Forecasting Consumer Price Index for All Urban Consumers: Apparel in U.S. City Average

Sampreeth Salveru

06/05/2020

Contents

1. Abstract

The Consumer Price Index (CPI) is a measure of the average change overtime in the prices paid by urban consumers for a market basket of consumer goods and services. This dataset is specifically for a market basket of apparel items for all urban consumers (CPI-U) in the U.S city average from 1990 to 2019. Using this data set, our goal is to forecast the CPI-U for apparel 12 months into the future using Box-Jenkings methodology and seasonal autoregressive integrated moving average (SARIMA) models. Forecasting these values will help us gain a better understanding in the fluctuations of CPI in apparel due to inflation and also seasonal changes.

2. Introduction

The Consumer Price Index For All Urban Consumers (CPI-U) measures the changes in the price of a basket of goods and services purchased by urban consumers. All urban consumers refers to all urban households in Metropolitan Statistical Areas (MSAs) and urban places of 2,500 inhabitants or more. CPI-U is used as an economic indicator to measure inflation or deflation in the economy and it is also used as a means of adjusting dollar values.

This data set is for all apparel retail prices for urban consumers. Along with the government, business executives and other private citizens also use CPI-U in order to make economic decisions. This dataset is interesting as it has fluctuations in values due to seasonal changes, which we would like to analyze and forecast. The models created to forecast future values are trained from a monthly non-seasonally adjusted dataset. The data was collected from 1990 to 2019 by the U.S. Bureau of Labor Statistics. The source for this data can be found at Federal Reserve Economic Database, St. Louis.

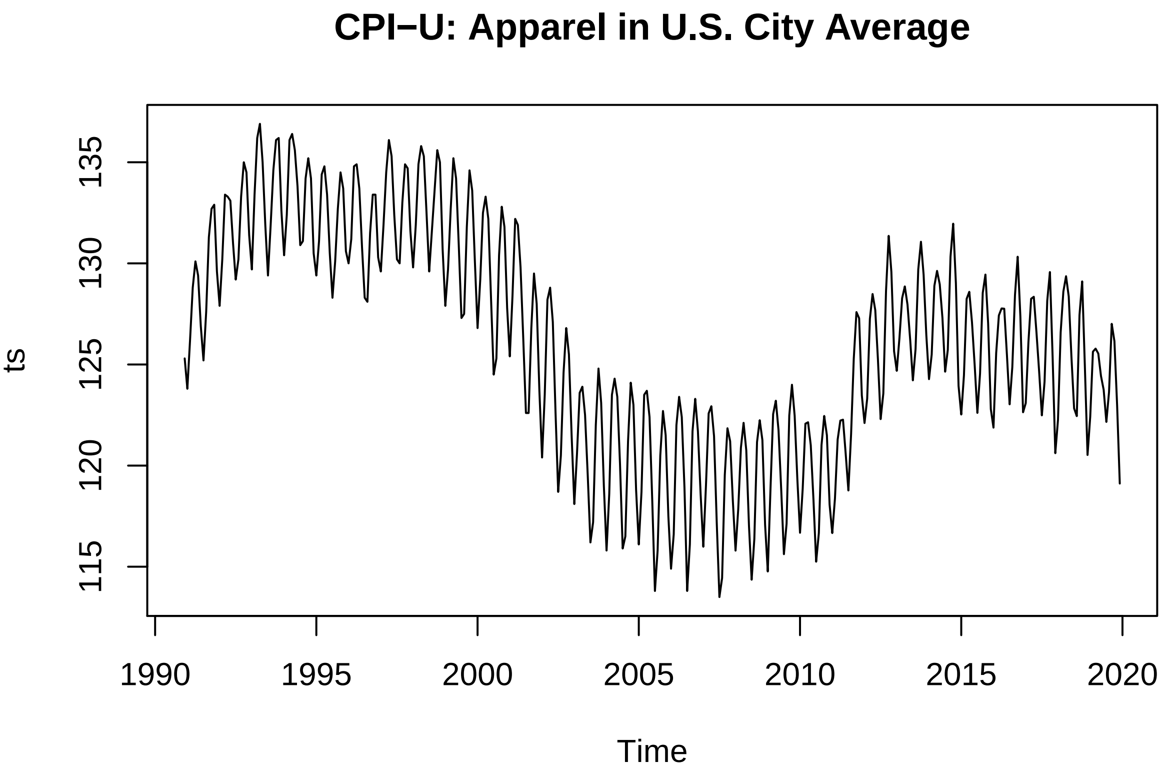
To analyze and forecast, we begin by loading the dataset and removing the last 12 values to compare our forecasts with. Then, we begin an exploratory analysis and difference the data to remove trend and seasonality. After differencing, we hope to use our now stationary time series to plot the autocorrelations and partial autocorrelation functions to help identify a seasonal autoregressive integrated moving average (SARIMA) model to forecast. We identify two potential models and conduct diagnostic checking on both to identify the best model that follows all of our assumptions. Using the fitted model we utilize it to forecast 12 future values. Finally, these predicted values are compared with the values we removed in the beginning in order to prove that the fitted model was successfull in forecasting.

All statistical analysis was performed on ‘RStudio’ utilizing various software libraries from ‘R’.

3. Analysis

3.1 Exploratory Data Analysis

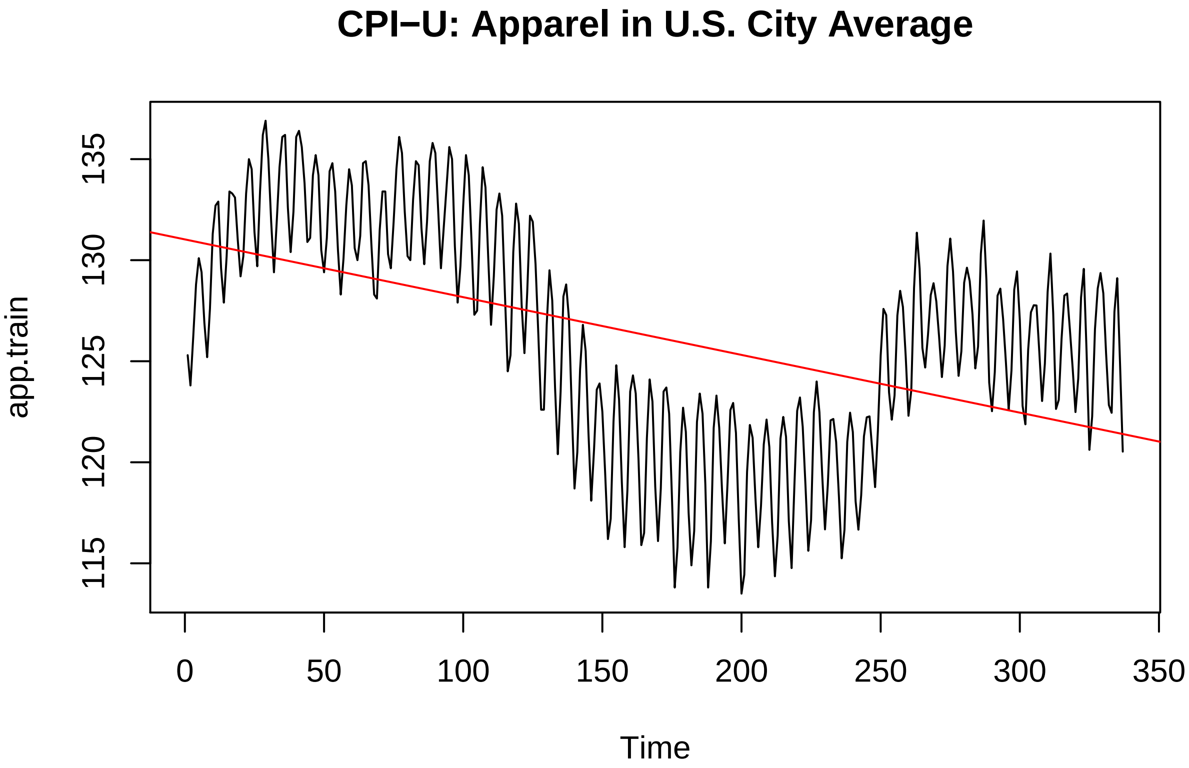
We begin by loading the time series into RStudio and plotting it.



From this plot we can see there is a trend and there is a sharp negative decline in behavior beginning in 2001 till 2009. This negative change is due to the U.S. recessions that took place in Mar 2001 - Nov 2001 and Dec 2007 - June 2009. There also seems to be a strong seasonal component and the variance changes slightly over time. In order to make this stationary we must transform and difference the data.

At this point the last 12 values of the data are removed to compare against the forecasted values.

3.2 Data Transformations

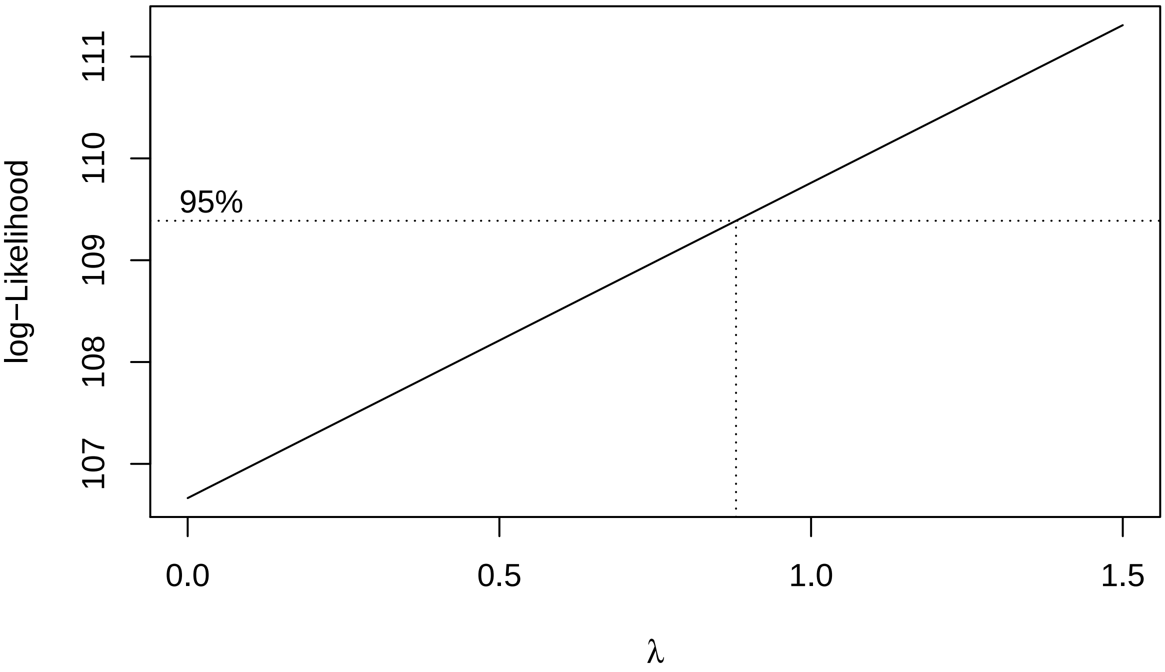


Once we added a regression line, a negative trend is clearly seen. We have to difference at lag = 1 to remove the trend and difference at lag = 12 to remove seasonality.

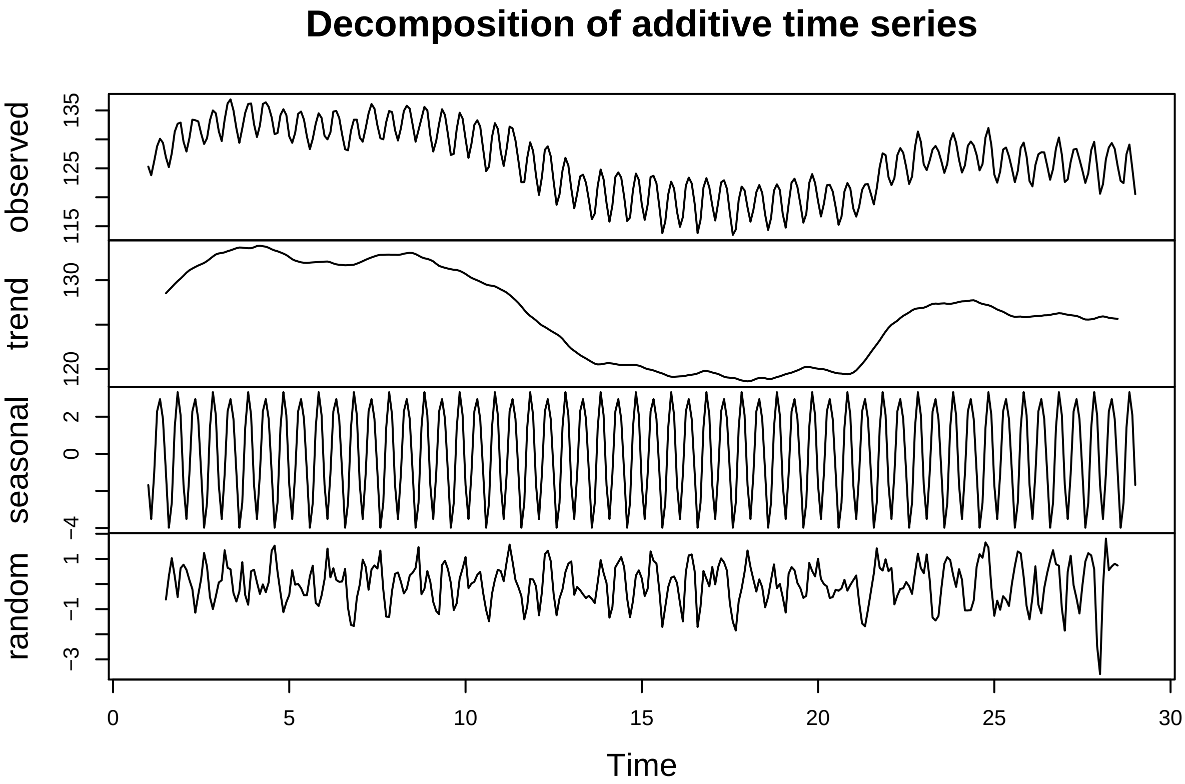
In order to stabilize variance and keep it constant, we will need to perform a Box-Cox transformation.

$$y_i^{(\lambda)} = \begin{cases} \dfrac{y_i^\lambda - 1}{\lambda}, & \text{if } \lambda \neq 0 \\ \log y_i, & \text{if } \lambda = 0 \end{cases}$$

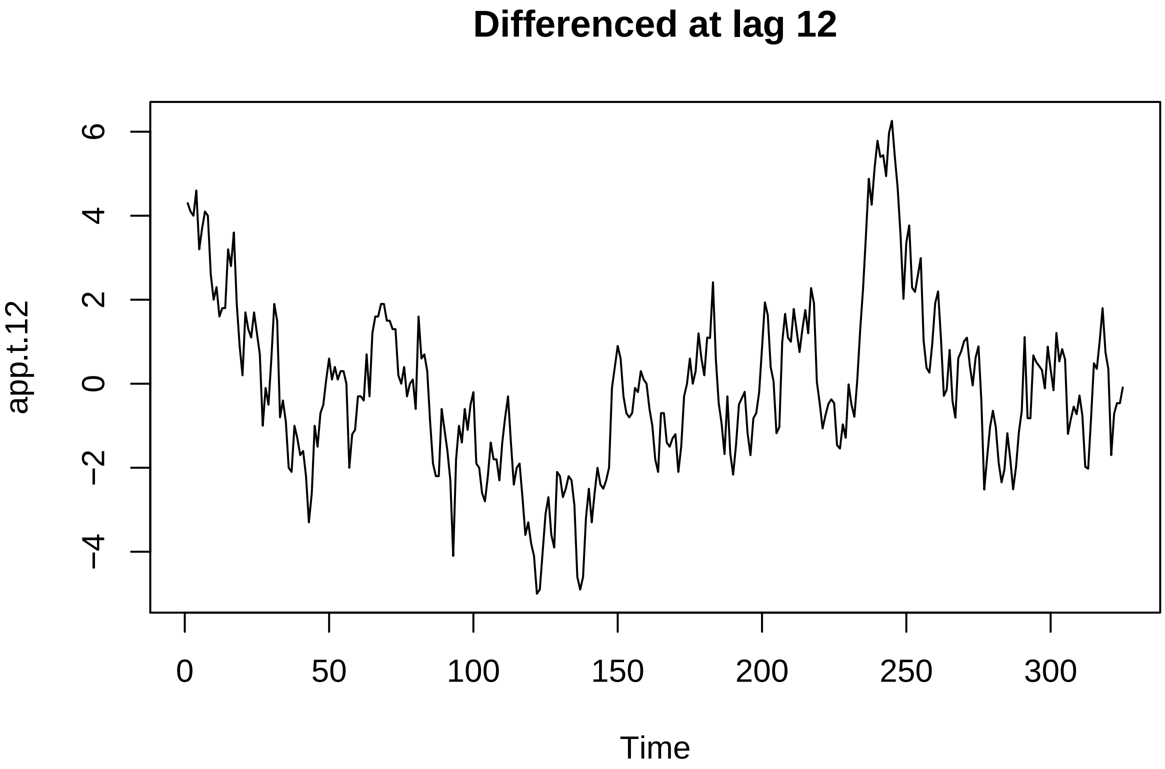
We use the boxcox() function to find our optimal λ value.



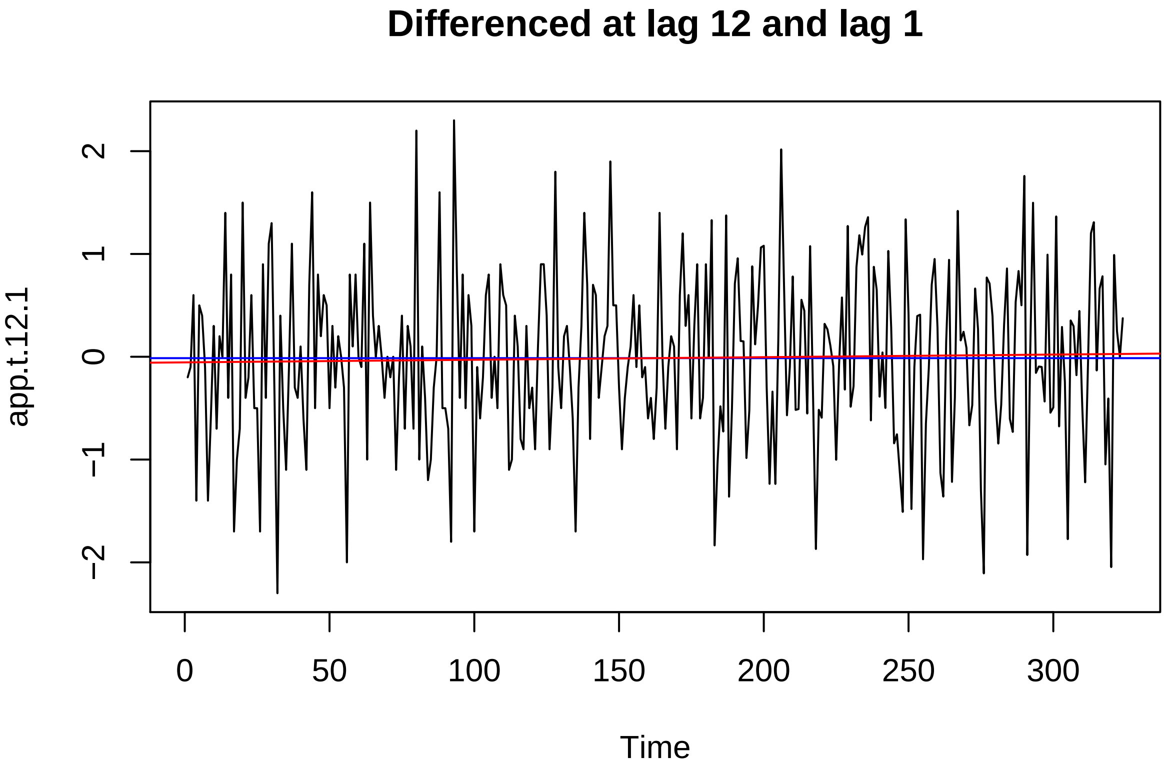
Based off of our Box-Cox plot, our optimal value is $\lambda = 1.5$. But we will choose $\lambda = 1$ because it is included in the interval and will make our analysis easier. A value of 1 means there is no transformation required, also there isn't a major change when using $\lambda = 1.5$.



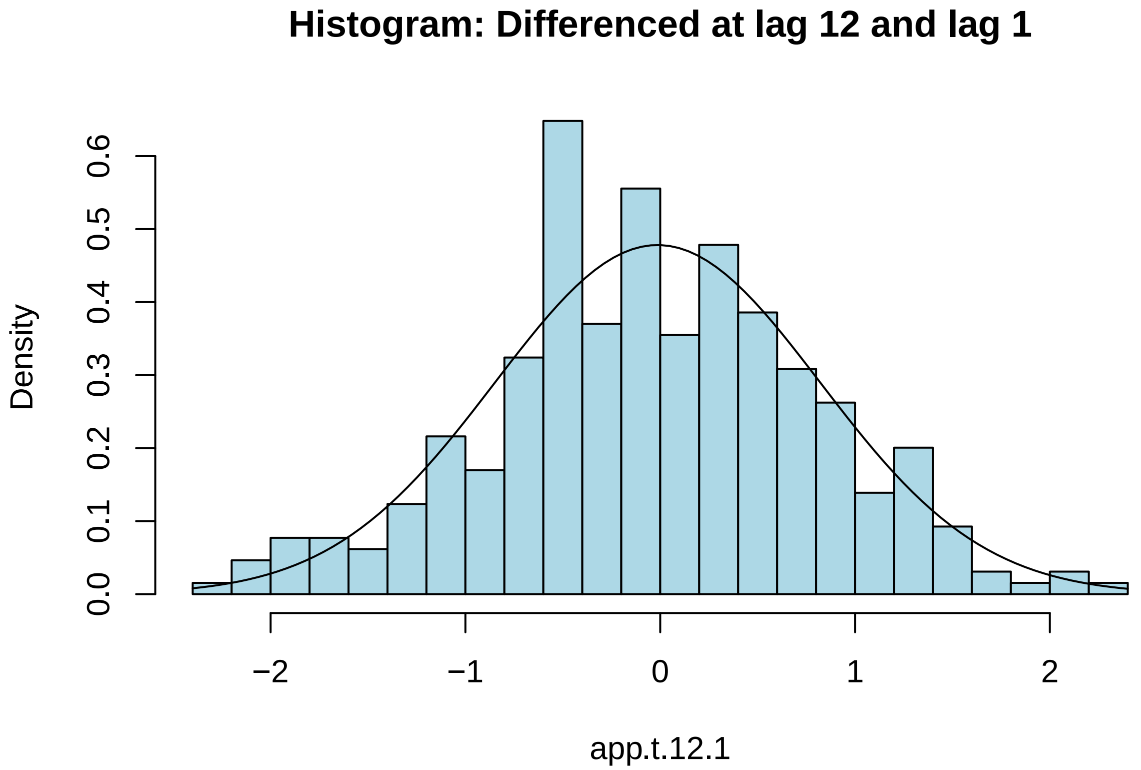
The decomposition of the time series shows our trend and seasonal components. We will now difference at lag = 1 and lag = 12 to remove them.



When differenced at lag = 12, the seasonality is removed. Our variance decreases from 32.38529 to 4.076279, so our differencing is working. The trend is still apparent so we now difference at lag = 1.



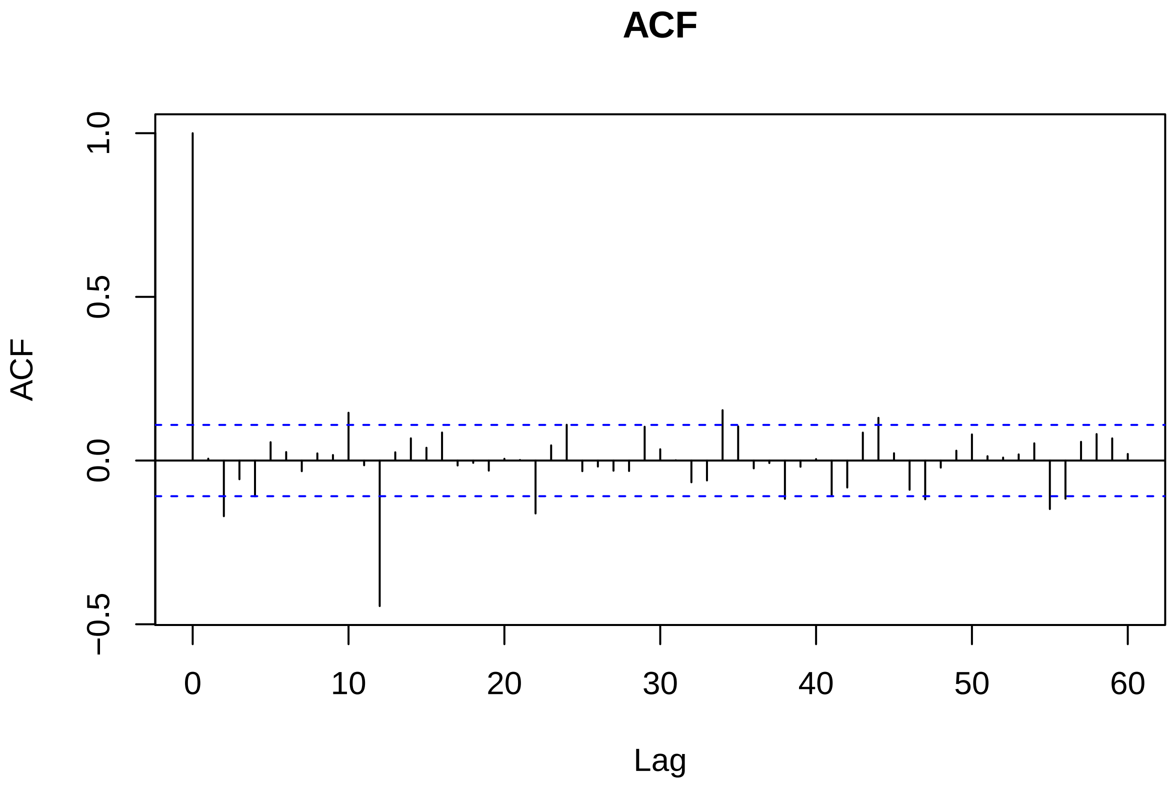
When differenced at lag = 1, the variance decreases from 4.076279 to 0.6960859. The mean is added in blue and the regression line is added in red. We see there is no trend and the mean is 0, the data seems to be stationary.

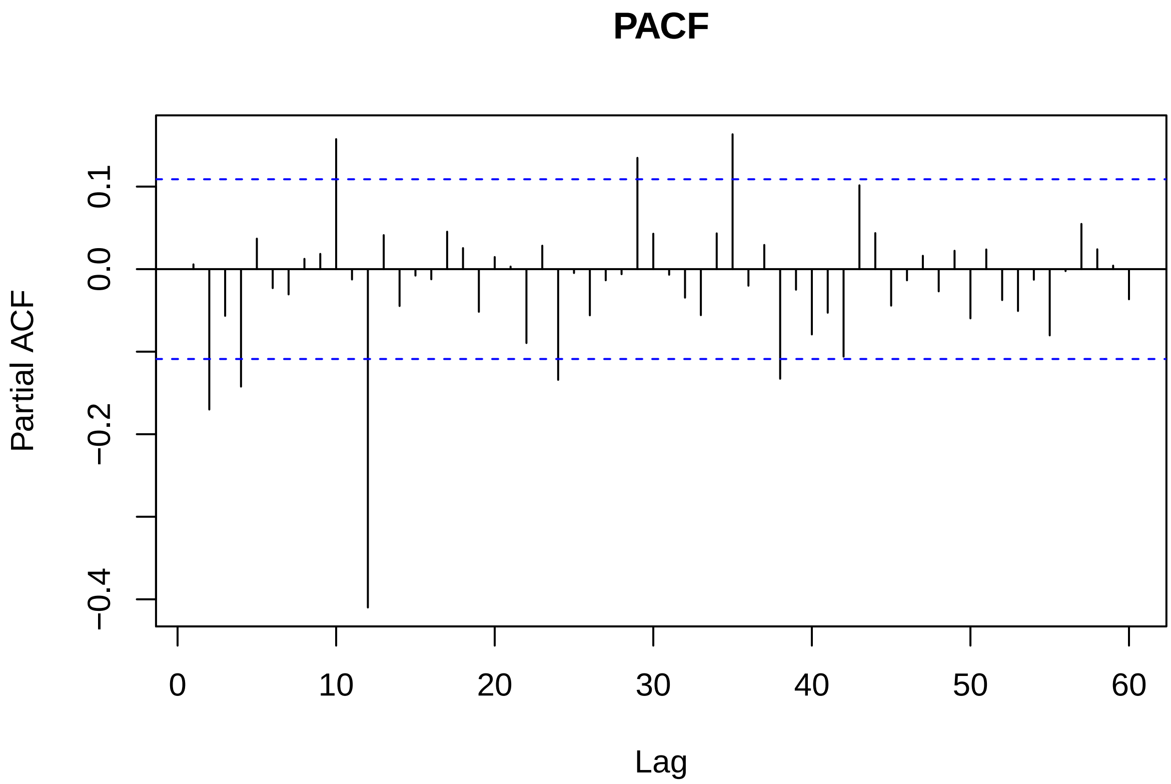


Looking at the histogram of our twice difference data, we notice it is symmetric and almost Gaussian.

3.3 ACF and PACF Analysis

Now that our data is stationary we can use the ACF and PACF to fit possible SARIMA models.





We can identify the seasonal components by examining lags $l = 12n$, $n \in \mathbb{N}$. The ACF cuts off after lag 12, so we can assume SMA(1). The PACF cuts off after lags 2 or 3, so we can assume either SAR(2) or SAR(3).

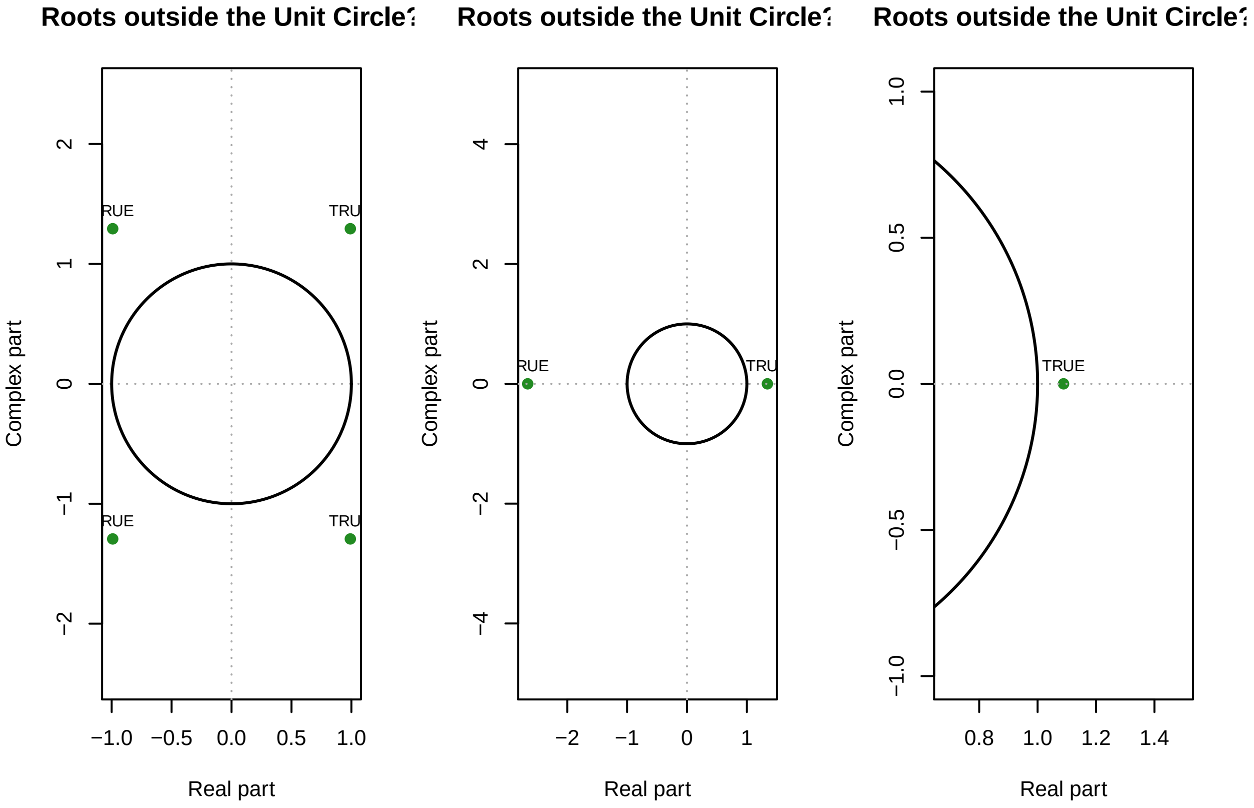
To identify the AR and MA orders, we will examine the lags 1 to 11. The ACF cuts off after lag 0, so we can assume MA(0). The PACF cuts off at lag 4, so we assume AR(4). We fit multiple models using our parameters from the ACF and PACF plots. We also consider models found by calculating AICc for a given model and choose one with the smallest AICc value. My first model is found using ACF and PACF, and my second model is obtained by using AICc.

We consider these two models and estimate the coefficients using ML estimation. For coefficients that have 0 in their confidence interval, we fix their value as 0 in order to maintain a lower AICc.

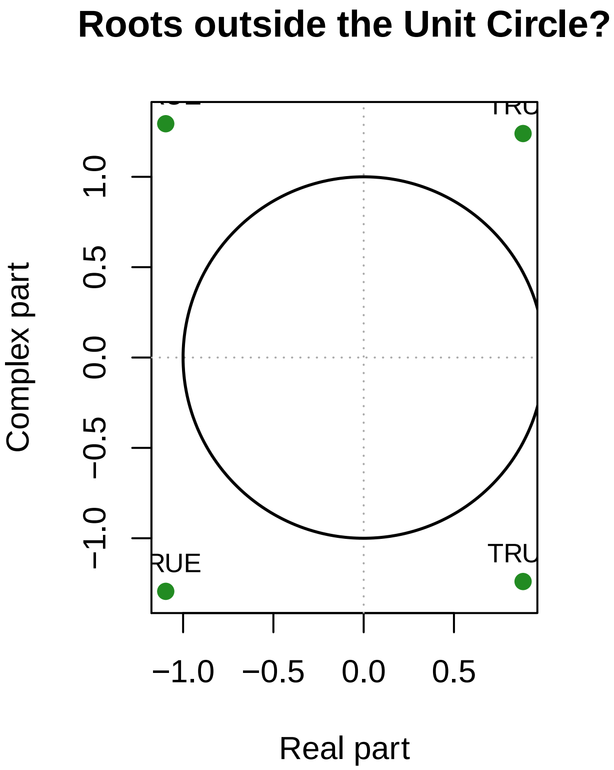
- 1. SARIMA $(4, 1, 0)$ x $(2, 1, 1)_{12}$
  - AICc = 715.9194
  - $(1 + 0.1957B^2 + 0.1418B^4)(1 - 0.3688B^{12} - 0.2799B^{24})\nabla_{12}\nabla Y_t = (1 - 0.9184B^{12})Z_t$
- 2. SARIMA $(4, 1, 0)$ x $(0, 1, 1)_{12}$
  - AICc = 716.2502
  - $(1 + 0.1984B^2 + 0.0641B^3 + 0.1502B^4)\nabla_{12}\nabla Y_t = (1 - 0.536B^{12})Z_t$

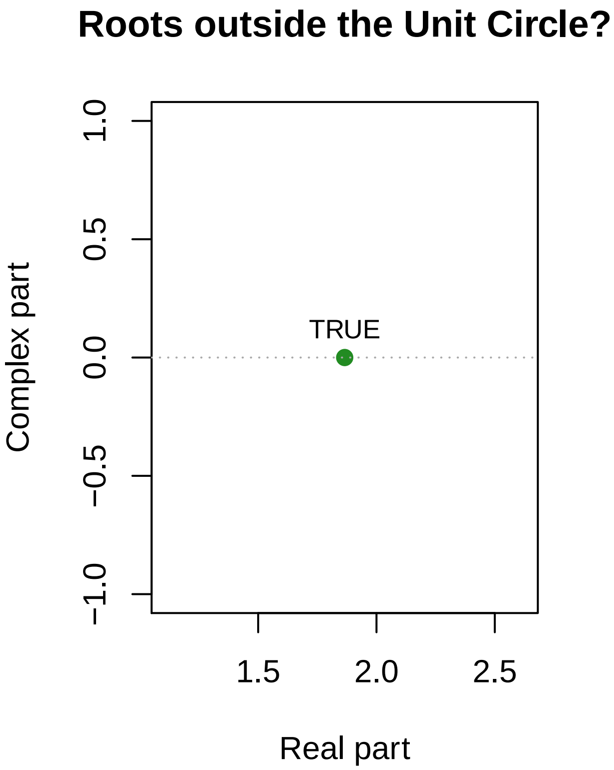
3.4 Model Diagnostics

We begin the diagnostic checking for both of our models by checking if they are causal and invertible. We check this by seeing if the roots of the polynomials are outside of the unit circle.



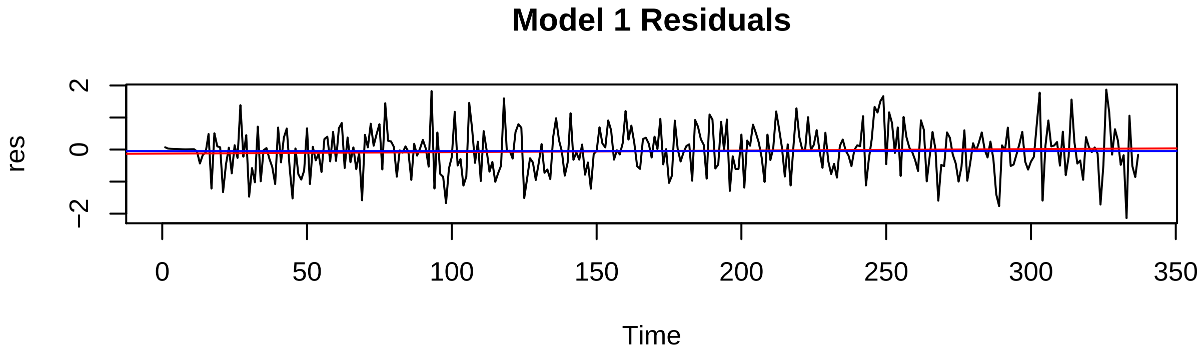
We plot the roots of the AR, SMA, and SAR separately. The roots for Model 1 are all outside of the unit circle so they are both causal and invertible.

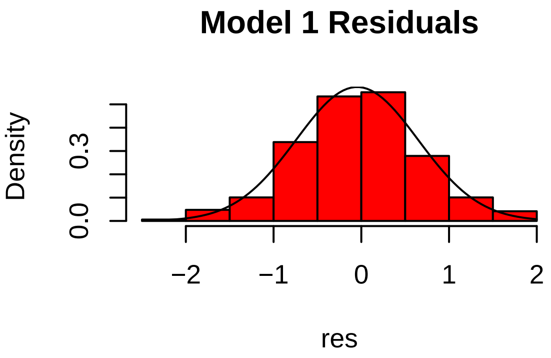


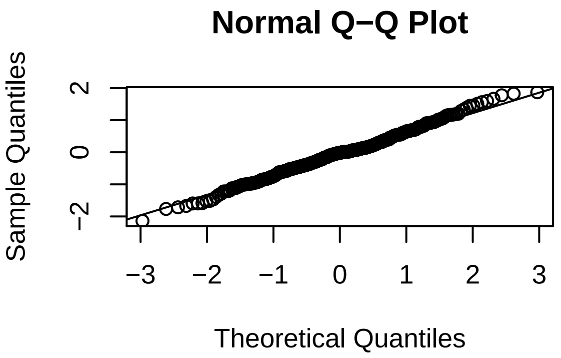


We plot the roots of the AR and SMA separately. The roots for Model 2 are all outside of the unit circle so they are both causal and invertible.

Now we will analyze the residuals of each model separately.
We begin with Model 1. $(1+0.1957B^2+0.1418B^4)(1-0.3688B^{12}-0.2799B^{24})\nabla_{12}\nabla Y_t = (1-0.9184B^{12})Z_t$







```
##
##  Shapiro-Wilk normality test
##
## data:  res
## W = 0.99542, p-value = 0.4262
```

The plot of the residuals for Model 1 appears to resemble white noise. The histogram and Q-Q plot appear to show that the residuals are approximately normal.

Performing the Shapiro-Wilk normality test gives us a p-value = 0.4262, which passes the test.

We also perform the Box-Pierce, Ljung-Box, and McLeod-Li tests on the residuals.

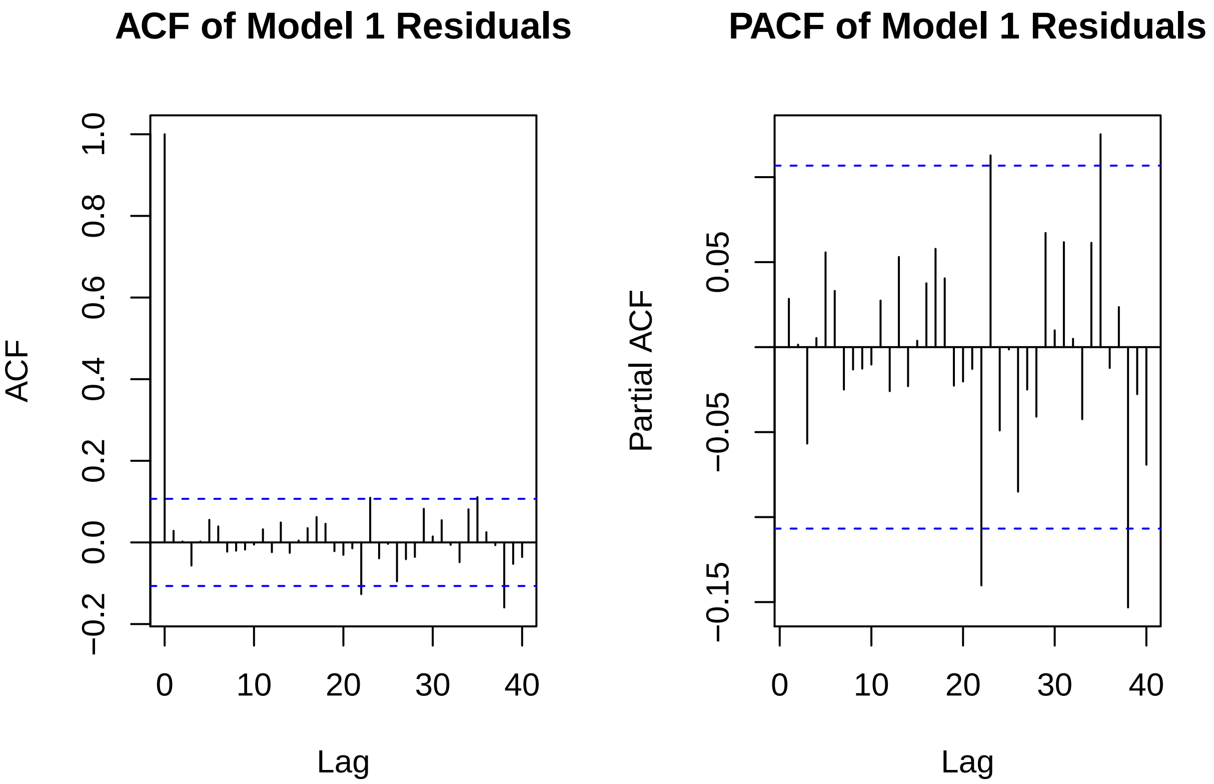
```
##
##  Box-Pierce test
##
## data:  res
## X-squared = 7.3334, df = 11, p-value = 0.7715
```

```
##
##  Box-Ljung test
##
## data:  res
## X-squared = 7.6151, df = 11, p-value = 0.7473
```

```
##
##  Box-Ljung test
##
## data:  res^2
## X-squared = 10.08, df = 18, p-value = 0.9293
```

The model passes all tests at the $\alpha = 0.05$ significance level.

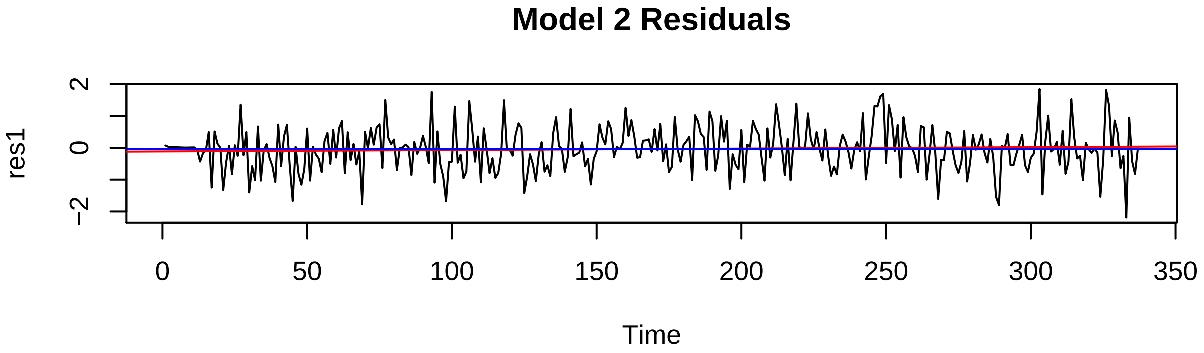
We also plot the ACF and PACF of the residuals to check if they resemble white noise.

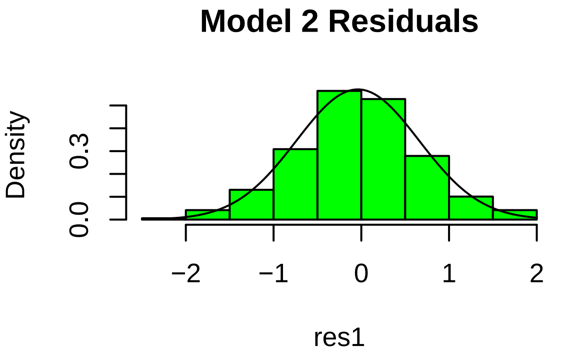


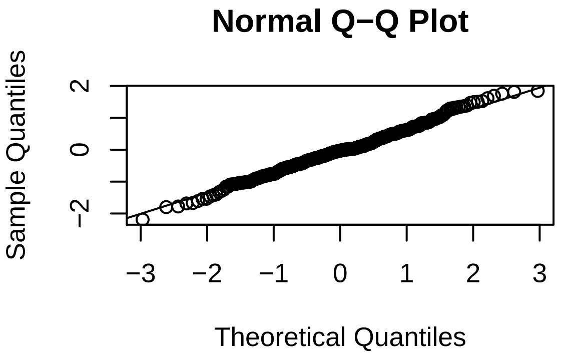
Besides the ACFs and PACFs extending slightly over the confidence intervals at lags 22 and 23, they resemble white noise.

This model passes diagnostics and can be used for forecasting.

We now check the residuals for Model 2. $(1 + 0.1984B^2 + 0.0641B^3 + 0.1502B^4)\nabla_{12}\nabla Y_t = (1 - 0.536B^{12})Z_t$







```
##
##  Shapiro-Wilk normality test
##
## data:  res1
## W = 0.99437, p-value = 0.2503
```

The plot of the residuals for Model 2 appears to resemble white noise. The histogram and Q-Q plot appear to show that the residuals are approximately normal.

Performing the Shapiro-Wilk normality test gives us a p-value = 0.2503, which passes the test.

We also perform the Box-Pierce, Ljung-Box, and McLeod-Li tests on the residuals.

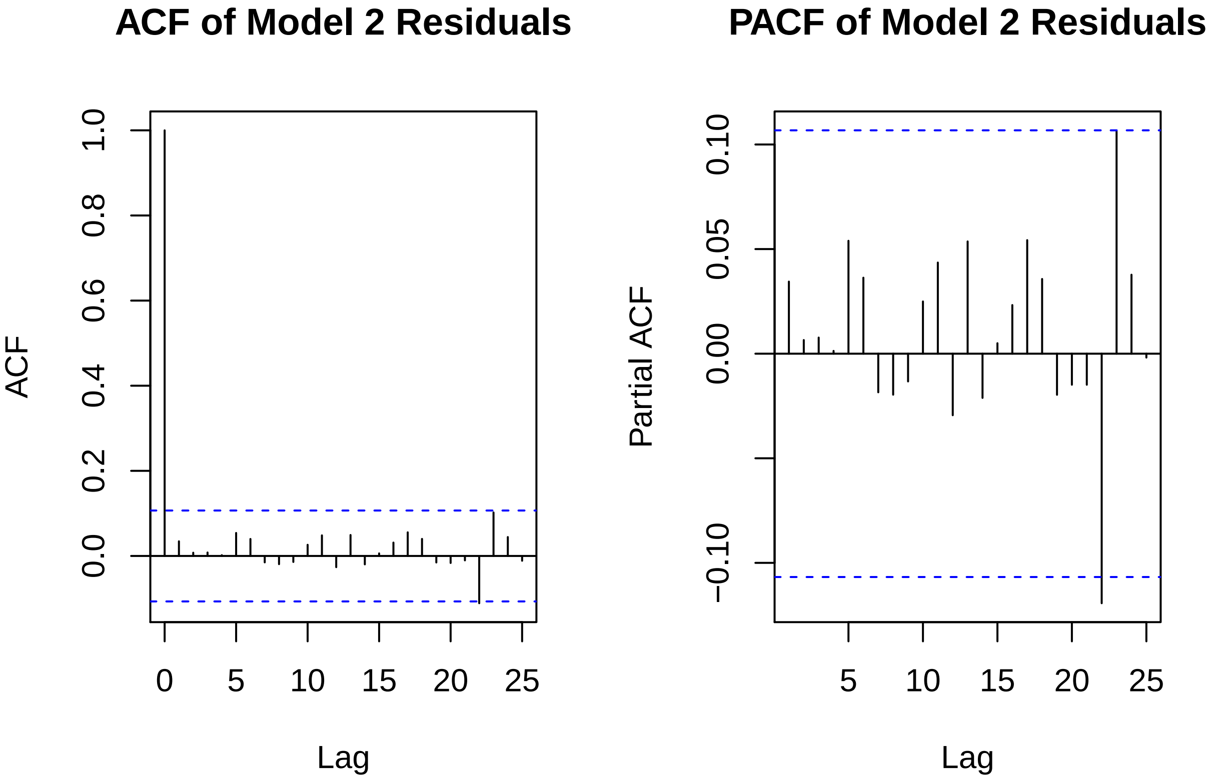
```
##
##  Box-Pierce test
##
## data:  res1
## X-squared = 6.3747, df = 13, p-value = 0.9314
```

```
##
##  Box-Ljung test
##
## data:  res1
## X-squared = 6.6305, df = 13, p-value = 0.9202
```

```
##
##  Box-Ljung test
##
## data:  res1^2
## X-squared = 8.2662, df = 18, p-value = 0.9744
```

The model passes all tests at the $\alpha = 0.05$ significance level.

We also plot the ACF and PACF of the residuals to check if they resemble white noise.



ACF of Model 2 Residuals

PACF of Model 2 Residuals

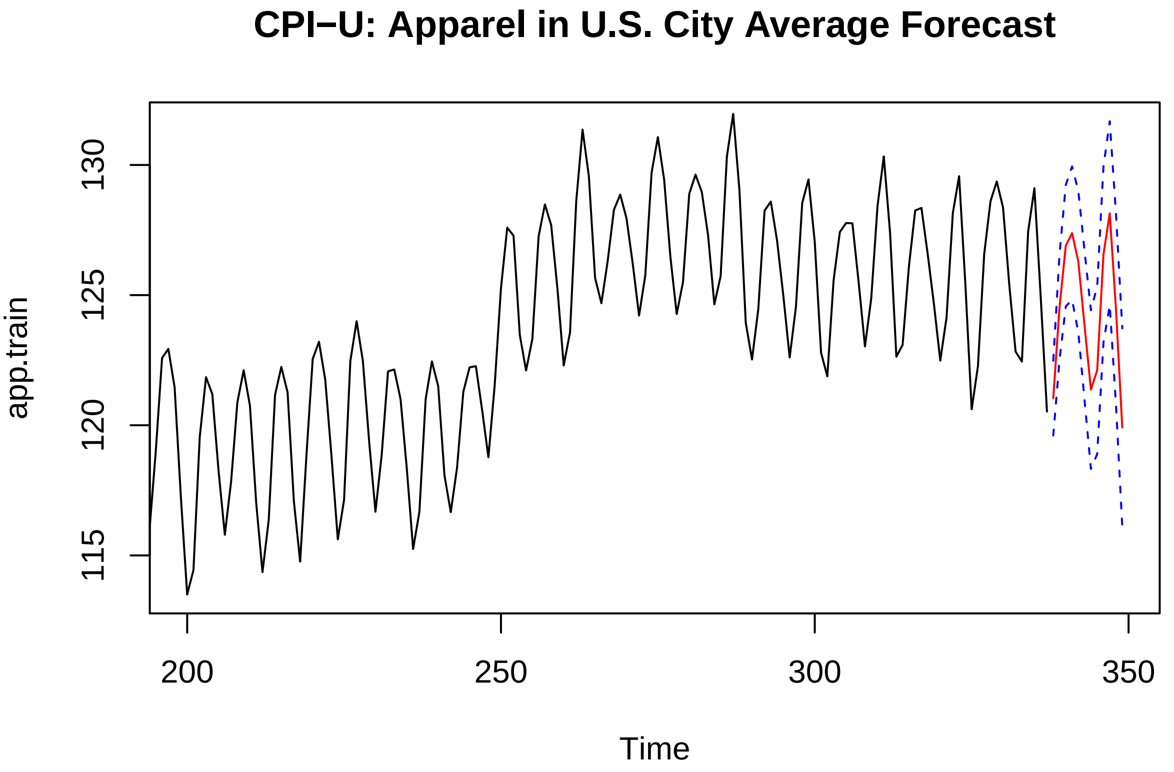
The ACF and PACF is slightly improved in this model. The lags that caused an issue in Model 1 are now in the confidence interval, except for lag 22 of the PACF.

Again, the model also passes all diagnostic tests and can be used for forecasting.

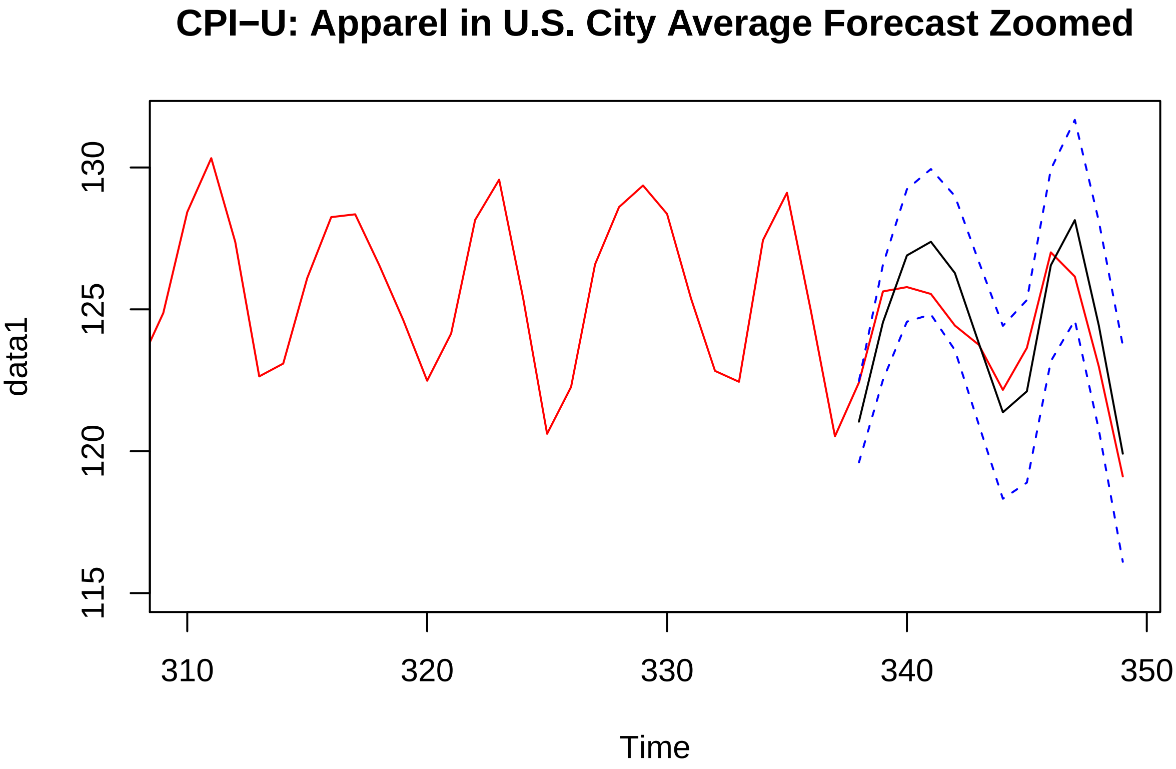
Since both of our models pass our diagnostic tests and their AICc values are similar, we can choose any model. We utilize the principle of parsimony and choose Model 2 as it has fewer parameters.

3.5 Forecasting

We selected our model and now we can begin forecasting.



The forecasted values are plotted in red and their confidence intervals are in blue.



This plot is zoomed in order to get a better view on the forecasting accuracy. The original values saved in the beginning are added in red and the forecasted values are in black. We conclude that the model provides great accuracy, as the forecasted values lie close to the original values and they are all included in the confidence interval.

4. Conclusion

We were successful in our analysis of the time series and selected two SARIMA models for forecasting. Since both passed all of our diagnostics we resorted to the principle of parsimony and chose the model with the least number of parameters. The equation for the model was:

$$(1 + 0.1984B^2 + 0.0641B^3 + 0.1502B^4)\nabla_{12}\nabla Y_t = (1 - 0.536B^{12})Z_t$$

This model was proven to be accurate in forecasting the CPI-U values for apparel 12 months into the future.

5. References

- Brockwell, PJ., Davis, RA. 2016. Introduction to Time Series and Forecasting.
- U.S. Bureau of Labor Statistics, Consumer Price Index
- Consumer Price Index for All Urban Consumers: Apparel in U.S. City Average
- Dr. Raya Feldman, UCSB PSTAT 174/274, Time Series Lecture Notes

Appendix

The R code used to create this report.

```r
library(forecast)
library(MASS)
library(astsa)
library(tseries)
library(ggplot2)
library(ggfortify)
library(qpcR)

#loading data
data <- read.csv("CPIAPPNS.csv")
data1 <- data$CPIAPPNS
ts <- ts(data1, start = c(1990,12), end = c(2019,12), frequency = 12)
plot.ts(ts, main = "CPI-U: Apparel in U.S. City Average")

#test and train split
app.train <- data1[c(1:337)]
app.test <- data1[c(338:349)]

plot.ts(app.train)
fit <- lm(app.train ~ as.numeric(1:length(app.train)))
abline(fit, col = "red")

#boxcox transfrom
bc <- boxcox(lm(app.train ~ as.numeric(1:length(app.train))), lambda = c(0,1.5))

y <- ts(as.ts(app.train), frequency = 12)
decomp <- decompose(y)
plot(decomp)

#differencing
varapptrain <- var(app.train)
app.t.12 <- diff(app.train, 12)
plot.ts(app.t.12, main = "Differenced at lag 12")
varappt12 <- var(app.t.12)

app.t.12.1 <- diff(app.t.12, 1)
plot.ts(app.t.12.1, main = "Differenced at lag 12 and lag 1")
varapp121 <- var(app.t.12.1)
abline(h=mean(app.t.12.1), col = "blue")
abline(lm(app.t.12.1 ~ as.numeric(1:length(app.t.12.1))), col = "red")

hist(app.t.12.1,breaks = 20, prob = T, col = "light blue",
     main = "Histogram: Differenced at lag 12 and lag 1")
m <- mean(app.t.12.1)
std <- sqrt(var(app.t.12.1))
curve(dnorm(x,m,std), add= TRUE, col = "black")

#acf and pacf analysis
acf(app.t.12.1, lag.max = 60, main = "ACF")
pacf(app.t.12.1, lag.max = 60, main = "PACF")
```

```r
#fitting models and checking AICc
fit <- Arima(app.train, order=c(4,1,0), seasonal = list(order = c(2,1,1),
                            period = 12),fixed = c(0,NA,0,NA,NA,NA,NA), method = "ML")
fitaicc <- AICc(arima(app.train, order=c(4,1,0), seasonal = list(order = c(2,1,1),
                            period = 12),fixed = c(0,NA,0,NA,NA,NA,NA), method = "ML"))
model <- Arima(app.train, order=c(4,1,0), seasonal = list(order = c(0,1,1),
                            period = 12),fixed = c(0,NA,NA,NA,NA),  method = "ML")
modelaicc <- AICc(arima(app.train, order=c(4,1,0), seasonal = list(order = c(0,1,1),
                           fixed = c(0,NA,NA,NA,NA), period = 12),  method = "ML"))

#checking roots
par(mfrow=c(1,3))
uc.check(pol_ = c(1,0,0.1957,0,0.1418),print_output = F)
uc.check(pol_ = (c(1,-0.3688,-0.2799)),print_output = F)
uc.check(pol_ =(c(1,-0.9184)),print_output = F)

par(mfrow=c(1,2))
uc.check(pol_ =(c(1,0,0.1984,0.0641,0.1502)),print_output = F)
uc.check(pol_ =(c(1,-0.5360)),print_output = F)

#residuals diagnostics model 1
res <- residuals(fit)
layout(matrix(c(1,1,2,3), 2, 2, byrow = TRUE))
plot.ts(res, main = "Model 1 Residuals")
hist(res, breaks = 10, main = "Model 1 Residuals", col = "red", prob = T)
curve(dnorm(x,mean(res),sqrt(var(res))), add= TRUE, col = "black")
qqnorm(res)
qqline(res)
shapiro.test(res)

Box.test(res, lag = 18, type = c("Box-Pierce"), fitdf = 7)
Box.test(res, lag = 18, type = c("Ljung-Box"), fitdf = 7)
Box.test(res^2, lag = 18, type = c("Ljung-Box"), fitdf = 0)

acf(res, main = "ACF of Model 1 Residuals", lag.max = 40)
pacf(res, main = "PACF of Model 1 Residuals", lag.max = 40)

#residuals diagnostics model 2
res1 <- residuals(model)
layout(matrix(c(1,1,2,3), 2, 2, byrow = TRUE))
plot.ts(res1, main = "Model 2 Residuals")
hist(res1, breaks = 10, main = "Model 2 Residuals", col = "green", prob = T)
curve(dnorm(x,mean(res1),sqrt(var(res1))), add= TRUE, col = "black")
qqnorm(res1)
qqline(res1)
shapiro.test(res1)

Box.test(res1, lag = 18, type = c("Box-Pierce"), fitdf = 5)
Box.test(res1, lag = 18, type = c("Ljung-Box"), fitdf = 5)
Box.test(res1^2, lag = 18, type = c("Ljung-Box"), fitdf = 0)

acf(res1, main = "ACF of Model 2 Residuals")
pacf(res1, main = "PACF of Model 2 Residuals")
```

```
#forecast(model)
pred.tr <- predict(model, n.ahead = 12)
U.tr <- pred.tr$pred + 2*pred.tr$se
L.tr <- pred.tr$pred - 2*pred.tr$se
ts.plot(app.train, xlim=c(200,length(app.train)+24), ylim = c(min(app.train),max(U.tr)),
        main = "CPI-U: Apparel in U.S. City Average Forecast")
lines(U.tr, col="blue", lty="dashed")
lines(L.tr, col="blue", lty="dashed")
lines((length(app.train)+1):(length(app.train)+12), pch = 1,pred.tr$pred,col="red")

ts.plot(data1,xlim =c(310,length(app.train)+12), ylim=c(115,max(U.tr)), col = "red",
        main = "CPI-U: Apparel in U.S. City Average Forecast Zoomed")
lines(U.tr,col ="blue",lty = "dashed")
lines(L.tr,col ="blue",lty = "dashed")
lines((length(app.train)+1):(length(app.train)+12), pred.tr$pred,pch = 1 ,col = "black")
```