

Toyota Vehicle Sales - PSTAT 174 Project

Jonathan Palada Rosal

2022-12-08

Abstract:

My motivation is to predict dips and peaks of the car sales throughout the year. This is important because if car manufacturers could have a relatively accurate prediction of when the most/least cars are sold, then they would know when to start receiving more vehicles and when to start sending out advertisements. It is difficult to make predictions due to the difference in sales throughout the months.

In order to have a model that accurately predicts the peaks and dips throughout the year I separated part of the data into a training set. This was done to train the model to make the most accurate predictions. After creating a series plot, I observed that there was a trend, a seasonal component, and the variances varied. I needed to remove those components by performing transformations and the process of differencing. After this was achieved, I made sure the model was invertible and approximately normal. Once that was done I was given the best model to predict was: SARIMA $(0, 1, 1) \times (0, 1, 1)_{12}$. Lastly, I plotted the predicted figures with this model and compared it to the actual values of the original data. I compared the values and concluded that this model is accurate enough to make predictions for future sales of Toyota vehicle sales.

Introduction:

Car sales fluctuates throughout the year due to more interest in specific months. The fluctuation makes it difficult for dealerships to predict the sales of future months. This causes a problem of whether or not these dealerships are going to make their car sales goal. The dataset that I used was a dataset that consisted the sales of cars in Norway from the year 2007 to the year 2017. The dataset contained the year of the sales, the month of the sales, the brand of the car, the quantity sold, and the percentage share total per month. I planned to use the data only from the brand Toyota. The reason why I wanted to use Toyota was because Toyota is one of the well-known car brands in the world. The focus of this project was the quantity sold, the year of the sales, and the month of the sales. In explicit terms, the problem I attempted to answer was when do they have the most/least amount of sales, in order to take advantage of when interest increases/decreases in buying a car. To answer this question, I made a model that will be able to predict future car sales for Toyota.

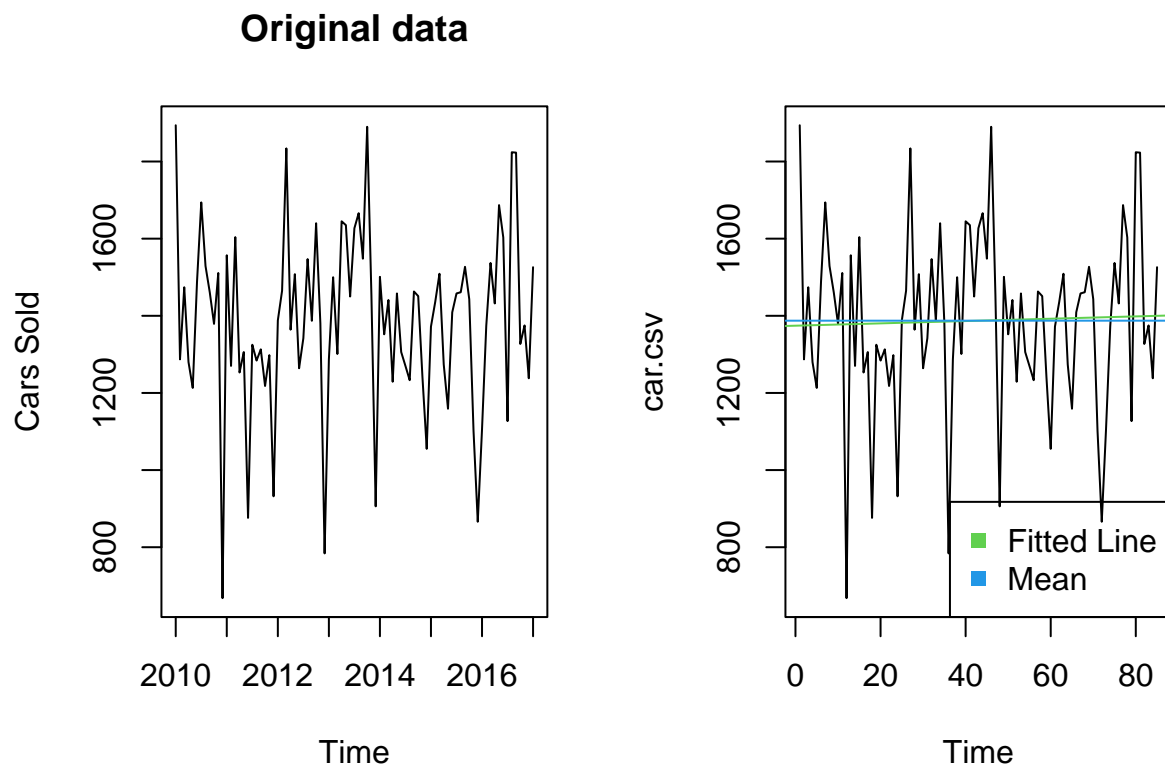
The first step I decided to do was to make a training set. I removed the last 8 points of the dataset to be compared with my forecasting. In order to forecast, we would want our training set to be stationary. Since my training set was not stationary I applied a Box-Cox transformation. After doing the transformation, I displayed some plots to conclude that there was a trend, and a seasonal component. Due to this, I used differencing at 12 then 1 to remove the trend and seasonal component. Examining the ACF and the PACF after differencing, I identified possible values for p's, P's, q's, and Q's. I thought all the values could possibly range from 0:1. I ran all the models to find the lowest AICc. I found the lowest AICc model to be: SARIMA $(0, 1, 1) \times (0, 1, 1)_{12}$. Using this model, I tested invertibility. Once I made sure it was invertible, I ran Shapiro-Wilk, Box-Pierce, and Box-Ljung tests which it all passed. Since the model passed diagnostic checking, this model could be used for forecasting. Examining the forecasted points, I can see that the points

lie within the 95% confidence intervals. After looking at the results of the forecasting, I can conclude that my model is accurate enough to predict future values.

Analysis:

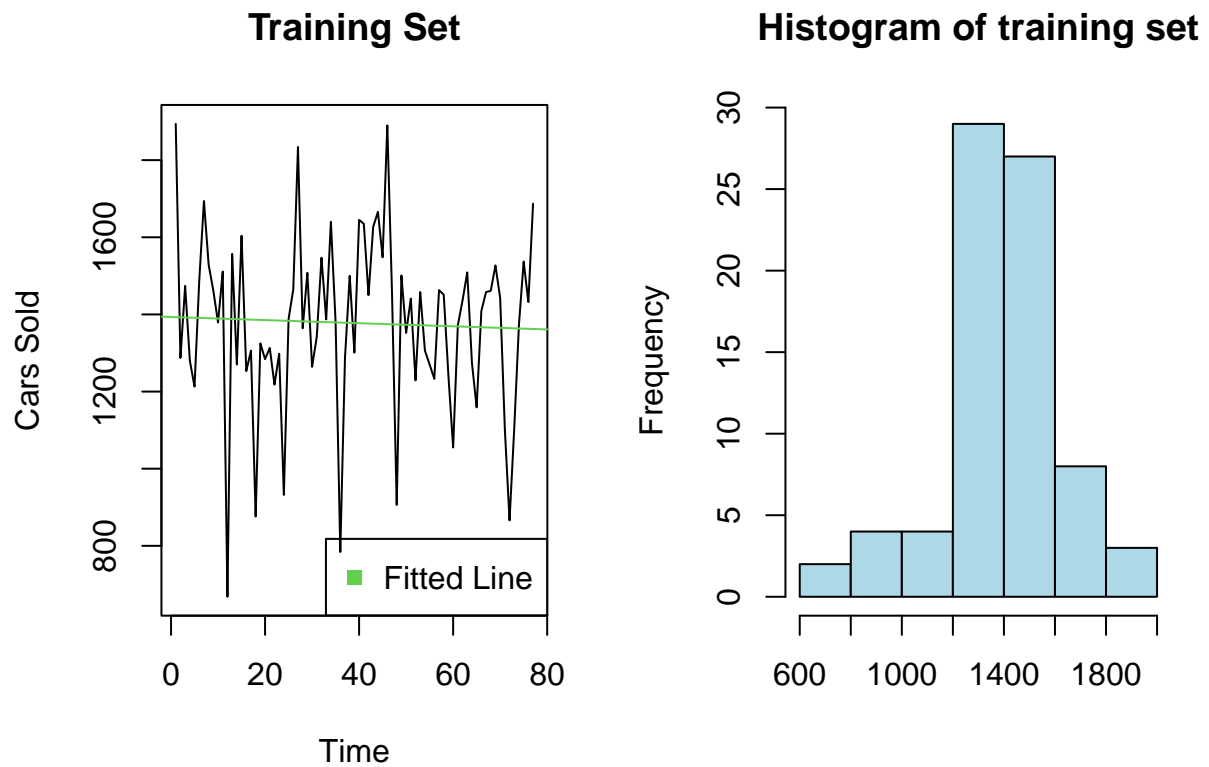
Plotting the original dataset:

I started by plotting a time series plot with the original data. However, I have modded the data by only having Toyota's sales and also by excluding the years, 2007, 2008, 2009. The reason why I cut off those years because there was a huge financial crisis in the years 2007-2009. Including this data would skew the results, and would not be consistent with the rest of the data.



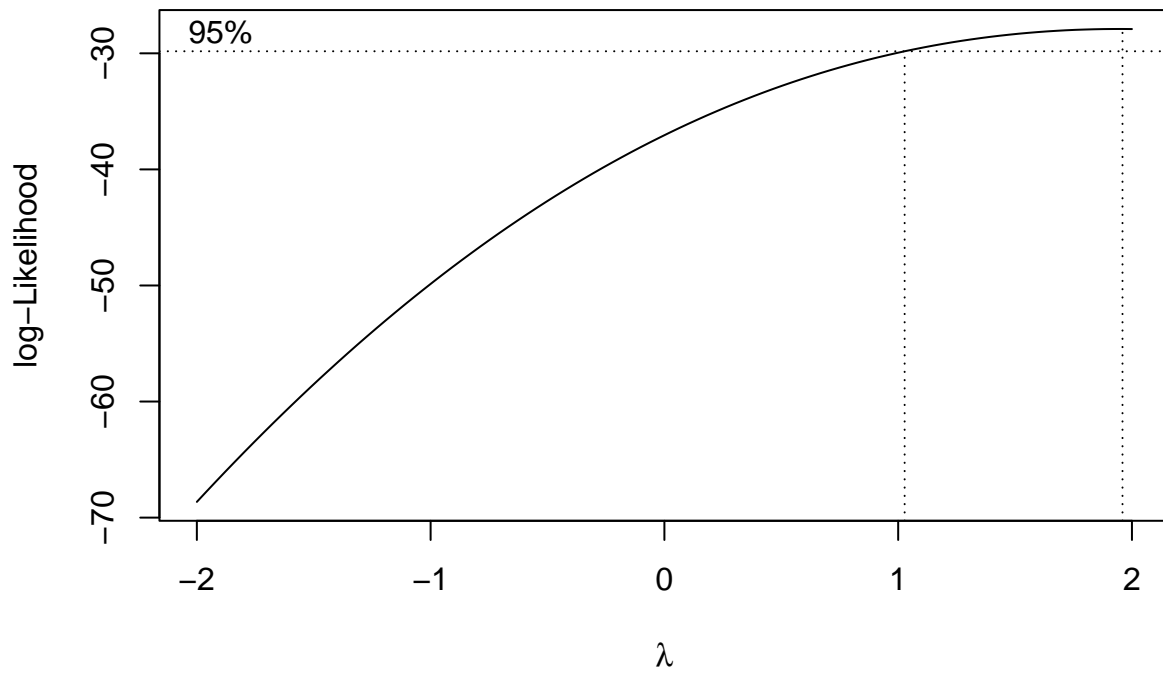
In order to make sure our model was relatively accurate I sectioned some of the data into a training set. I removed the last 8 observations, in order to compare the forecasted values with the original data values later.

Plotting the training set:

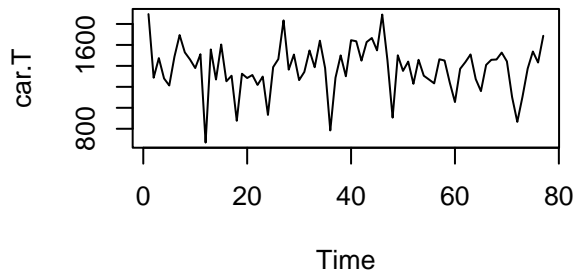


After observing the time series plot, I could see that it had an increasing and decreasing trend in the data. Based on the fitted line, the trend was slightly decreasing. There also seemed to be a relatively strong seasonal component. The variance I noticed seemed to change at different time intervals. The histogram looked relatively normal and almost symmetric. Based on these observations, I would say it would be best to perform a Box-Cox transformation.

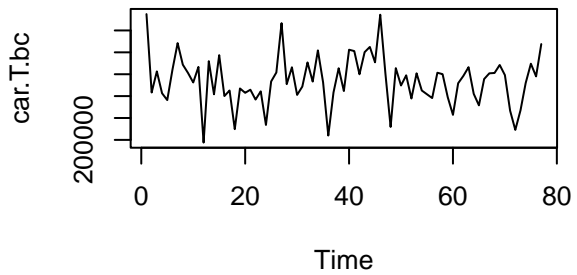
Box-Cox/LOG transformation:



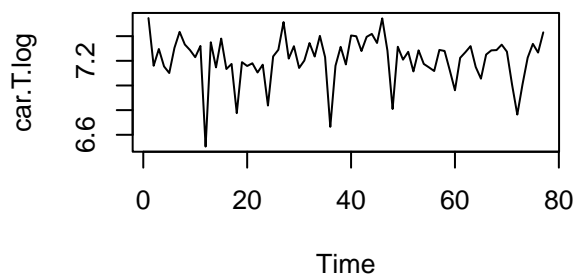
Original, Variance: 52,757.68



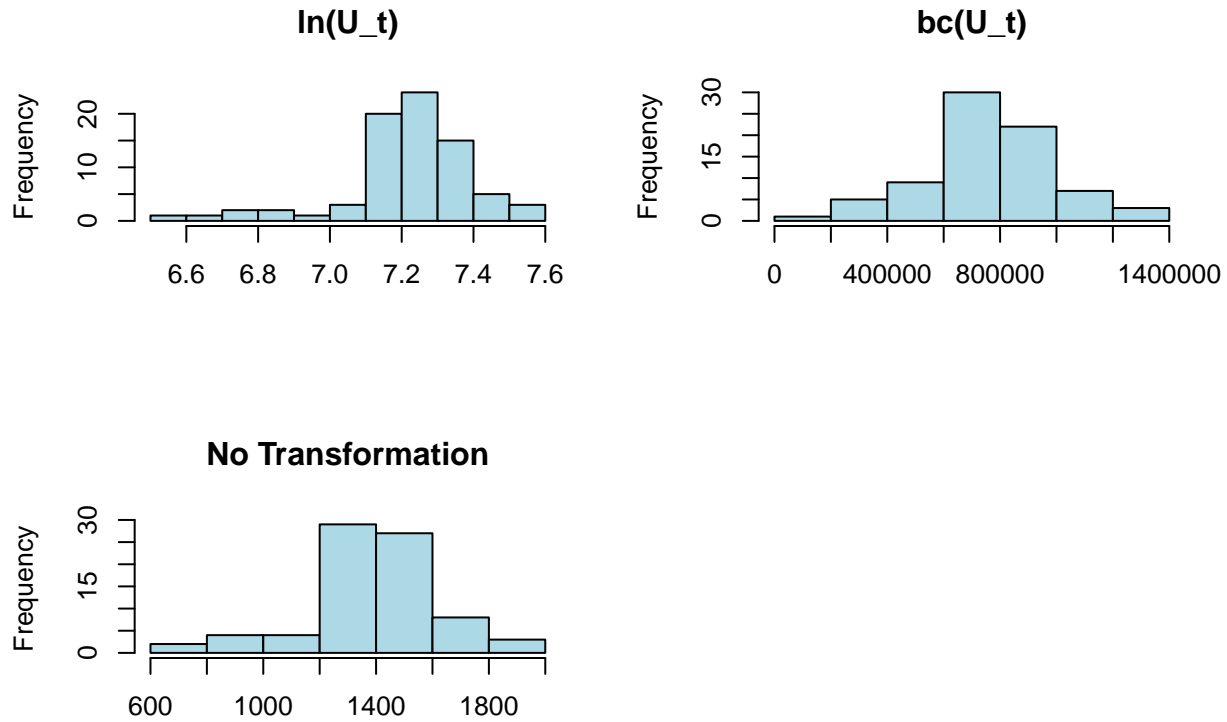
Box-Cox, Variance: 51,463,429,615



LOG, Variance: 0.03446

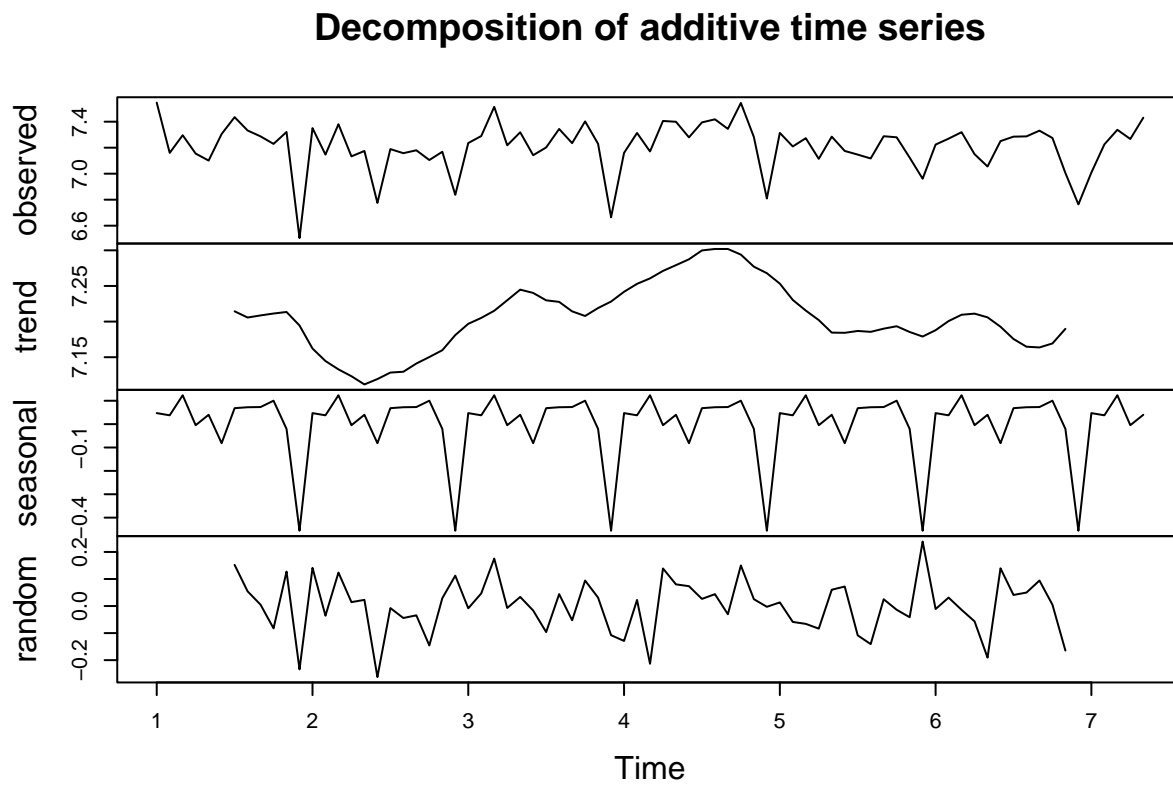


Looking at the 95% confidence interval for the true λ , we could see that the suggested transformations were $Y_t = \frac{1}{\lambda}(X_t^\lambda - 1)$ or no transformation. I also took the LOG to compare to the Box-Cox and the original. Looking at the plots we could see that the LOG plot actually didn't have as sharp dips and tips. This was most prominent around time 20, 50 and 60. Also when looking at the variances, we could see the Box-Cox transformation drastically increases the variance. However, the LOG transformation drastically decreases the variance.



The histogram that looked most normally distributed was the Box-Cox transformation. The histogram of the LOG transformation looked the least like a normal distribution. Based on the histograms, variances, and plots I have decided to go with the LOG transformation because it lowered the variance drastically.

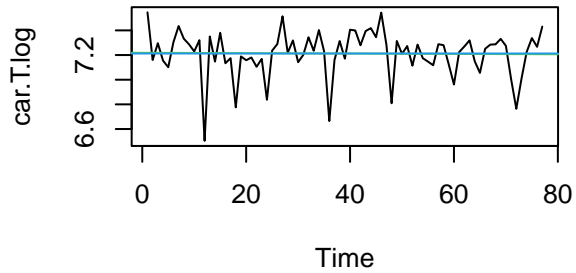
Decomposition of LOG transformed training set:



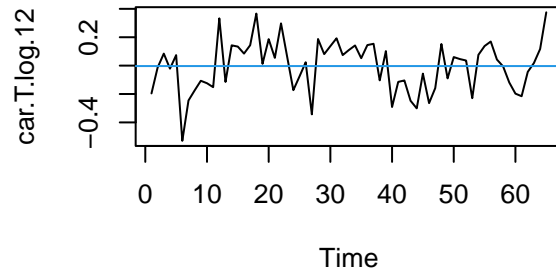
There seemed to be a slight trend and a clear seasonal component. To correct this I performed differencing.

Differencing:

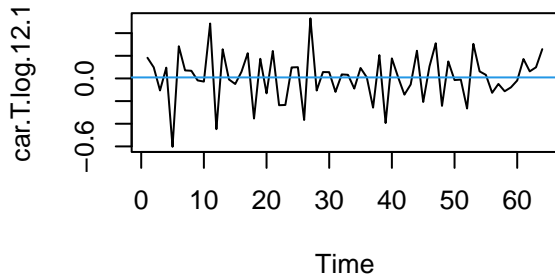
NO difference, Variance: 0.03446



Differenced at 12, Variance: 0.03199



Differenced at 12 then 1, Variance:0.044

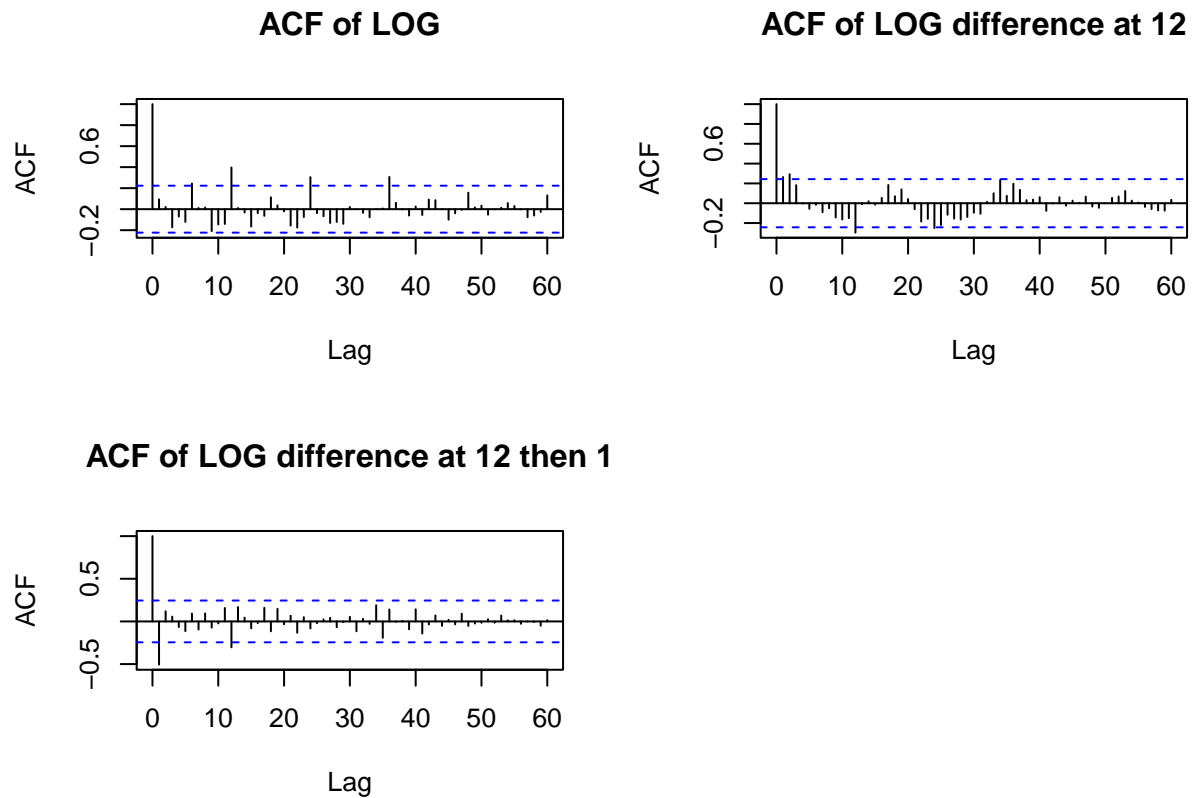


Green Line: Fitted Line

Blue Line: Mean

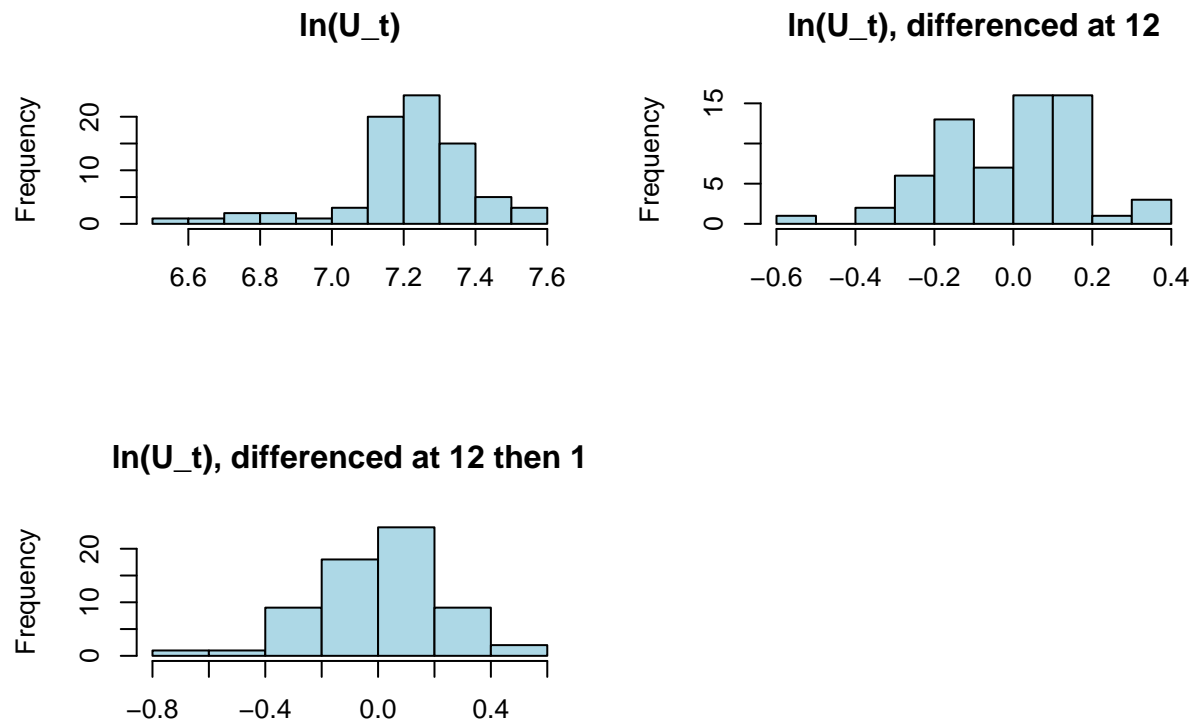
Looking at the different plots we could see that the plot differenced at only lag 12 had the least apparent trend, and the least apparent seasonality. This plot also had the lowest variance (0.03198874) and the mean closest to zero (-0.003156617).

Plots of differences:



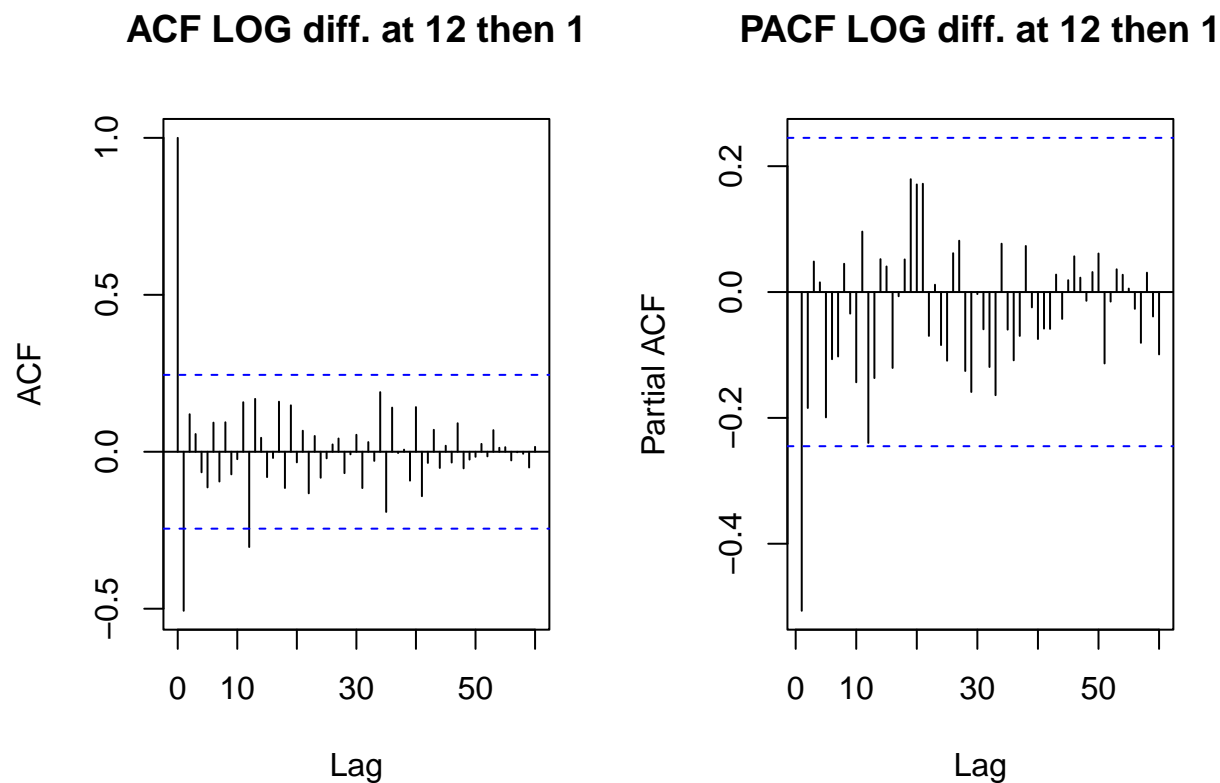
Comparing the ACF's, the non-differenced and differenced at 12 still contained some seasonality, and the ACF decayed slowly which indicated non-stationarity. The ACF differenced at 12 then 1 had no apparent seasonality and the decay corresponded to a stationary process.

Comparing the histograms at different differences:



Comparing histograms, it was clear that the histogram differenced at 12 then at 1 produced the most symmetrical and normal-like distribution. Therefore, I continued with the model that was LOG and differenced at 12 then 1.

ACF and PACF of log then differenced at 12 then 1:



Looking at the ACF and PACF I made the following observations:

ACF outside confidence intervals: 1,12

PACF outside the confidence intervals: 1,12?

List of candidate models to try:

SARIMA for log: s=12, D=1, d=1, Q=0 or 1, q=0, P=0 or 1, p=0.

Trying models now:

##	p	q	P	Q	AICc
## 1	0	0	0	0	-15.91183
## 2	1	0	0	0	-33.20894
## 3	0	1	0	0	-35.44341
## 4	1	1	0	0	-33.81377
## 5	0	0	1	0	-22.17962
## 6	1	0	1	0	-39.05980
## 7	0	1	1	0	-41.31836
## 8	1	1	1	0	-39.23425
## 9	0	0	0	1	-24.83465
## 10	1	0	0	1	-44.34148
## 11	0	1	0	1	-47.91584
## 12	1	1	0	1	-45.75403

```
## 13 0 0 1 1 -22.76444
## 14 1 0 1 1 -42.24412
## 16 1 1 1 1 -43.52801
```

When conducting a code to find the best model based on the lowest AICc, I found that when $p=0$, $q=1$, $P=0$, $Q=1$ we got the lowest AICc of -47.91584. But as an alternative model I also recorded the second lowest AICc of -45.75403 when $p=1$, $q=1$, $P=0$, $Q=1$.

Models:

This is the first model when we have $p=1$, $q=1$, $P=0$, $Q=1$.

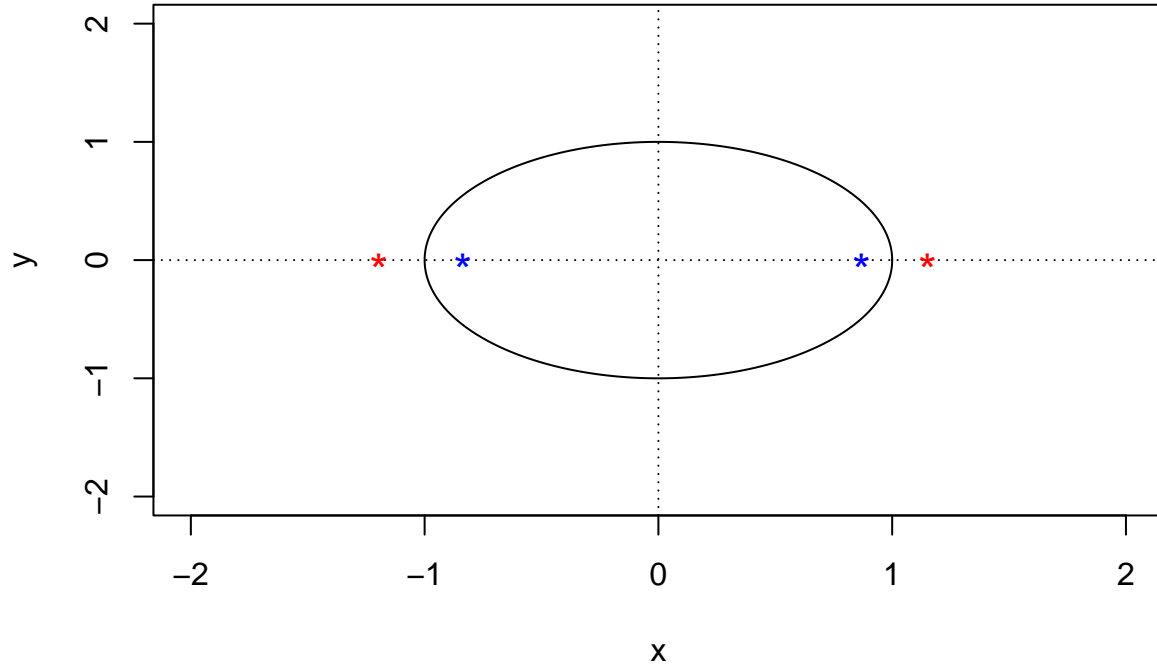
```
##
## Call:
## arima(x = car.T.log, order = c(1, 1, 1), seasonal = list(order = c(0, 1, 1),
##   period = 12), method = "ML")
##
## Coefficients:
##          ar1          ma1          sma1
##      -0.0322  -0.7265  -0.6743
## s.e.   0.4869   0.4947   0.1789
##
## sigma^2 estimated as 0.02219:  log likelihood = 27.04,  aic = -46.08
```

This is the second model when we have $p=0$, $q=1$, $P=0$, $Q=1$.

```
##
## Call:
## arima(x = car.T.log, order = c(df$p[final], 1, df$q[final]), seasonal = list(order = c(df$P[final],
##   1, df$Q[final]), period = 12), method = "ML")
##
## Coefficients:
##          ma1          sma1
##      -0.7569  -0.6724
## s.e.   0.1378   0.1754
##
## sigma^2 estimated as 0.02218:  log likelihood = 27.04,  aic = -48.08
```

Checking invertibility:

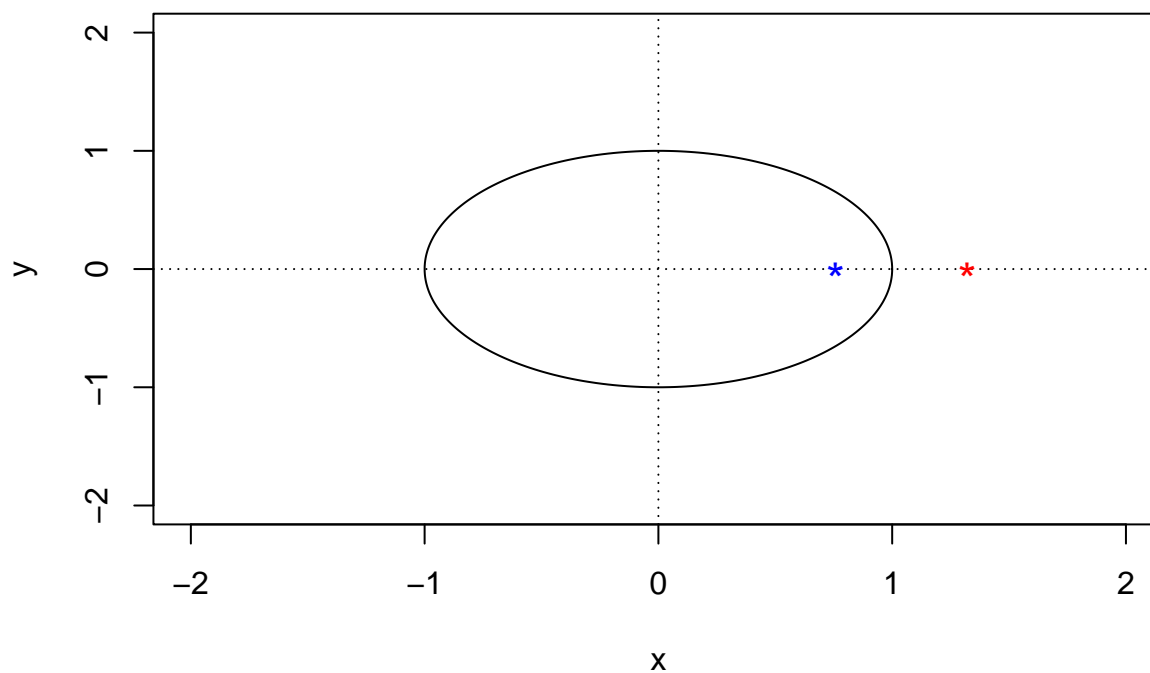
First Model:



```
## [1] 1.151276-0i -1.195598+0i
```

We know the model is invertible because $|\theta_1| < 1$ and $|\underline{\theta}| < 1$. All roots (in red) are outside of the unit circle; thus, both models are causal and invertible. Blue stars correspond to inverse roots, should be inside unit circle. No roots should be on a unit circle. With the first model we get 1.14358 and -1.18730. When taking the absolute value of those values, they are greater than 1 and outside of the unit circle as well. Therefore it is invertible.

Second Model:



```
## [1] 1.321178+0i
```

With the second model, the root is 1.321178 which is greater than 1 and we can see that the root is outside the unit circle which also confirms that it is invertible.

Diagnostic Checking 1st model:

```
##
##  Shapiro-Wilk normality test
##
## data:  res
## W = 0.97535, p-value = 0.1397

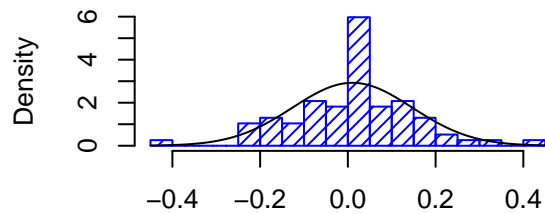
##
##  Box-Pierce test
##
## data:  res
## X-squared = 8.4075, df = 6, p-value = 0.2097

##
##  Box-Ljung test
##
```

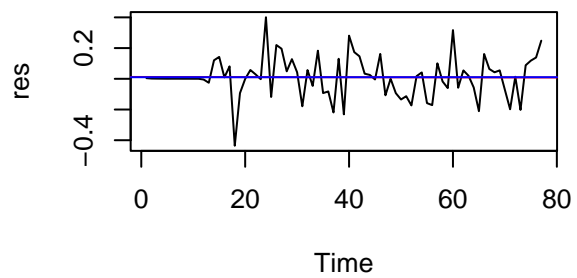
```
## data:  res
## X-squared = 9.2181, df = 6, p-value = 0.1617

##
## Box-Ljung test
##
## data:  res^2
## X-squared = 8.3904, df = 9, p-value = 0.4953
```

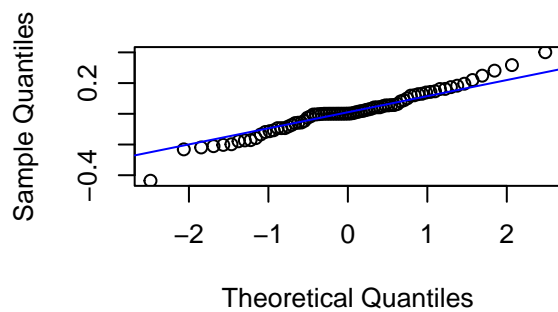
Histogram of res



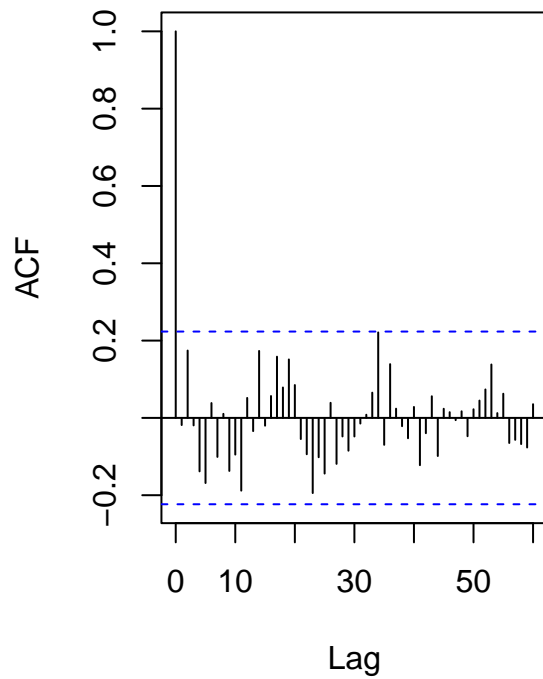
Time-Series Plot



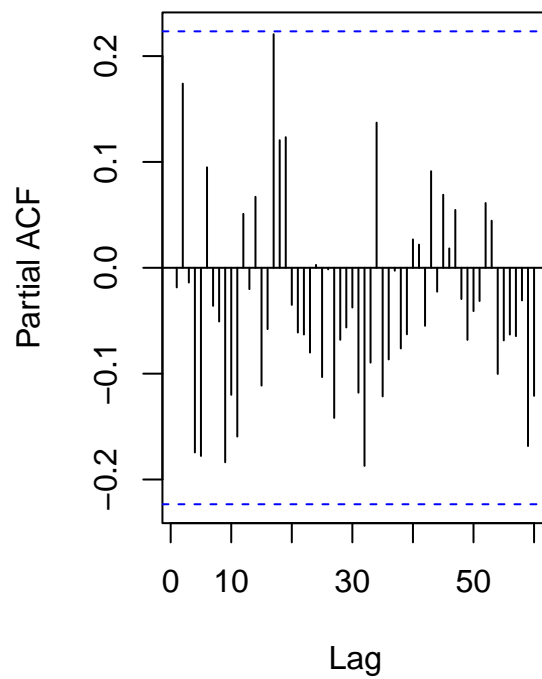
Normal Q-Q Plot for Model



ACF of Residuals



PACF of Residuals



```
##
## Call:
## ar(x = res, aic = TRUE, order.max = NULL, method = c("yule-walker"))
##
##
## Order selected 0   sigma^2 estimated as  0.01859
```

Diagnostic Checking 2nd model:

```
##
##  Shapiro-Wilk normality test
##
## data:  res
## W = 0.97677, p-value = 0.1707

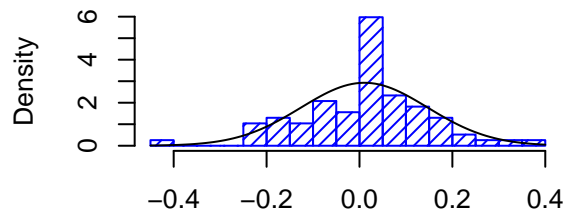
##
##  Box-Pierce test
##
## data:  res
## X-squared = 8.6017, df = 7, p-value = 0.2825

##
##  Box-Ljung test
##
```

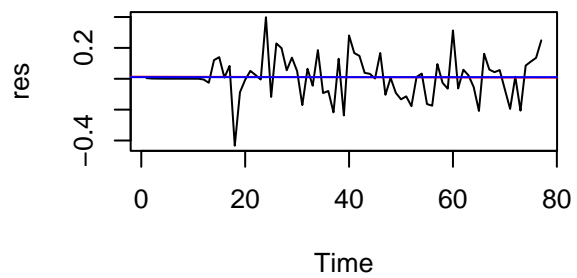
```
## data:  res
## X-squared = 9.4229, df = 7, p-value = 0.2237

##
## Box-Ljung test
##
## data:  res^2
## X-squared = 8.2131, df = 9, p-value = 0.5128
```

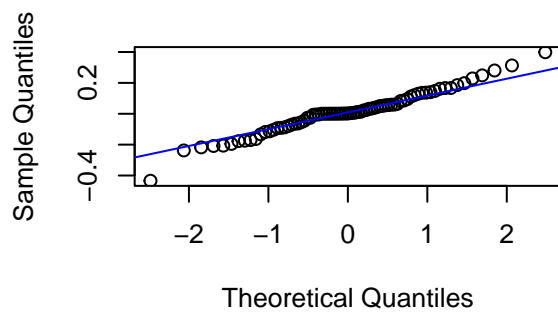
Histogram of res

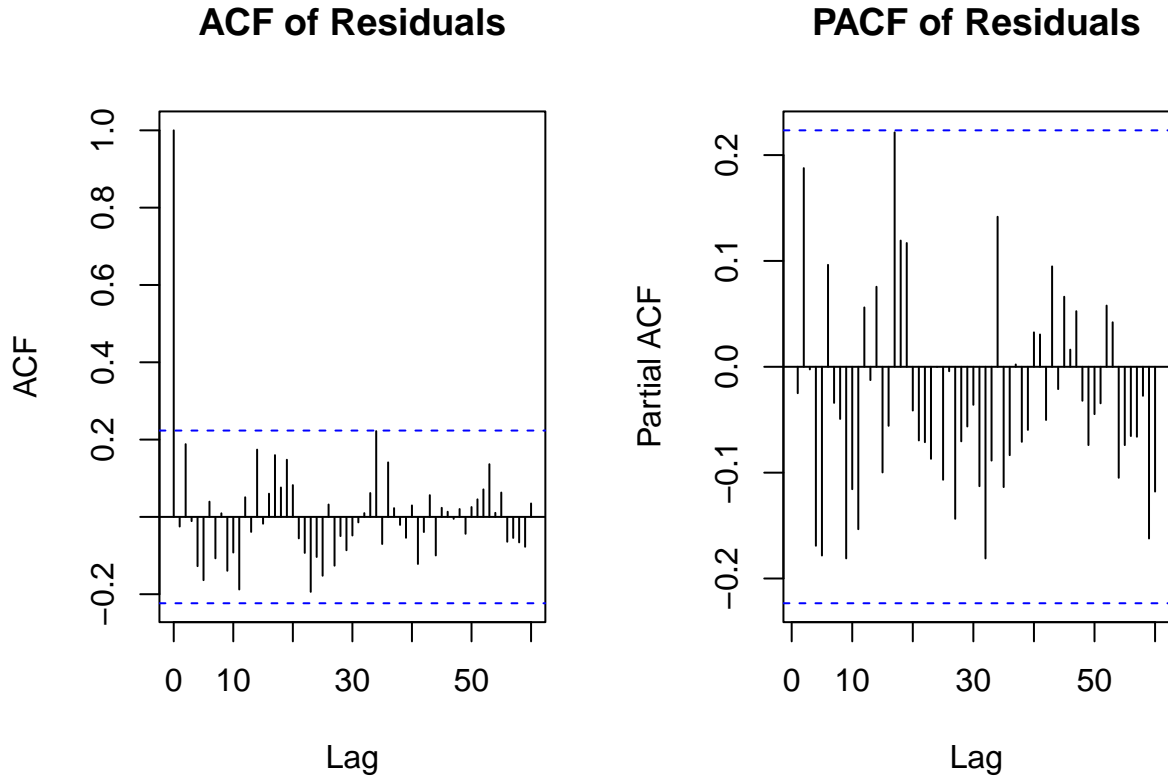


Time-Series Plot



Normal Q-Q Plot for Model





```
##
## Call:
## ar(x = res, aic = TRUE, order.max = NULL, method = c("yule-walker"))
##
##
## Order selected 0   sigma^2 estimated as  0.01858
```

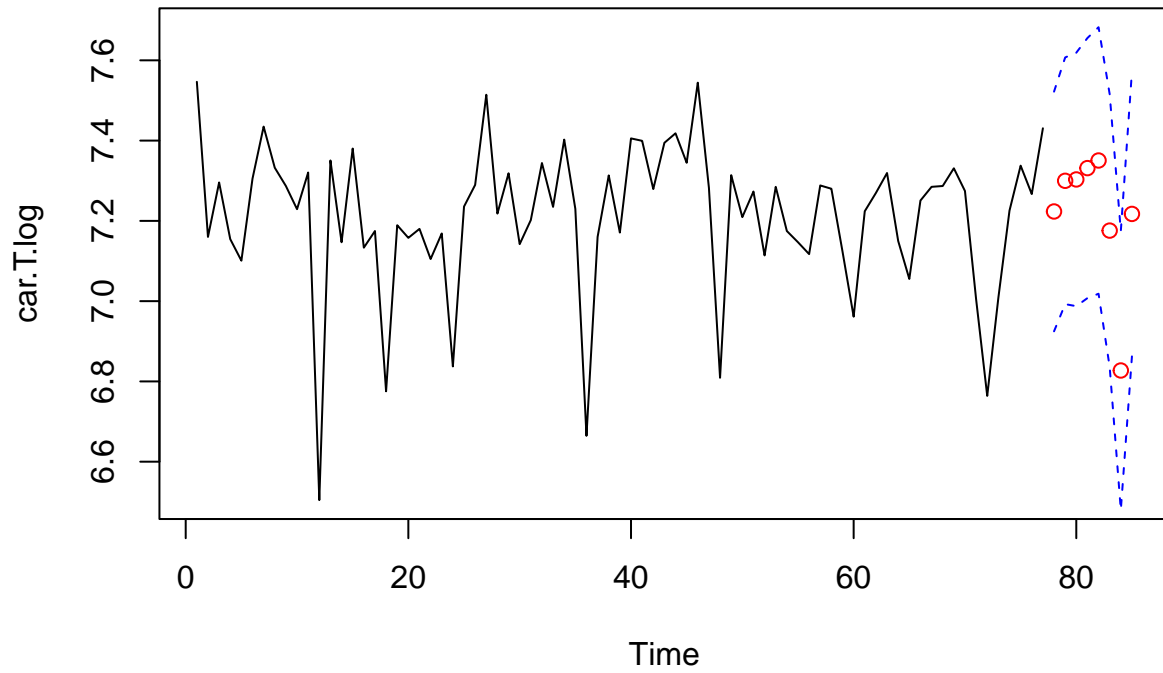
When performing the different types of tests on the two models, we saw that the models passed all the tests because its p-values are greater than the significance level of 0.05. Looking at the time series plot we saw that there was no trends, no change in variances nor seasonal components. Looking at the Q-Q plots and histograms, we could see that they were approximately normally distributed. The ACF's also resembled white noise since there was no lags outside the confidence interval. Both models were also invertible. Even though both models passed all the tests, I decided to go with the model with the lowest AICc. The reasoning for this decision, was not only did it have a lower AICc, it also was slightly closer to a normal distribution. Another benefit of the second model was that it had one less coefficient. Based on all these testings, I concluded that the second model was best for forecasting.

$$\nabla_1 \nabla_{12} \ln(U_t) = (1 - 0.7569_{(0.1378)} B)(1 - 0.6724_{(0.1754)} B^{12}) Z_t$$

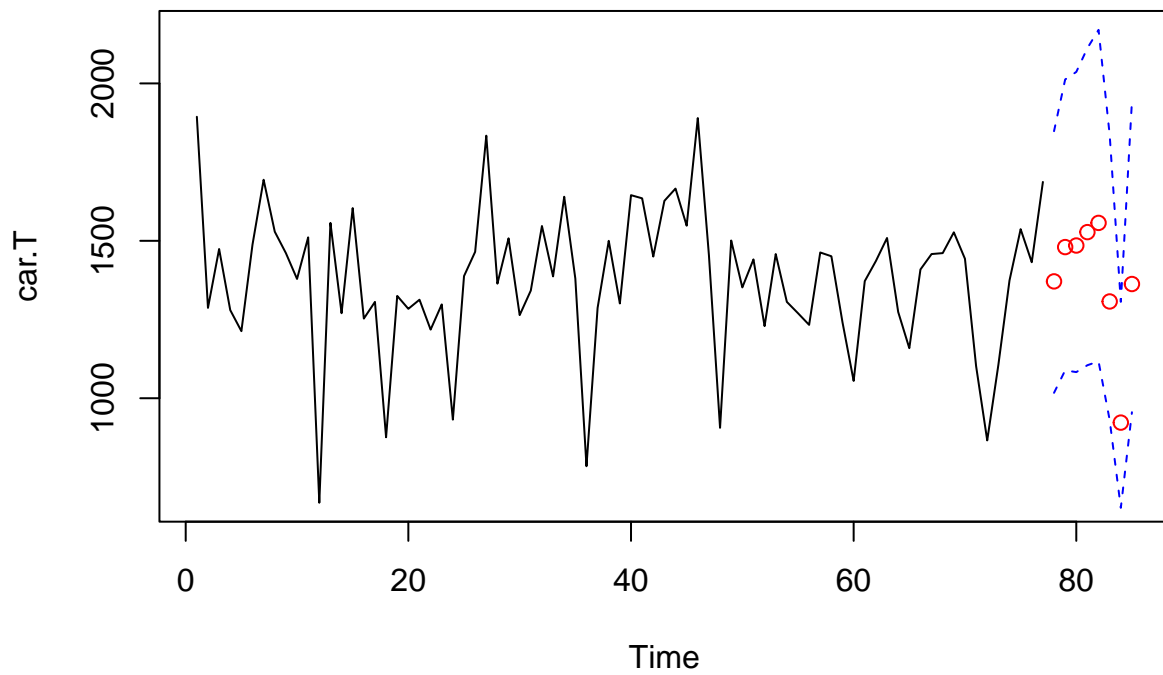
$$\hat{\sigma}_Z^2 = 0.02218$$

Forecasting:

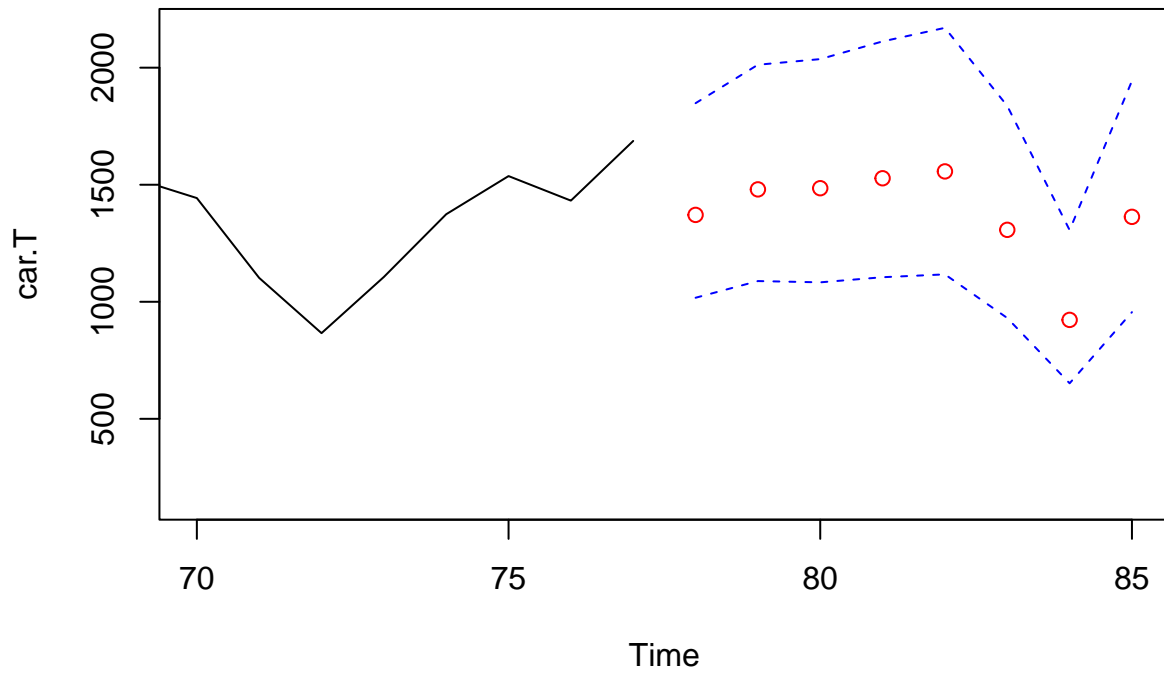
Log Differenced Training Set



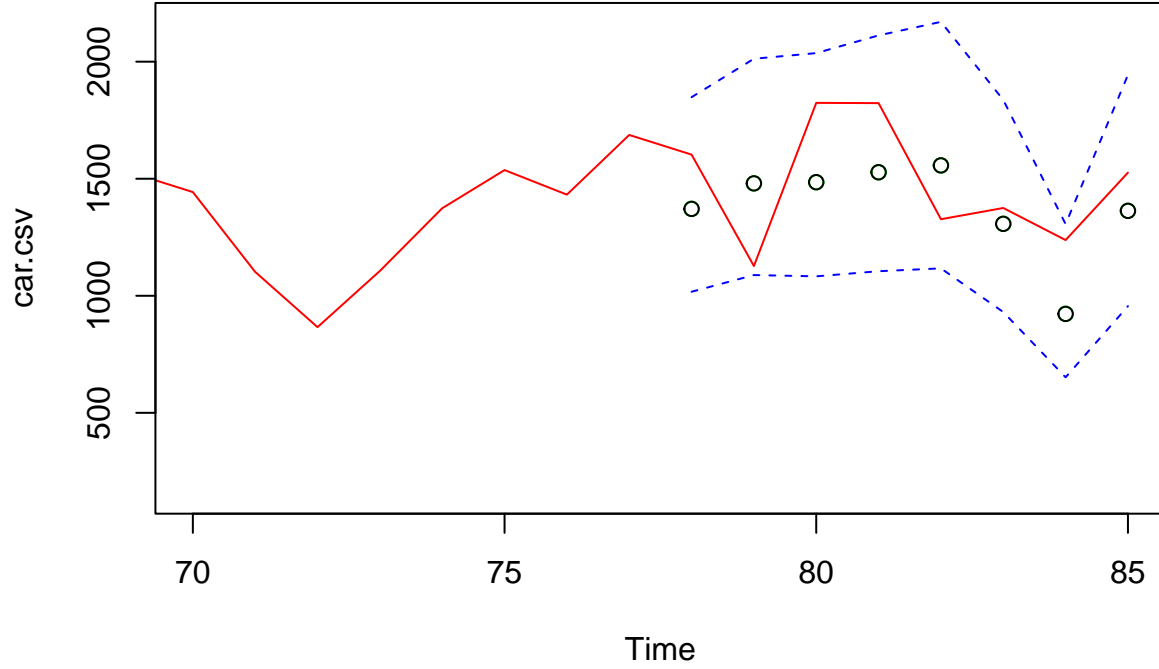
Original Training



Focusing on predicted data:



True values and predicted data:



I forecasted the 8 points with the confidence intervals using the decided model. On the same plot, I plotted the original points from the dataset to compare. Based on the plot, we could see that the original line was within the confidence intervals of the forecasted values. Based on this, we can confidently say that this model accurately forecasts the data.

Conclusions:

Summary:

The goal of this project was to provide an accurate model that predicted the future Toyota vehicle sales of each month. This goal was achieved with this model:

$$\nabla_1 \nabla_{12} \ln(U_t) = (1 - 0.7569_{(0.1378)} B)(1 - 0.6724_{(0.1754)} B^{12}) Z_t$$

In order to get this model, we divided the data set, log transformed it, differenced it at 12 then 1, and diagnostic checked it. After this was done this model was able to make relative accurate predictions that were close to the true values of the original data set. Using this model, Toyota dealerships could have an accurate projection of what their car sales will be like. With this information, they could make adjustments to maximize profits for each month.

Acknowledgements:

I would like to thank my TA Ming Hu for providing me the proper coding techniques and procedures to work on this project. I especially want to thank my professor Raya Feldman for teaching me the information needed to apply a time-series plot, and create a proper model for forecasting. I also want to thank my professor for taking the time to provide assistance outside of class with my endeavors throughout this project.

References:

Dataset:

dmi3kno. (2017, February 18). New car sales in Norway. Kaggle. Retrieved November 29, 2022, from <https://www.kaggle.com/datasets/dmi3kno/newcarsalesnorway>

Confirmation of Financial Crisis:

Greenbaum, S. I., Thakor, A. V., & Boot, A. W. A. (Eds.). (2015, September 4). The 2007–2009 financial crisis and other financial crises. Contemporary Financial Intermediation (Third Edition). Retrieved November 29, 2022, from <https://www.sciencedirect.com/science/article/pii/B9780124051966000148#:~:text=The%20financial%20crisis%20of%202007%E2%80%932009%20is%20widely%20regarded,the%20United%20States%29%20was%20bought%20out%20by%20Citigroup>

Appendix:

```
## Analysis
#Import Data
car.csv <- read.csv("C:\\Users\\Jonat\\OneDrive\\schoolwork\\PSTAT 174\\Final Project\\norway_new_car_sales.csv")
#car_1 <- car /> filter(Make=="Toyota")
car.csv <- car.csv[car.csv$Make=="Toyota",]
car.csv <- na.omit(car.csv)
car.csv <- subset(car.csv, !(Year %in% c(2007, 2008, 2009)))
car.csv <- car.csv[,-0:-3]
car.csv <- car.csv[,-2] #85 observations
car.T <- car.csv[c(1:77)]
car.Testing <- car.csv[c(78:85)]

## Plotting the original dataset
par(mfrow=c(1,2))
car = ts(car.csv, c(2010,1), c(2017,1),12)
ts.plot(car,main = "Original data", ylab = "Cars Sold")
plot.ts(car.csv)
nt=length(car.csv)
fit <- lm(car.csv ~ as.numeric(1:nt)); abline(fit, col="3")
abline(h = mean(car.csv), col="4")
legend("bottomright", legend = c("Fitted Line", "Mean"), pch = rep(15, 4), col = 3:4)

## Plotting the training set:
par(mfrow=c(1,2))
plot.ts(car.T, main = "Training Set", ylab = "Cars Sold")
fit <- lm(car.T ~ as.numeric(1:length(car.T)));abline(fit, col="3")
legend("bottomright", legend = c("Fitted Line"), pch = rep(15, 4), col = 3)
```

```

hist(car.T, col = "light blue", xlab = "", main = "Histogram of training set") #Looks very roughly normal

## Transformations:
bcTransform <- boxcox(car.T~ as.numeric(1:length(car.T)))
lambda=bcTransform$x[which(bcTransform$y == max(bcTransform$y))]
car.T.bc = (1/lambda)*(car.T^lambda-1)
car.T.log <- log(car.T)
par(mfrow=c(2,2))
plot.ts(car.T, main = "Original, Variance: 52,757.68")
plot.ts(car.T.bc, main = "Box-Cox, Variance: 51,463,429,615")
plot.ts(car.T.log, main = "LOG, Variance: 0.03446") #0.03445896

var(car.T)
var(car.T.bc) # highest variance
var(car.T.log) # lowest variance
par(mfrow=c(2,2))

hist(car.T.log, col="light blue", xlab="", main="ln(U_t)") # Has the lowest variance
hist(car.T.bc, col="light blue", xlab="", main="bc(U_t)") #looks the most symmetric and normally distributed
hist(car.T, col = "light blue", xlab = "", main = "No Transformation")

## Decomposition of LOG transformed training set:
y <- ts(as.ts(car.T.log), frequency = 12)
decomposition <- decompose(y)
plot(decomposition)

## Differencing:
par(mfrow=c(2,2))
plot.ts(car.T.log, main = "NO difference, Variance: 0.03446") #0.03445896
fit1 <- lm(car.T.log ~ as.numeric(1:length(car.T.log)));abline(fit1, col=3)
abline(h=mean(car.T.log), col=4)

car.T.log.12 <- diff(car.T.log, lag =12)
plot.ts(car.T.log.12, main = "Differenced at 12, Variance: 0.03199") #0.03198874
fit2 <- lm(car.T.log.12 ~ as.numeric(1:length(car.T.log.12)));abline(fit1, col=3)
abline(h=mean(car.T.log.12), col=4)
#This is the model I am going to choose so far. There is no apparent seasonality, has the lowest variance

car.T.log.12.1 <- diff(car.T.log.12, lag =1)
plot.ts(car.T.log.12.1, main = "Differenced at 12 then 1, Variance:0.04488") #0.04487819
fit3 <- lm(car.T.log.12.1 ~ as.numeric(1:length(car.T.log.12.1)));abline(fit1, col=3)
abline(h=mean(car.T.log.12.1), col=4)

var(car.T.log) #0.03445896
mean(car.T.log) #7.212436
var(car.T.log.12) #0.03198874 went lower
mean(car.T.log.12) #-0.003156617
var(car.T.log.12.1) #variance went higher, 0.04487819 went higher
mean(car.T.log.12.1) #0.008926943 went lower

## Plots of differences:
par(mfrow=c(2,2))
acf(car.T.log, lag.max = 60, main = "ACF of LOG")

```

```

acf(car.T.log.12, lag.max = 60, main = "ACF of LOG difference at 12")
acf(car.T.log.12.1, lag.max = 60, main = "ACF of LOG difference at 12 then 1") #ACF decays correspondin

## Comparing the histograms at different differences:
par(mfrow=c(2,2))
hist(car.T.log, col="light blue", xlab="", main="ln(U_t)")
hist(car.T.log.12, col="light blue", xlab="", main="ln(U_t), differenced at 12")
hist(car.T.log.12.1, col="light blue", xlab="", main="ln(U_t), differenced at 12 then 1") #This looks a

## ACF and PACF of log then differenced at 12 then 1:
par(mfrow=c(1,2))
acf(car.T.log.12.1, lag.max = 60, main = "ACF of LOG difference at 12 then 1")
#ACF outside confidence intervals: 1,12
pacf(car.T.log.12.1, lag.max = 60, main = "PACF of LOG difference at 12 then 1")
#PACF outside the confidence intervals: 1,12?

#List of candidate models to try:
#SARIMA for log: s=12, D=1, d=1, Q=0 or 1, q=0, P=0 or 1, p=0.

## Trying models now:
df <- expand.grid(p=0:1, q=0:1, P=0:1, Q=0:1)
df <- cbind(df, AICc=NA)
# Compute AICc:
for (i in 1:nrow(df)) {
  sarima.obj <- NULL
  try(arima.obj <- arima(car.T.log, order=c(df$p[i], 1, df$q[i]),
seasonal=list(order=c(df$P[i], 1, df$Q[i]), period=12),
method="ML"))
  if (!is.null(arima.obj)) { df$AICc[i] <- AICc(arima.obj) }
  # print(df[i, ])
}
#df[which.min(df$AICc), ]
df[(df$AICc),] # second lowest AIC -45.75403 when p=1, q=1, P=0, Q=1
# when p=0, q=1, P=0, Q=1 we get the lowest AIC -47.91584

final <- which.min(df$AICc)
fit4.1 <- arima(car.T.log, order=c(1, 1, 1),
seasonal=list(order=c(0, 1, 1), period=12),
method="ML")
fit4.1

fit4 <- arima(car.T.log, order=c(df$p[final], 1, df$q[final]),
seasonal=list(order=c(df$P[final], 1, df$Q[final]), period=12),
method="ML")
fit4

### First Model:
plot.roots(NULL,polyroot(c(1, -0.0322, -0.7365)))
polyroot(c(1,-0.0322, -0.7365)) # both roots are outside of the unit circle

### Second Model:
plot.roots(NULL,polyroot(c(1, -0.7569)))#the one root is 1.321178+0i
# which is outside the unit circle which

```

```

polyroot(c(1,-0.7569))

## Diagnostic Checking 1st model:
res <- residuals(fit4.1)
shapiro.test(res)
# p-value is greater than 0.05 so it passes the shapiro.test
Box.test(res, lag = 9, type = c("Box-Pierce"), fitdf = 3)
Box.test(res, lag = 9, type = c("Ljung-Box"), fitdf = 3)
Box.test(res^2, lag = 9, type = c("Ljung-Box"), fitdf = 0)
#passes all the tests, all p-values are larger than 0.05
par(mfrow=c(2,2))
hist(res,density=20,breaks=20, col="blue", xlab="", prob=TRUE)
# Approximately normal and almost symmetric.
m <- mean(res)
std <- sqrt(var(res))
curve( dnorm(x,m,std), add=TRUE )
plot.ts(res,main = "Time-Series Plot")
fitt <- lm(res ~ as.numeric(1:length(res))); abline(fitt, col="red")
abline(h=mean(res), col="blue")
qqnorm(res,main= "Normal Q-Q Plot for Model")
qqline(res,col="blue")
# Looks approximately normal
par(mfrow=c(1,2))
acf(res, lag.max=60, main = "ACF of Residuals")
pacf(res, lag.max=60, main = "PACF of Residuals")
# All ACF and PACF are within confidence intervals and can be counted as zero.
#acf(res^2, lag.max=60, main = "ACF of Residuals^2")
ar(res, aic = TRUE, order.max = NULL, method = c("yule-walker"))
# Fitted residuals to AR(0), i.e. WN
# passed diagnostic checking ready to be used for forecasting

## Diagnostic Checking 2nd model:
res <- residuals(fit4)
shapiro.test(res)
# p-value is greater than 0.05 so it passes the shapiro.test
Box.test(res, lag = 9, type = c("Box-Pierce"), fitdf = 2)
Box.test(res, lag = 9, type = c("Ljung-Box"), fitdf = 2)
Box.test(res^2, lag = 9, type = c("Ljung-Box"), fitdf = 0)
#passes all the tests, all p-values are larger than 0.05
par(mfrow=c(2,2))
hist(res,density=20,breaks=20, col="blue", xlab="", prob=TRUE)
# Approximately normal and almost symmetric.
m <- mean(res)
std <- sqrt(var(res))
curve( dnorm(x,m,std), add=TRUE )
plot.ts(res,main = "Time-Series Plot")
fitt <- lm(res ~ as.numeric(1:length(res))); abline(fitt, col="red")
abline(h=mean(res), col="blue")
qqnorm(res,main= "Normal Q-Q Plot for Model")
qqline(res,col="blue")
# Looks approximately normal
par(mfrow=c(1,2))
acf(res, lag.max=60, main = "ACF of Residuals")

```



```

pacf(res, lag.max=60, main = "PACF of Residuals")
# All ACF and PACF are within confidence intervals and can be counted as zero.
#acf(res^2, lag.max=60, main = "ACF of Residuals^2")
ar(res, aic = TRUE, order.max = NULL, method = c("yule-walker"))
# Fitted residuals to AR(0), i.e. WN
# passed diagnostic checking ready to be used for forecasting

## Forecasting:
fit4
forecast(fit4)

#par(mfrow=c(1,2))
pred.tr <- predict(fit4, n.ahead = 8)
U.tr = pred.tr$pred + 2*pred.tr$se #upper bound
L.tr = pred.tr$pred - 2*pred.tr$se #lower bound
ts.plot(car.T.log, xlim=c(1,length(car.T.log)+8), ylim = c(min(car.T.log),max(U.tr)), main = "Log Differenced Data")
lines(U.tr, col="blue", lty="dashed")
lines(L.tr, col="blue", lty="dashed")
points((length(car.T.log)+1):(length(car.T.log)+8), pred.tr$pred, col="red")

pred.orig <- exp(pred.tr$pred)
U = exp(U.tr)
L = exp(L.tr)
ts.plot(car.T, xlim=c(1, length(car.T)+8), ylim = c(min(car.T), max(U)), main = "Original Training Data")
lines(U, col="blue", lty="dashed")
lines(L, col="blue", lty="dashed")
points((length(car.T)+1):(length(car.T)+8), pred.orig, col="red")

## Focusing on predicted data:
ts.plot(car.T, xlim = c(70,length(car.T)+8), ylim = c(150,max(U)))
lines(U, col="blue", lty="dashed")
lines(L, col="blue", lty="dashed")
points((length(car.T)+1):(length(car.T)+8), pred.orig, col="red")

## True values and predicted data:
ts.plot(car.csv, xlim = c(70,length(car.T)+8), ylim = c(150,max(U)), col="red")
lines(U, col="blue", lty="dashed")
lines(L, col="blue", lty="dashed")
points((length(car.T)+1):(length(car.T)+8), pred.orig, col="green")
points((length(car.T)+1):(length(car.T)+8), pred.orig, col="black")

```