Time Series Analysis on Post Graduate Unemployment Rate

PSTAT 174

Spring 2020

Justin Hsiang

Abstract

With the recent effects of COVID-19 on the world, one of the larger concerns among people is the skyrocketing unemployment rate. I specifically decided to focus on the unemployment of future college graduates as this is the age group that I personally fall into and will have to worry about the most as it is personally becoming a strong source of anxiety for new college grads all over the world. In my time series analysis, I want to be able to forecast what the unemployment would look like for the following 2 years which is around the time that people in my year will be heading into the job market. I specifically chose college graduates as it makes the situation a lot more relatable for what the future holds for my peers and me.

The steps I will be taking to forecast this data are: exploratory analysis, model identification, diagnostic testing and forecasting. The data I had forecasted was that the unemployment would go down but remain higher than originally. In my conclusion, I continue to see unemployment rates correcting to even lower rates; however, for the next few years it seems like unemployment will remain relatively high in general.

1 Introduction

In the recent months, a global pandemic has struck the world and one of the biggest consequences in correspondence was the skyrocketing unemployment rate. Prior to this time, unemployment was seen to be going in a relatively downwards direction since 2010. With this, I would like to forecast what unemployment will be looking like in the coming years post-pandemic. Many people around my age are currently worried about the job market and will be entering it soon whether it be this year or next year. I would like to conduct a time series analysis to see how the unemployment rate will recover after the pandemic and if it shows any promise to the college population.

To begin with, I used the dataset of college graduates and began by plotting the data. The first thing I noticed was that there was a clear outlier in April of 2020 which was representative of the effects of COVID-19 causing unemployment rates to skyrocket. I decided to leave this datapoint in just because my ultimate goal is to see how the data can recover post-pandemic while also hoping that the outlier will influence the forecasting to still make it somewhat accurate. The four steps of my time series analysis include: exploratory analysis, model identification, diagnostic testing and forecasting. Now the ultimate goal for each of these steps are the following.

1. **Exploratory Analysis:** Looking for trends and cyclical patterns in the data to make necessary transformations. The goal of this is to get data that averages out around 0 and has a pretty constant variance (imagine how much a value differs from the average of 0). I was able to do this through a log-transformation of the dataset as well as getting rid of seasonal and underlying trends occurring throughout the data.
2. **Model Identification:** Analyze autocorrelation and partial autocorrelation graphs to find a specific statistical model in the form of an equation that will help us forecast the data. Autocorrelation is determining the correlation between two time values with the time in between being called "lag". As for partial autocorrelation, it measures the same thing but with the values in between taken into account for. In this specific instance, the model we will be trying to find coefficients for is a SARIMA(p,d,q)x(P,D,Q)[12]. After model candidates are discovered, we will be looking for those with the lowest AIC value and continue through diagnostic testing with those models. My model was discovered to be SARIMA(1,1,1)x(2,1,1)[12]
3. **Diagnostic Testing:** Testing the sample residuals(difference between a measured value and predicted value) of the data with tests such as Shapiro-Wilkes Test that tests for normality(bell-shaped curve an assumption we

need to meet), and the Portmaneau tests(Ljung-Box, Box-Pierce and McLeod-Li) that will test for independence.

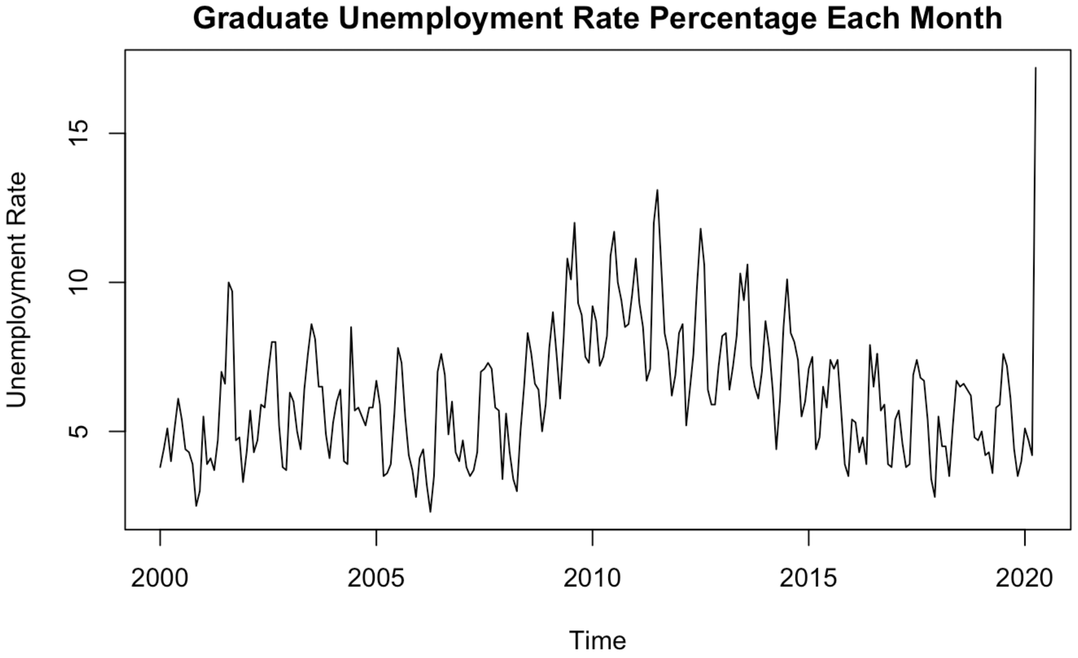
4. **Forecasting:** Using a training set of 216 observations or 18 years to test the model on and then using then attempting to predict the last 28 observations(up until the present) and comparing them to the present true values. Then I will use the model I chose to forecast the next 24 observations or next 2 years.

The results I received were very similar to my understanding of how things would work. Essentially, the productions showed that the unemployment rate would fall back down; however, it would still be higher than the original unemployment rate by a solid 10%. This is synonymous with my suspicions of the job market at first as the unemployment rate will definitely not be able to go down to as low as it was from the current loss as there are already many companies completely going out of business. The entry for barrier into the job market for college graduates with bachelor's degrees in the next coming years will be relatively high compared to recent years.

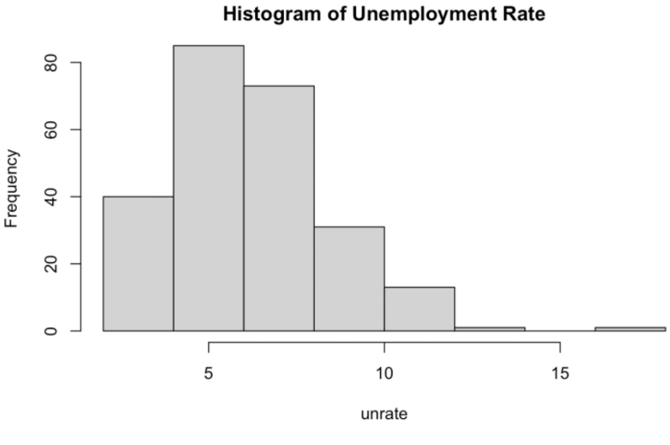
The dataset I used to address this specific issue is a dataset from the Federal Reserve Bank that shows the unemployment rate of college students with bachelor's degrees in the age range of 20 to 24 years old from the years January 1, 2000 to the present. I used RStudio to analyze the data and conduct the time series analysis.

2.1 Exploratory Analysis

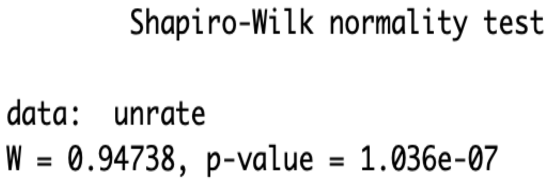
First, we will view the time series plot of the unemployment rate in college graduates with bachelor's degrees (aged 20-24) from January 1, 2000 to April 1, 2020.



In terms of specific trends, there are no noticeable ones to begin with.
I will lookout for normality by plotting a histogram of the data and look for a bell-shape curve.



This doesn't really follow the bell-shape curve and just to be sure, I conducted a Shapiro-Wilk test in R that will tell us if there is sufficient evidence to say that the graph is not normal - implying the need for further transformation. The null hypothesis for a Shapiro-Wilk test is that the data is not normal. Upon conducting the Shapiro-Wilk test we get:

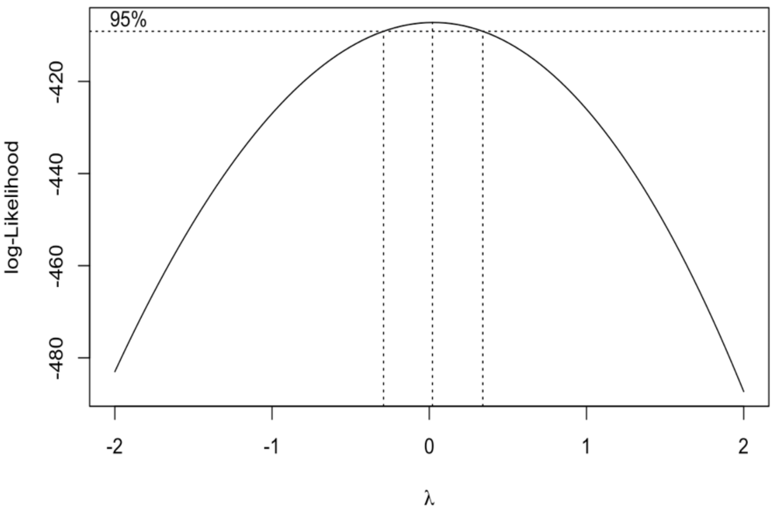


```
        Shapiro-Wilk normality test

data:  unrate
W = 0.94738, p-value = 1.036e-07
```

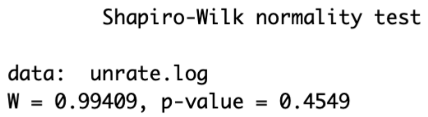
The p-value of 1.036e-07 is less than the alpha value of .05. What this means is that we can conclude that the data is not normal with 95% confidence.

Due to this development, I will conduct a Box-Cox test to tell me what sort of transformation I will make. Plotting the Box-Cox plot I get this graphic:

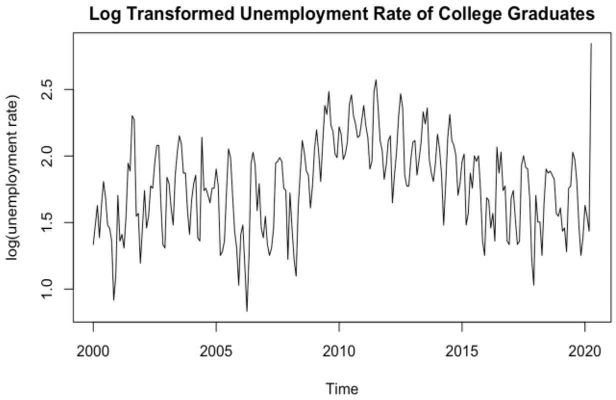
From this plot, it shows that a value of lambda = 0 is best within the 95% confidence interval. This will essentially mean that I will conduct a log-transform on the data by taking the log() of the entire dataset. Hopefully, this will give me a normal dataset. After doing that I get the new time series and will conduct another Shapiro-Wilk test to test for normality.

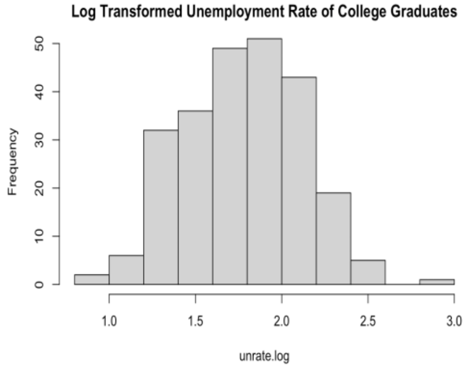


Below is the Shapiro-Wilk test, the log-transformed time series as well as the histogram.



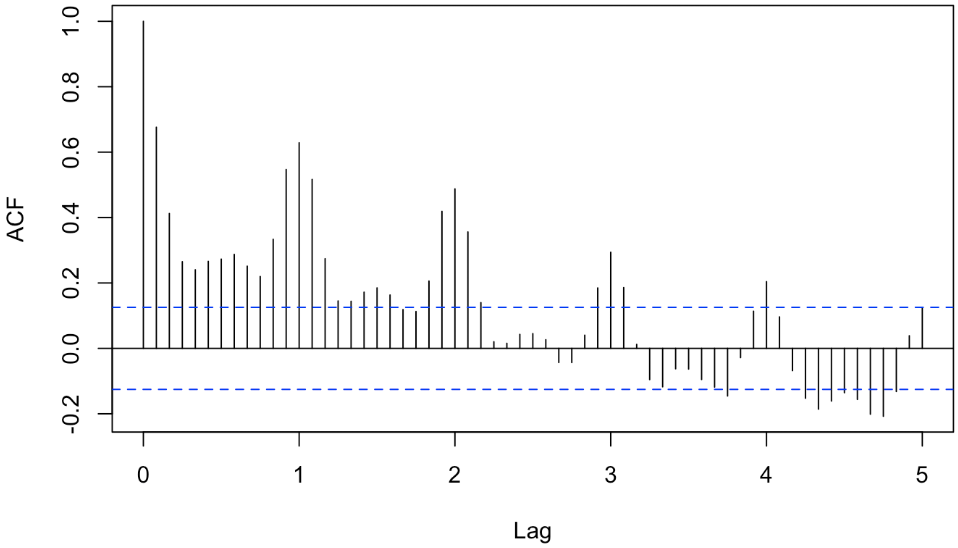
```
        Shapiro-Wilk normality test

data:  unrate.log
W = 0.99409, p-value = 0.4549
```



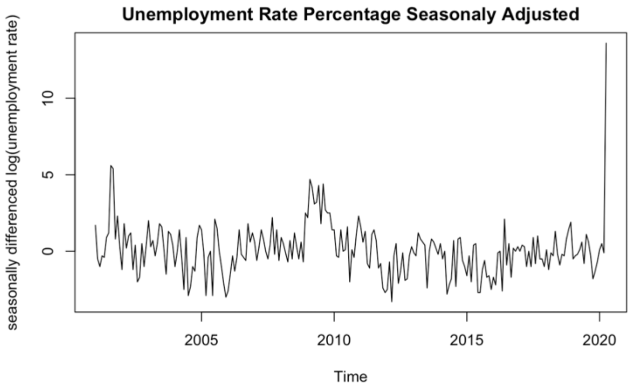


From the histogram, it is already much more bell-shaped than the original showing that the data is relatively normal. With the Shapiro Wilks test, the p-value is .4549 which is greater than the alpha = .05. This means that there is not sufficient evidence to say that the log transformed time series is not normal.

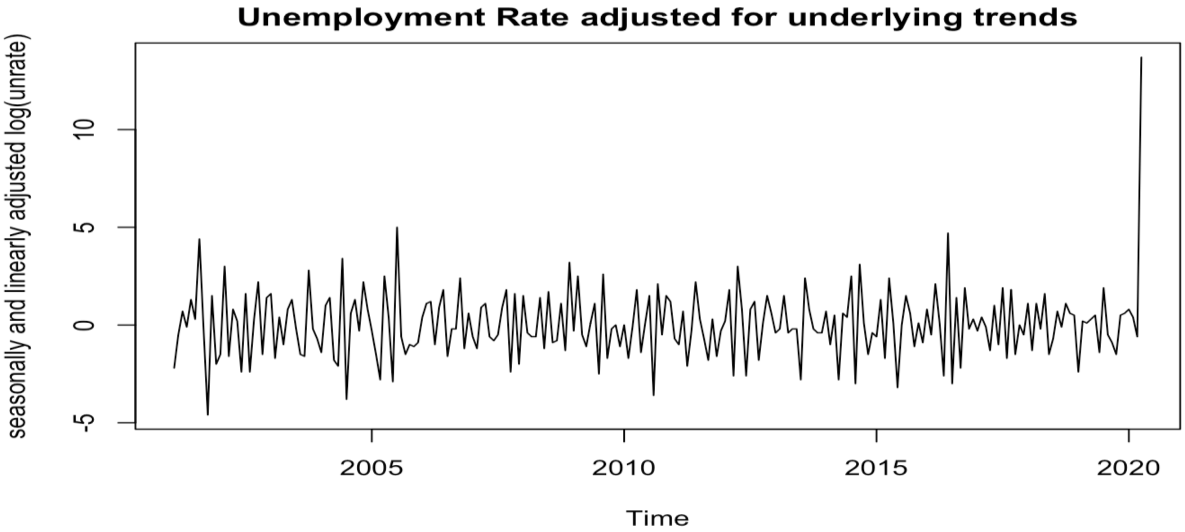
Now, I will plot the ACF of the log transformed data to conduct analysis for seasonality.



Through the ACF, a cyclical pattern can be observed. In between the numbers 0 and 1 there are 12 lines(lags) that represent the months in a year. With this observation and the pattern being shown on a yearly basis, there is a clear seasonal trend being observed. To get rid of this seasonal trend, we will difference the data at a lag of 12.



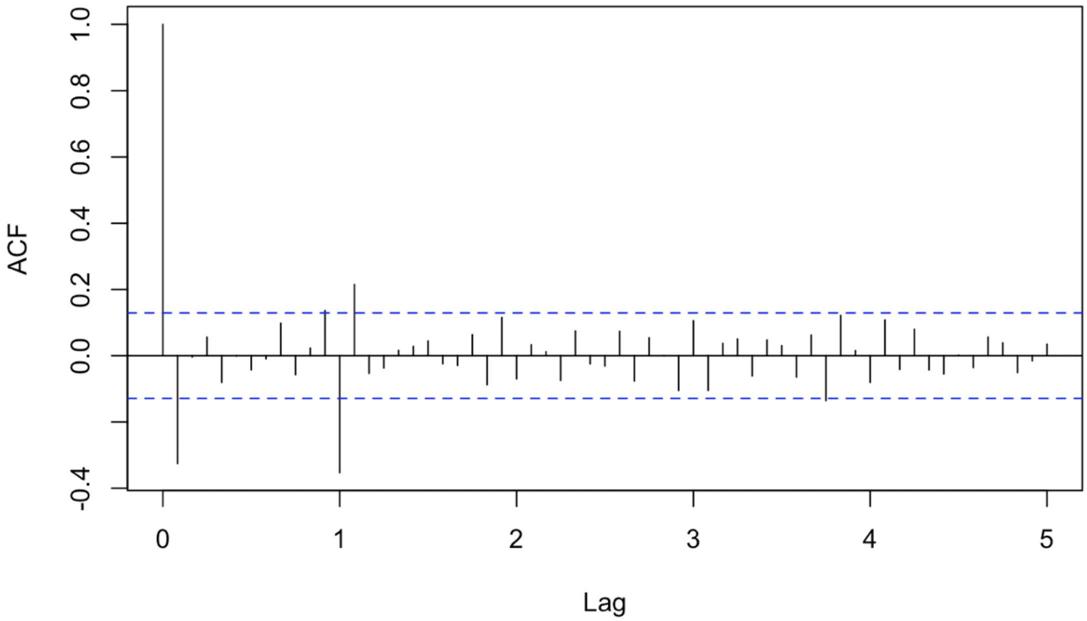
This shows the data to be relatively flatter and almost stationary. Meaning that the mean of the data seems to almost be 0 however there still seem to be a downward trend. To account for this, I will difference again at lag 1 just to get rid of any underlying linear trends.

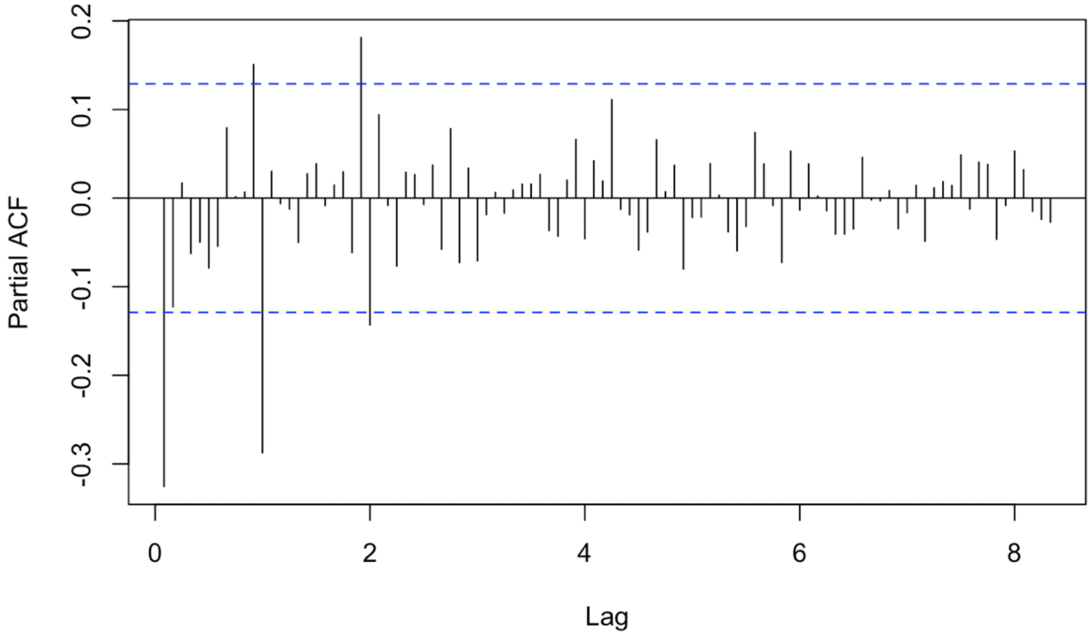


This plot now shows a much more stationary plot that could have a mean of 0 besides the outlier due to COVID-19. This will be the basis of our model identification process now.

2.2 Model Identification

I have decided to use a SARIMA(p,d,q)x(P,D,Q)[12] model as we have seasonally adjusted are data and SARIMA stands for Seasonal Auto-regressive Integrated Moving Average Model. First, I will plot the ACF and the PACF of the de-seasonalized, de-trended and log transformed unemployment rate data.





SARIMA(p,d,q)(P,D,Q)[12] model is a seasonally autoregressive intergrated moving average model which supports AR components and MA components with a seasonal component as well. To begin with,we will model the seaonal portion of the SARIMA model (P,D,Q).

We applied one seasonal differencing so D=1 at lag s = 12. The ACF shows a strong peak at h = 1s and does not seem to show any stronger peaks. So a good choice for the MA part would be Q=1. As for the AR part, the PACF shows strong peaks at h = 1s, 2s with the PACF approaching zero as lag increases. This would mean a good choice for P would be 2. Now too model the non-seasonal part(p, d, q). We will focus on the lags between each lag s which will be considered the within season lags. We applied one differencing to remove any linear trend so d = 1. The ACF seems to tall off at lag = 1 so a good choice for the MA part would be q = 0 or q = 1. For the AR portion, the within lag cuts off once at 1. A good choice for p would be p = 1.

The potential models that I have identified are:

SARIMA(1,1,0) x (2,1,1)[12]
SARIMA(1,1,1) x (2,1,1)[12]

It seems like I am confident with most of my coefficients on the seasonal side and know that d=1 for the non-seasonal portion. I then went and tested other values of p and q in correspondence to the data to see which would give me the lowest AICc value on R. Essentially, the lowest AICc value amongst the model candidates will give me the best fit model.

The two lowest AICc values I found were :

SARIMA(0,1,3) x (2,1,1)[12] with an AICc value of -64.021

SARIMA(1,1,1) x (2,1,1)[12] with an AICc value of -65.740

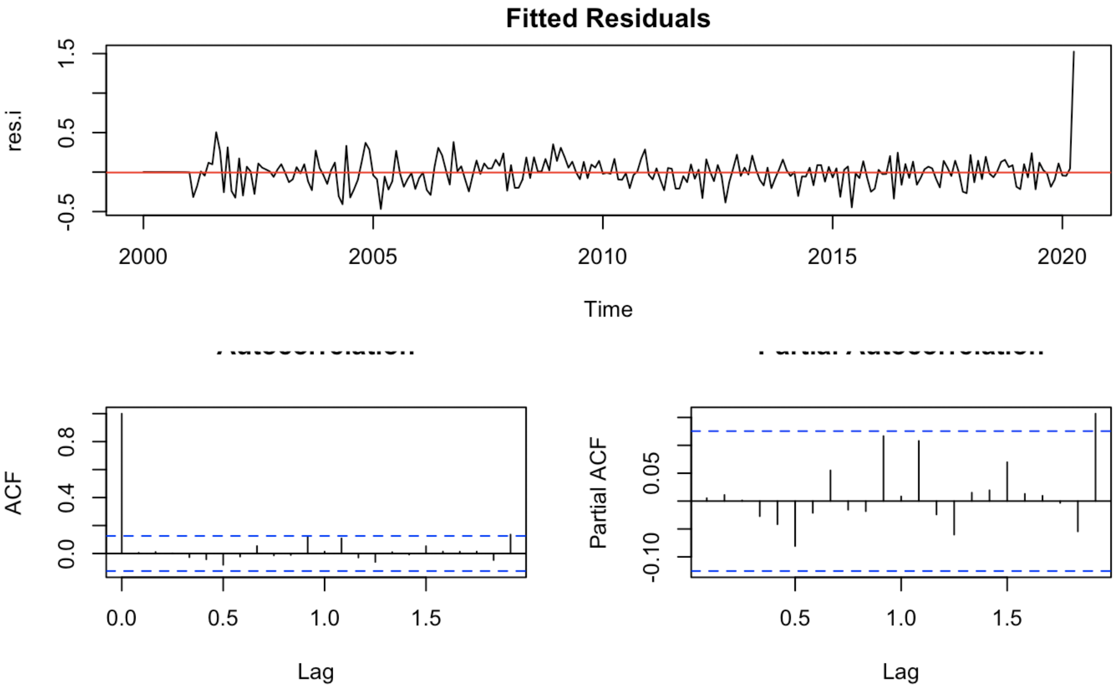
I will proceed with diagnotic testing for these two models.

2.3 Diagnostic Testing:

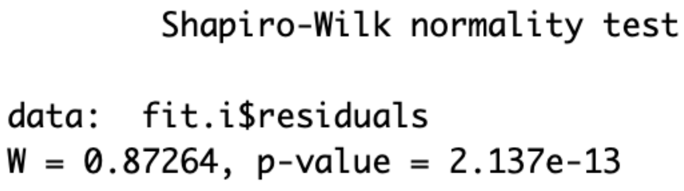
For the first model:

SARIMA(0,1,3) x (2,1,1)[12]

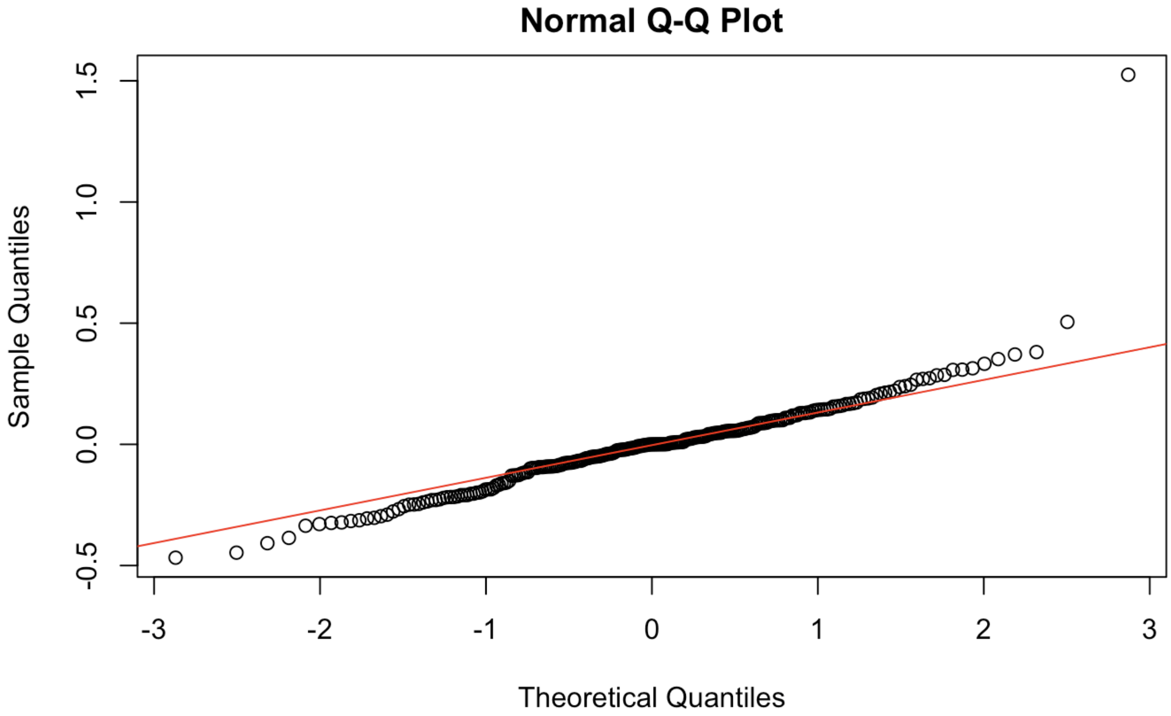
I will first begin by plotting the fitted residuals, the ACF of the residuals and the PACF of the residuals for this model.



The fitted residuals seem to have a mean of 0 with no distinct trend. It seems to follow the white-noise assumption based on having no trend with constant variance and mean of 0. Next, to test the residuals for normality, I will plot a QQ-plot of the fitted residuals as well as conduct a Shapiro-Wilks test for normality.

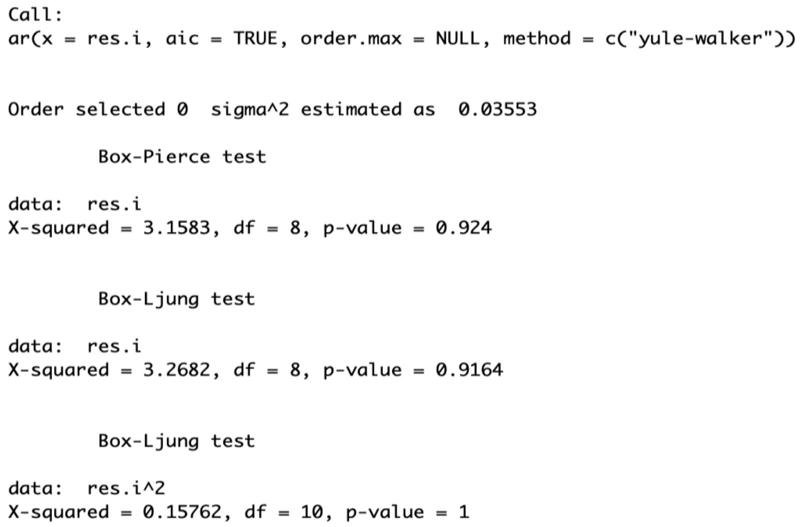


```
        Shapiro-Wilk normality test

data:  fit.i$residuals
W = 0.87264, p-value = 2.137e-13
```



Since the points on the line generally follow the red diagonal, the fitted residuals seem to be normal. In addition with the Shapiro-Wilks test it is clear that the p-value is less than alpha = .05. Meaning that there is sufficient evidence to say that the fitted residuals are normal.

Next, I will conduct a Yule-Walker test for White-Noise as well as Portmaneu test for independence(serial correlation).



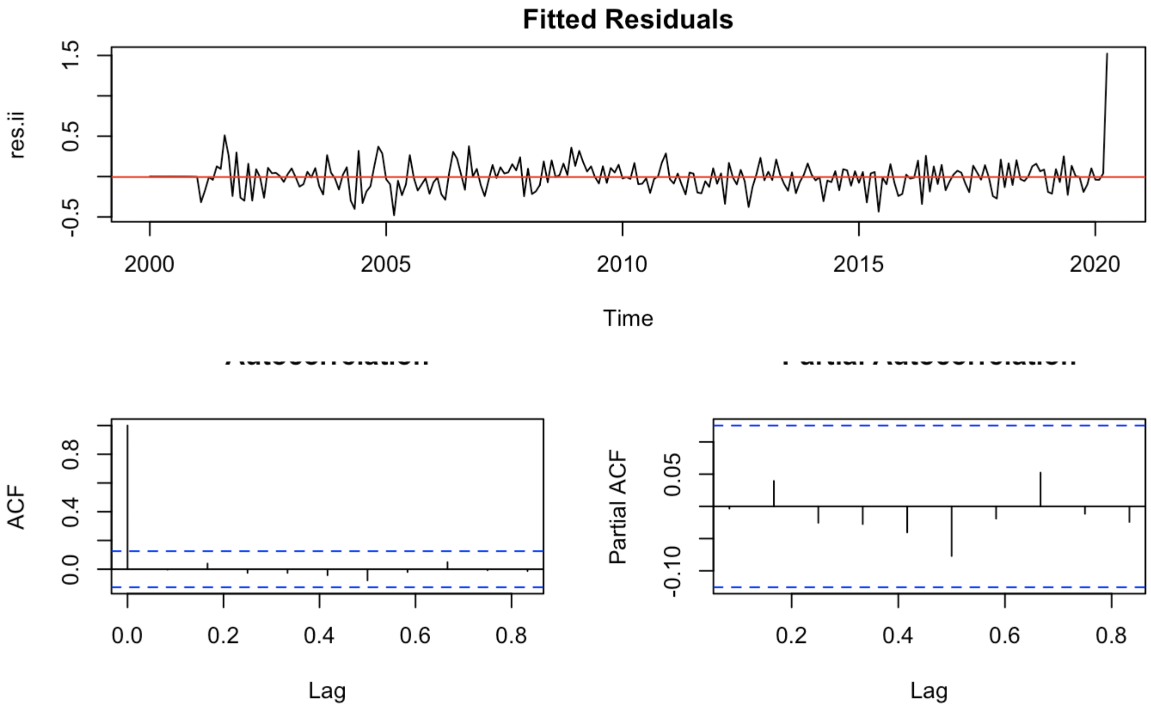
```
Call:
ar(x = res.i, aic = TRUE, order.max = NULL, method = c("yule-walker"))


Order selected 0  sigma^2 estimated as  0.03553

        Box-Pierce test

data:  res.i
X-squared = 3.1583, df = 8, p-value = 0.924


        Box-Ljung test

data:  res.i
X-squared = 3.2682, df = 8, p-value = 0.9164


        Box-Ljung test

data:  res.i^2
X-squared = 0.15762, df = 10, p-value = 1
```

For the first line, the Yule-Walker test was used which determined that the residuals of our model was determined to be an AR(1) model. This means that the residuals of the model is a White Noise model and meets one of our assumptions. The next assumption that we must satisfy are for non-linearity. The following 3 tests are Portmaneau tests that test for independence. The null hypothesis being that the residuals are non-linear. With p-values all over alpha = .05, we can fail to reject the fact that the residuals are non-linear. In this case, non-linearity and independence mean the same thing and there is not sufficient evidence to prove that the residuals for the model aren't non-linear.

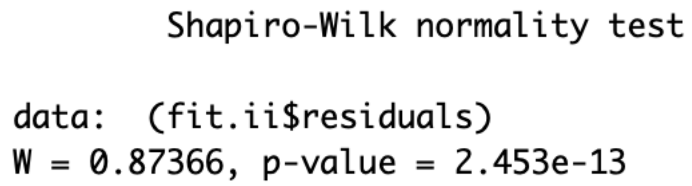
For the second model:

SARIMA(1,1,1) x (2,1,1)[12]

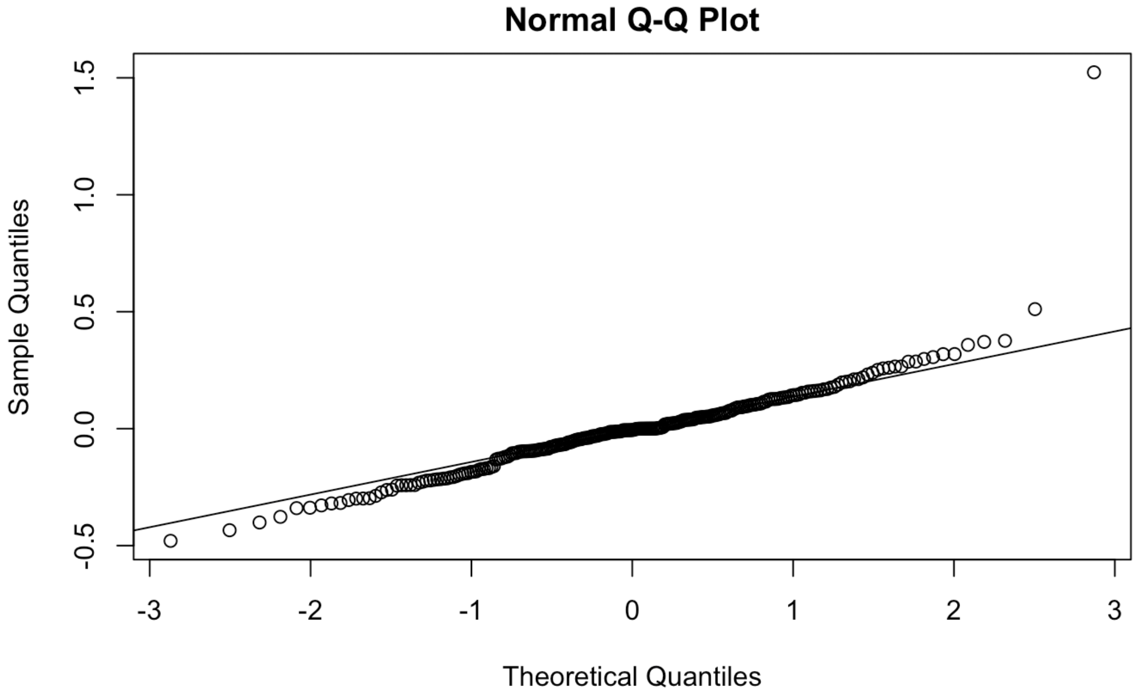
I will first begin by plotting the fitted residuals, the ACF of the residuals and the PACF of the residuals for this model.



Similar to the first model, this one also seems to be white noise as well on first glance and we will continute to test for normality with the Shapiro Wilks test and a normal QQ Plot.

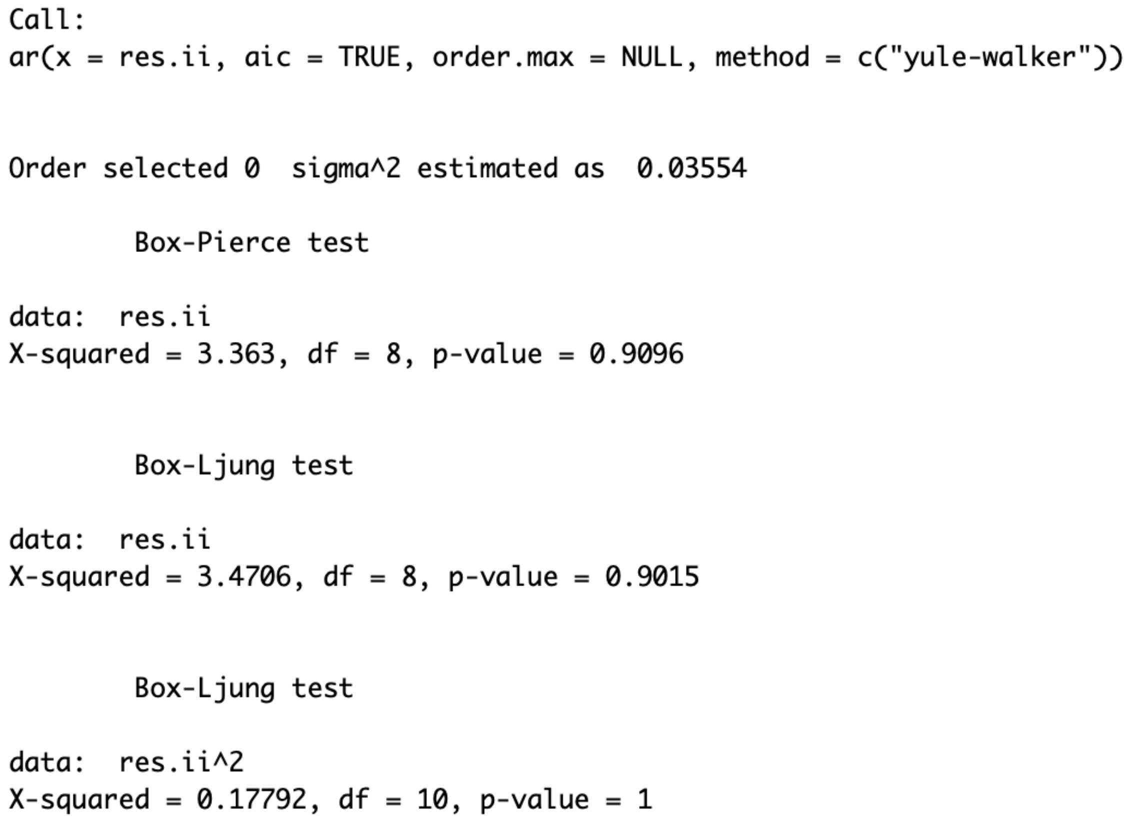


```
        Shapiro-Wilk normality test

data:  (fit.ii$residuals)
W = 0.87366, p-value = 2.453e-13
```



Once again, the residuals for our second model also seem to pass the test for normality as the datapoints roughly follow the diagonal line with the exception of outliers. The Shapiro-Wilk test also shows that we can reject non-normality with 95% confidence as the p-value 2.453-13 is less than alpha = .05.

Next, I will again use the Yule-Walker test to test for White Noise as well as using the Portmaneau tests to test for independence.



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


Order selected 0  sigma^2 estimated as  0.03554

        Box-Pierce test

data:  res.ii
X-squared = 3.363, df = 8, p-value = 0.9096


        Box-Ljung test

data:  res.ii
X-squared = 3.4706, df = 8, p-value = 0.9015


        Box-Ljung test

data:  res.ii^2
X-squared = 0.17792, df = 10, p-value = 1
```

First off, the Yule-Walker test tells us that the model is AR(0) with the "Order selected 0" line. This can ultimaely tell us that the residuals of the second model follow White Noise which is one of the assumptions that we needed confirm.

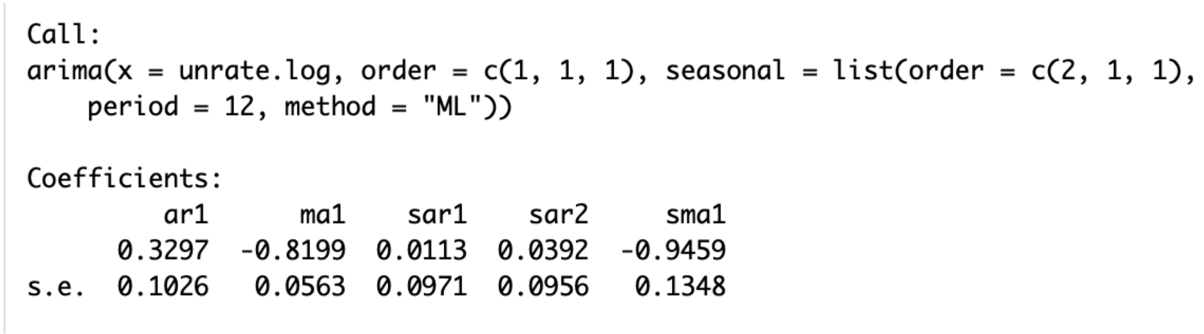
We also need to confirm the assumption of independence in the residuals. Through the Portmaneau tests, it is clear that the p-values are all over alpha = .05 and there is not enough evidence to reject non-linearity(independence).

With both of these model's residuals diagnostic testing not breaking any of our assumptions that we needed, I will have to choose the model based off the lower AICc value that we began with.

SARIMA(0,1,3) x (2,1,1)[12] with an AICc value of -64.021

SARIMA(1,1,1) x (2,1,1)[12] with an AICc value of -65.740

Since SARIMA(1,1,1) x (2,1,1)[12] had the lower AICc value, this is the model that I will be forecasting with.



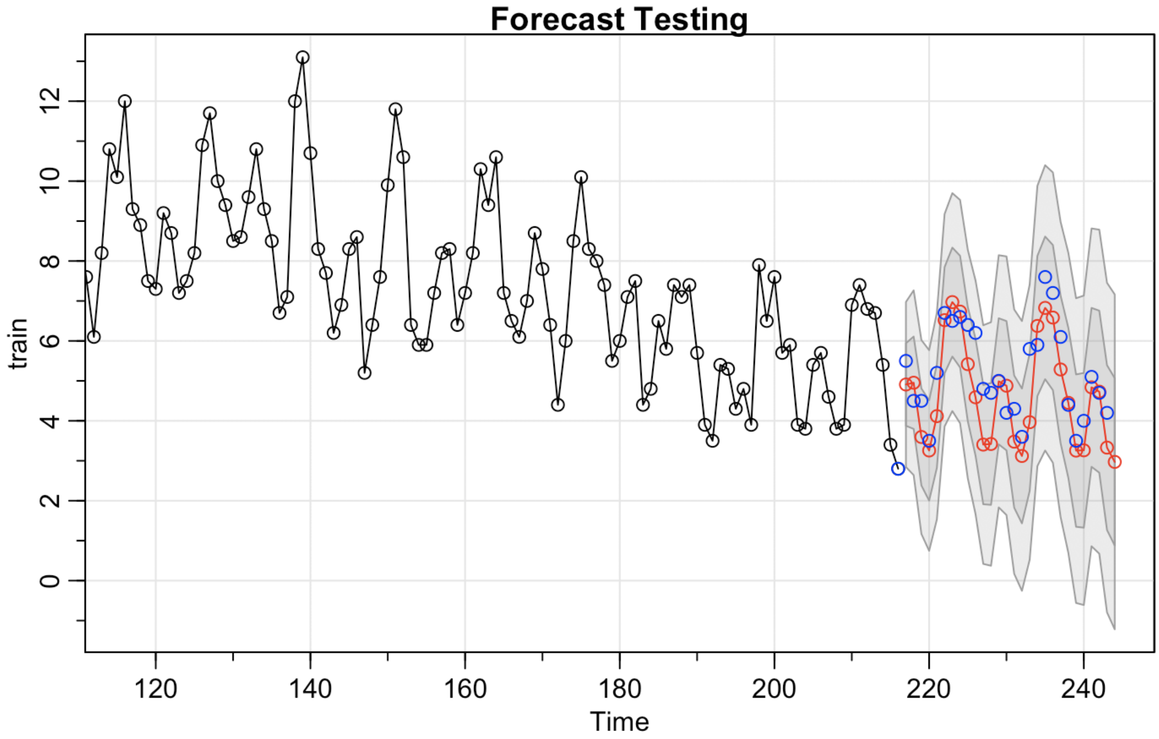
```
Call:
arima(x = unrate.log, order = c(1, 1, 1), seasonal = list(order = c(2, 1, 1),
    period = 12, method = "ML"))

Coefficients:
         ar1      ma1    sar1    sar2     sma1
      0.3297  -0.8199  0.0113  0.0392  -0.9459
s.e.  0.1026   0.0563  0.0971  0.0956   0.1348
```

These are the estimated coefficients of my equation and I will get that SARIMA(1,1,1)x(2,1,1)[12] can be represented by the equation:

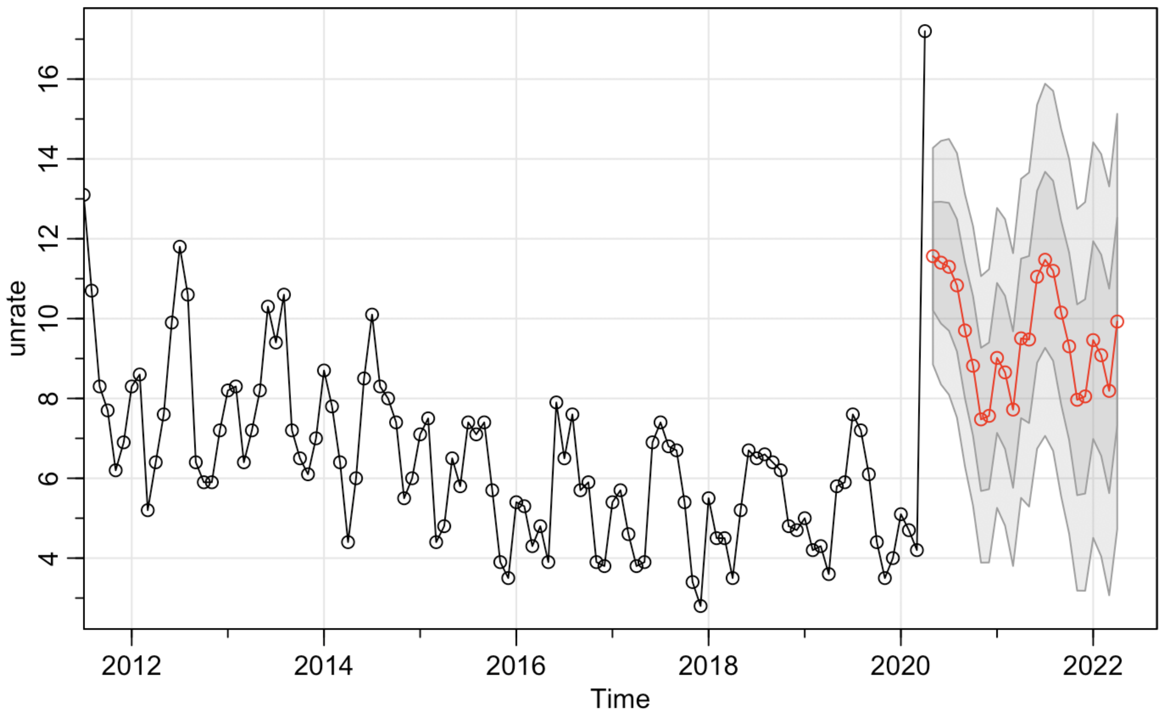
$$\boldsymbol{X_t - .3297X_{t-1} - .0133X_{t-12} - .0392X_{t-24} = Z_t - .8199Z_{t-1} - .9459Z_{t-12}}$$

2.4 Forecasting

As I mentioned in the introduction, I will be forecasting the data by splitting it into training and testing subsets where I will test how accurate the model SARIMA(1,1,1)x(2,1,1)[12] is. I chose to train the data from the years 2000 to 2018 which makes up a total of 216 observations. The last 28 observations of the data I used our model to forecast it and this is the results we got:



The dots in blue are the true values of our data set while the red values are the forecasted values based on our model. As you can see the forecast using this model is pretty accurate. The dark gray area represents a 99% confidence interval in which almost all of the true blue values are in the confidence interval for the forecast. So with seeing the accuracy of our model, we will proceed to forecast the following two years.



This is the overall forecast of the original data with not that large of a 99% confidence interval. This is saying that with 99% confidence the unemployment rate should average out to be 10% in the following two years. The seasonality of the unemployment will remain intact however it seems that the unemployment rate for college students will not completely be able to drop back down to what it was prior to the COVID-19 pandemic.

3 Conclusion

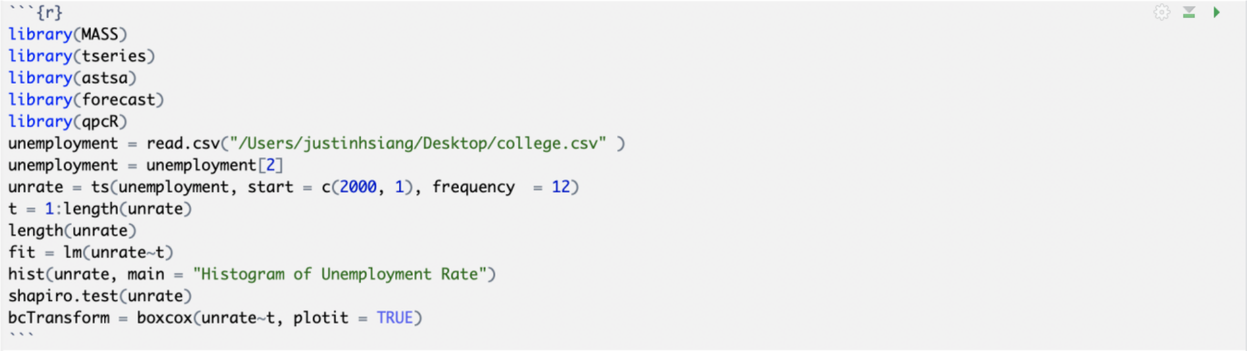
The forecast for the unemployment rate in regard to 20-24 year old bachelor's degrees college graduates shows a lot of promise after the COVID-19 pandemic passes. However, this is not taking into account the possibility for a second wave of the virus leading to another national emergency again. In my forecast, the unemployment rate will fall to about 10% and continue its original trend. While this is not ideal at all as the only time unemployment rates for college graduated have reached 10% in the past 20 years is during the housing crash of 2008. The SARIMA(1,1,1)x(2,1,1)[12] model proved to be the best model as it showed the future values with COVID-19 taken into account with a relatively small 99% confidence interval.

I do think myself that the unemployment rate noted in the forecast makes sense as we are primed to enter an economic recession right now with many small business permanently shutting down and layoffs happening for big businesses as well.

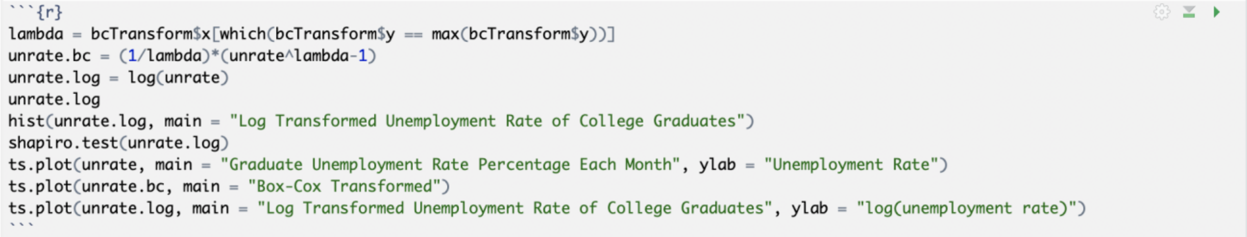
4. References

“Unemployment Rate - College Graduates - Bachelor's Degree, 20 to 24 Years.” *FRED*, 5 June 2020, fred.stlouisfed.org/series/CGBD2024.

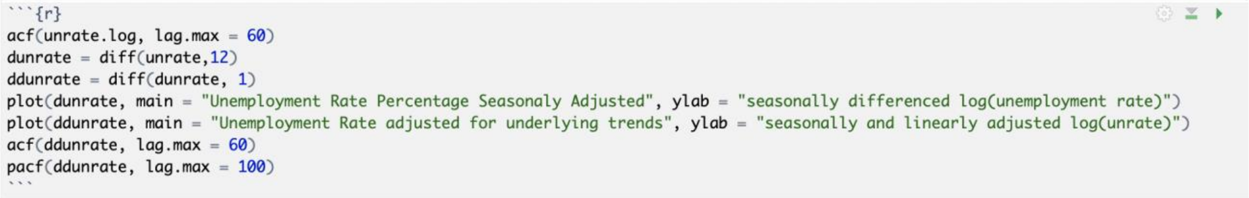
5. Appendix



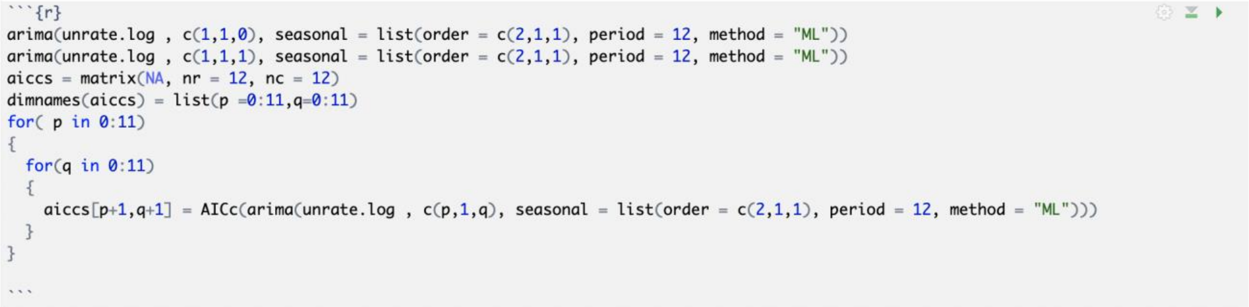
~~~
```{r}
library(MASS)
library(tseries)
library(astsa)
library(forecast)
library(qpcR)
unemployment = read.csv("/Users/justinhsiang/Desktop/college.csv" )
unemployment = unemployment[2]
unrate = ts(unemployment, start = c(2000, 1), frequency  = 12)
t = 1:length(unrate)
length(unrate)
fit = lm(unrate~t)
hist(unrate, main = "Histogram of Unemployment Rate")
shapiro.test(unrate)
bcTransform = boxcox(unrate~t, plotit = TRUE)
```
~~~



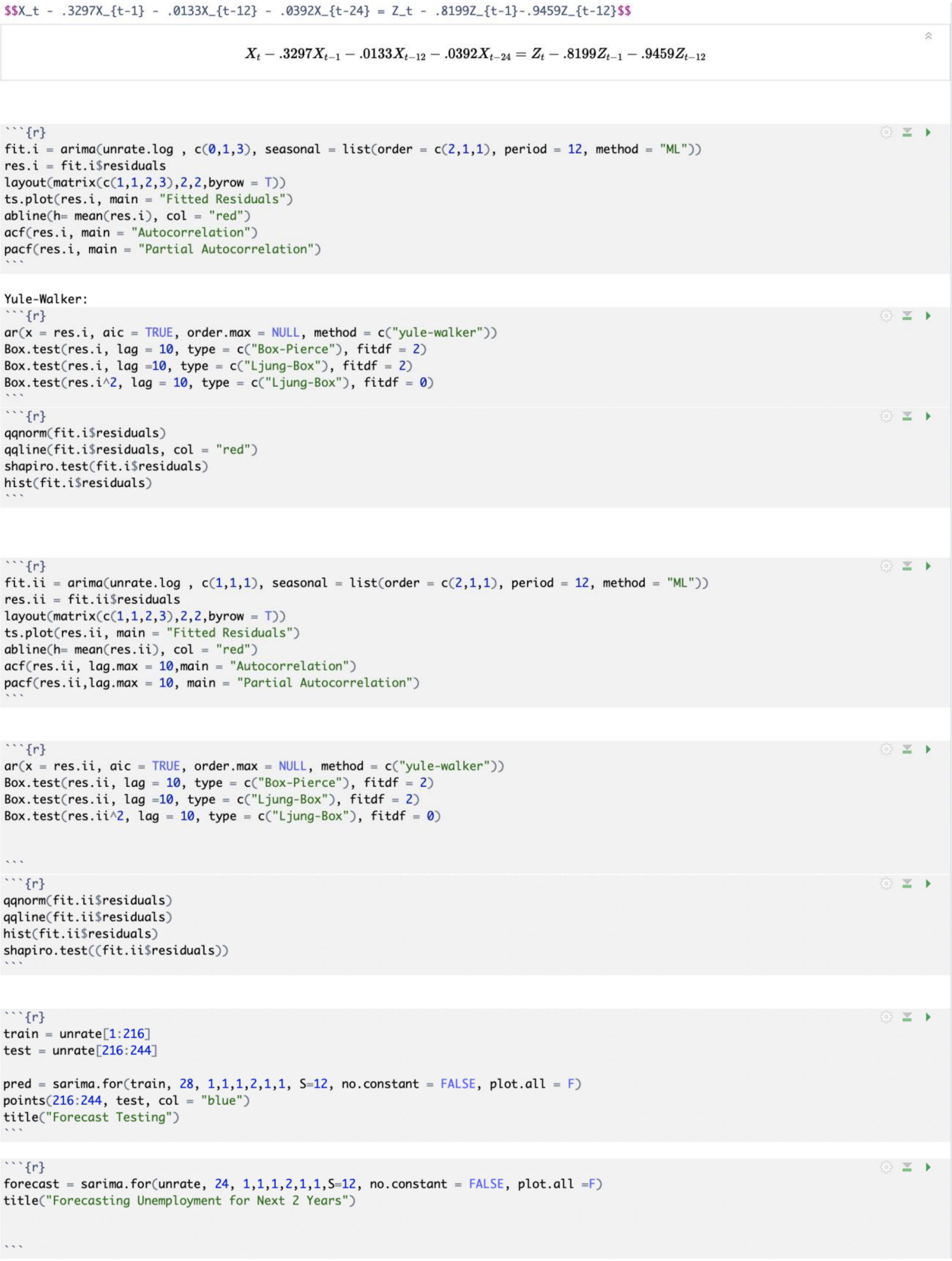
~~~
```{r}
lambda = bcTransform$x[which(bcTransform$y == max(bcTransform$y))]
unrate.bc = (1/lambda)*(unrate^lambda-1)
unrate.log = log(unrate)
unrate.log
hist(unrate.log, main = "Log Transformed Unemployment Rate of College Graduates")
shapiro.test(unrate.log)
ts.plot(unrate, main = "Graduate Unemployment Rate Percentage Each Month", ylab = "Unemployment Rate")
ts.plot(unrate.bc, main = "Box-Cox Transformed")
ts.plot(unrate.log, main = "Log Transformed Unemployment Rate of College Graduates", ylab = "log(unemployment rate)")
```
~~~



~~~
```{r}
acf(unrate.log, lag.max = 60)
dunrate = diff(unrate,12)
ddunrate = diff(dunrate, 1)
plot(dunrate, main = "Unemployment Rate Percentage Seasonaly Adjusted", ylab = "seasonally differenced log(unemployment rate)")
plot(ddunrate, main = "Unemployment Rate adjusted for underlying trends", ylab = "seasonally and linearly adjusted log(unrate)")
acf(ddunrate, lag.max = 60)
pacf(ddunrate, lag.max = 100)
```
~~~



~~~
```{r}
arima(unrate.log , c(1,1,0), seasonal = list(order = c(2,1,1), period = 12, method = "ML"))
arima(unrate.log , c(1,1,1), seasonal = list(order = c(2,1,1), period = 12, method = "ML"))
aiccs = matrix(NA, nr = 12, nc = 12)
dimnames(aiccs) = list(p =0:11,q=0:11)
for( p in 0:11)
{
  for(q in 0:11)
  {
    aiccs[p+1,q+1] = AICc(arima(unrate.log , c(p,1,q), seasonal = list(order = c(2,1,1), period = 12, method = "ML")))
  }
}

```
~~~



```
$$X_t - .3297X_{t-1} - .0133X_{t-12} - .0392X_{t-24} = Z_t - .8199Z_{t-1}-.9459Z_{t-12}$$
```

$$X_t - .3297X_{t-1} - .0133X_{t-12} - .0392X_{t-24} = Z_t - .8199Z_{t-1} - .9459Z_{t-12}$$

~~~
```{r}
fit.i = arima(unrate.log , c(0,1,3), seasonal = list(order = c(2,1,1), period = 12, method = "ML"))
res.i = fit.i$residuals
layout(matrix(c(1,1,2,3),2,2,byrow = T))
ts.plot(res.i, main = "Fitted Residuals")
abline(h= mean(res.i), col = "red")
acf(res.i, main = "Autocorrelation")
pacf(res.i, main = "Partial Autocorrelation")
```
~~~

Yule-Walker:

~~~
```{r}
ar(x = res.i, aic = TRUE, order.max = NULL, method = c("yule-walker"))
Box.test(res.i, lag = 10, type = c("Box-Pierce"), fitdf = 2)
Box.test(res.i, lag =10, type = c("Ljung-Box"), fitdf = 2)
Box.test(res.i^2, lag = 10, type = c("Ljung-Box"), fitdf = 0)
~~~
```{r}
qqnorm(fit.i$residuals)
qqline(fit.i$residuals, col = "red")
shapiro.test(fit.i$residuals)
hist(fit.i$residuals)
```

```{r}
fit.ii = arima(unrate.log , c(1,1,1), seasonal = list(order = c(2,1,1), period = 12, method = "ML"))
res.ii = fit.ii$residuals
layout(matrix(c(1,1,2,3),2,2,byrow = T))
ts.plot(res.ii, main = "Fitted Residuals")
abline(h= mean(res.ii), col = "red")
acf(res.ii, lag.max = 10,main = "Autocorrelation")
pacf(res.ii,lag.max = 10, main = "Partial Autocorrelation")
```

```{r}
ar(x = res.ii, aic = TRUE, order.max = NULL, method = c("yule-walker"))
Box.test(res.ii, lag = 10, type = c("Box-Pierce"), fitdf = 2)
Box.test(res.ii, lag =10, type = c("Ljung-Box"), fitdf = 2)
Box.test(res.ii^2, lag = 10, type = c("Ljung-Box"), fitdf = 0)


```
```{r}
qqnorm(fit.ii$residuals)
qqline(fit.ii$residuals)
hist(fit.ii$residuals)
shapiro.test((fit.ii$residuals))
```

```{r}
train = unrate[1:216]
test = unrate[216:244]

pred = sarima.for(train, 28, 1,1,1,2,1,1, S=12, no.constant = FALSE, plot.all = F)
points(216:244, test, col = "blue")
title("Forecast Testing")
```

```{r}
forecast = sarima.for(unrate, 24, 1,1,1,2,1,1,S=12, no.constant = FALSE, plot.all =F)
title("Forecasting Unemployment for Next 2 Years")


```
```
```