Time Series Analysis of Daily Confirmed COVID-19 Cases in the US

Yuki Yamazaki

PSTAT 174

Spring 2020

Table of Contents

Abstract

"Prediction is very difficult, especially if it's about the future." (Nils Bohr)

Forecasts are made to allow people to see a somewhat accurate future and can be very helpful, especially during a pandemic. During the current COVID-19 global pandemic, many people wonder whether the worst of it is over or if there is more to come. Based on past data, constructing a model to forecast future data is not impossible by utilizing time series techniques. Through this project, I was able to see the future.

Introduction

There were very few people, if any, that predicted the global pandemic currently happening this year. We are all in quarantine and cannot not go through our usual lifestyle anymore. Many sources show all the data of how many people tested positive for COVID-19 and try to predict if the worst of it has passed or not. For this project I will be analyzing the daily confirmed cases of COVID-19 in the US from the end of 2019 to the middle of May of 2020. The *Our World in Data* website I acquired the dataset from had up-to-date data of confirmed cases, death rates and recovery rates for many countries as well as the global data. I will be focusing solely on the number of confirmed cases in the United States of America.

Through this project, I plan to forecast the number of confirmed cases in the US using time series analysis methods. By identifying and estimating the model for the time series dataset and performing a diagnostic checking, at the end I will be able to predict how the number of confirmed cases will look in the future.

Original Time Series Plot

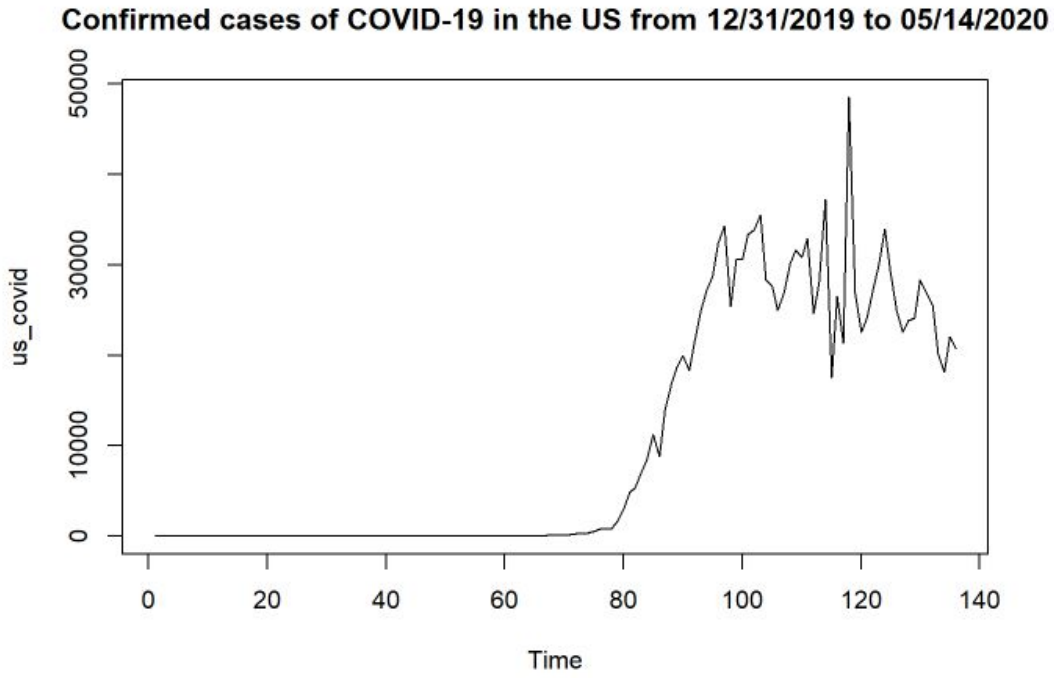

Figure 1: Time Series Plot of Confirmed Cases of COVID-19

As seen in Figure 1, the data starts with zero confirmed cases and does not change for a while until around two months later (the Time axis is in days; 60 days is about two months), when the COVID-19 starts spreading. Afterwards, there is a clear upward trend which makes this time series not stationary. The max number of daily confirmed cases (for now) is reached around Day 120, which is around the end of April. The histogram of the data (on the next page) is badly skewed and asymmetric, meaning non-stationarity. If stationary, the histogram should be symmetric and resemble a Gaussian distribution. The data must be stationary in order to perform time series analysis.

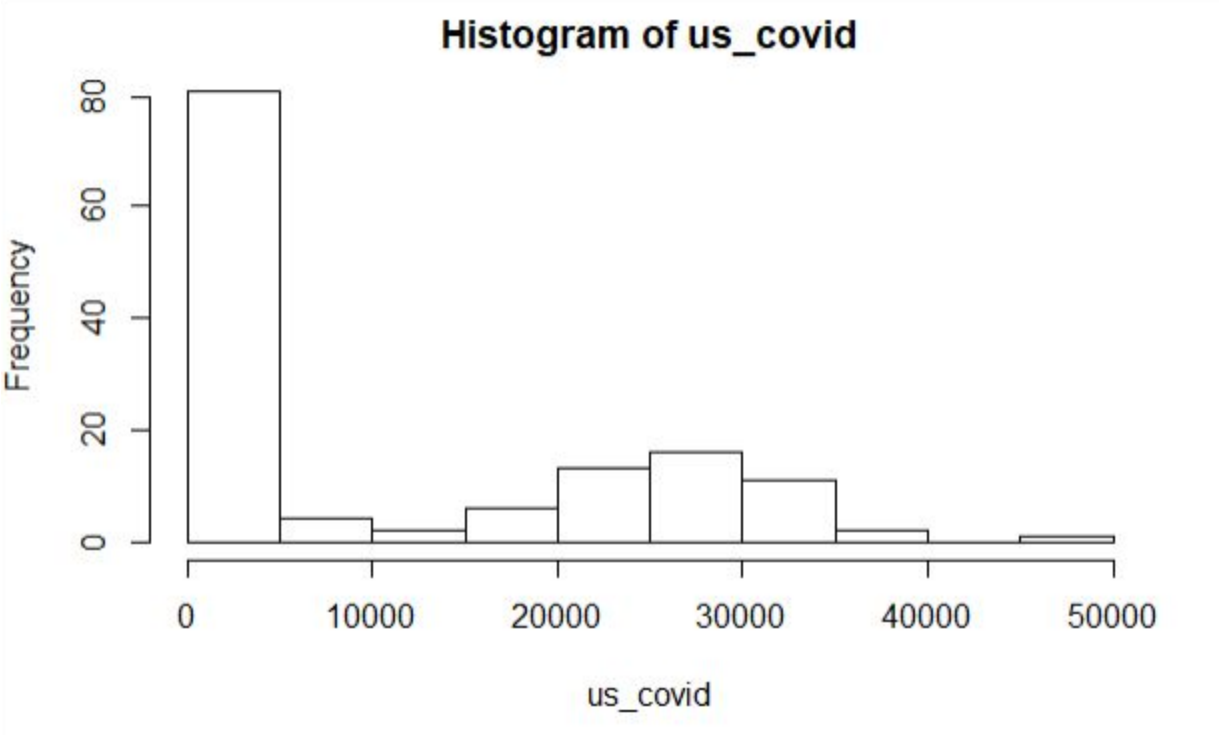

Figure 2: Histogram of Confirmed COVID-19 Cases

Once the virus spread, it swept across the world exponentially. However, there does not seem to be any seasonality. To make this time series stationary, the trend must be removed. The next section will remove any trends and/or seasonality.

Removing Trend and Seasonality

In order to remove the trend, we must take the difference of the series by lag 1 by using the diff() function in R. A transformation is not needed because the variable is mostly stable. Also, since the data contains many zeros, log transformation will not work.

```
# differencing data by lag 1
d = diff(us_covid, 1)
ts.plot(d, main = "De-trended us_covid")
```

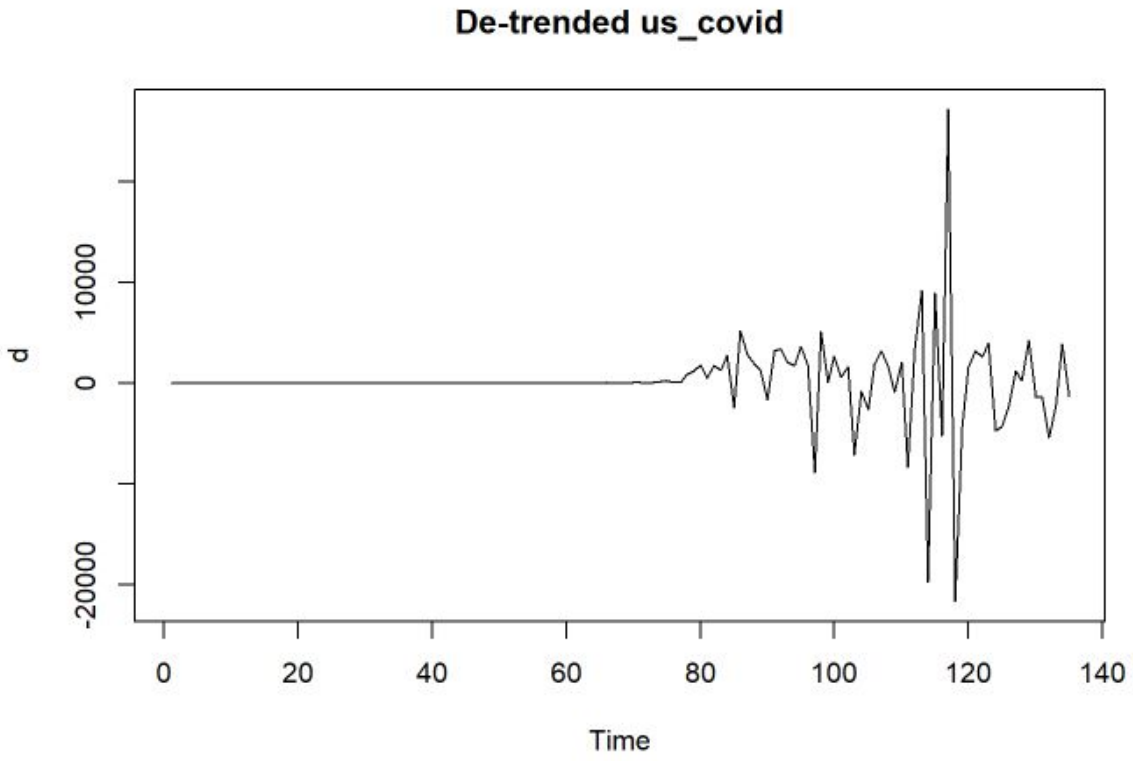

Figure 3: Data after taking the first difference (detrended)

Since seasonality was not found in the original data, differencing by lag 1 should be enough to make the dataset into a stationary time series. The variance is more stable as well. The histogram of the detrended data (on the next page) should be symmetric and almost a Gaussian distribution with mean 0.

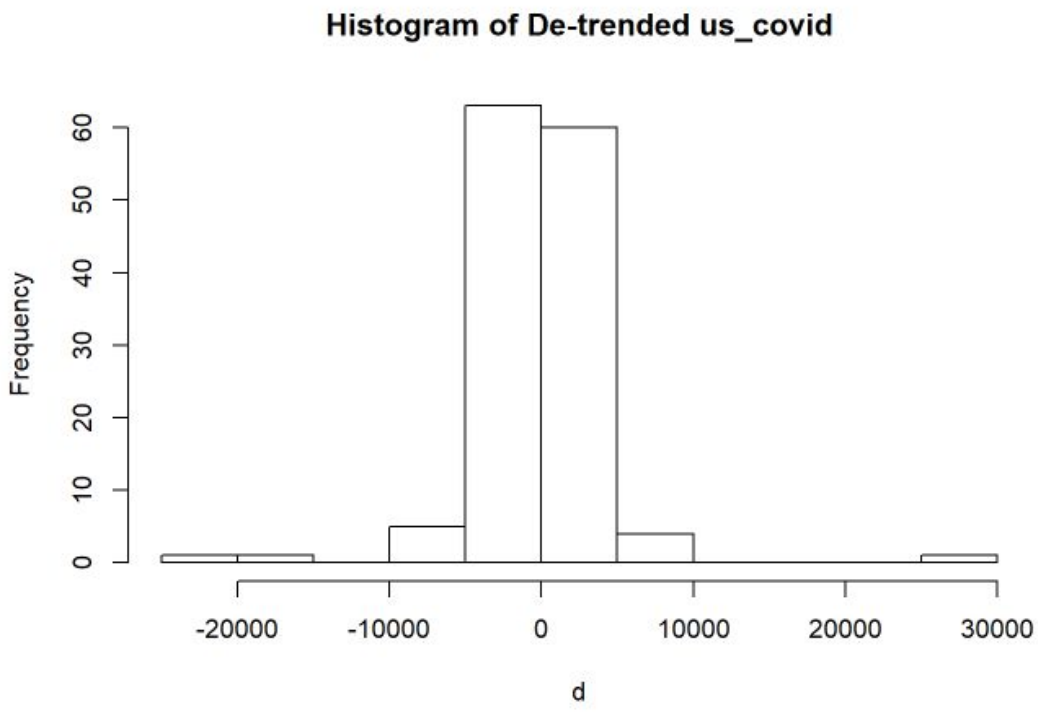

Figure 4: Histogram of detrended data

Using the adf.test() that uses the Augmented Dickey-Fuller Test to test for stationarity, I will confirm that the detrended time series is stationary. The result of the test along with the code follows:

```
adf.test(d)
```

```
##
##  Augmented Dickey-Fuller Test
##
## data:  d
## Dickey-Fuller = -5.634, Lag order = 5, p-value = 0.01
## alternative hypothesis: stationary
```

Figure 5: Result of the Augmented Dickey-Fuller Test

This test runs a hypothesis test for whether there is a unit root in a time series sample with the alternative saying there is not, meaning the series is stationary. With a $\alpha = 0.05$ significance level, the test shows that

the p-value is 0.01, which is less than the significance level meaning the null hypothesis is rejected. Therefore, this detrended time series is confirmed stationary.

Analysis of ACF and PACF

ACF stands for Autocovariance Function, while the PACF stands for Partial ACF. The ACF and PACF of the differenced time series was found through:

```
# plot ACF and PACF
acf(d)
pacf(d)
```

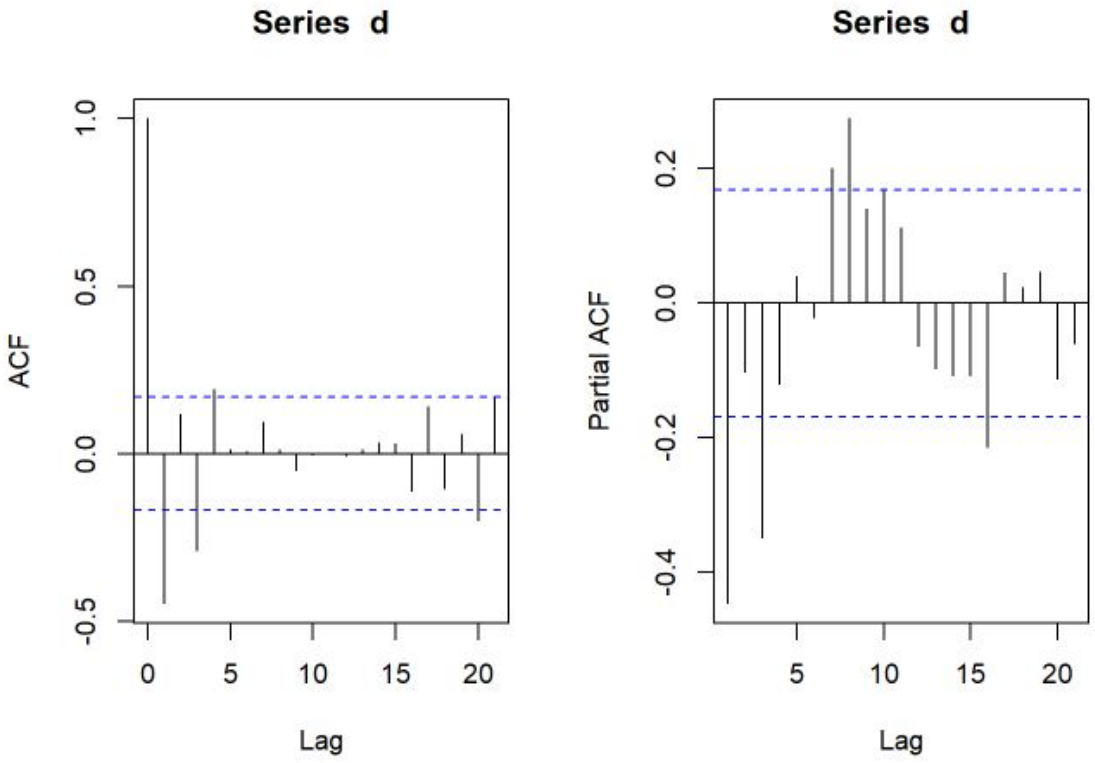

Figure 6: ACF and PACF plots of the differenced time series

From these plots, one can try to figure out the model of the time series based on where the plots tail off or cut off. The ACF decay corresponds to a stationary process, which is a good thing. According to Figure 4, the ACF cuts off after lag 4 in the ACF plot, which is a characteristic of a Moving-average (MA) model.

The PACF does not cut off anywhere in the PACF plot, so I do not think it can be a pure Autoregressive (AR) model. Judging by the ACF plot, I identify this time series as a MA(4) model.

In order to verify my prediction and select the best model, I will calculate the AICc (Akaike Information Criterion, Corrected for Bias) statistic to determine the best model. The AICc is defined as:

$$\text{AICc} = -2 \ln L(\theta_q, \varphi_p, S(\theta_q, \varphi_p) / n) + 2n(p + q + 1) / (n - p - q - 2),$$

where p and q are chosen so the AICc is minimized, which determines the best model.

I will run the auto.arima() function in R to see which model is determined the best by R.

```
auto.arima(us_covid, trace = TRUE)
```

```
##
##  ARIMA(2,1,2) with drift         : 2605.16
##  ARIMA(0,1,0) with drift         : 2637.942
##  ARIMA(1,1,0) with drift         : 2610.206
##  ARIMA(0,1,1) with drift         : 2601.523
##  ARIMA(0,1,0)                    : 2636.065
##  ARIMA(1,1,1) with drift         : 2603.613
##  ARIMA(0,1,2) with drift         : 2603.615
##  ARIMA(1,1,2) with drift         : 2604.051
##  ARIMA(0,1,1)                    : 2600.749
##  ARIMA(1,1,1)                    : 2602.787
##  ARIMA(0,1,2)                    : 2602.79
##  ARIMA(1,1,0)                    : 2608.611
##  ARIMA(1,1,2)                    : 2603.127
##
##  Best model: ARIMA(0,1,1)


## Series: us_covid
## ARIMA(0,1,1)
##
## Coefficients:
##          ma1
##       -0.5507
## s.e.   0.0626
##
## sigma^2 estimated as 13274908:  log likelihood=-1298.33
## AIC=2600.66   AICc=2600.75   BIC=2606.47
```

Figure 7: Result generated by auto.arima() function

According to the results, the model differenced by lag 1 was correct, but my predicted model of MA(4), or ARIMA(0,1,4) was not there. However, I was informed by Professor Feldman that this function "often gives bizarre results," so I will try a different approach just in case.

To confirm the results, I will use the AICc() function to calculate the AICc and find which combination of p and q values will minimize the AICc statistic. The testing will consist of all pairs of p and q values from 0 to 5.

```
# calculate AICc
aiccs <- matrix(NA, nr = 6, nc = 6)
dimnames(aiccs) = list(p = 0:5, q = 0:5)
for (i in 0:5)
      for (j in 0:5)
            aiccs[i+1, j+1] = AICc(arima(us_covid, order = c(i,1,j), method =
"ML"))
aiccs
```

```
##    q
## p          0        1        2        3        4        5
##   0 2636.035 2600.688 2602.697 2604.726 2571.596 2572.912
##   1 2608.550 2602.694 2603.001 2602.183 2572.809 2571.522
##   2 2609.323 2604.707 2604.109 2598.521 2574.786 2573.674
##   3 2594.839 2595.814 2583.135 2580.803 2574.052 2575.668
##   4 2595.313 2596.782 2572.773 2574.994 2576.235 2578.605
##   5 2597.077 2591.869 2574.994 2575.476 2577.976 2579.469
```

Figure 8: AICc Statistics of ARIMA(p,1,q)

According to these results, the models with the smallest AICc statistics were ARIMA(1,1,4), ARIMA(4,1,2), ARIMA(0,1,4), and ARIMA(1,1,5), in that order. My prediction from earlier is considered and the AICc statistics calculated in this test is less than the value from the auto.arima() function, so I will be going with the latter results. From parsimony (increase of the number of parameters), ARIMA(0,1,4) seems to be the best model, but I will check ARIMA(1,1,5) since it has the lowest AICc statistic just in case.

Model A: ARIMA(0,1,4)

Model B: ARIMA(1,1,5)

Fitted Model / Diagnostic Checking

I have two models to fit and test: Model A and Model B. I will estimate the coefficients and fit the model using the arima() function. The number of daily confirmed cases is X_t, t = 1, 2,...., 138.

```
# Model A
arima(us_covid), order = c(0,1,4), method = "ML")
```

```
##
## Call:
## arima(x = us_covid, order = c(0, 1, 4), method = "ML")
##
## Coefficients:
##           ma1     ma2      ma3     ma4
##       -0.7371  0.0374  -0.2527  0.5313
## s.e.   0.0716  0.0950   0.1014  0.0713
##
## sigma^2 estimated as 9995533:  log likelihood = -1280.65,  aic = 2571.29
```

Figure 9: arima() of Model A

The model is:

$$\nabla X_t = (1 - 0.7371_{(0.0716)}B + 0.0374_{(0.0950)}B^2 - 0.2527_{(0.1014)}B^3 + 0.5313_{(0.0713)}B^4)Z_t$$

```
# Model B
arima(us_covid), order = c(1,1,5), method = "ML")
```

```
##
## Call:
## arima(x = us_covid, order = c(1, 1, 5), method = "ML")
##
## Coefficients:
##          ar1      ma1     ma2      ma3     ma4      ma5
##       0.8858  -1.6034  0.6613  -0.2713  0.6684  -0.3420
## s.e.  0.0785   0.1165  0.1607   0.1566  0.1643   0.1121
##
## sigma^2 estimated as 9634470:  log likelihood = -1278.44,  aic = 2570.87
```

Figure 10: arima() of Model B

The model is:

$$(1 - 0.8858_{(0.0785)}B)\nabla X_t = (1 - 1.6034_{(0.1165)}B + 0.6613_{(0.1607)}B^2 - 0.2713_{(0.1566)}B^3 + 0.6684_{(0.1643)}B^4 - 0.3420_{(0.1121)}B^5)Z_t$$

In order to perform a diagnostic check to see how good the fit is, the chosen models must be an invertible and causal ARMA(p,q) models.

Using the plot.roots() function provided, I can verify that the models are indeed invertible and causal. If all the roots of the AR component are outside the unit circle, the model is stationary and causal. If all the roots of the MA component are outside the unit circle, the model is invertible.

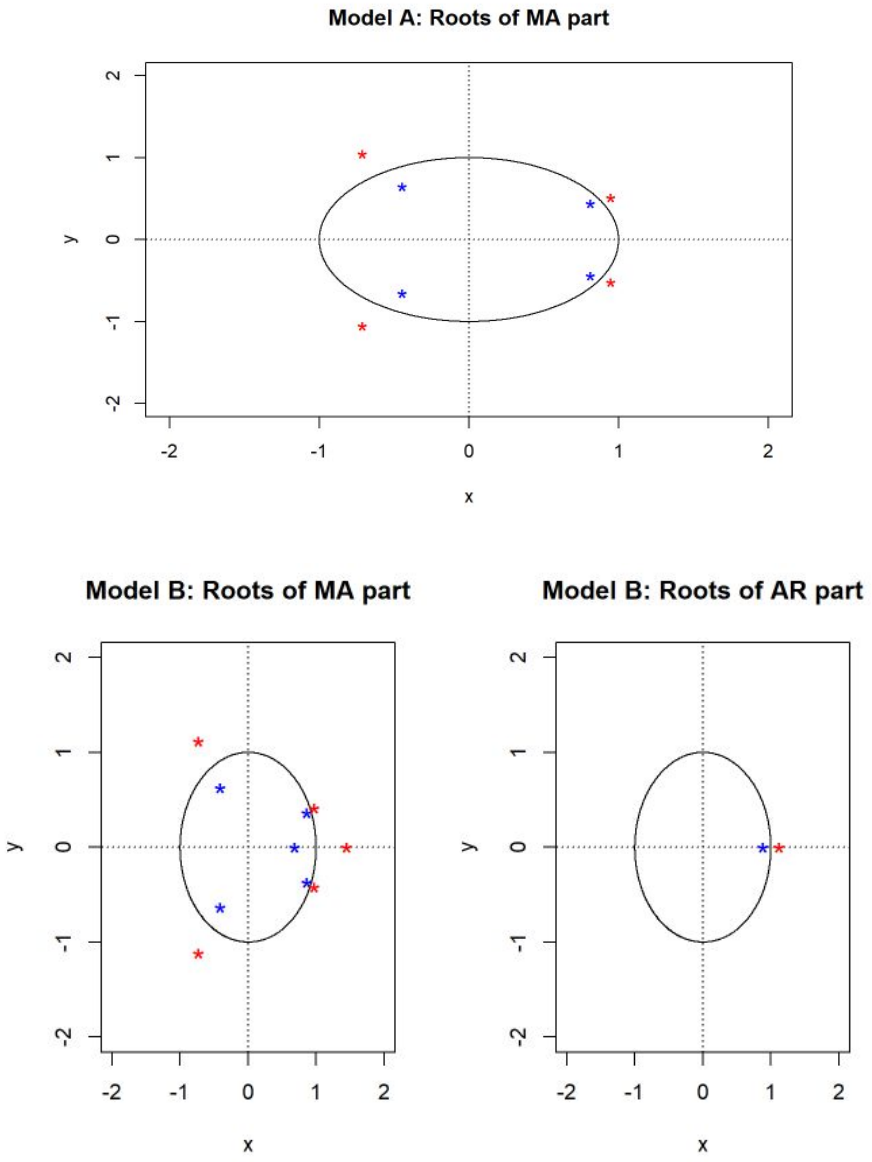

Figure 11: Invertibility and Causality Check for Model A and B

Both models pass the check, so they can go through diagnostic checking. Now we can move on to residual analysis, where the residual of the fitted models are analyzed.

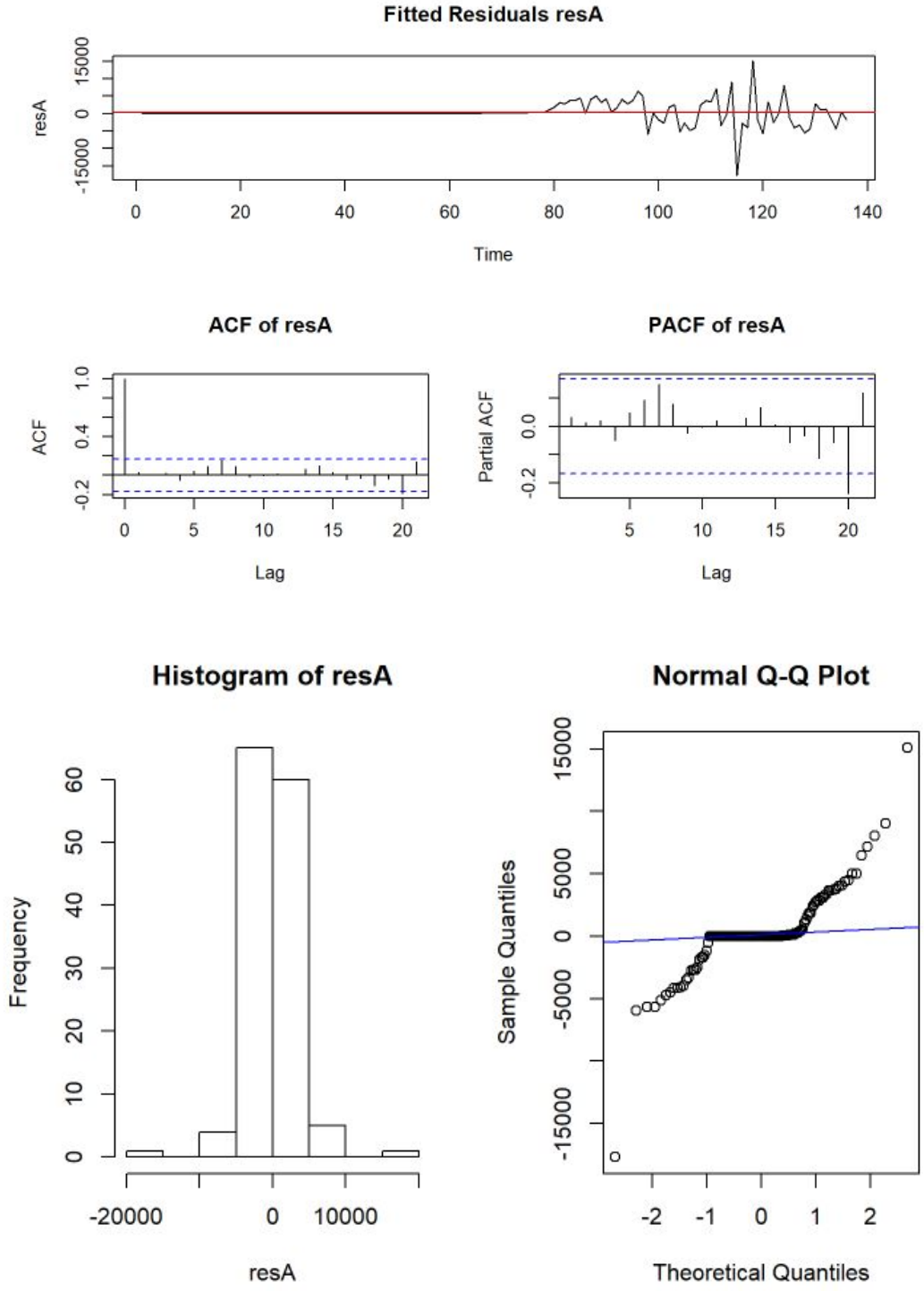

Figure 12: Residual Analysis of Model A

There are no trends, seasonality, or change of variance in the fitted residuals plot with the mean being around zero. The histogram and Q-Q plot seems alright as well. The results may not be perfect because of the many zeros in the data, but I will work with what I can. The ACF and PACF of the residuals seem to be within the confidence intervals, which is good.

```
# independence of residuals for Model A
resA <- residuals(modelA)
Box.test(resA, lag = 12, type = c("Box-Pierce"), fitdf = 4)
Box.test(resA, lag = 12, type = c("Ljung-Box"), fitdf = 4)
Box.test(resA^2, lag = 12, type = c("Ljung-Box"), fitdf = 0)
shapiro.test(resA)
ar(resA, aic = TRUE, order.max = NULL, method = c(“yule-walker”))
```

```
        Box-Pierce test

data:  resA
X-squared = 6.3941, df = 8, p-value = 0.6032


        Box-Ljung test

data:  resA
X-squared = 6.8206, df = 8, p-value = 0.5561


        Box-Ljung test

data:  resA^2
X-squared = 54.748, df = 12, p-value = 0.8692


        Shapiro-Wilk normality test

data:  resA
W = 0.76454, p-value = 1.7e-13


Call:
ar(x = resA, aic = TRUE, order.max = NULL, method = c("yule-walker"))


Order selected 0  sigma^2 estimated as  9932727
```

Figure 13: Test for Independence of Residuals of Model A

The p-value is larger than 0.05 for most of the tests, except the Shapiro-Wilk normality test. It can possibly be due to the multiple number of zeros in the data, but at least most of the p-values are greater

than 0.05. The fitted residuals are estimated as an AR(0) model, which means it is a white noise. The Shapiro-Wilk test is concerning, but everything else seems to pass the diagnostic checking. If Model B does not have better results, I will use Model A for forecasting. Now, moving on to Model B.

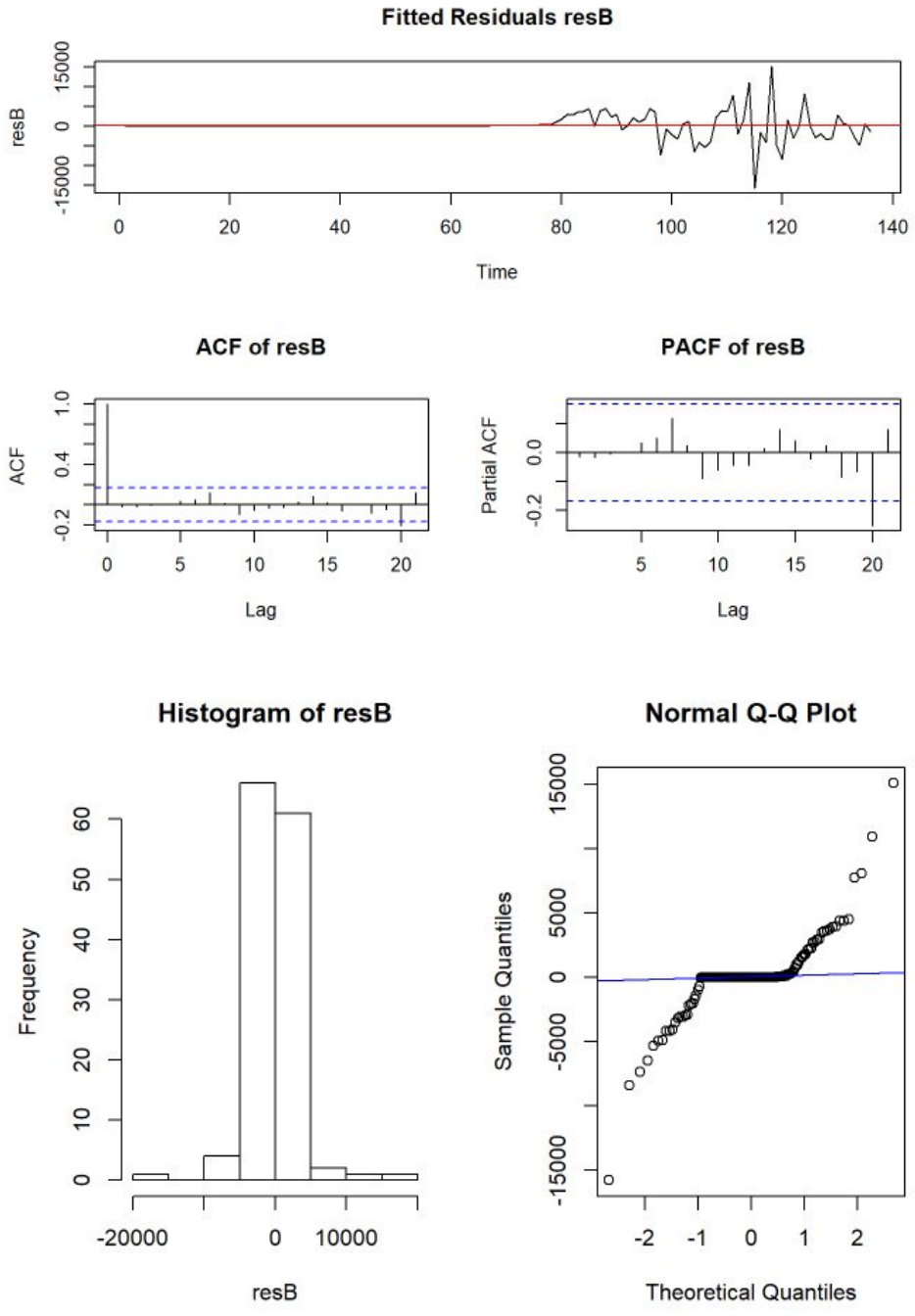

Figure 14: Residual Analysis of Model B

Like Model A, there are no trends, seasonality, or change of variance in the fitted residuals plot with the mean being around zero. The histogram and Q-Q plot seems satisfactory. Similarly, the ACF and PACF of the residuals seem to be within the confidence intervals. There does not seem to be a significant difference from the Model A residual analysis results.

```
resB <- residuals(modelB)
Box.test(resB, lag = 12, type = c("Box-Pierce"), fitdf = 4)
Box.test(resB, lag = 12, type = c("Ljung-Box"), fitdf = 4)
Box.test(resB^2, lag = 12, type = c("Ljung-Box"), fitdf = 0)
shapiro.test(resB)
ar(resB, aic = TRUE, order.max = NULL, method = c(“yule-walker”))
```

```
        Box-Pierce test

data:  resB
X-squared = 4.346, df = 6, p-value = 0.63


        Box-Ljung test

data:  resB
X-squared = 4.681, df = 6, p-value = 0.5853


        Box-Ljung test

data:  resB^2
X-squared = 75.502, df = 12, p-value = 2.953e-11


        Shapiro-Wilk normality test

data:  resB
W = 0.74512, p-value = 4.45e-14


Call:
ar(x = resA, aic = TRUE, order.max = NULL, method = c("yule-walker"))
```

Figure 15: Test for Independence of Residuals of Model B

Although the residual is fitted as a white noise model, two of the four tests have p-values less than 0.05. I conclude that Model B does not pass the diagnostic checking and will use Model A for forecasting, which follows an ARIMA(0,1,4) model and matches the predicted model based on the ACF/PACF plots:

$$\text{Model A: } \nabla X_t = (1 - 0.7371B + 0.0374B^2 - 0.2527B^3 + 0.5313B^4)Z_t$$

Forecasting with Confidence Intervals

In order to confirm the forecasts, I got the number of confirmed COVID-19 cases from May 15, 2020 (the day after the last day in the dataset) to May 22, 2020 (total of eight days). The forecast for the eight days are:

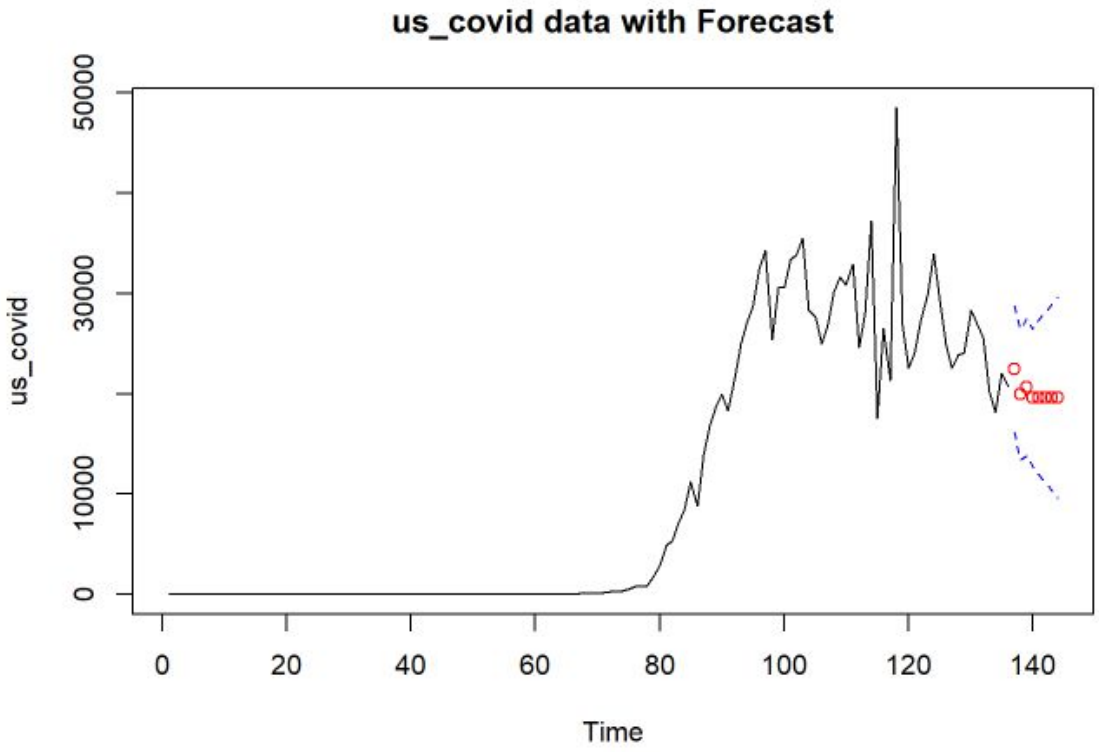

Figure 16: Forecasted Data

The prediction interval is calculated by:

$$P_n X_{n+h} \pm \Phi_{1-\alpha/2}\, \sigma_n(h),$$

where $P_n X_{n+h}$ is the predicted value, Φ is the z-score (for 95% prediction interval ($\alpha = 0.05$), $\Phi = 1.96$), and σ_n is the standard error. All the forecasts seem to be within the prediction interval. Now we can compare it to the actual data of May 15 to May 22. The plot of showing both will be on the next page.

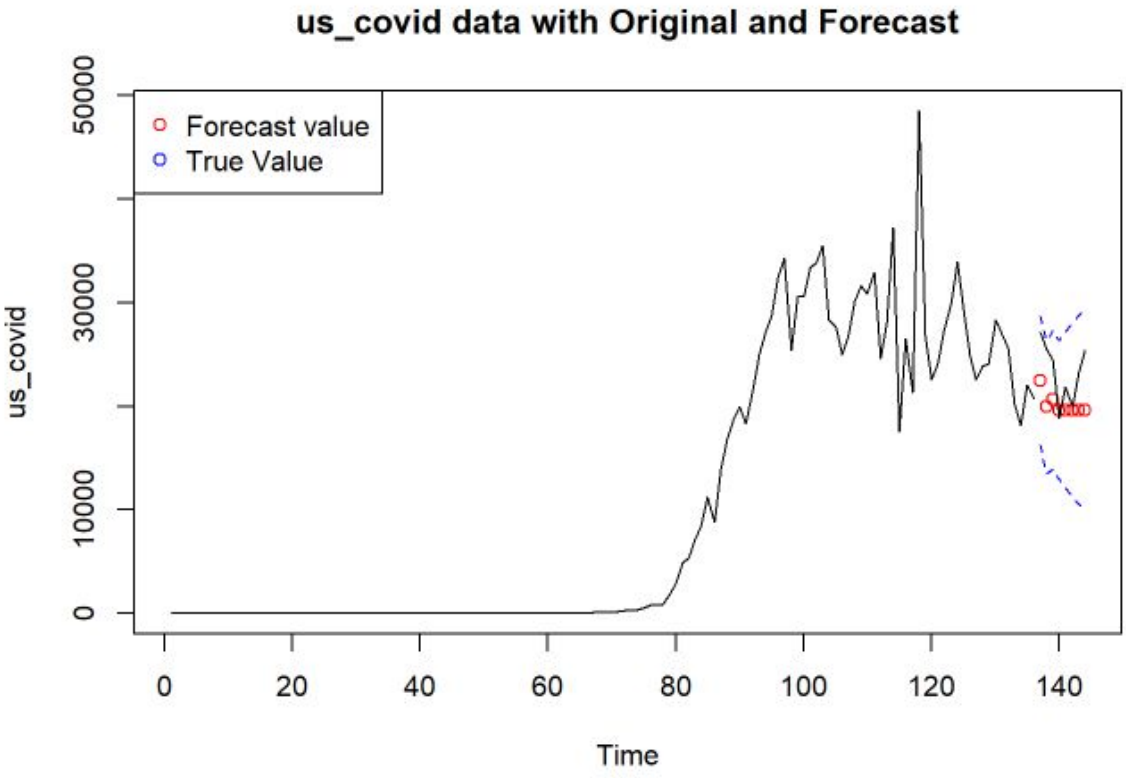

Figure 17: Original and Forecast Data

The original series was barely in the 95% prediction interval, but it is hard to say the forecast was accurate. Some days got close, but the other days are quite far off. Further forecasting to predict how the pandemic will go will be difficult with this model.

Conclusion

While I was not able to get the best forecast, it was not significantly far off. The model suggests that the number of confirmed cases will not change much. The actual data moved up and down, but overall the number of confirmed cases did not change significantly. The amount of zero cases in the data may have skewed some of the analysis, which may have caused the model not passing one of the diagnostic checks. As a result, Model A ($\nabla X_t = (1 - 0.7371B + 0.0374B^2 - 0.2527B^3 + 0.5313B^4)Z_t$) was the best one I could have used. Judging by the ACF/PACF plots and the AICc statistics, the ARIMA(0,1,4) model selected seemed suitable and appropriate. Though the outcome of this project did not predict the cases as accurately as I wanted to, it is definitely a start. As much as predicting the track of a pandemic is certainly difficult, but there can be improvements and revisions I can make based off of this work to make more accurate predictions. I truly hope the quarantine will end soon and the world will be cured of COVID-19. I miss being on campus at UC Santa Barbara and hope that everyone can come back in the fall. This PSTAT 174 class was challenging, especially during these times, but it was very helpful and I am glad to have taken it. I would like to acknowledge Dr. Raya Feldman for giving me advice on this project as well as teaching the course and providing material under these stressful conditions working online.

Reference

European Centre for Disease Prevention and Control. “Daily Confirmed COVID-19 Cases.” May 2020. “https://ourworldindata.org/grapher/daily-cases-covid-19?country=USA+OWID_WRL”

Appendix

```
library(tsdl)
library(forecast)
library(qpcR)

# reads the data of 12/31/2019 - 05/14/2020
us_covid <- scan("c://Users/yyama/OneDrive/Documents/PSTAT
174/US_covid19_confirmed_cases.csv")
ts.plot(us_covid, main = "Confirmed cases of COVID-19 in the US from
12/31/2019 to 05/14/2020")
hist(us_covid)

# differencing data by lag 1
d = diff(us_covid, 1)
ts.plot(d, main = "De-trended us_covid")
hist(d, main = "Histogram of De-trended us_covid")

library(tseries)
# Augmented Dickey-Fuller Test
adf.test(d)

op <- par(mfrow = c(1,2))
# ACF and PACF plots of detrended data
acf(d)
pacf(d)
par(op)

# find best model
# auto.arima() is unreliable
auto.arima(us_covid, trace = TRUE)
# calculates AICc
aiccs <- matrix(NA, nr = 6, nc = 6)
dimnames(aiccs) = list(p = 0:5, q = 0:5)
for (i in 0:5)
  for (j in 0:5)
  {
      aiccs[i+1,j+1] = AICc(arima(us_covid, order = c(i,1,j), method = "ML"))
  }
aiccs
# (1,4), (4,2), (0,4), (1,5)

# fitting Model A
arima(us_covid, order = c(0,1,4), method = "ML")
# fitting Model B
arima(us_covid, order = c(1,1,5), method = "ML")
```

```
# check for causality and invertibility
source("plot.roots.R")
# Model A
plot.roots(NULL, polyroot(c(1,-0.7371,0.0374,-0.2527,0.5313)), main = "Model
A: Roots of MA part") # invertible
# Model B
op <- par(mfrow = c(1,2))
plot.roots(NULL, polyroot(c(1,-1.6034,0.6613,-0.2713,0.6684,-0.3420)), main =
"Model B: Roots of MA part") # invertible
plot.roots(NULL, polyroot(c(1,-0.8858)), main = "Model B: Roots of AR part")
# stationary, causal
op <- par(mfrow = c(1,2))

# Diagnostic checking of Model A
modelA <- arima(us_covid, order = c(0,1,4))
resA <- residuals(modelA)
layout(matrix(c(1,1,2,3),2,2,byrow=T))
ts.plot(resA, main = "Fitted Residuals resA")
abline(h = mean(resA), col = "red")
acf(resA, main = "ACF of resA")
pacf(resA, main = "PACF of resA")
opar <- par(no.readonly = T)
par(mfrow=c(1,2))
hist(resA)
qqnorm(resA)
qqline(resA, col = "blue")
Box.test(resA, lag = 12, type = c("Box-Pierce"), fitdf = 4)
Box.test(resA, lag = 12, type = c("Ljung-Box"), fitdf = 4)
Box.test(resA^2, lag = 12, type = c("Ljung-Box"), fitdf = 0)
shapiro.test(resA)
ar(resA, aic = TRUE, order.max = NULL, method = c("yule-walker"))

# Diagnostic checking of Model B
modelB <- arima(us_covid, order = c(1,1,5))
resB <- residuals(modelB)
layout(matrix(c(1,1,2,3),2,2,byrow=T))
ts.plot(resB, main = "Fitted Residuals resB")
abline(h = mean(resB), col = "red")
acf(resB, main = "ACF of resB")
pacf(resB, main = "PACF of resB")
opar <- par(no.readonly = T)
par(mfrow=c(1,2))
hist(resB)
qqnorm(resB)
qqline(resB, col = "blue")
Box.test(resB, lag = 12, type = c("Box-Pierce"), fitdf = 6)
Box.test(resB, lag = 12, type = c("Ljung-Box"), fitdf = 6)
Box.test(resB^2, lag = 12, type = c("Ljung-Box"), fitdf = 0)
```

```
shapiro.test(resB)
ar(resA, aic = TRUE, order.max = NULL, method = c("yule-walker"))

# forecasting
pred.tr <- predict(modelA, n.ahead = 8)
upp <- pred.tr$pred + 1.96 * pred.tr$se
low <- pred.tr$pred - 1.96 * pred.tr$se
ts.plot(us_covid, xlim = c(1, length(us_covid) + 8), main = "us_covid data
with Forecast")
lines(upp, col = "blue", lty = "dashed")
lines(low, col = "blue", lty = "dashed")
points((length(us_covid) + 1):(length(us_covid) + 8), pred.tr$pred, col =
"red")

# confirmed cases from 05/15 - 05/22
fut <- c(27143,25508,24487,18873,21841,19970,23285,25434)
ts.plot(us_covid, xlim = c(1, 144), main = "us_covid data with Original and
Forecast")
points((length(us_covid) + 1):(length(us_covid) + 8), pred.tr$pred, col =
"red")
lines((length(us_covid) + 1):(length(us_covid) + 8), fut, col = "black")
lines(upp, col = "blue", lty = "dashed")
lines(low, col = "blue", lty = "dashed")
legend("topleft", pch = 1,col=c("red","blue"),
legend=c("Forecast value", "True Value"))
```