Car Sales Prediction

James(Changhwan) Han (3923257)

12/2/2022

Contents

Car Sales Prediction

```
# Required Library
library(MASS)
library(forecast)
library(qpcR)
library(ggplot2)
library(ldsr) # perform inverse Box-Cox transform
```

Abstract

The "New Car Sales in Norway" dataset describes monthly car sales between 2007 and 2016. As an international student who flies back to my country a lot, I noticed that the prices for flight were way more expensive in certain months. And I was curious if car sales have same logic in it. "Are cars more expensive in certain months?"

In order to validate my assumption I used time series including transforming data and make a model to predict future car sales. After fitting a model, I performed diagnostic checking to see if the model is validate. From the prediction, I couldn't find any differences between months but it would give us better insights with having more data.

Introduction

The dataset includes a total of 120 observations from January 2007 to December 2016. I was always wondering when the best time is to buy a new car and this dataset caught my attention. My goal in this project is to predict car sales, however, considering the lack of observation, I used 12 observations of 2016 as a testset to validate the prediction.

In order to predict car sales, I used time series techniques including box-cox transformation, comparing acfs/pacfs, differencing, AICc computation, and diagnosis checking. After doing all the model transformations, I compared three different models out of 11 possible models, and chose one model that had the best result in diagnosis checking. All 11 possible models had low p-values for Shapiro-test so the model that had the highest p-value of 0.04635 and passed all the diagnostic tests were chosen. Differencing at different lags or applying different values of lambda for Box-Cox transformation didn't improve the model performance.

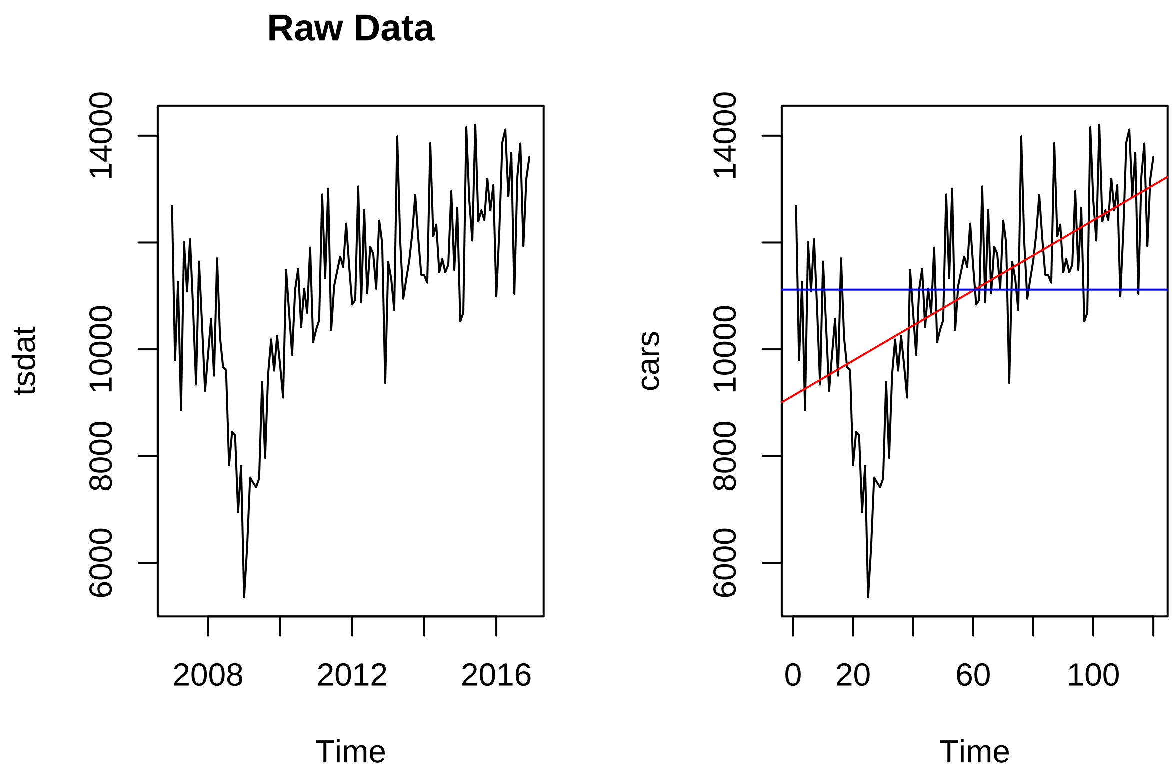
Both predictions of transformed data and original data were within the confidence interval. However, the prediction was almost linear and was not best at giving meaningful insight but having more data would have possibly given better insights.
The dataset was collected from Kaggle, https://www.kaggle.com/datasets/dmi3kno/newcarsalesnorway and R was used throughout the project.

Sections

Car Sales Data

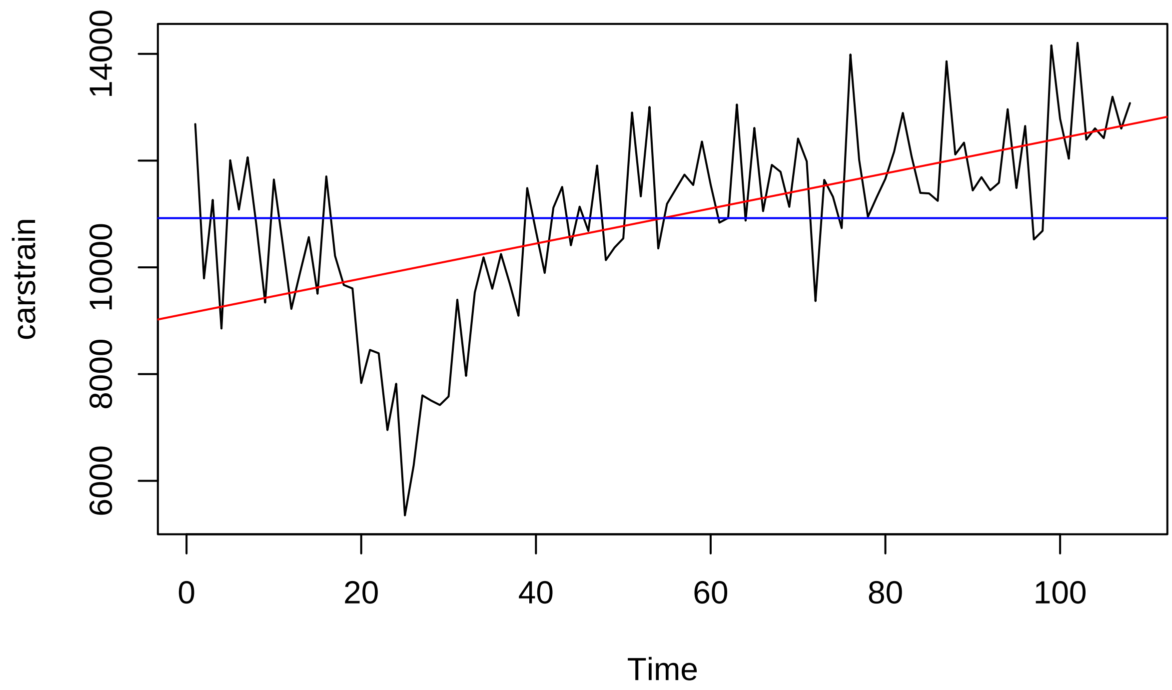
```r
# load data
cars <- scan("norway_new_car_sales_by_month.txt")
```

```r
par(mfrow=c(1,2))

# plot of data with years on x-axis
tsdat <- ts(cars, start = c(2007,1), end = c(2016,12), frequency = 12)

ts.plot(tsdat, main = "Raw Data")

# plot of data with time on x-axis
plot.ts(cars)

fit <- lm(cars ~ as.numeric(1:length(cars)))
# plot trend
abline(fit,col="red")
# plot mean
abline(h=mean(cars), col="blue")
```



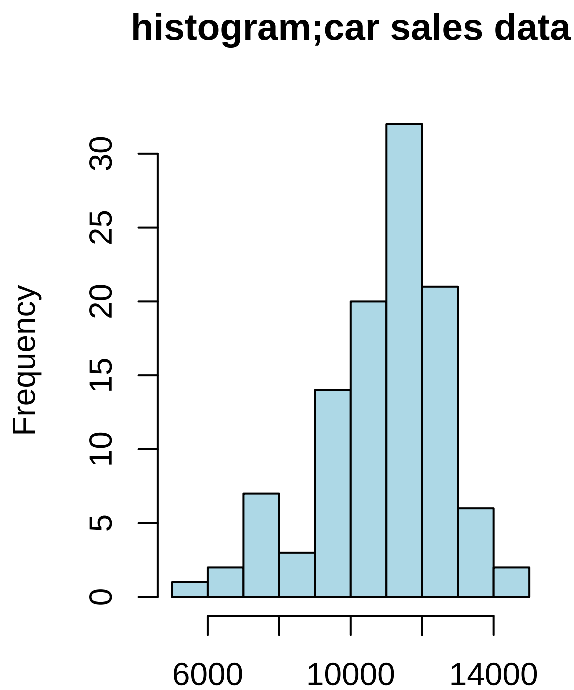
Two plots represent car sales having year and time on x-axis respectively. From January 2007 to December 2016, there are 120 observations.

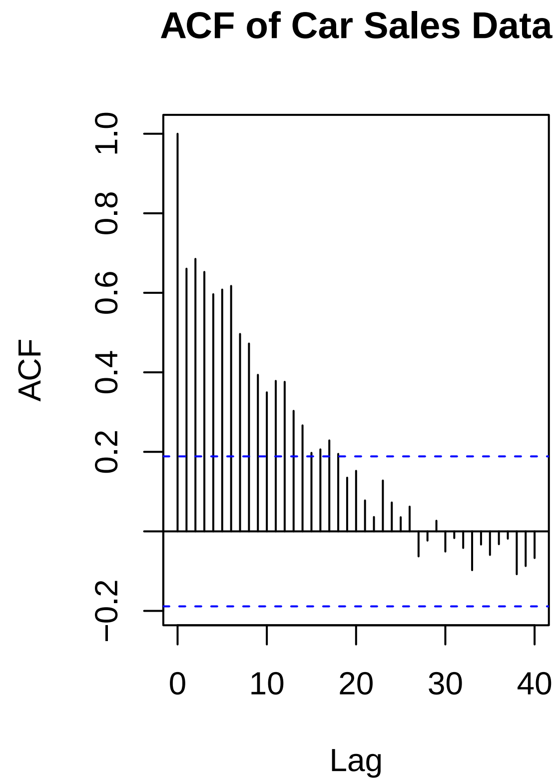
```
# split the model : train/test
# we are going to work with carstrain , {U_t, t=1,2,...,120}
# we check validity of the model with cars.test
carstrain = cars[c(1:108)]
cars.test = cars[(c(109:120))]

# plot train set of the model
plot.ts(carstrain)

fit <- lm(carstrain~ as.numeric(1:length(carstrain)))

# plot trend and mean respectively
abline(fit, col="red")
abline(h=mean(carstrain), col="blue")
```



Since I do not have any new data and to check validity of the model I create, I started with creating a test/train set. Train set corresponds to 108 observations of the first 9 years and test set corresponds to the 12 observations of the last year, 2016. I am going to use the trainset to build a model throughout the project.
There doesn't seem to be a seasonality and there is a downward trend in the beginning. However, after that, I was able to see upward trend in car sales.

```
par(mfrow=c(1,2))

# histogram of carstrain
hist(carstrain, col="light blue", xlab="", main="histogram;car sales data")


acf(carstrain, lag.max=40, main="ACF of Car Sales Data")
```

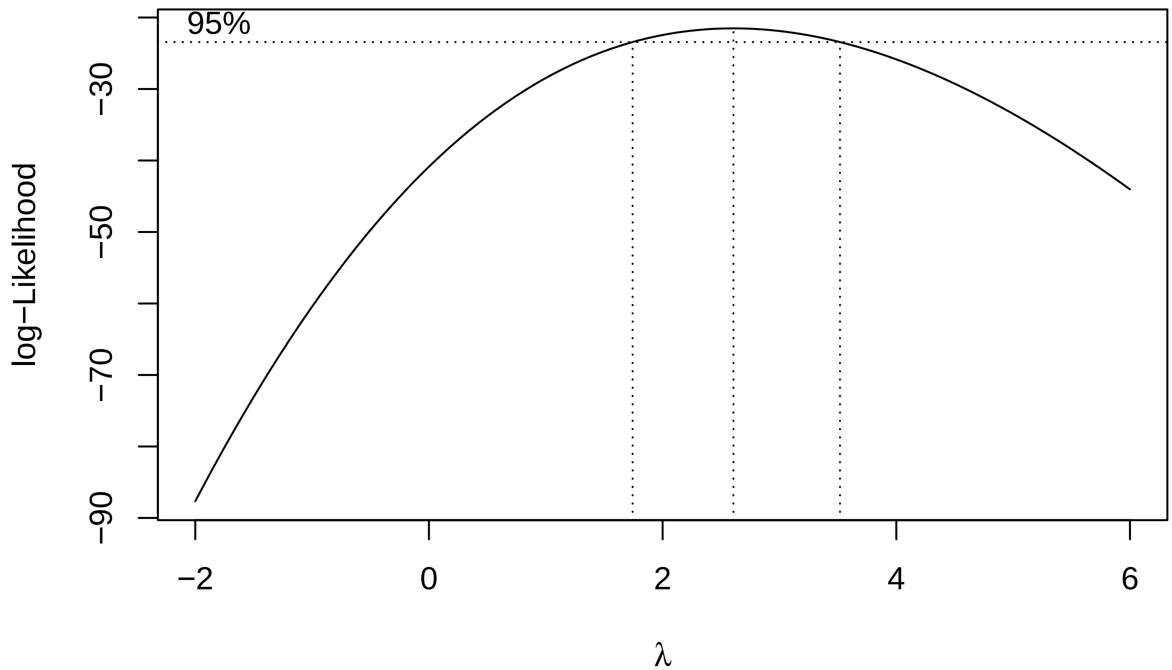




The histogram of "car sales" train set is highly right skewed and Acfs remain large in the beginning and there doesn't seem to be a seasonality.

Box-Cox Transformation

```r
# perform box-cox transformation to make the data normally distributed
bcTransform <- boxcox(carstrain ~ as.numeric(1:length(carstrain)), lambda= seq(-2,6, by = 0.5))
```




```
bcTransform$x[which(bcTransform$y == max(bcTransform$y))]
```

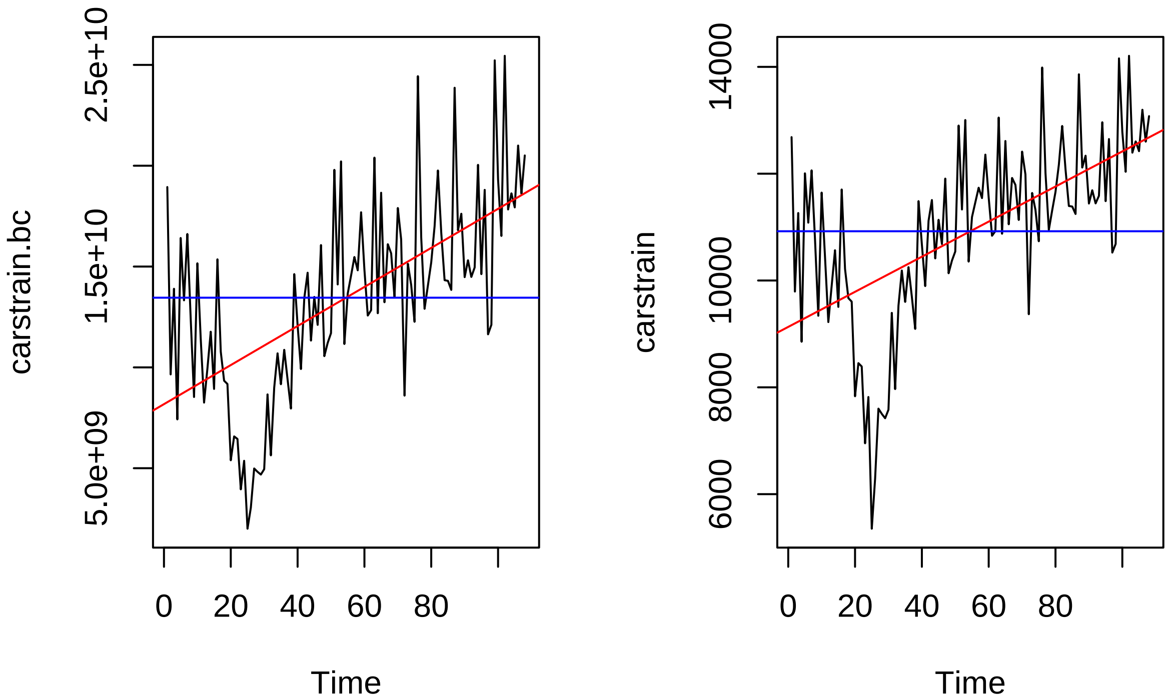
```
## [1] 2.606061
```

```
# lambda = 2.606061
lambda = bcTransform$x[which(bcTransform$y == max(bcTransform$y))]
```

Since the data is highly skewed, I tried Box-cox transformation to normalize the data. "BcTransform" command gives value of $\lambda = 2.6061$

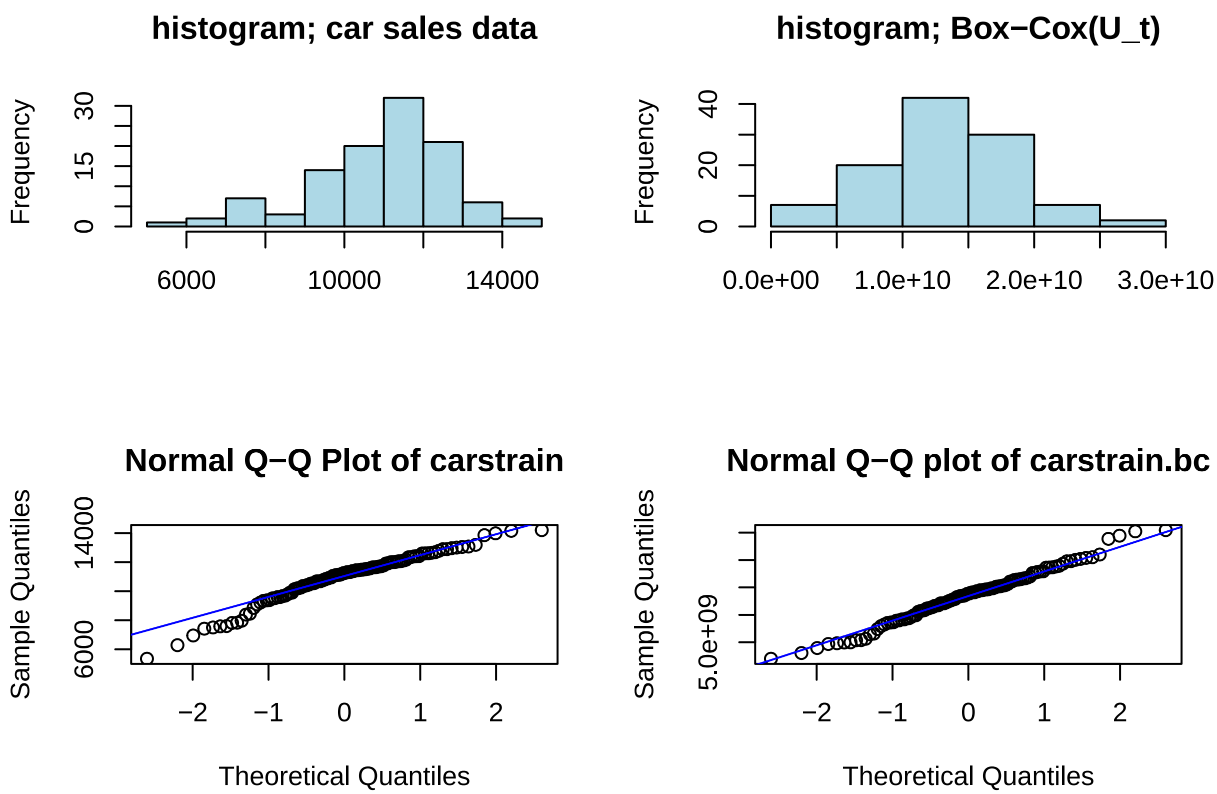
```
par(mfrow=c(1,2))

lambda = bcTransform$x[which(bcTransform$y == max(bcTransform$y))]

# Box-Cox transformation
carstrain.bc = (1/lambda) * (carstrain^lambda-1)

# plot of U_t after Box-Cox transformation
plot.ts(carstrain.bc)
fit <- lm(carstrain.bc~ as.numeric(1:length(carstrain.bc)))
abline(fit, col="red")
abline(h=mean(carstrain.bc), col="blue")

# plot of U_t before Box-Cox transformation
```

```r
plot.ts(carstrain)

fit <- lm(carstrain~ as.numeric(1:length(carstrain)))

abline(fit, col="red")
abline(h=mean(carstrain), col="blue")
```



Since the value of λ used for Box-Cox transformation was large, overall variance increased, however we could expect to have normalized data and we could see this by plotting a histogram.

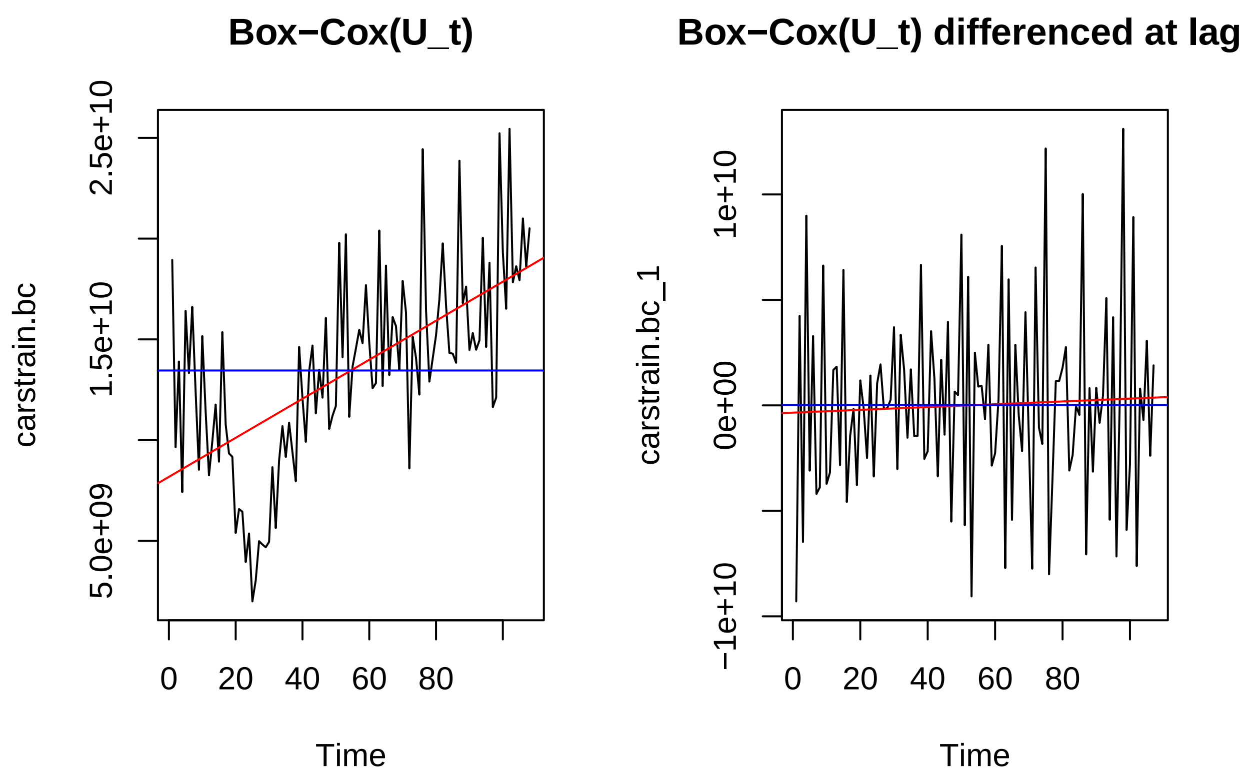
```r
par(mfrow=c(2,2))

hist(carstrain, col="light blue", xlab="", main="histogram; car sales data")
hist(carstrain.bc, col="light blue", xlab="", main="histogram; Box-Cox(U_t)")

qqnorm(carstrain, main = "Normal Q-Q Plot of carstrain")
qqline(carstrain, col = "blue")

qqnorm(carstrain.bc, main = "Normal Q-Q plot of carstrain.bc")
qqline(carstrain.bc, col = "blue")
```



Before Box-Cox transformation, the data was highly right skewed. After Box-Cox transformation, the data is more centered to the middle and seems more symmetric. We could also confirm this by comparing Q-Q plot before and after Box-Cox transformation.

Make the data stationary(remove trend/seasonality)

```
par(mfrow=c(1,2))

carstrain.bc_1 <- diff(carstrain.bc, lag=1)

plot.ts(carstrain.bc, main="Box-Cox(U_t)")
fit <- lm(carstrain.bc ~ as.numeric(1:length(carstrain.bc))); abline(fit, col="red")
abline(h=mean(carstrain.bc), col="blue")

plot.ts(carstrain.bc_1, main="Box-Cox(U_t) differenced at lag 1")
fit <- lm(carstrain.bc_1 ~ as.numeric(1:length(carstrain.bc_1))); abline(fit, col="red")
abline(h=mean(carstrain.bc_1), col="blue")
```




```
var(carstrain.bc)
```

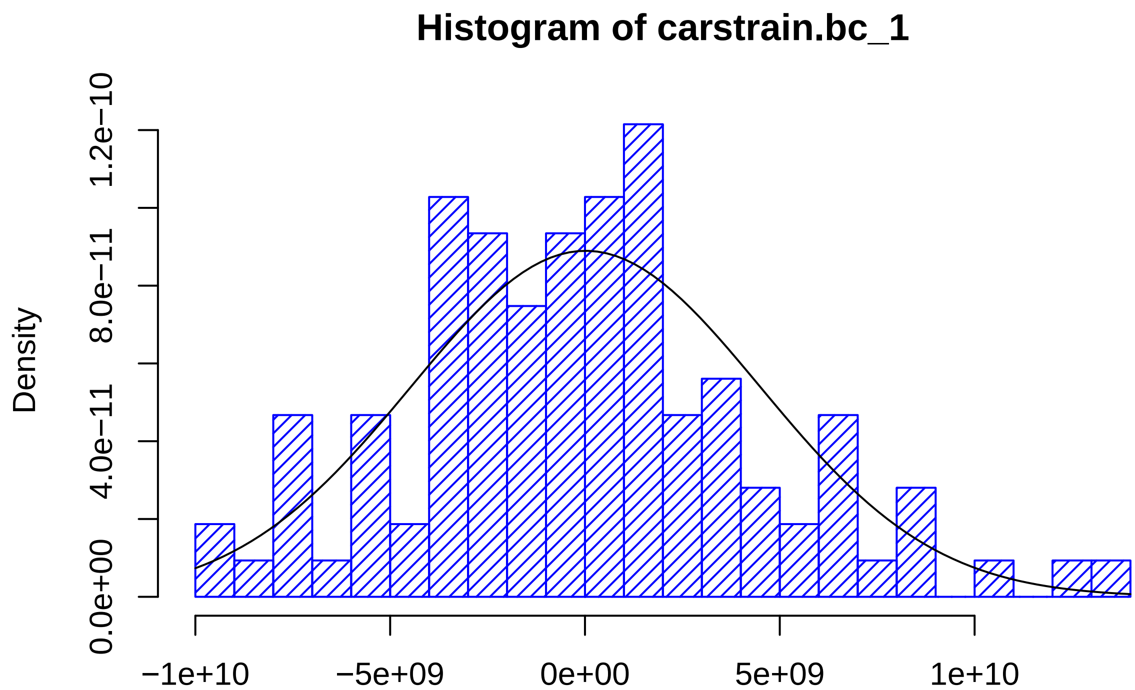
```
## [1] 2.368118e+19
```

```
var(carstrain.bc_1)
```

```
## [1] 2.011165e+19
```

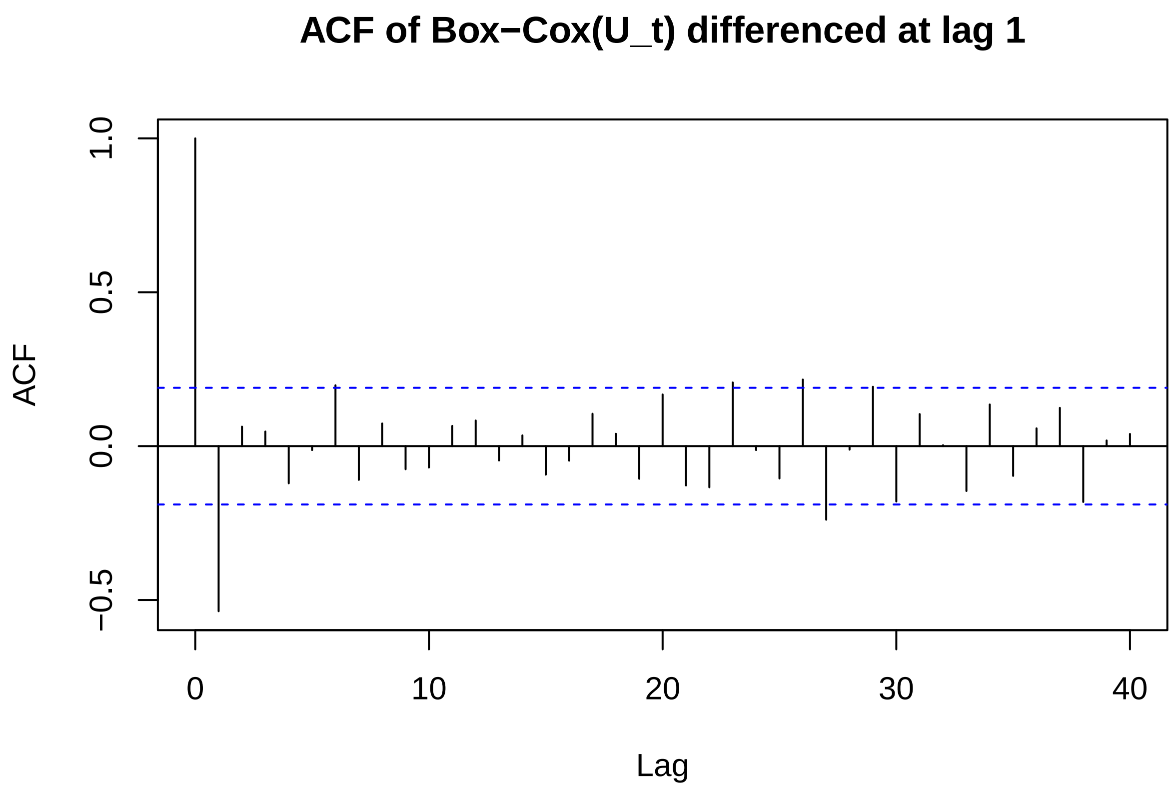
Differencing the transformed model at lag 1 removed the trend and the data looks stationary. Also, the variance is lower after removing the trend. However, differencing one more time at lag 1 gives us higher variance that leads to overdifferencing so I didn't proceed to further differencing.

```
hist(carstrain.bc_1, density=20,breaks=20, col="blue", xlab="", prob=TRUE)
m1 <- mean(carstrain.bc_1)
std1 <- sqrt(var(carstrain.bc_1))
curve(dnorm(x,m1,std1), add=TRUE )
```

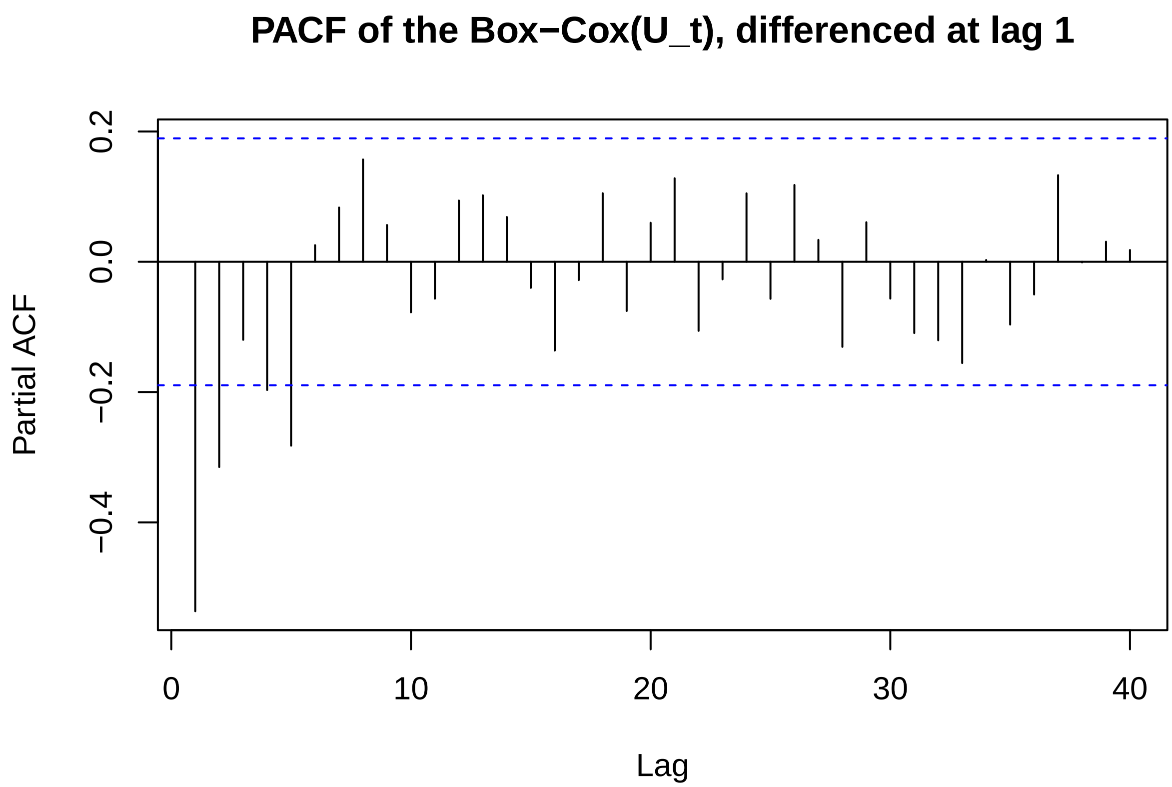


histogram of ∇_1Box_Cox(U_t) looks symmetric and normally distributed.

```
acf(carstrain.bc_1, lag.max=40, main="ACF of Box-Cox(U_t) differenced at lag 1")
```




```
pacf(carstrain.bc_1, lag.max=40, main="PACF of the Box-Cox(U_t), differenced at lag 1")
```



Now, analysis of ACF/PACF could give us what p and q to choose for ARIMA model. There's a spike outside of the confidence interval at lag 1 from the ACF and PACF suggests $p = 5$. Therefore, list of candidate models would be ARIMA model, p ranging from 0 to 5 and q ranging from 0 to 1.

Possible models

```
AICc(arima(carstrain.bc, order=c(1,1,0), method= "ML"))
```

```
## [1] 5024.757
```

```
AICc(arima(carstrain.bc, order=c(2,1,0), method= "ML"))
```

```
## [1] 5014.768
```

```
AICc(arima(carstrain.bc, order=c(3,1,0), method= "ML"))
```

```
## [1] 5014.144
```

```
AICc(arima(carstrain.bc, order=c(4,1,0), method= "ML"))
```

```
## [1] 5011.455
```

```r
AICc(arima(carstrain.bc, order=c(5,1,0), method= "ML"))
```

```
## [1] 5002.888
```

```r
AICc(arima(carstrain.bc, order=c(0,1,1), method= "ML"))
```

```
## [1] 5005.797
```

```r
AICc(arima(carstrain.bc, order=c(1,1,1), method= "ML"))
```

```
## [1] 5005.662
```

```r
AICc(arima(carstrain.bc, order=c(2,1,1), method= "ML"))
```

```
## [1] 5007.64
```

```r
AICc(arima(carstrain.bc, order=c(3,1,1), method= "ML"))
```

```
## [1] 5009.673
```

```r
AICc(arima(carstrain.bc, order=c(4,1,1), method= "ML"))
```

```
## [1] 5008.881
```

```r
AICc(arima(carstrain.bc, order=c(5,1,1), method= "ML"))
```

```
## [1] 5005.103
```

By comparing AICcs of possible models, we can narrow down the possible models. ARIMA(5,1,0), ARIMA(5,1,1), and ARIMA(1,1,1) had the lowest AICcs so I'm going to compare these three possible models. Also, I'm going to denote these model A, B, and C respectively.

```r
arima(carstrain.bc, order=c(5,1,0), method= "ML") # model A
```

```
##
## Call:
## arima(x = carstrain.bc, order = c(5, 1, 0), method = "ML")
##
## Coefficients:
##           ar1      ar2      ar3      ar4      ar5
##       -0.8896  -0.6438  -0.5041  -0.4686  -0.3174
## s.e.   0.0934   0.1189   0.1258   0.1181   0.0937
##
## sigma^2 estimated as 1.039e+19:  log likelihood = -2495.15,  aic = 5002.3
```

```r
arima(carstrain.bc, order=c(5,1,1), method= "ML") # model B
```

```
##
## Call:
## arima(x = carstrain.bc, order = c(5, 1, 1), method = "ML")
##
## Coefficients:
##           ar1      ar2      ar3      ar4      ar5     ma1
##       -0.9314  -0.6781  -0.5273  -0.4823  -0.3266  0.0460
## s.e.   0.2639   0.2354   0.1865   0.1434   0.1062  0.2728
##
## sigma^2 estimated as 1.039e+19:  log likelihood = -2495.14,  aic = 5004.27
```

```r
arima(carstrain.bc, order=c(1,1,1), method= "ML") # model C
```

```
##
## Call:
## arima(x = carstrain.bc, order = c(1, 1, 1), method = "ML")
##
## Coefficients:
##           ar1      ma1
##       -0.1793  -0.6841
## s.e.   0.1177   0.0778
##
## sigma^2 estimated as 1.138e+19:  log likelihood = -2499.77,  aic = 5005.55
```

ARIMA(5,1,0), model A in algebraic form would be

$\nabla_1 Box{-}Cox(U_t) = (1{+}0.8896B{+}0.6438B^2{+}0.5041B^3{+}0.4686B^4{+}0.3174B^5)(1{-}B)X_t = Z_t, \hat{\sigma_z}^2 = 1.039e{+}19$

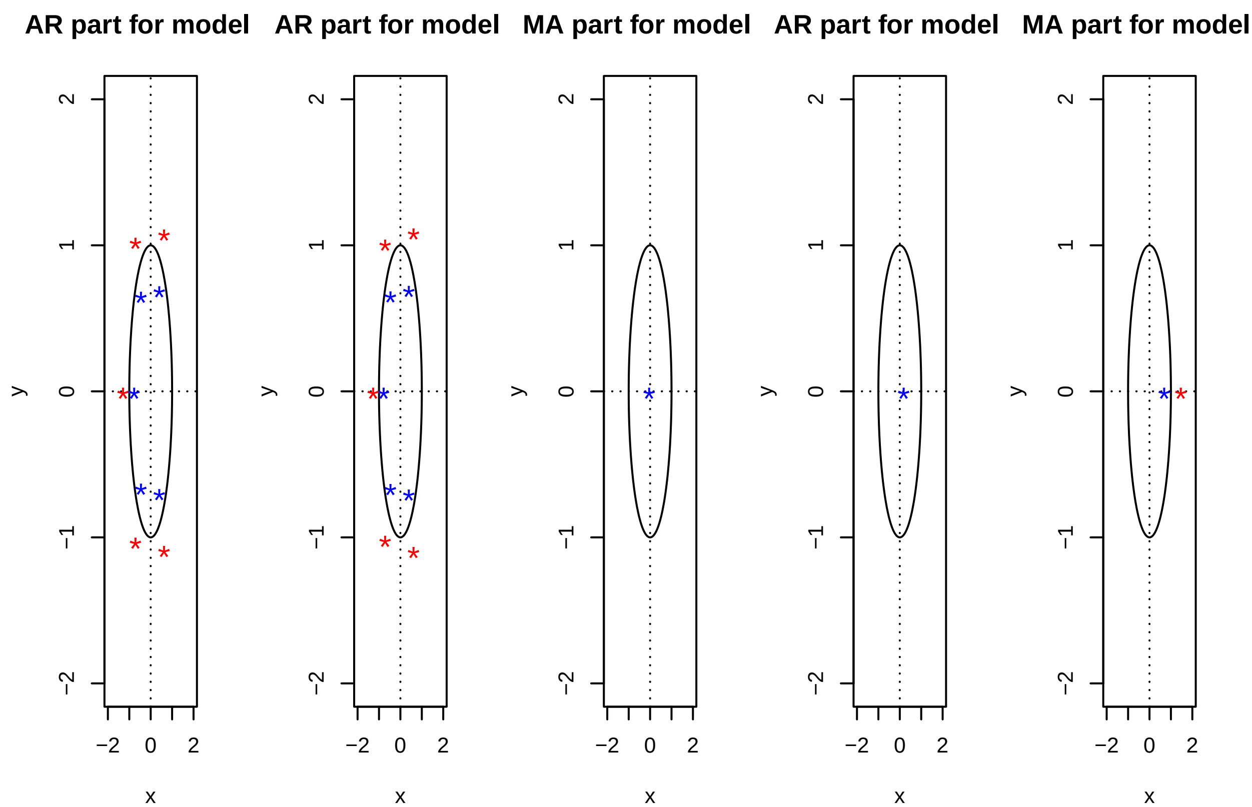
ARIMA(5,1,1), model B in algebraic form would be

$\nabla_1 Box{-}Cox(U_t) = (1{+}0.9314B{+}0.6781B^2{+}0.5273B^3{+}0.4823B^4{+}0.3266B^5)(1{-}B)X_t = (1{-}0.0460B)Z_t, \hat{\sigma_z}^2 = 1.039e{+}19$

ARIMA(1,1,1), model C in algebraic form would be

$$\nabla_1 Box - Cox(U_t) = (1 + 0.1793B)(1 - B)X_t = (1 - 0.6841)Z_t, \hat{\sigma_z}^2 = 1.138e + 19$$

```r
par(mfrow=c(1,5))

source("plot.roots.R.txt")

# AR part for model A
plot.roots(NULL,polyroot(c(1, 0.8896,0.6438,0.5041,0.4686,0.3174)), main="AR part for model A")

# AR/MA part respectively for model B
plot.roots(NULL,polyroot(c(1, 0.9314,0.6781,0.5273,0.4823,0.3266)), main="AR part for model B")
plot.roots(NULL,polyroot(c(1, 0.0460)), main="MA part for model B")

# AR/MA part respectively for model C
plot.roots(NULL,polyroot(c(1, -0.1793)), main="AR part for model C")
plot.roots(NULL,polyroot(c(1, -0.6841)), main="MA part for model C")
```



All three models are stationary, causal, and invertible since roots of both AR/MA parts for all models lie outside unit circles.

Model fitting and Diagnostic Checking

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

# Model A
fit1 <- arima(carstrain.bc, order=c(5,1,0), method= "ML")
res1 <- residuals(fit1)
hist(res1,density=20,breaks=20, col="blue", xlab="", prob=TRUE, main = "Histogram of res_A")

m1 <- mean(res1)
std1 <- sqrt(var(res1))
curve( dnorm(x,m1,std1), add=TRUE )

plot.ts(res1)
fitt <- lm(res1 ~ as.numeric(1:length(res1))); abline(fitt, col="red")
abline(h=mean(res1), col="blue")
qqnorm(res1,main= "Normal Q-Q Plot of res_A")
qqline(res1,col="blue")

acf(res1, lag.max=40)
pacf(res1, lag.max=40)
```

```r
shapiro.test(res1)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  res1
## W = 0.97557, p-value = 0.04397
```

```r
Box.test(res1, lag=10, type = c("Box-Pierce"), fitdf=5)
```

```
##
##  Box-Pierce test
##
## data:  res1
## X-squared = 3.4301, df = 5, p-value = 0.634
```

```r
Box.test(res1, lag=10, type = c("Ljung-Box"), fitdf=5)
```

```
##
##  Box-Ljung test
##
## data:  res1
## X-squared = 3.7057, df = 5, p-value = 0.5925
```

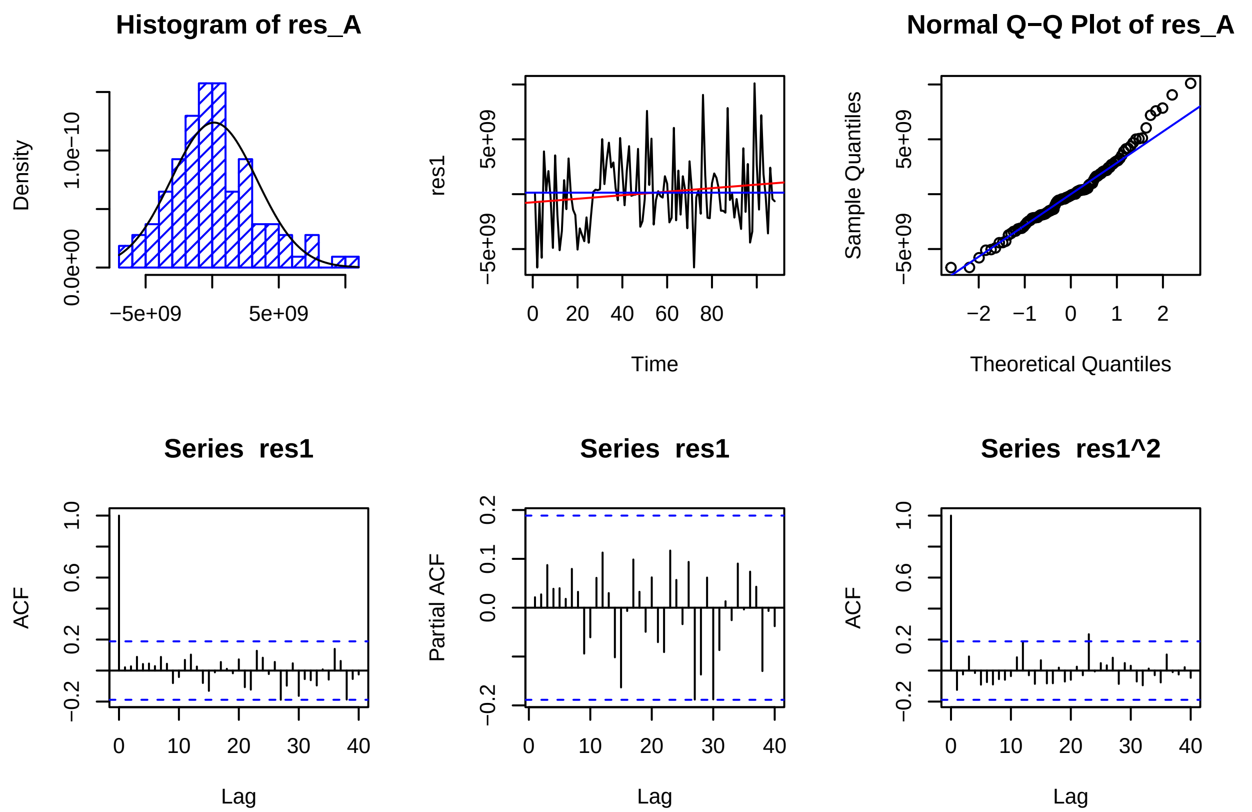
```r
Box.test(res1^2, lag=10, type = c("Ljung-Box"), fitdf=0)
```

```
##
##  Box-Ljung test
##
## data:  res1^2
## X-squared = 6.2428, df = 10, p-value = 0.7945
```

```r
ar(res1, aic = TRUE, order.max = NULL, method = c("yule-walker"))
```

```
##
## Call:
## ar(x = res1, aic = TRUE, order.max = NULL, method = c("yule-walker"))
##
##
## Order selected 0  sigma^2 estimated as  1.037e+19
```

```r
acf(res1^2, lag.max=40)
```



For residuals of model A, there's a slight trend but it is negligible. Both histogram and Q-Q plot shows that res_A is normally distributed. Also, all acf and pacf of residuals are within confidence intervals and can be counted as zeros. In addition, ACF of $(residuals)^2$ shows nonlinear dependence. Lastly, Model A passes all the diagnostic testings but Shapiro-Wilk normality test, having p-value(0.04397) less than 0.05.

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

# Model B
fit2 <- arima(carstrain.bc, order=c(5,1,1), method= "ML")
res2 <- residuals(fit2)
hist(res2,density=20,breaks=20, col="blue", xlab="", prob=TRUE, main="Histogram of res_B")

m2 <- mean(res2)
std2 <- sqrt(var(res2))
curve( dnorm(x,m2,std2), add=TRUE )

plot.ts(res2)
fitt <- lm(res2 ~ as.numeric(1:length(res2))); abline(fitt, col="red")
abline(h=mean(res2), col="blue")
qqnorm(res2,main= "Normal Q-Q Plot of res_B")
qqline(res2,col="blue")

acf(res2, lag.max=40)
pacf(res2, lag.max=40)

shapiro.test(res2)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  res2
## W = 0.97585, p-value = 0.04635
```

```r
Box.test(res2, lag=10, type = c("Box-Pierce"), fitdf=6)
```

```
##
##  Box-Pierce test
##
## data:  res2
## X-squared = 3.3765, df = 4, p-value = 0.4969
```

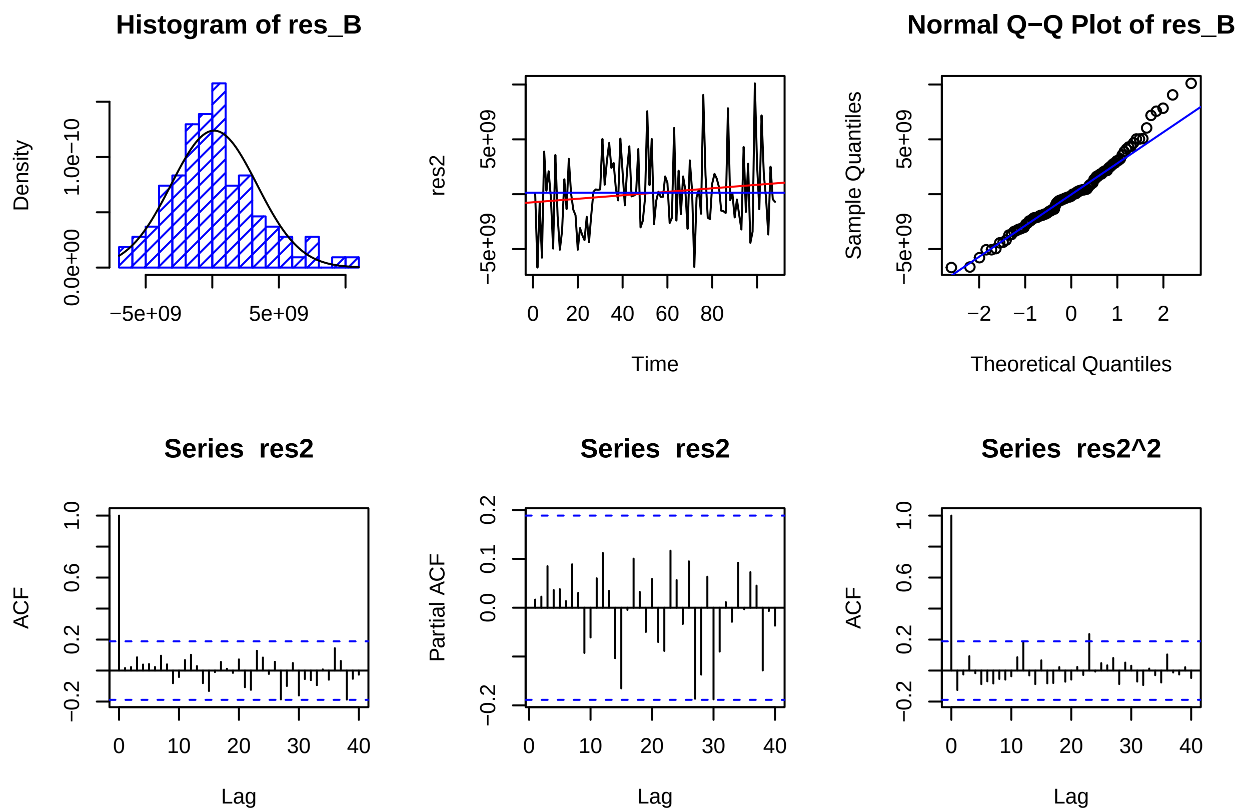
```r
Box.test(res2, lag=10, type = c("Ljung-Box"), fitdf=6)
```

```
##
##  Box-Ljung test
##
## data:  res2
## X-squared = 3.6536, df = 4, p-value = 0.4549
```

```r
Box.test(res2^2, lag=10, type = c("Ljung-Box"), fitdf=0)
```

```
##
##  Box-Ljung test
##
## data:  res2^2
## X-squared = 6.0149, df = 10, p-value = 0.814
```

```r
acf(res2^2, lag.max=40)
```




```r
ar(res2, aic = TRUE, order.max = NULL, method = c("yule-walker"))
```

```
##
## Call:
## ar(x = res2, aic = TRUE, order.max = NULL, method = c("yule-walker"))
##
##
## Order selected 0  sigma^2 estimated as  1.037e+19
```

Model B also looks good, there's a slight trend but it's negligible. Also, histogram and Q-Q plot shows that the residual of model B is normally distributed. All acf and pacf of residuals are within confidence intervals and can be counted as zeros as well. Just like model A, model B passes all the tests but Shapiro-Wilk normality test, having p-value(0.04635) less than 0.05.

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

# Model C
fit3 <- arima(carstrain.bc, order=c(1,1,1), method= "ML")
res3 <- residuals(fit3)
hist(res3,density=20,breaks=20, col="blue", xlab="", prob=TRUE, main="Histogram of res_C")

m3 <- mean(res3)
std3 <- sqrt(var(res3))
curve( dnorm(x,m3,std3), add=TRUE )
```

```r
plot.ts(res3)
fitt <- lm(res3 ~ as.numeric(1:length(res3))); abline(fitt, col="red")
abline(h=mean(res3), col="blue")
qqnorm(res3,main= "Normal Q-Q Plot of res_C")
qqline(res3,col="blue")

acf(res3, lag.max=40)
pacf(res3, lag.max=40)

shapiro.test(res3)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  res3
## W = 0.96835, p-value = 0.01121
```

```r
Box.test(res3, lag=10, type = c("Box-Pierce"), fitdf=2)
```

```
##
##  Box-Pierce test
##
## data:  res3
## X-squared = 13.828, df = 8, p-value = 0.08636
```

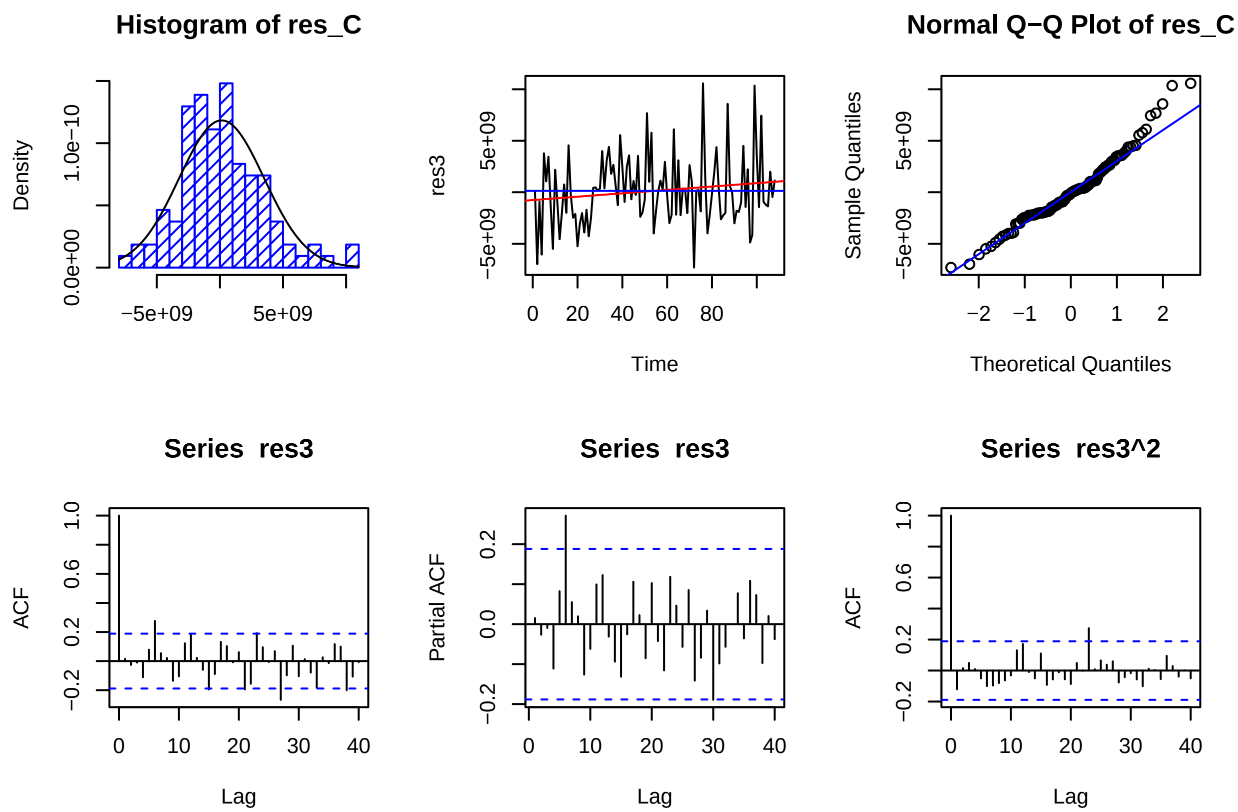
```r
Box.test(res3, lag=10, type = c("Ljung-Box"), fitdf=2)
```

```
##
##  Box-Ljung test
##
## data:  res3
## X-squared = 14.995, df = 8, p-value = 0.05925
```

```r
Box.test(res3^2, lag=10, type = c("Ljung-Box"), fitdf=0)
```

```
##
##  Box-Ljung test
##
## data:  res3^2
## X-squared = 5.8491, df = 10, p-value = 0.8278
```

```r
acf(res3^2, lag.max=40)
```




```r
ar(res3, aic = TRUE, order.max = NULL, method = c("yule-walker"))
```

```
##
## Call:
## ar(x = res3, aic = TRUE, order.max = NULL, method = c("yule-walker"))
##
##
## Order selected 0  sigma^2 estimated as  1.136e+19
```

Model C also has same results as model A and B, normally distributed residuals and ACF/PACFs are fine. Also, model C passed all the diagnostics testings but Shapiro-Wilk normality test. However, model C had the lowest p-value(0.01121) which is far away from 0.05.

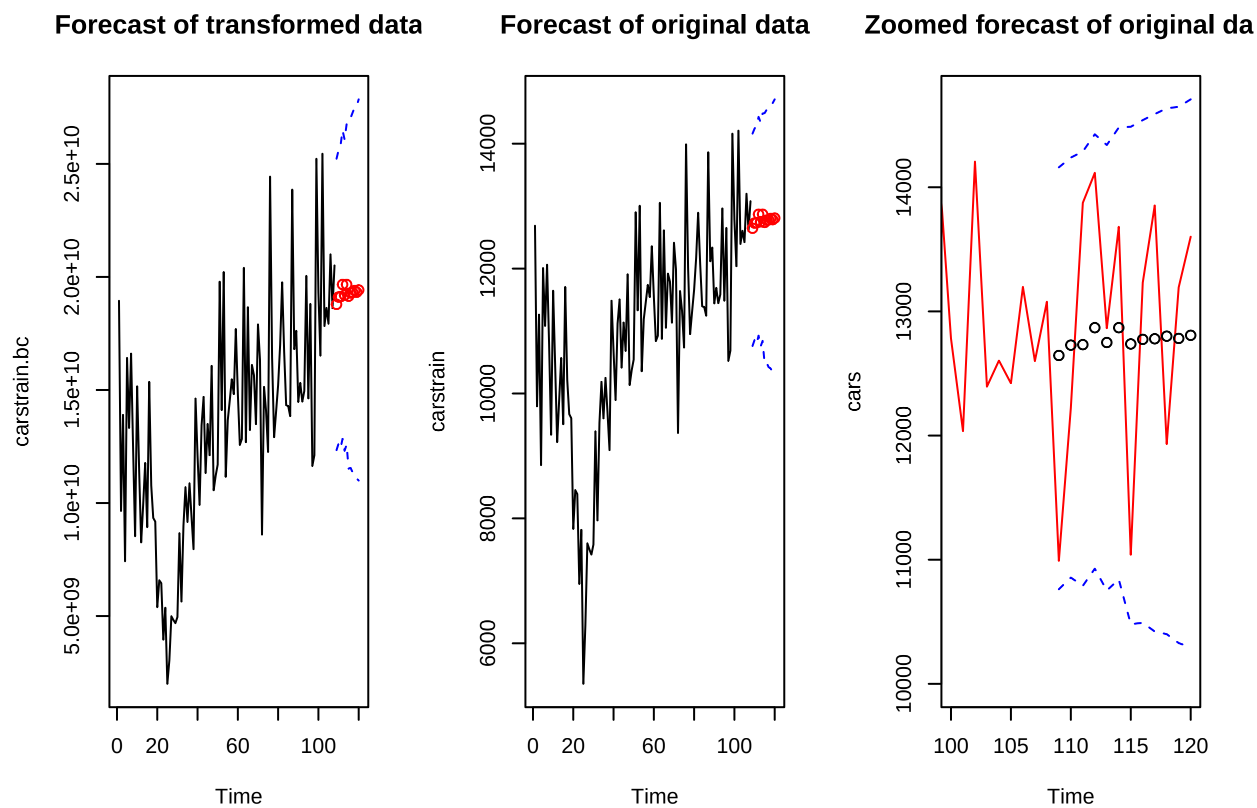
I decided to choose model B, ARIMA(5,1,1) considering that it all passed the diagnostic testings and had the highest p-value of 0.04635 that is as close to 0.05.

Forecasting using model B

```r
par(mfrow=c(1,3))

fit.B <- arima(carstrain.bc, order=c(5,1,1), method= "ML")
forecast(fit.B)
```

```
##     Point Forecast       Lo 80       Hi 80       Lo 95       Hi 95
## 109    18787611625 14657009616 22918213634 12470400996 25104822254
## 110    19112460879 14954801596 23270120161 12753869720 25471052037
## 111    19126882055 14831525799 23422238311 12557701549 25696062561
## 112    19666282073 15293960183 24038603964 12979392786 26353171360
## 113    19194143860 14782623781 23605663939 12447306117 25940981603
## 114    19666478335 15126069005 24206887665 12722521494 26610435177
## 115    19149224477 14258817322 24039631633 11669992153 26628456802
## 116    19294832596 14332024382 24257640810 11704872407 26884792784
## 117    19312417656 14209849973 24414985339 11508713804 27116121507
## 118    19396460679 14201439217 24591482141 11451360975 27341560383
```

```
# produce graph with 12 forecasts on transformed data
pred.tr <- predict(fit.B, n.ahead=12)
U.tr= pred.tr$pred + 2*pred.tr$se # upper bound of prediction interval
L.tr= pred.tr$pred - 2*pred.tr$se # lower bound of prediction interval

ts.plot(carstrain.bc, xlim=c(1,length(carstrain.bc)+12), ylim = c(min(carstrain.bc),max(U.tr)), main="Fo
lines(U.tr, col="blue", lty="dashed")
lines(L.tr, col="blue", lty="dashed")
points((length(carstrain.bc)+1):(length(carstrain.bc)+12), pred.tr$pred, col="red")

# produce graph with 12 forecasts on original data
pred.orig <- inv_boxcox(pred.tr$pred, lambda)
U= inv_boxcox(U.tr, lambda)
L= inv_boxcox(L.tr, lambda)

ts.plot(carstrain, xlim=c(1,length(carstrain)+12), ylim = c(min(carstrain),max(U)), main="Forecast of o
lines(U, col="blue", lty="dashed")
lines(L, col="blue", lty="dashed")
points((length(carstrain)+1):(length(carstrain)+12), pred.orig, col="red")

# plot zoomed forecasts and true values(in car)
ts.plot(cars, xlim = c(100,length(carstrain)+12), ylim = c(10000,max(U)), col="red", main="Zoomed foreca
lines(U, col="blue", lty="dashed")
lines(L, col="blue", lty="dashed")
points((length(carstrain)+1):(length(carstrain)+12), pred.orig, col="black") #forecasts
```



Conclusion

Every model passed all the diagnostic checkings but failed Shapiro-Wilk normality test. Model A,B, and C had a p-value of 0.04397, 0.04635, and 0.1121 respectively. I chose model B for final model since it has the the largest value of p-value among the three.

Final model for the Box-Cox transform of original data: $Box-Cox(U_t)$ follows ARIMA(5,1,1) model. And the model in algebraic form would be

$$\nabla_1 Box-Cox(U_t) = (1+0.9314B+0.6781B^2+0.5273B^3+0.4823B^4+0.3266B^5)(1-B)X_t = (1-0.0460B)Z_t, \hat{\sigma_z}^2 = 1.039e+19$$

Finally, both forecasts of transformed data and original data were within the confidence interval. However, the prediction was almost linear and was not best at giving meaningful insight. Going back to the beginning, differencing the model at different lags and applying different λ for Box-Cox transformation didn't improve the model performance. Considering the small amount of data, having more data would have possibly given better prediction.

Reference

Introduction to Time Series and Forecasting, by P. Brockwell and R. Davis, Springer
Time Series Analysis with R Examples, by R. H. Shumway and D. S. Stoffer, Springer
https://www.kaggle.com/datasets/dmi3kno/newcarsalesnorway

Appendix

```
knitr::opts_chunk$set(echo = TRUE)
# Required Library
library(MASS)
library(forecast)
library(qpcR)
library(ggplot2)
library(ldsr) # perform inverse Box-Cox transform
# load data
cars <- scan("norway_new_car_sales_by_month.txt")
par(mfrow=c(1,2))

# plot of data with years on x-axis
tsdat <- ts(cars, start = c(2007,1), end = c(2016,12), frequency = 12)

ts.plot(tsdat, main = "Raw Data")

# plot of data with time on x-axis
plot.ts(cars)

fit <- lm(cars ~ as.numeric(1:length(cars)))
# plot trend
abline(fit,col="red")
# plot mean
abline(h=mean(cars), col="blue")
# split the model : train/test
# we are going to work with carstrain , {U_t, t=1,2,...,120}
# we check validity of the model with cars.test
carstrain = cars[c(1:108)]
cars.test = cars[(c(109:120))]

# plot train set of the model
plot.ts(carstrain)

fit <- lm(carstrain~ as.numeric(1:length(carstrain)))

# plot trend and mean respectively
abline(fit, col="red")
abline(h=mean(carstrain), col="blue")
par(mfrow=c(1,2))

# histogram of carstrain
hist(carstrain, col="light blue", xlab="", main="histogram;car sales data")


acf(carstrain, lag.max=40, main="ACF of Car Sales Data")
# perform box-cox transformation to make the data normally distributed
bcTransform <- boxcox(carstrain ~ as.numeric(1:length(carstrain)), lambda= seq(-2,6, by = 0.5))

bcTransform$x[which(bcTransform$y == max(bcTransform$y))]

# lambda = 2.606061
```

```
lambda = bcTransform$x[which(bcTransform$y == max(bcTransform$y))]
par(mfrow=c(1,2))

lambda = bcTransform$x[which(bcTransform$y == max(bcTransform$y))]

# Box-Cox transformation
carstrain.bc = (1/lambda) * (carstrain^lambda-1)

# plot of U_t after Box-Cox transformation
plot.ts(carstrain.bc)
fit <- lm(carstrain.bc~ as.numeric(1:length(carstrain.bc)))
abline(fit, col="red")
abline(h=mean(carstrain.bc), col="blue")

# plot of U_t before Box-Cox transformation
plot.ts(carstrain)

fit <- lm(carstrain~ as.numeric(1:length(carstrain)))

abline(fit, col="red")
abline(h=mean(carstrain), col="blue")
par(mfrow=c(2,2))

hist(carstrain, col="light blue", xlab="", main="histogram; car sales data")
hist(carstrain.bc, col="light blue", xlab="", main="histogram; Box-Cox(U_t)")

qqnorm(carstrain, main = "Normal Q-Q Plot of carstrain")
qqline(carstrain, col = "blue")

qqnorm(carstrain.bc, main = "Normal Q-Q plot of carstrain.bc")
qqline(carstrain.bc, col = "blue")
par(mfrow=c(1,2))

carstrain.bc_1 <- diff(carstrain.bc, lag=1)

plot.ts(carstrain.bc, main="Box-Cox(U_t)")
fit <- lm(carstrain.bc ~ as.numeric(1:length(carstrain.bc))); abline(fit, col="red")
abline(h=mean(carstrain.bc), col="blue")

plot.ts(carstrain.bc_1, main="Box-Cox(U_t) differenced at lag 1")
fit <- lm(carstrain.bc_1 ~ as.numeric(1:length(carstrain.bc_1))); abline(fit, col="red")
abline(h=mean(carstrain.bc_1), col="blue")

var(carstrain.bc)
var(carstrain.bc_1)
hist(carstrain.bc_1, density=20,breaks=20, col="blue", xlab="", prob=TRUE)
m1 <- mean(carstrain.bc_1)
std1 <- sqrt(var(carstrain.bc_1))
curve(dnorm(x,m1,std1), add=TRUE )
acf(carstrain.bc_1, lag.max=40, main="ACF of Box-Cox(U_t) differenced at lag 1")
pacf(carstrain.bc_1, lag.max=40, main="PACF of the Box-Cox(U_t), differenced at lag 1")
AICc(arima(carstrain.bc, order=c(1,1,0), method= "ML"))
AICc(arima(carstrain.bc, order=c(2,1,0), method= "ML"))
```

```r
AICc(arima(carstrain.bc, order=c(3,1,0), method= "ML"))
AICc(arima(carstrain.bc, order=c(4,1,0), method= "ML"))
AICc(arima(carstrain.bc, order=c(5,1,0), method= "ML"))
AICc(arima(carstrain.bc, order=c(0,1,1), method= "ML"))
AICc(arima(carstrain.bc, order=c(1,1,1), method= "ML"))
AICc(arima(carstrain.bc, order=c(2,1,1), method= "ML"))
AICc(arima(carstrain.bc, order=c(3,1,1), method= "ML"))
AICc(arima(carstrain.bc, order=c(4,1,1), method= "ML"))
AICc(arima(carstrain.bc, order=c(5,1,1), method= "ML"))
arima(carstrain.bc, order=c(5,1,0), method= "ML") # model A
arima(carstrain.bc, order=c(5,1,1), method= "ML") # model B
arima(carstrain.bc, order=c(1,1,1), method= "ML") # model C
par(mfrow=c(1,5))

source("plot.roots.R.txt")

# AR part for model A
plot.roots(NULL,polyroot(c(1, 0.8896,0.6438,0.5041,0.4686,0.3174)), main="AR part for model A")

# AR/MA part respectively for model B
plot.roots(NULL,polyroot(c(1, 0.9314,0.6781,0.5273,0.4823,0.3266)), main="AR part for model B")
plot.roots(NULL,polyroot(c(1, 0.0460)), main="MA part for model B")

# AR/MA part respectively for model C
plot.roots(NULL,polyroot(c(1, -0.1793)), main="AR part for model C")
plot.roots(NULL,polyroot(c(1, -0.6841)), main="MA part for model C")
par(mfrow=c(2,3))

# Model A
fit1 <- arima(carstrain.bc, order=c(5,1,0), method= "ML")
res1 <- residuals(fit1)
hist(res1,density=20,breaks=20, col="blue", xlab="", prob=TRUE, main = "Histogram of res_A")

m1 <- mean(res1)
std1 <- sqrt(var(res1))
curve( dnorm(x,m1,std1), add=TRUE )

plot.ts(res1)
fitt <- lm(res1 ~ as.numeric(1:length(res1))); abline(fitt, col="red")
abline(h=mean(res1), col="blue")
qqnorm(res1,main= "Normal Q-Q Plot of res_A")
qqline(res1,col="blue")

acf(res1, lag.max=40)
pacf(res1, lag.max=40)

shapiro.test(res1)
Box.test(res1, lag=10, type = c("Box-Pierce"), fitdf=5)
Box.test(res1, lag=10, type = c("Ljung-Box"), fitdf=5)
Box.test(res1^2, lag=10, type = c("Ljung-Box"), fitdf=0)
ar(res1, aic = TRUE, order.max = NULL, method = c("yule-walker"))

acf(res1^2, lag.max=40)
```

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

# Model B
fit2 <- arima(carstrain.bc, order=c(5,1,1), method= "ML")
res2 <- residuals(fit2)
hist(res2,density=20,breaks=20, col="blue", xlab="", prob=TRUE, main="Histogram of res_B")

m2 <- mean(res2)
std2 <- sqrt(var(res2))
curve( dnorm(x,m2,std2), add=TRUE )

plot.ts(res2)
fitt <- lm(res2 ~ as.numeric(1:length(res2))); abline(fitt, col="red")
abline(h=mean(res2), col="blue")
qqnorm(res2,main= "Normal Q-Q Plot of res_B")
qqline(res2,col="blue")

acf(res2, lag.max=40)
pacf(res2, lag.max=40)

shapiro.test(res2)
Box.test(res2, lag=10, type = c("Box-Pierce"), fitdf=6)
Box.test(res2, lag=10, type = c("Ljung-Box"), fitdf=6)
Box.test(res2^2, lag=10, type = c("Ljung-Box"), fitdf=0)

acf(res2^2, lag.max=40)
ar(res2, aic = TRUE, order.max = NULL, method = c("yule-walker"))
par(mfrow=c(2,3))

# Model C
fit3 <- arima(carstrain.bc, order=c(1,1,1), method= "ML")
res3 <- residuals(fit3)
hist(res3,density=20,breaks=20, col="blue", xlab="", prob=TRUE, main="Histogram of res_C")

m3 <- mean(res3)
std3 <- sqrt(var(res3))
curve( dnorm(x,m3,std3), add=TRUE )

plot.ts(res3)
fitt <- lm(res3 ~ as.numeric(1:length(res3))); abline(fitt, col="red")
abline(h=mean(res3), col="blue")
qqnorm(res3,main= "Normal Q-Q Plot of res_C")
qqline(res3,col="blue")

acf(res3, lag.max=40)
pacf(res3, lag.max=40)

shapiro.test(res3)
Box.test(res3, lag=10, type = c("Box-Pierce"), fitdf=2)
Box.test(res3, lag=10, type = c("Ljung-Box"), fitdf=2)
Box.test(res3^2, lag=10, type = c("Ljung-Box"), fitdf=0)

acf(res3^2, lag.max=40)
```

```
ar(res3, aic = TRUE, order.max = NULL, method = c("yule-walker"))
par(mfrow=c(1,3))

fit.B <- arima(carstrain.bc, order=c(5,1,1), method= "ML")
forecast(fit.B)

# produce graph with 12 forecasts on transformed data
pred.tr <- predict(fit.B, n.ahead=12)
U.tr= pred.tr$pred + 2*pred.tr$se # upper bound of prediction interval
L.tr= pred.tr$pred - 2*pred.tr$se # lower bound of prediction interval

ts.plot(carstrain.bc, xlim=c(1,length(carstrain.bc)+12), ylim = c(min(carstrain.bc),max(U.tr)), main="Fo
lines(U.tr, col="blue", lty="dashed")
lines(L.tr, col="blue", lty="dashed")
points((length(carstrain.bc)+1):(length(carstrain.bc)+12), pred.tr$pred, col="red")

# produce graph with 12 forecasts on original data
pred.orig <- inv_boxcox(pred.tr$pred, lambda)
U= inv_boxcox(U.tr, lambda)
L= inv_boxcox(L.tr, lambda)

ts.plot(carstrain, xlim=c(1,length(carstrain)+12), ylim = c(min(carstrain),max(U)), main="Forecast of o
lines(U, col="blue", lty="dashed")
lines(L, col="blue", lty="dashed")
points((length(carstrain)+1):(length(carstrain)+12), pred.orig, col="red")

# plot zoomed forecasts and true values(in car)
ts.plot(cars, xlim = c(100,length(carstrain)+12), ylim = c(10000,max(U)), col="red", main="Zoomed foreca
lines(U, col="blue", lty="dashed")
lines(L, col="blue", lty="dashed")
points((length(carstrain)+1):(length(carstrain)+12), pred.orig, col="black") #forecasts
```