

summary_table_before_covid[1,] <- list("ARIMA", 13.08, 8.81, 1.02)
summary_table_before_covid[2,] <- list("KNN", 44.04, 33.78, 3.17)
summary_table_before_covid[3,] <- list("Neural Network", 13.01, 8.77, 1.02)

summary_table_after_covid[1,] <- list("ARIMA", 16.64, 10.44, 1.09)
summary_table_after_covid[2,] <- list("KNN", 45.97, 35.78, 3.36)
summary_table_after_covid[3,] <- list("Neural Network", 14.71, 9.82, 1.03)

kable(summary_table_before_covid, caption =
        "Summary of Models for data before COVID-19") %>%
  kable_styling(bootstrap_options =
     c("striped", "hover", "condensed", "responsive"), full_width = F, fixed_thead = T )
```

```
kable(summary_table_after_covid, caption =
      "Summary of Models for data after COVID-19") %>%
 kable_styling(bootstrap_options =
     c("striped", "hover", "condensed", "responsive"), full_width = F, fixed_thead = T )
```

Based on the above summary of model performance parameters, we can conclude that Neural Network Model performs better than the ARIMA and the KNN Model for both the datasets. Therefore, our final model is going to be the Neural Network Model to forecast the stock prices for the next two months.

Conclusion

From the above summary of model performance parameters, we can see that Neural Network Model performs better than the ARIMA and the KNN Model for both the datasets. Hence, we will use the Neural Network Model to forecast the stock prices for the next two months.

Final Model : Before COVID-19

We now forecast the values for March and April using the data till February. Next, we will compare the forecasted price with the actual price to check if there is any significant effect related to COVID-19

```
forecast_during_covid <-
  data.frame("Date" = row.names(tail(AMZN_data_after_covid, n = 40)),
                    "Actual Values" = tail(AMZN_data_after_covid$AMZN.Close, n = 40),
                    "Forecasted Values" = dnn_forecast_before_covid$mean[
    c(-1,-7,-8,-14,-15,-21,-22,-28,-29,-35,-36,-41,-42,-43,-49,-50,-56,-57,-59,-60,-61)])



summary(forecast_during_covid)
```

```
##      Date           Actual.Values  Forecasted.Values
##  Length:40          Min.   :3035   Min.   :1644
##  Class :character   1st Qu.:3126   1st Qu.:1662
##  Mode  :character   Median :3167   Median :1675
##                     Mean   :3170   Mean   :1673
##                     3rd Qu.:3204   3rd Qu.:1688
##                     Max.   :3322   Max.   :1696
```

Based on the table above, we conclude that the actual values of Amazon Stock are almost *Double* and much higher than the forecasted values. This means that there was a reason to make the stock prices go much higher almost close to doubling. This effect of course the COVID-19 effect on the Amazon daily price.

Final Model : After COVID-19

```
forecast_after_covid <- data.frame("Date" =
        (seq.Date(as.Date("2020-04-27"), as.Date("2020-06-30"),by = "day")),
                                  "Price" = dnn_forecast_after_covid$mean )

summary(forecast_after_covid)
```

```
##       Date                 Price
##  Min.   :2020-04-27   Min.   :3130
##  1st Qu.:2020-05-13   1st Qu.:3132
##  Median :2020-05-29   Median :3134
##  Mean   :2020-05-29   Mean   :3134
##  3rd Qu.:2020-06-14   3rd Qu.:3136
##  Max.   :2020-06-30   Max.   :3138
```

From the table, we can conclude that after the 1st wave of COVID-19, the prices of Amazon Stock continue to rise and maintain their high positions suggesting a new high for Amazon stock prices.

Based on this study, we can conclude that COVID-19 did influence the Amazon stock price to clim to new highs and Amazon stock will continue to maintain this new high until new waves of COVID-19 will drive the stock price to much higher levels as seen before during the winter of 2020.