DSCC/CSC/TCS 462 : HW0

Aradhya Mathur

2022-09-05

```
library(readr)
library(ggplot2)
library(moments)
```

Question 1. Getting familiar with the dataset via exploratory data analysis install.packages("ggplot2")s. a. Read the data into RStudio and summarize the data with the `summary()` function.

```
data1 <- read_csv("car_sales.csv")
```

```
## Rows: 152 Columns: 11
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr (2): Manufacturer, Model
## dbl (9): price, Engine_size, Horsepower, Wheelbase, Width, Length, Curb_weig...
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
summary(data1) # Summarize
```

```
##  Manufacturer          Model               price        Engine_size
##  Length:152         Length:152         Min.   : 9235   Min.   :1.000
##  Class :character   Class :character   1st Qu.:17889   1st Qu.:2.300
##  Mode  :character   Mode  :character   Median :22747   Median :3.000
##                                        Mean   :27332   Mean   :3.049
##                                        3rd Qu.:31939   3rd Qu.:3.575
##                                        Max.   :85500   Max.   :8.000
##    Horsepower      Wheelbase         Width           Length
##  Min.   : 55.0   Min.   : 92.6   Min.   :62.60   Min.   :149.4
##  1st Qu.:147.5   1st Qu.:102.9   1st Qu.:68.38   1st Qu.:177.5
##  Median :175.0   Median :107.0   Median :70.40   Median :186.7
##  Mean   :184.8   Mean   :107.4   Mean   :71.09   Mean   :187.1
##  3rd Qu.:211.2   3rd Qu.:112.2   3rd Qu.:73.10   3rd Qu.:195.1
##  Max.   :450.0   Max.   :138.7   Max.   :79.90   Max.   :224.5
##   Curb_weight     Fuel_capacity   Fuel_efficiency
##  Min.   :1.895   Min.   :10.30   Min.   :15.00
##  1st Qu.:2.965   1st Qu.:15.78   1st Qu.:21.00
##  Median :3.336   Median :17.20   Median :24.00
##  Mean   :3.376   Mean   :17.96   Mean   :23.84
##  3rd Qu.:3.821   3rd Qu.:19.80   3rd Qu.:26.00
##  Max.   :5.572   Max.   :32.00   Max.   :45.00
```

```
data2 <- data1[c('price')]
Price <- as.numeric(data2$price)
```

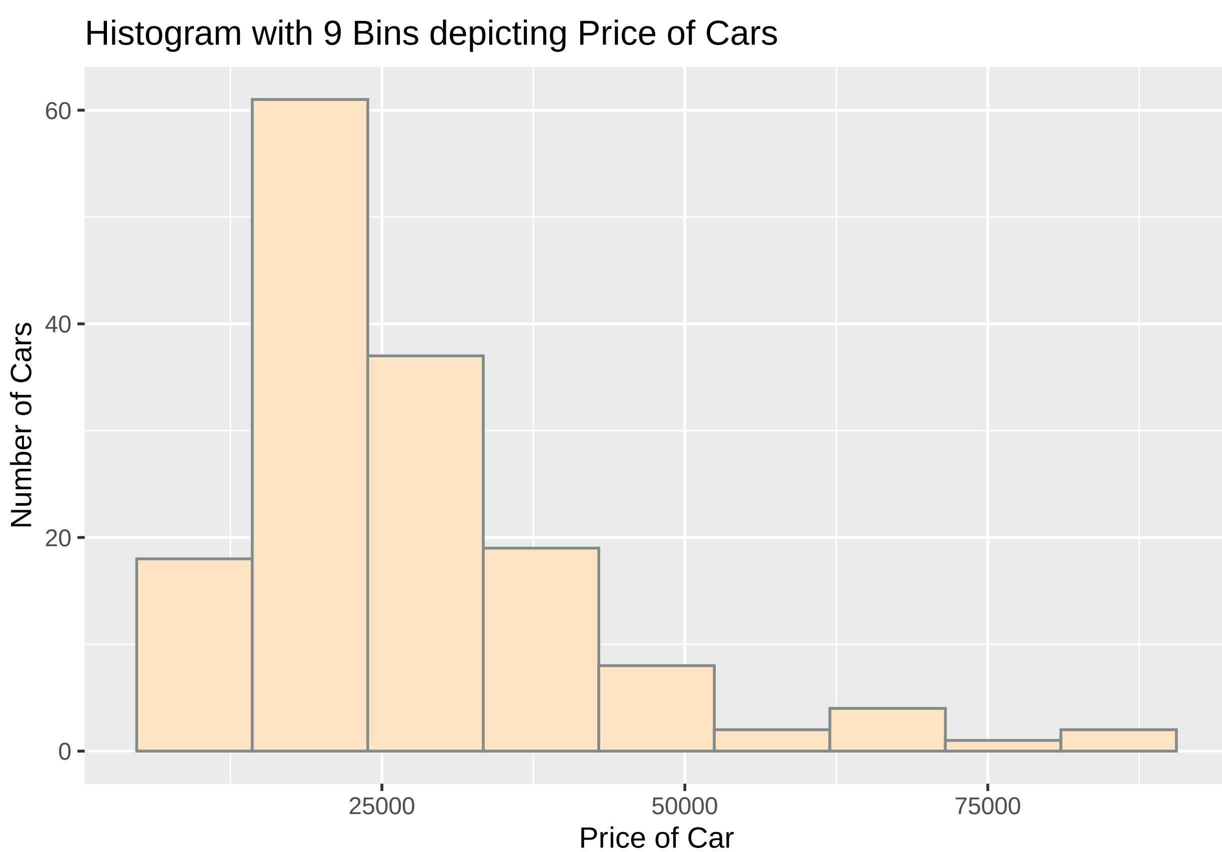
b. How many bins does Sturges' formula suggest we use for a histogram of `price`? Show your work

```
n <- 152 # 152 observations are seen in data2
bins <- ceiling(log2(n)) + 1 # Using Sturges' Formlua
bins <- ceiling(log2(152)) + 1 #Substituting value of n
bins <- ceiling(7.24792751344) + 1 # log2(152) = 7.24792751344
bins <- 8 + 1 # ceiling(7.24792751344) = 8
bins
```

```
## [1] 9
```

c. Create a histogram of `price` using the number of bins suggested by Sturges' formula in 1b. Make sure to appropriately title the histogram and label the axes. Comment on the center, shape, and spread.

```
histo <- ggplot(data1,aes(x=price)) + geom_histogram(bins = bins,color="azure4",
                                                     fill="bisque") +
  ggtitle("Histogram with 9 Bins depicting Price of Cars") +
  labs(y= "Number of Cars", x = "Price of Car")
histo
```



This histogram is positive skewed (right skew), unimodal, asymmetric. Median should be used to find the center because of skewness.

2. Measures of center and spread for the selling price of cars.

    a. Calculate the mean, median, and 10% trimmed mean of the selling price. Report the mean, median, and 10% trimmed mean on the histogram. In particular, create a red vertical line on the histogram at the mean, and report the value of the mean in red next to the line using the form "$\bar{x} =$". Create a blue vertical line on the histogram at the median, and report the value of the median in blue next to the line using the form "$\tilde{x} =$". Create a green vertical line on the histogram at the 10% trimmed mean, and report the value of the 10% trimmed mean in green next to the line using the form "$\bar{x}_{10} =$" (to get $\bar{x}_{10}$ to print on the plot, use `bar(x)[10]` within the `paste()` function).

```
mean <- mean(data2$price)
mean #is the mean
```

```
## [1] 27331.82
```

```
median <- median(data2$price)
median #is the median
```

```
## [1] 22747
```

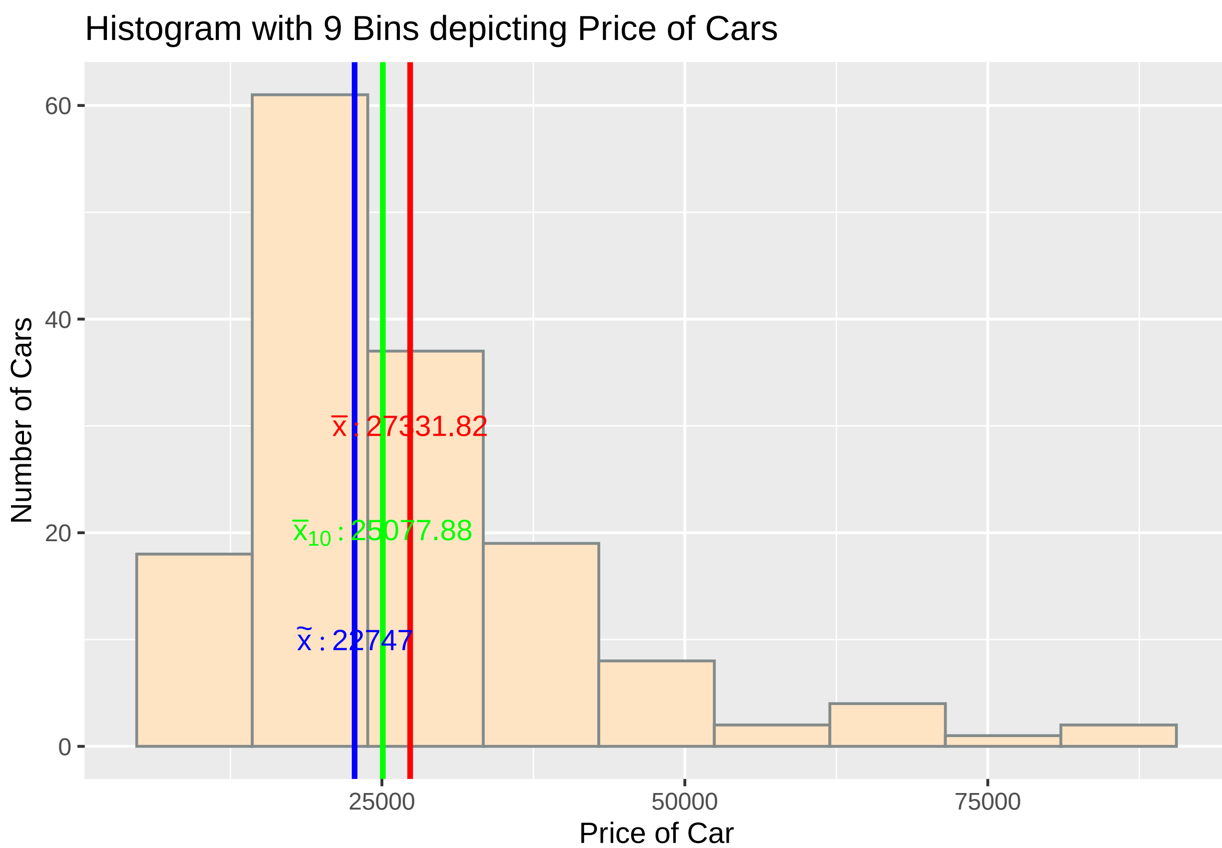
```
trim <- mean(data2$price, trim=0.1)
trim #is the trimmed mean
```

```
## [1] 25077.88
```

```
data1 <- read_csv("car_sales.csv")
```

```
## Rows: 152 Columns: 11
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr (2): Manufacturer, Model
## dbl (9): price, Engine_size, Horsepower, Wheelbase, Width, Length, Curb_weig...
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
histo + geom_vline(aes(xintercept=mean(price)), color="red", size=1) +
geom_vline(aes(xintercept=median(price)), color="blue", size=1) +
geom_vline(aes(xintercept=mean(price,trim=0.1)), color="green", size=1) +
annotate(geom = "text", x = mean, y = 30, parse = TRUE,
         label =paste("bar(x) :", mean), size = 4, col = "red") +
  annotate(geom = "text",x = median,y = 10, parse = TRUE,
           label = paste("tilde(x) :", median), size = 4,  col = "blue") +
  annotate(geom = "text", x = trim, y = 20,parse = TRUE,
           label =paste("bar(x)[10] :", trim), size = 4, col = "green")
```



b. Calculate and report the 25th and 75th percentiles.

```
quantile <- quantile(data2$price, probs = c(.25, .75))
quantile #are the 25th and 75th Percentile
```

```
##      25%      75%
## 17888.75 31938.75
```

c. Calculate and report the interquartile range.

```
IQR <- IQR(data2$price)
IQR  #is the IQR
```

```
## [1] 14050
```

d. Calculate and report the standard span, the lower fence, and the upper fence.

```
span <- IQR*1.5
span #is the Standard Span
```

```
## [1] 21075
```

```r
percentile1 <- quantile(data2$price, probs = c(.25))
percentile2 <- quantile(data2$price, probs = c(.75))
lfence <- percentile1 - (IQR*1.5)
lfence #is the Lower Fence
```

```
##      25%
## -3186.25
```

```r
ufence <- percentile2 + (IQR*1.5)
ufence #is the Upper Fence
```

```
##      75%
## 53013.75
```

e. Are there any outliers? Subset the outlying points. Use code based on the following:

```r
#
outlier1 <- data2[data2$price >= ufence, ]
outlier2 <- data2[data2$price <= lfence, ]
outlier1
```

```
## # A tibble: 9 x 1
##   price
##   <dbl>
## 1 71020
## 2 74970
## 3 69725
## 4 54005
## 5 62000
## 6 85500
## 7 82600
## 8 69700
## 9 60105
```

```r
outlier2 #Are the outliers
```

```
## # A tibble: 0 x 1
## # ... with 1 variable: price <dbl>
```

Yes, there were 9 outliers.

f. Calculate and report the variance, standard deviation, and coefficient of variation of car prices

```r
var <- var(data2$price)
var #is the Variance
```

```
## [1] 207898012
```

```r
sd <- sd(data2$price)
sd #is the Standard Deviation
```

```
## [1] 14418.67
```

```r
cv <- sd/mean
cv #is the  is the coefficient of variation.
```

```
## [1] 0.5275414
```

g. We have seen from the histogram that the data are skewed. Calculate and report the skewness. Comment on this value and how it matches with what you visually see in the histogram.

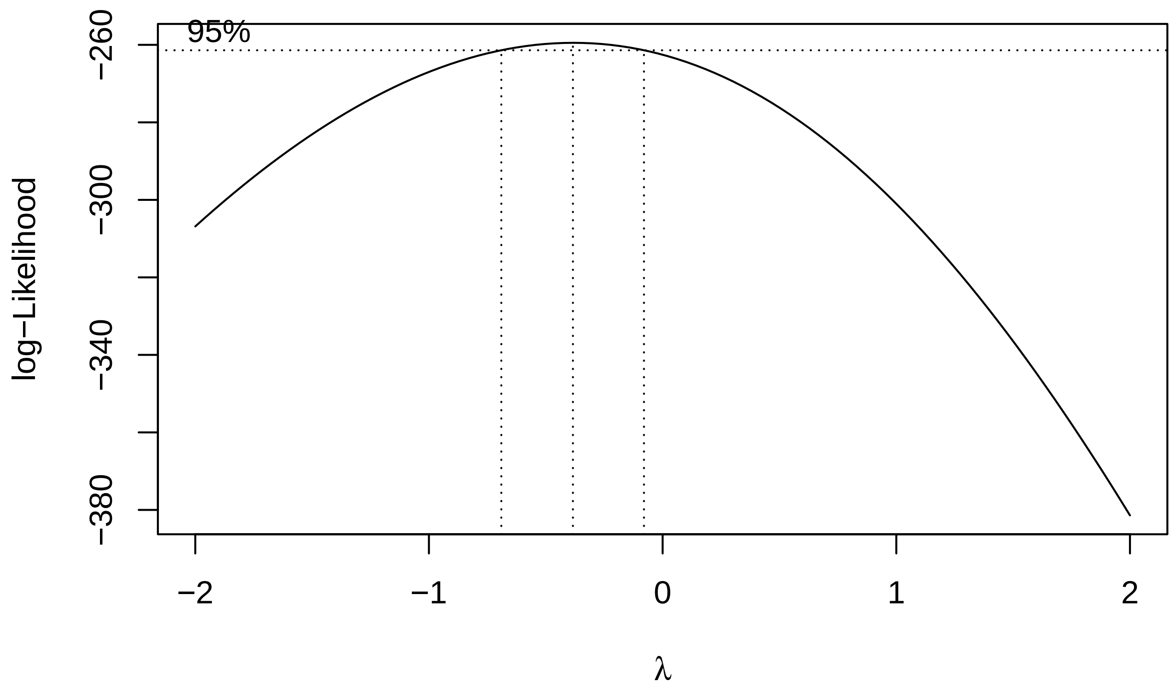
```r
skew <- skewness(Price)
skew #is the Skewness
```

```
## [1] 1.760286
```

As initially observed the histogram was right skewed (positive skew) and the value generated (1.76) confirms the observation.

Question 3: Transforming the data. a. Use a Box-Cox power transformation to appropriately transform the data. In particular, use the `boxcox()` function in the `MASS` library. Report the recommended transformation. Do not apply this transformation to the data yet. (Note: the `boxcox` function automatically produces a plot. You do NOT need to make this in `ggplot2`.)

```r
library(MASS)
bxcx <- boxcox(data1$price ~ 1)
```




```
lambda <- bxcx$x[bxcx$y==max(bxcx$y)]
lambda #finding lambda
```

```
## [1] -0.3838384
```

b. Apply the exact Box-Cox recommended transformation (rounded to four decimal places) to the data (this transformation is hereon referred to as the Box-Cox transformed data). Use the `summary()` function to summarize the results of this transformation.

```
lambda <- round(lambda,4) #rounding off to four decimal places
lambda
```

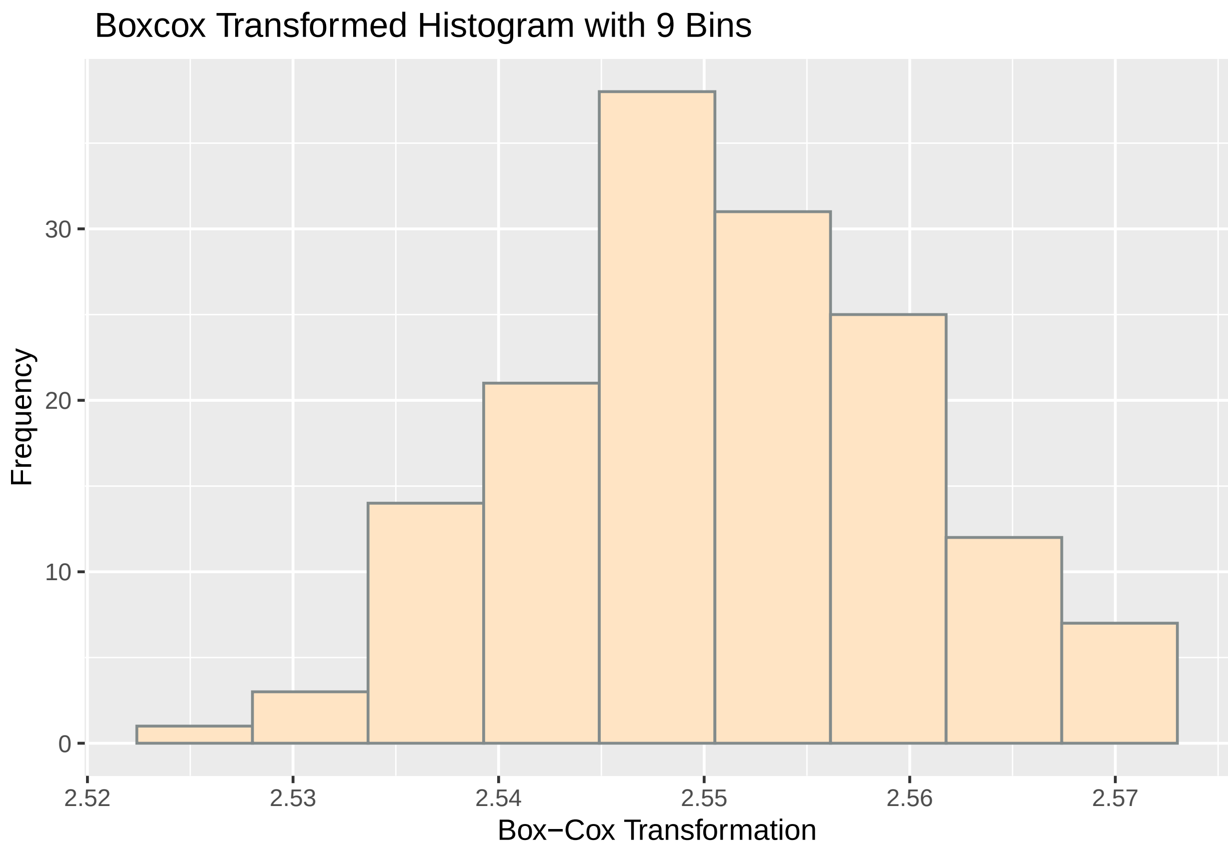
```
## [1] -0.3838
```

```
boxdata <- ( data1$price^lambda -1 )/lambda
summary(boxdata) #Summarize
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   2.527   2.545   2.550   2.551   2.557   2.572
```

c. Create a histogram of the Box-Cox transformed data using the number of bins suggested by Sturges' formula. On this histogram, report the mean, median, and 10% trimmed mean using the same formatting options as in part 2a above. Comment on the center, shape, and spread.

```r
boxhist <- ggplot(data2,aes(x=boxdata)) +
  geom_histogram(bins = bins,color="azure4",fill="bisque") +
  ggtitle(" Boxcox Transformed Histogram with 9 Bins") +
  labs(y= "Frequency", x = "Box-Cox Transformation")
boxhist
```




```r
tmean <- mean(boxdata)
tmean # Mean of BoxCox transformed data
```

```
## [1] 2.550807
```

```r
tmedian <- median(boxdata)
tmedian # Median of BoxCox transformed data
```

```
## [1] 2.550098
```

```r
ttrim <- mean(boxdata, trim=0.1)
ttrim # Trimmed mean of of BoxCox transformed data
```

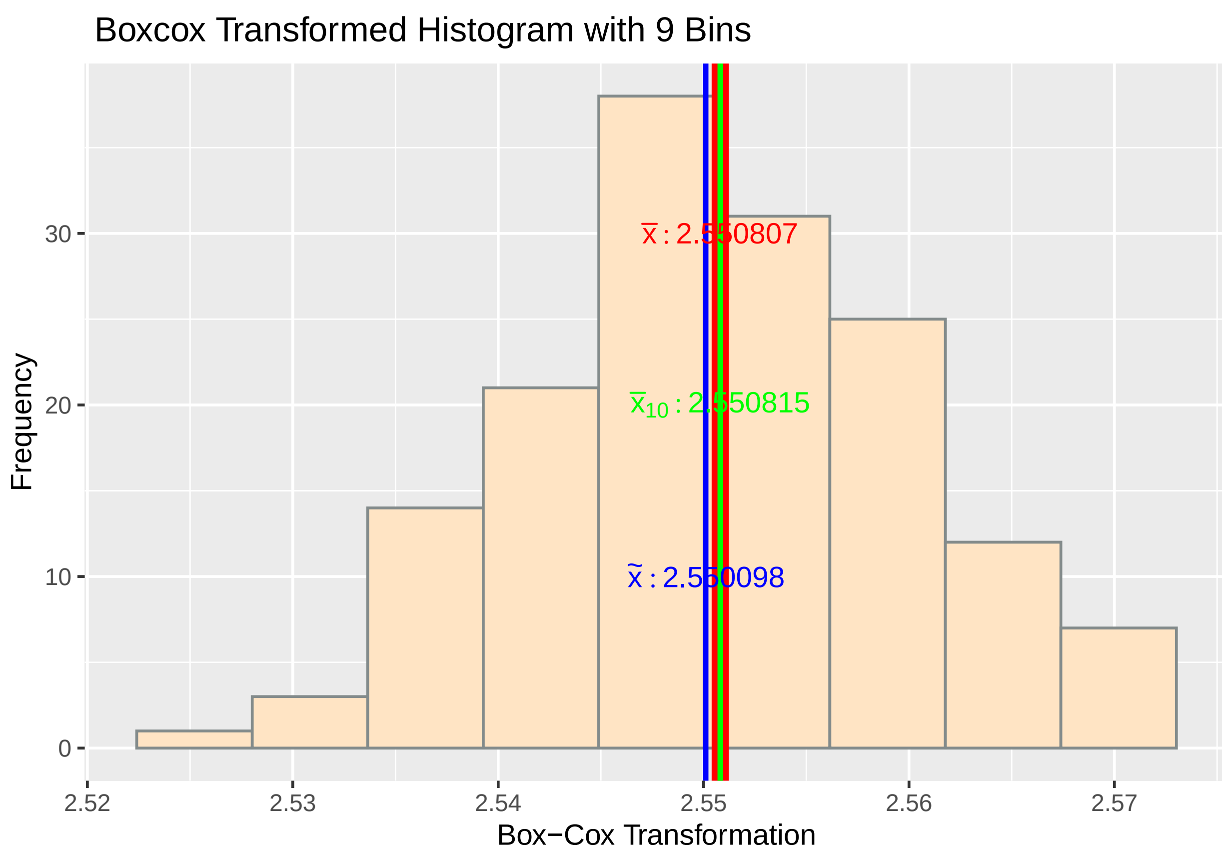
```
## [1] 2.550815
```

```r
boxhist +
  geom_vline(aes(xintercept=mean(boxdata)), color="red",lwd=3)  +

  geom_vline(aes(xintercept=median(boxdata)), color="blue", lwd=1) +
  geom_vline(aes(xintercept=mean(boxdata,trim=0.1)), color="green",  lwd=1)+

  annotate(geom = "text", x = tmean, y = 30, parse = TRUE,
           label= paste("bar(x) :", tmean), size = 4, col ="red") +

  annotate(geom = "text", x = tmedian, y = 10, parse = TRUE,
           label=paste("tilde(x) :", tmedian), size = 4, color= "blue") +
  annotate(geom = "text", x = ttrim, y = 20, parse = TRUE,
           label=paste("bar(x)[10] :", ttrim), size = 4, col="green")
```



Histogram is Unimodal, Symmetric. Transformed data closely follows normal distribution.

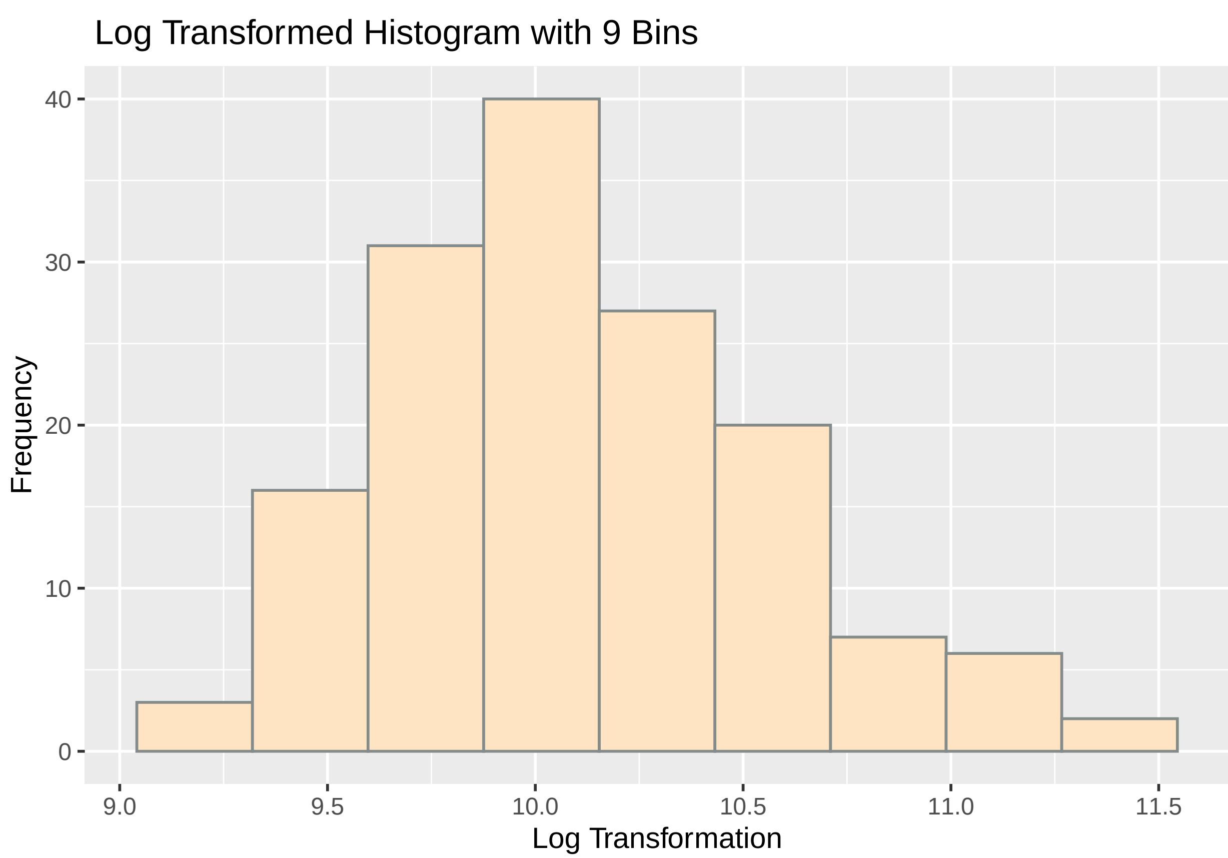
d. As an alternative to the Box-Cox transformation, let's also use a log transformation. Apply the log transformation to the original ‘price‘ data (this transformation is hereon referred to as the log transformed data). Use the ‘summary()‘ function to summarize the results of this transformation.

```r
logdata <- log(data1$price)
summary(logdata) #Summarize
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   9.131   9.792  10.032  10.105  10.372  11.356
```

e. Create a histogram of the log transformed data using the number of bins suggested by Sturges' formula. On this histogram, report the mean, median, and 10% trimmed mean using the same formatting options as in part 2a and 3c above. Comment on the center shape and spread.

```r
loghist <- ggplot(data1,aes(x=logdata)) + geom_histogram(bins = bins,
color="azure4",fill="bisque") + ggtitle(" Log Transformed Histogram with 9 Bins") +
labs(y= "Frequency", x = "Log Transformation")
loghist
```



```r
lmean <- mean(logdata)
lmean # Mean of Log transformed data
```

```
## [1] 10.10457
```

```r
lmedian <- median(logdata)
lmedian # Median of Log transformed data
```

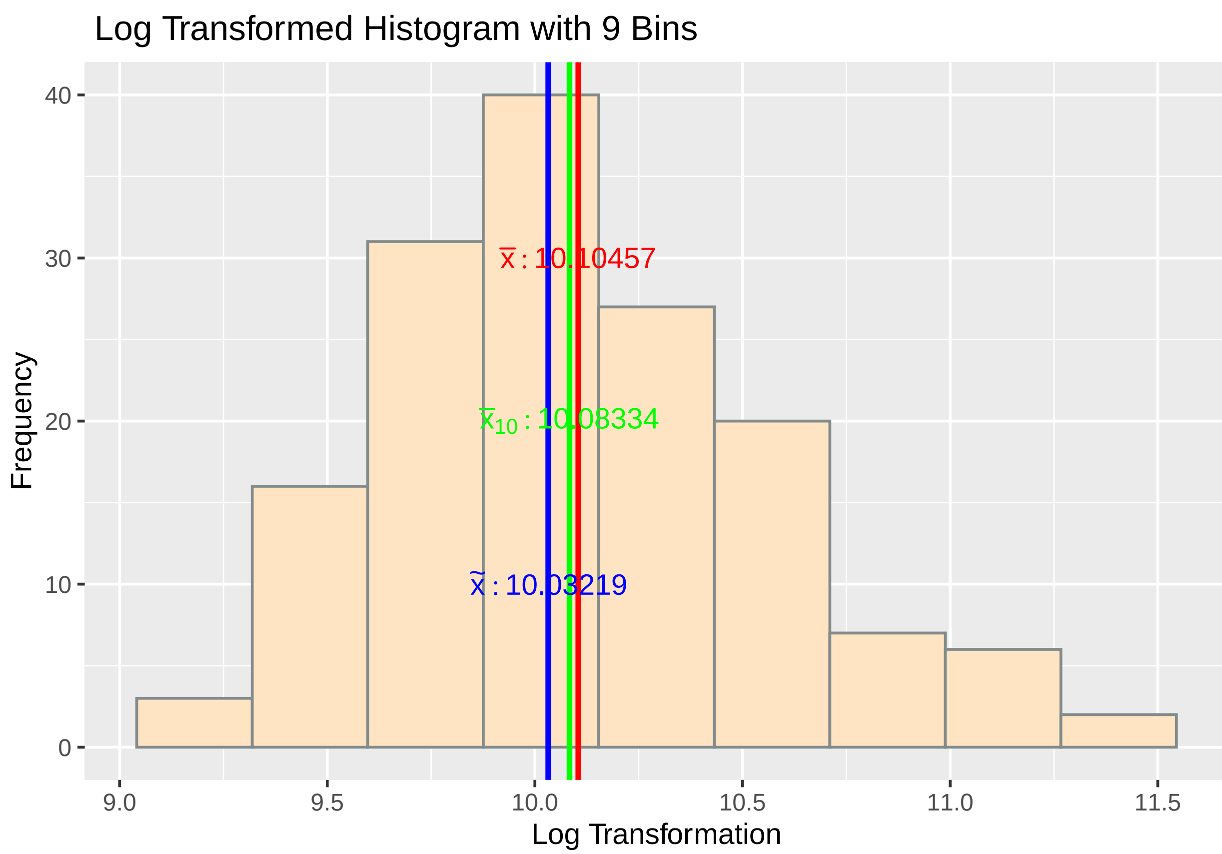
```
## [1] 10.03219
```

```r
ltrim <- mean(logdata, trim=0.1)
ltrim # Trimmed Mean of Log transformed data
```

```
## [1] 10.08334
```

```r
loghist + geom_vline(aes(xintercept=mean(logdata)), color="red", size=1) +
geom_vline(aes(xintercept=median(logdata)), color="blue", size=1) +
geom_vline(aes(xintercept=mean(logdata,trim=0.1)), color="green", size=1) +


annotate(geom = "text", x = lmean, y = 30, parse = TRUE,
label = paste("bar(x) :" ,lmean), size = 4, col="red") +
annotate(geom = "text", x = lmedian, y = 10, parse = TRUE,
label =paste("tilde(x) :", lmedian), size = 4, col="blue") +
annotate(geom = "text", x =ltrim, y = 20, parse = TRUE,
label =paste("bar(x)[10] :", ltrim), size = 4, col="green")
```



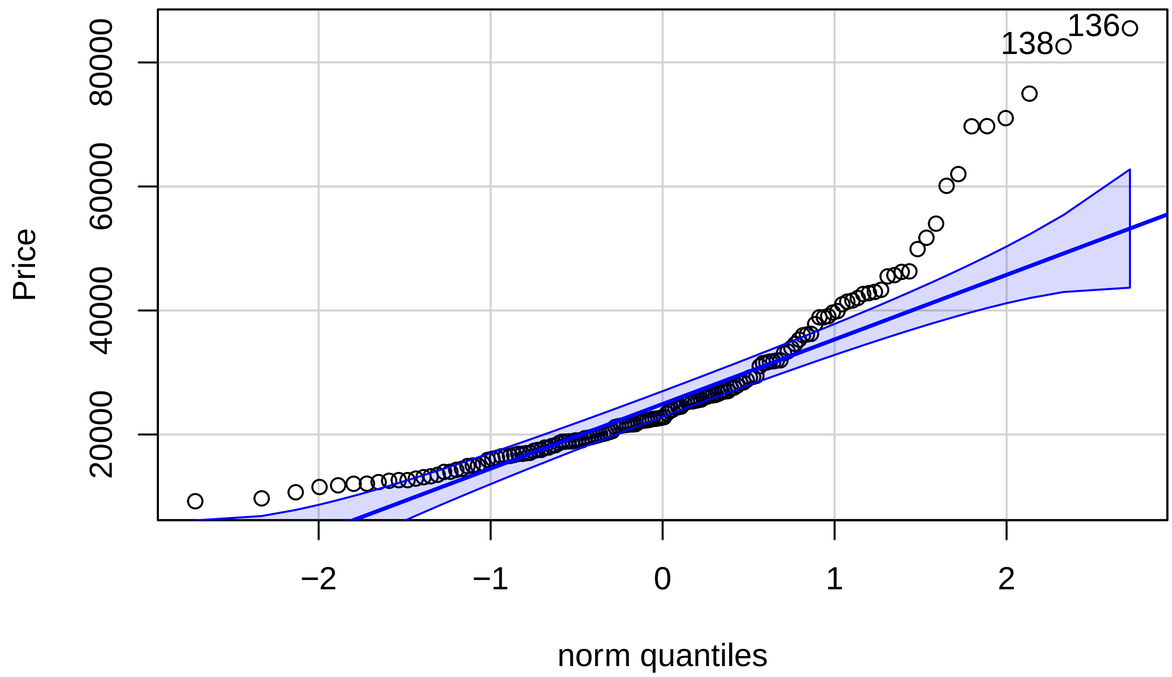
This histogram is unimodal, symmetric and follows normal distibution.

f. Create a qqplot for the original data, a qqplot for the Box-Cox transformed data, and a qqplot of the log transformed data. Comment on the results.

```
library(car) #importing library
```

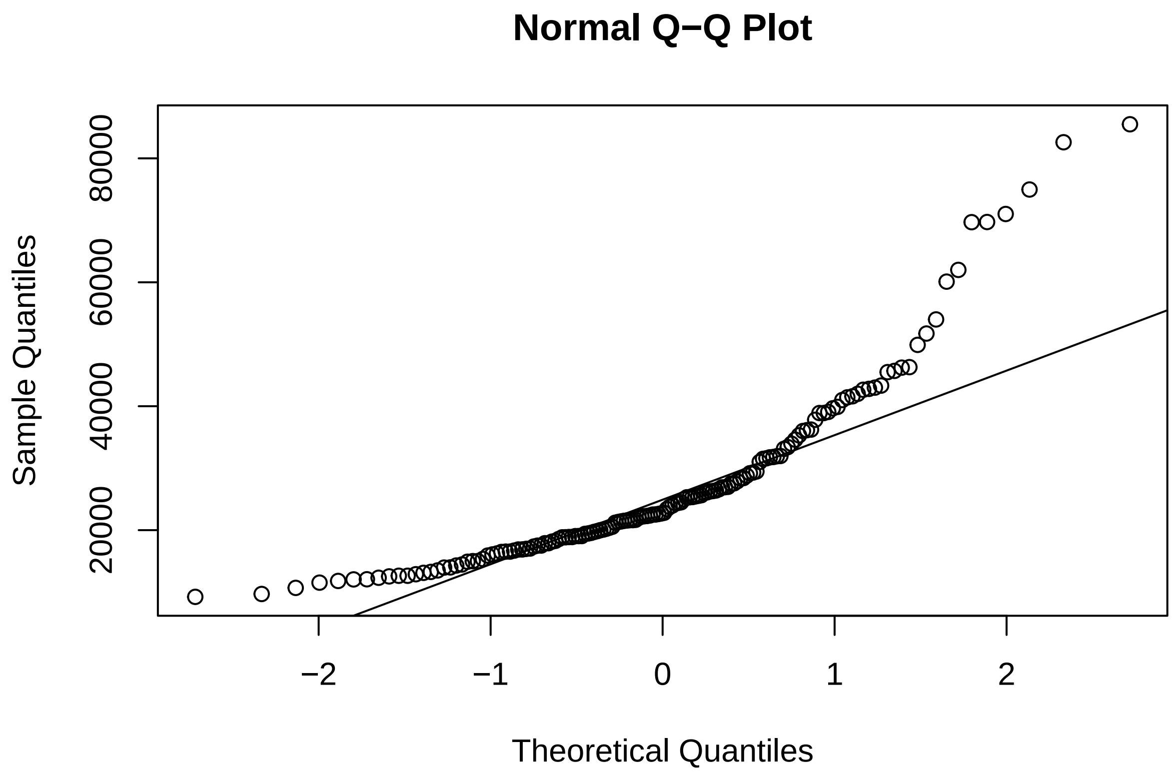
```
## Loading required package: carData
```

```
qqPlot(Price) #qqPlot of the original data
```

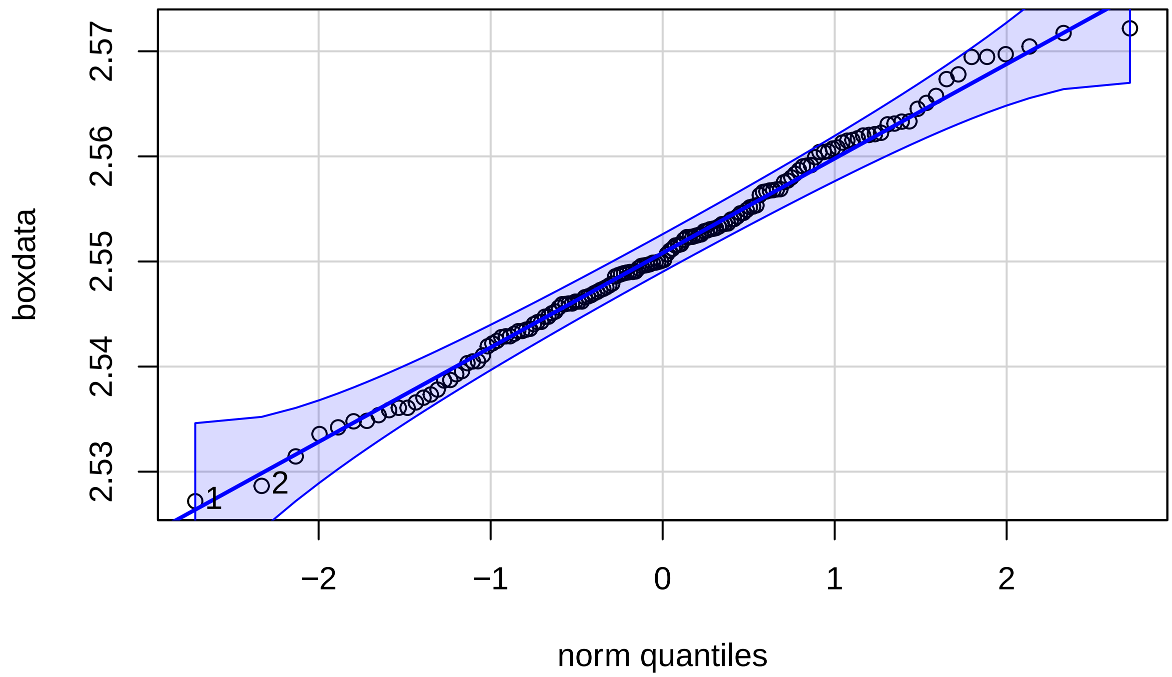



```
## [1] 136 138
```

```
qqnorm(Price); qqline(Price)
```

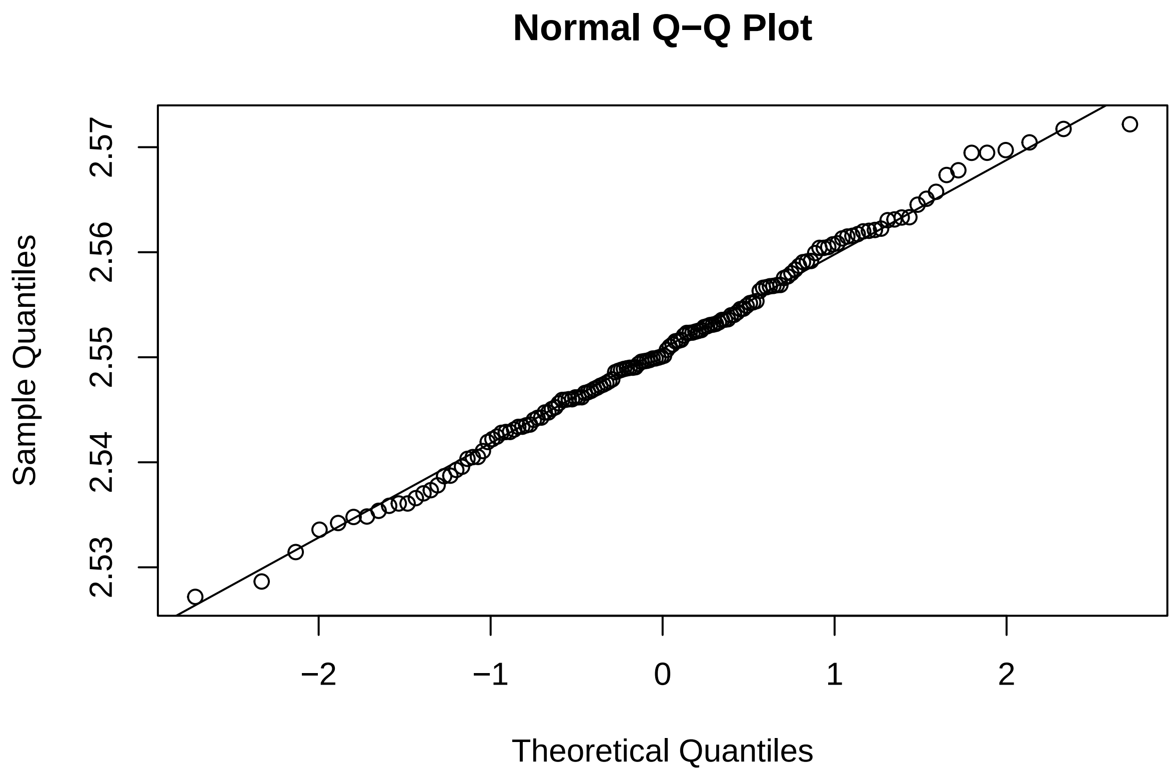



```
qqPlot(boxdata) #qqPlot of BoxCox transformed data
```

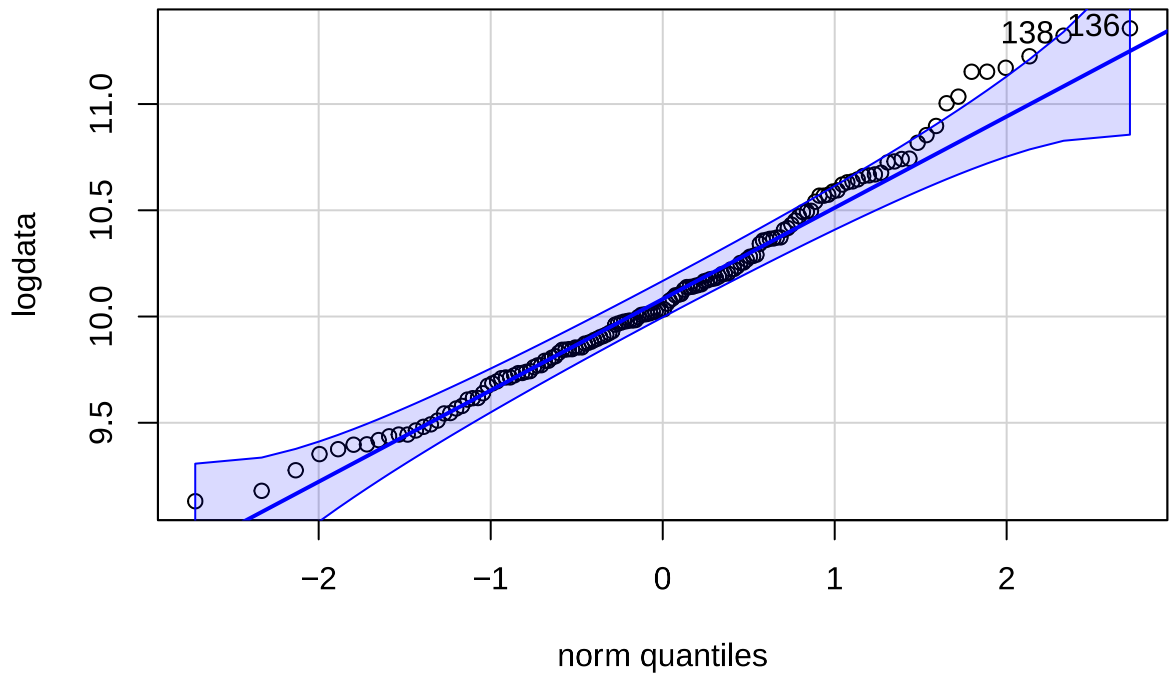



```
## [1] 1 2
```

```
qqnorm(boxdata); qqline(boxdata)
```

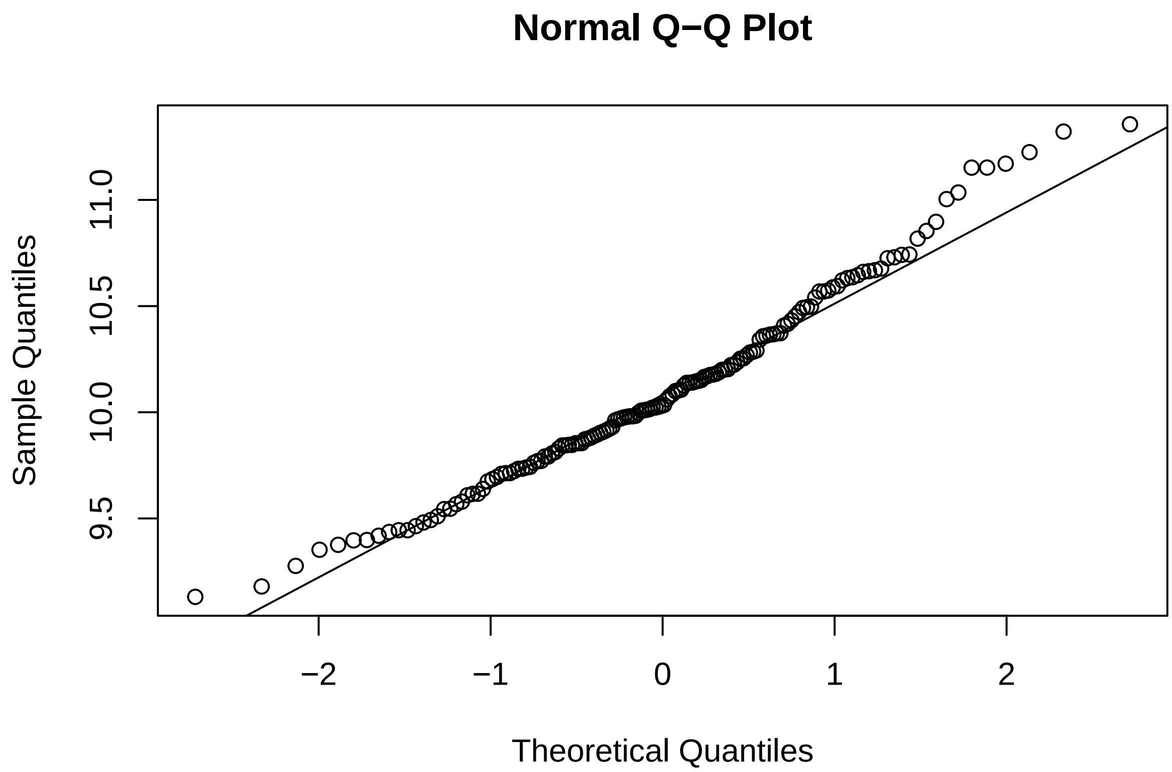



```
qqPlot(logdata) #qqPlot of Log Transformed data
```




```
## [1] 136 138
```

```
qqnorm(logdata); qqline(logdata)
```




```
The original data doesn't follow normal distribution and the shape suggests
positive (right) skewness. Box cox transformed data closely follows normal
distribution as the points are approximately on the line y=x.
Log transformed data also follows normal distribution but there are more
outliers than box cox transformed data.
``
```

g. Evaluate the empirical rule for the original data, the Box-Cox transformed data, and the log transformed data. In particular, make a table similar to that on slide 71 of the Chapter 2 notes. Comment on the results. Do either of the transformed data seem to be "better" to work with? Note, you can use code similar to the following to answer this question:

```r
sd
```

```
## [1] 14418.67
```

```r
sdbox <- sd(boxdata)
sdlog <- sd(logdata)

mat_rix <- matrix(NA, nrow=9, ncol=5)

colnames(mat_rix) <- c("k","xbar-k*s", "xbar+k*s", "Theoretical %","Actual %")

rownames(mat_rix) <- c("Original-Data","","","Box-Cox-Transformed","","","Log-Transformed","","")
```

```
mat_rix[,1] <- c(1,2,3)

mat_rix[,4]<-c(68,95,99.7)

# xbar -k*s

mat_rix[1,2] <- mean - sd
mat_rix[2,2] <- mean - 2*sd
mat_rix[3,2] <- mean - 3*sd
mat_rix[4,2] <- tmean - sdbox
mat_rix[5,2] <- tmean - sdbox*2
mat_rix[6,2] <- tmean - sdbox*3
mat_rix[7,2] <- lmean - sdlog*1
mat_rix[8,2] <- lmean - sdlog*2
mat_rix[9,2] <- lmean - sdlog*3

# xbar + k*s

mat_rix[1,3] <- mean + sd
mat_rix[2,3] <- mean + sd*2
mat_rix[3,3] <- mean + sd*3
mat_rix[4,3] <- tmean + sdbox
mat_rix[5,3] <- tmean + sdbox*2
mat_rix[6,3] <- tmean + sdbox*3
mat_rix[7,3] <- lmean + sdlog
mat_rix[8,3] <- lmean + sdlog*2
mat_rix[9,3] <- lmean + sdlog*3

mat_rix[1,5] <- sum(Price >=mean-1*sd
                    &  Price<= mean+1*sd)/length(Price)*100
mat_rix[2,5] <- sum(Price >=mean-2*sd
                    &  Price<= mean+2*sd)/length(Price)*100

mat_rix[3,5] <- sum(Price >=mean-3*sd
                    &  Price<= mean+3*sd)/length(Price)*100

mat_rix[4,5] <- sum(boxdata >=tmean-1*sdbox
                    &  boxdata<= tmean+1*sdbox)/length(boxdata)*100

mat_rix[5,5] <- sum(boxdata >=tmean-2*sdbox
                    &  boxdata<= tmean+2*sdbox)/length(boxdata)*100

mat_rix[6,5] <- sum(boxdata >=tmean-3*sdbox
                    &  boxdata<= tmean+3*sdbox)/length(boxdata)*100

mat_rix[7,5] <- sum(logdata >=lmean-1*sdlog
                    &  logdata<= lmean+1*sdlog)/length(logdata)*100

mat_rix[8,5] <- sum(logdata >=lmean-2*sdlog
                    &  logdata<= lmean+2*sdlog)/length(logdata)*100

mat_rix[9,5] <- sum(logdata >=lmean-3*sdlog
                    &  logdata<= lmean+3*sdlog)/length(logdata)*100
```

```r
library(knitr)
    kable(x=mat_rix, digits=2,row.names=T, format="markdown")
```

	k	xbar-k*s	xbar+k*s	Theoretical %	Actual %
Original-Data	1	12913.15	41750.49	68.0	78.95
	2	-1505.52	56169.16	95.0	94.74
	3	-15924.18	70587.83	99.7	97.37
Box-Cox-Transformed	1	2.54	2.56	68.0	66.45
	2	2.53	2.57	95.0	94.08
	3	2.52	2.58	99.7	100.00
Log-Transformed	1	9.65	10.56	68.0	66.45
	2	9.19	11.02	95.0	94.08
	3	8.73	11.48	99.7	100.00

Both transformed data are better to work than the original data. In particular Box Cox transformed data is superior to work with as the actual and theoretical % in range are very close. Even using histograms and qqplot, we can identify that box cox transformed data follows normal distribution and hence it is better to work with.

h. In your own words, provide some intuition about (1) why car price may not follow a normal distribution, and (2) why it may be useful to transform the data into a form that more closely follows a normal distribution.

1) Generally, it has been seen that real life data doesn't follow normal distribution. Considering prices of real cars and classifying them into luxury cars, exotic cars, daily drives and cheap cars, we can observe that luxury cars and exotic cars are rare compared to the other segment of cars. This was rightly observed in the histogram too, histogram was right skewed which implied that cars with lower price range were significantly more compared to cars with high price range. Hence, price doesn't follow a normal distribution.

2) Possibility of prediction is higher and more accurate if the data transformed follows normal distribution closely. Basically, any data which is transformed and follows normal distribution means that most of the values are near the mean of the dataset.

Short Answers:

- About how long did this assignment take you? Did you feel it was too long, too short, or reasonable? Around 4-5 hours in total, apart from that I studied for 2-3 hours to have a better understanding of the concepts and some functions in R Language. I felt the assignment was reasonable.
- Who, if anyone, did you work with on this assignment? No One
- What questions do you have relating to any of the material we have covered so far in class?
  Basics like mean, median, SD, Variance, CV, Histograms and some advanced topics like Box Cox and Log transformation were covered in the class.