# DSCC/CSC/TCS 462 Assignment 0

Due Thursday, September 8, 2022 by 3:59 p.m.

Daxiang Na

09.10.2022

This assignment will cover material from Lectures 1 and 2. You are expected to use the `ggplot2` library in `R` for completing all the graphics. To learn more about graphics using `ggplot2`, please read through the guide available here: http://www.cookbook-r.com/Graphs/. This is a wonderful open source textbook that walks through examples of many different graphics in `ggplot2`. If you have not done so already, start by installing the library. In the R console (i.e. NOT in your .RMD file), run the code `install.packages("ggplot2")`. Then, in your .RMD file, load the library as follows:

```
library(ggplot2)
```

For this first assignment, we will use the "car_sales.csv" dataset, which includes information about 152 different cars. In particular, we will mainly focus on the selling price of cars throughout this assignment.

1. Getting familiar with the dataset via exploratory data analysis.

    a. Read the data into RStudio and summarize the data with the `summary()` function.

```
df <- read.csv("car_sales.csv")
summary(df)
```

```
##  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
```

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

```
k = log2(152) + 1
print(paste0("Result of Sturges' formula is ", k))
```

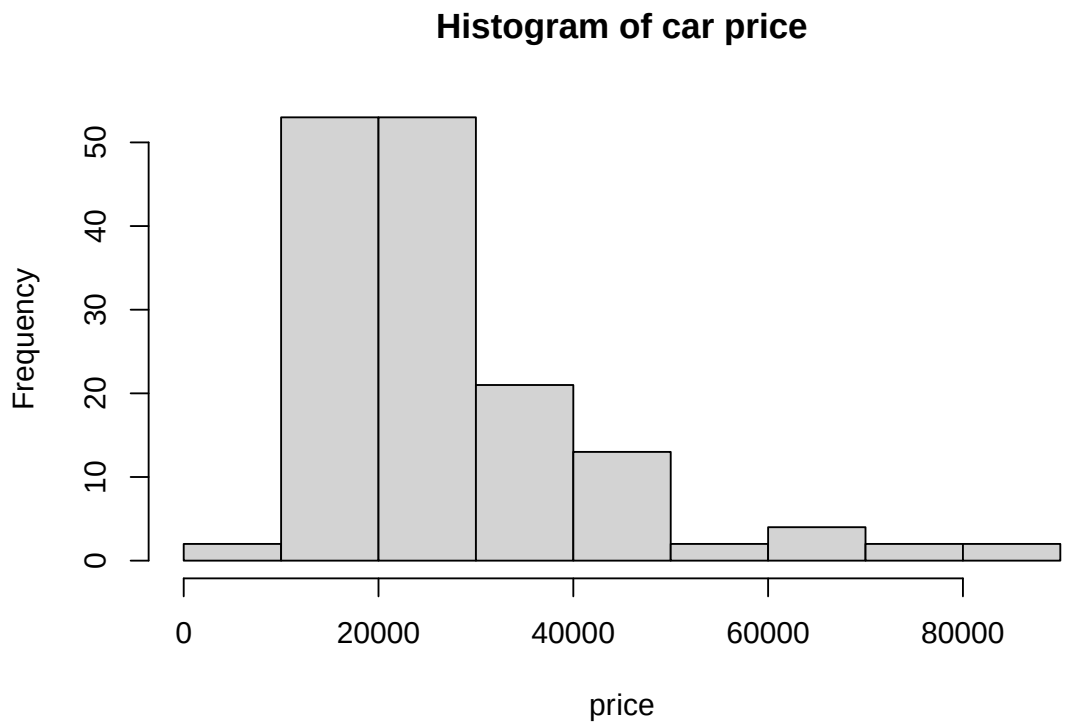
```
## [1] "Result of Sturges' formula is 8.24792751344359"
```

```
print("Number of bins is 9 after rounding up.")
```

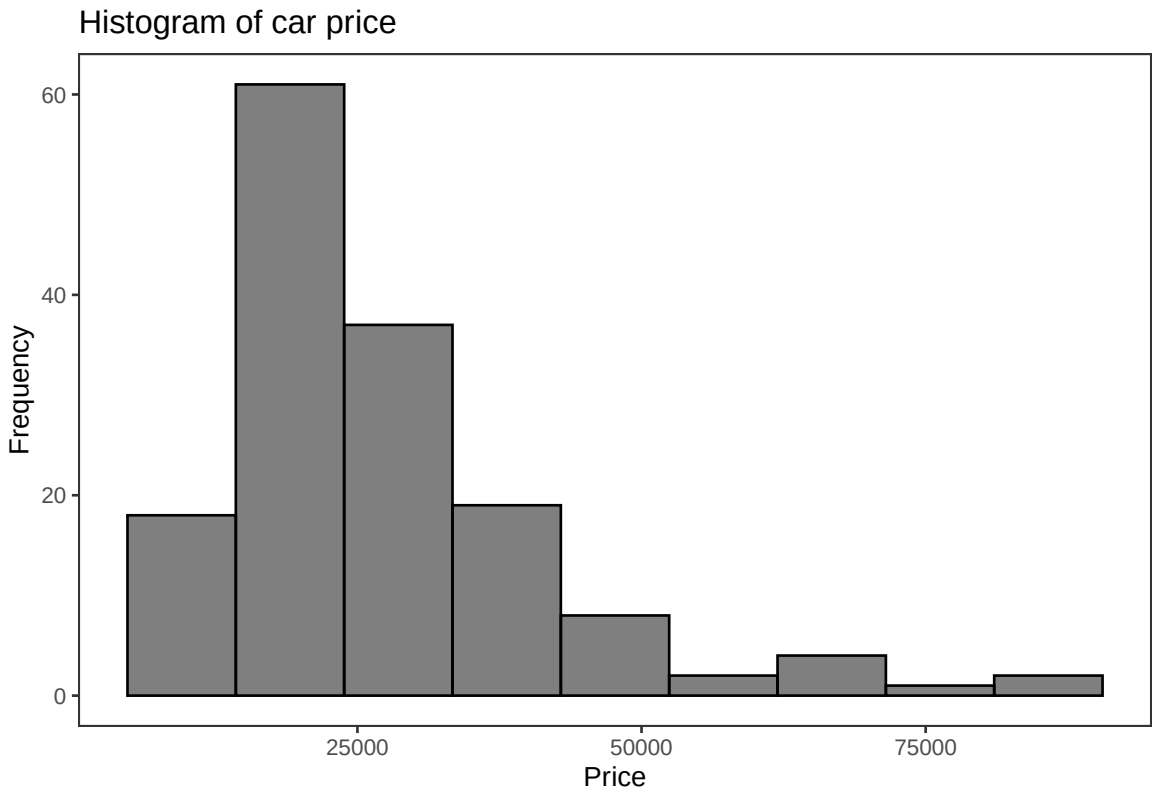
```
## [1] "Number of bins is 9 after rounding up."
```

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.

```
hist(df$price, breaks = "Sturges", xlab = "price",
    main = "Histogram of car price")
```




```
ggplot(df, aes(x = price)) + geom_histogram(bins = 9,
    color = "black", fill = "black", alpha = 0.5) +
    theme_bw() + theme(panel.grid = element_blank()) +
    labs(title = "Histogram of car price", x = "Price",
        y = "Frequency")
```



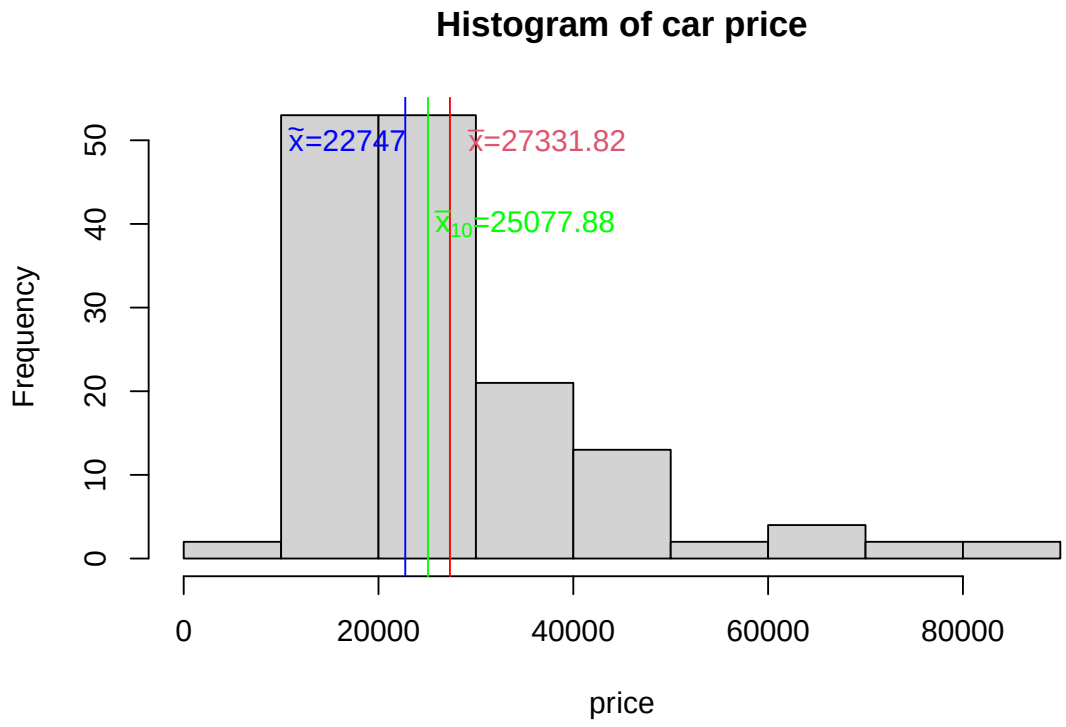

This histogram has one center. Its shape is asymmetric, with the peak closer to the left (right skewed), most of the data points appear to be close to the center.

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(df$price)
median <- median(df$price)
trimmed_mean <- mean(df$price, trim = 0.1)
hist(df$price, breaks = "Sturges", xlab = "price",
    main = "Histogram of car price")
abline(v = mean, col = "red")
text(mean + 10000, 50, substitute(paste(bar(x), "=",
    m), list(m = round(mean, 3))), col = 2)
```

```
abline(v = median, col = "blue")
text(median - 6000, 50, substitute(paste(tilde(x),
    "=", m), list(m = round(median, 3))), col = "blue")
abline(v = trimmed_mean, col = "green")
text(trimmed_mean + 10000, 40, substitute(paste(bar(x)[10],
    "=", m), list(m = round(trimmed_mean, 3))), col = "green")
```



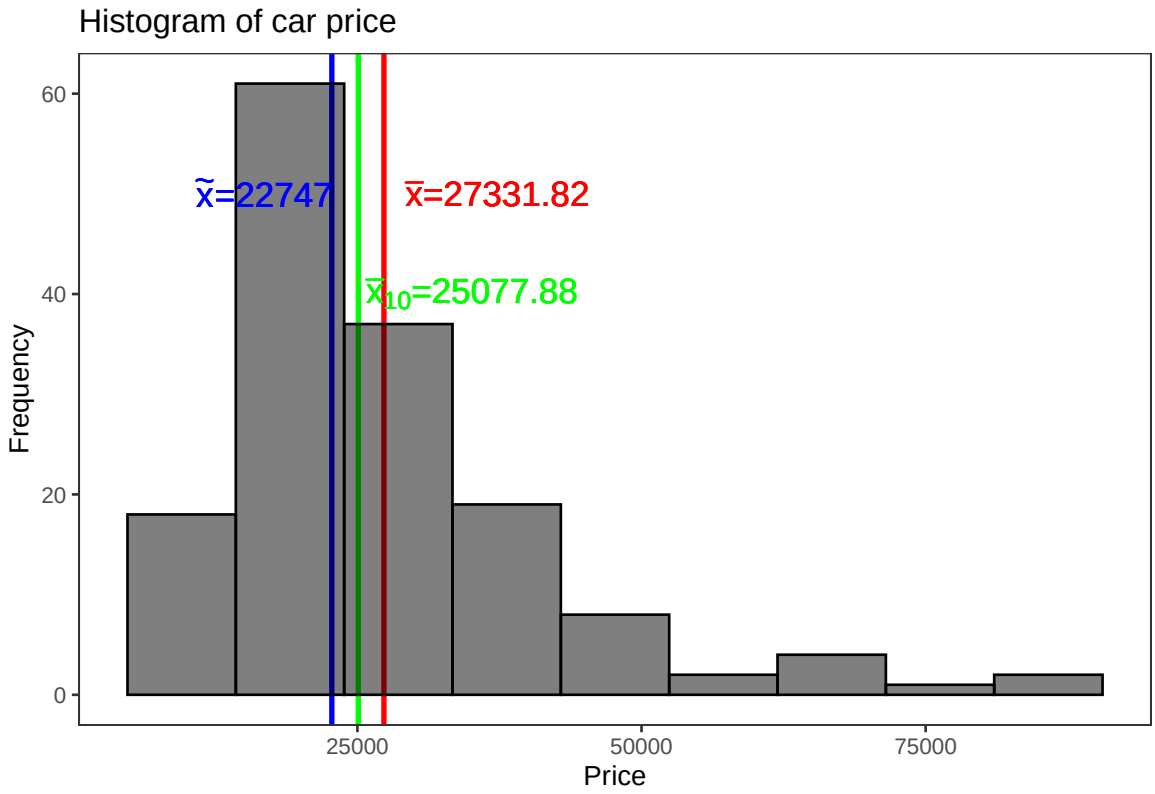

```
ggplot(df, aes(x = price)) + geom_vline(xintercept = c(mean(df$price),
    median(df$price), mean(df$price, trim = 0.1)),
    color = c("red", "blue", "green"), size = 1) +
    geom_histogram(bins = 9, color = "black", fill = "black",
        alpha = 0.5) + annotate("text", x = mean +
    10000, y = 50, label = substitute(paste(bar(x),
    "=", m), list(m = round(mean, 3))), color = "red",
    size = 5) + annotate("text", x = median - 6000,
    y = 50, label = substitute(paste(tilde(x), "=",
        m), list(m = round(median, 3))), color = "blue",
    size = 5) + annotate("text", x = trimmed_mean +
    10000, y = 40, label = substitute(paste(bar(x)[10],
    "=", m), list(m = round(trimmed_mean, 3))), color = "green",
    size = 5) + theme_bw() + theme(panel.grid = element_blank()) +
    labs(title = "Histogram of car price", x = "Price",
```

```
        y = "Frequency")
```

```
## Warning in is.na(x): is.na() applied to non-(list or vector) of type 'language'

## Warning in is.na(x): is.na() applied to non-(list or vector) of type 'language'

## Warning in is.na(x): is.na() applied to non-(list or vector) of type 'language'
```




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

```
Q1 <- quantile(df$price, 0.25)
Q3 <- quantile(df$price, 0.75)
print(paste0("The 25th percentile is ", Q1))
```

```
## [1] "The 25th percentile is 17888.75"
```

```
print(paste0("The 75th percentile is ", Q3))
```

```
## [1] "The 75th percentile is 31938.75"
```

c. Calculate and report the interquartile range.

```r
distance <- Q3 - Q1
print(paste0("The interquartile range is ", distance))
```

```
## [1] "The interquartile range is 14050"
```

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

```r
lower_fence <- Q1 - 1.5 * (Q3 - Q1)
upper_fence <- Q3 + 1.5 * (Q3 - Q1)
standard_span <- 1.5 * (Q3 - Q1)
print(paste0("Lower fence is ", lower_fence, "."))
```

```
## [1] "Lower fence is -3186.25."
```

```r
print(paste0("Upper fence is ", upper_fence, "."))
```

```
## [1] "Upper fence is 53013.75."
```

```r
print(paste0("Standard Span is ", standard_span, "."))
```

```
## [1] "Standard Span is 21075."
```

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

```r
upper <- df[df$price >= upper_fence, "price"]  #upper outliers
lower <- df[df$price <= lower_fence, "price"]  #lower outliers
# Use upper and lower fence values from part g.
upper <- paste0(upper)
upper <- paste0(upper, collapse = ",")
print(paste0("upper outliers are ", upper, "."))
```

```
## [1] "upper outliers are 71020,74970,69725,54005,62000,85500,82600,69700,60105."
```

```r
print("There is no lower outlier.")
```

```
## [1] "There is no lower outlier."
```

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

```
variance <- var(df$price)  # variance of car prices
std <- sd(df$price)  # standard deviation
CV <- std/mean  # coefficient of variation of car prices
print(paste0("variance of car prices is ", variance))
```

```
## [1] "variance of car prices is 207898011.65698"
```

```
print(paste0("standard deviation of car prices is ",
    std))
```

```
## [1] "standard deviation of car prices is 14418.6688587047"
```

```
print(paste0("coefficient of variation of car prices is ",
    CV))
```

```
## [1] "coefficient of variation of car prices is 0.527541437389256"
```

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.

```
library(moments)
skew <- skewness(df$price)
print(paste0("The skewness of car prices is ", skew))
```

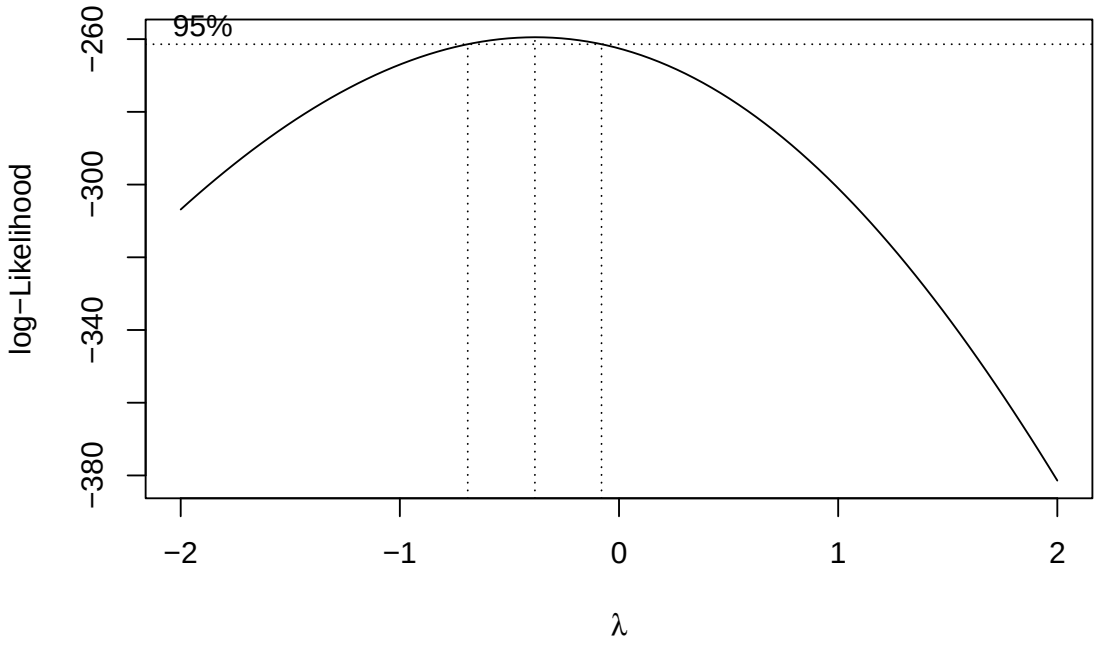
```
## [1] "The skewness of car prices is 1.76028644928878"
```

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`.)

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.

```
library(MASS)
bc1 <- boxcox(df$price ~ 1)
```



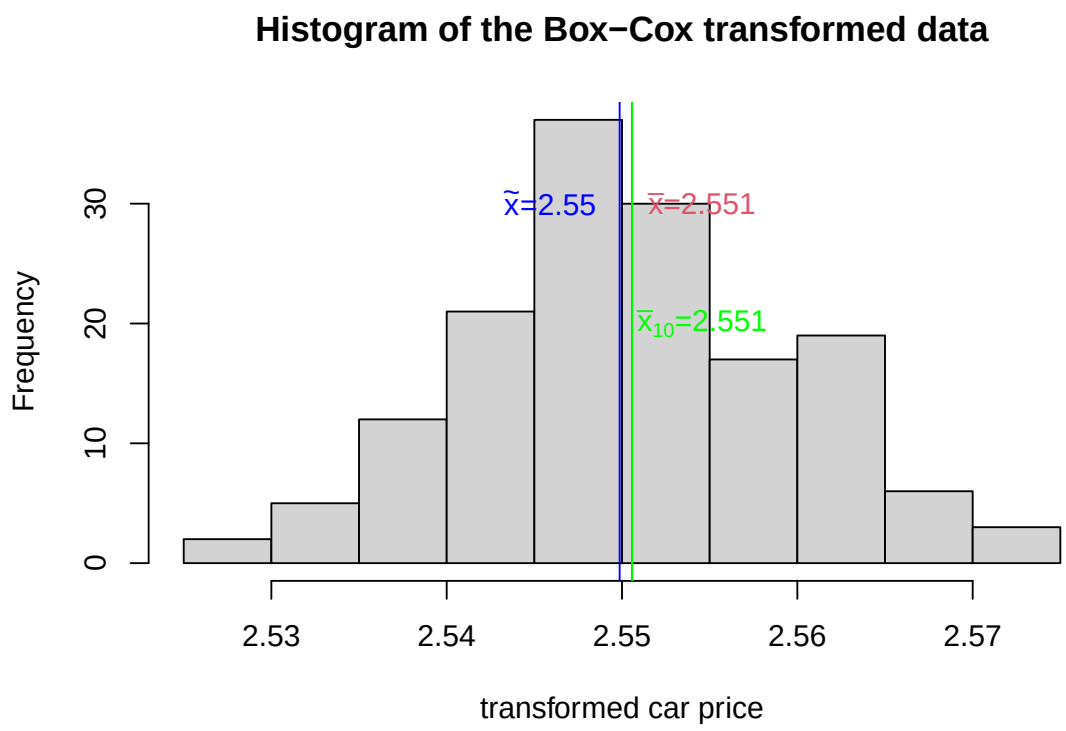

```
lambda <- bc1$x[bc1$y == max(bc1$y)]
trans <- (df$price^lambda - 1)/lambda
summary(trans)
```

```
##    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.

```
mean1 <- mean(trans)
median1 <- median(trans)
trimmed_mean1 <- mean(trans, trim = 0.1)
hist(trans, breaks = "Sturges", xlab = "transformed car price",
    main = "Histogram of the Box-Cox transformed data")
abline(v = mean1, col = "red")
```

```
text(mean1 + 0.004, 30, substitute(paste(bar(x), "=",
    m), list(m = round(mean1, 3))), col = 2)
abline(v = median1, col = "blue")
text(median1 - 0.004, 30, substitute(paste(tilde(x),
    "=", m), list(m = round(median1, 3))), col = "blue")
abline(v = trimmed_mean1, col = "green")
text(trimmed_mean1 + 0.004, 20, substitute(paste(bar(x)[10],
    "=", m), list(m = round(trimmed_mean1, 3))), col = "green")
```



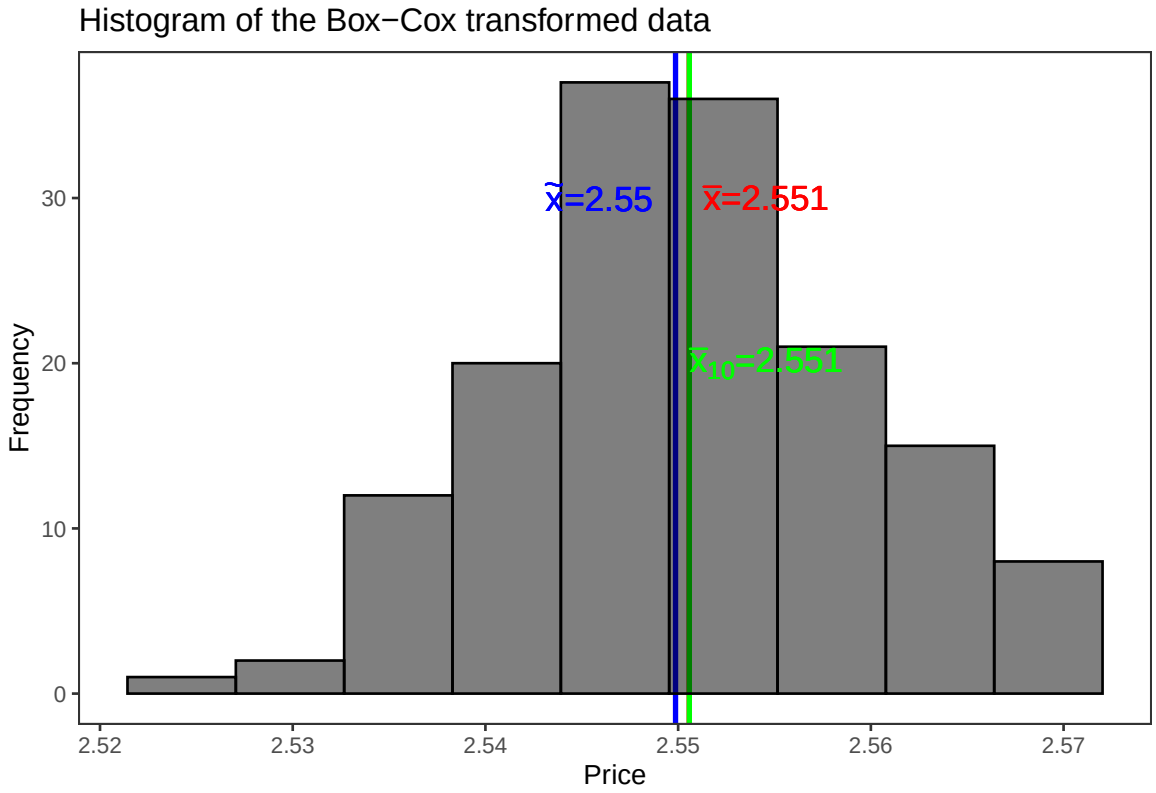

```
df1 <- df
df1[, "price"] <- trans
ggplot(df1, aes(x = price)) + geom_vline(xintercept = c(mean(df1$price),
    median(df1$price), mean(df1$price, trim = 0.1)),
    color = c("red", "blue", "green"), size = 1) +
    geom_histogram(bins = 9, color = "black", fill = "black",
        alpha = 0.5) + annotate("text", x = mean1 +
    0.004, y = 30, label = substitute(paste(bar(x),
    "=", m), list(m = round(mean1, 3))), color = "red",
    size = 5) + annotate("text", x = median1 - 0.004,
    y = 30, label = substitute(paste(tilde(x), "=",
        m), list(m = round(median1, 3))), color = "blue",
    size = 5) + annotate("text", x = trimmed_mean1 +
```

```
    0.004, y = 20, label = substitute(paste(bar(x)[10],
    "=", m), list(m = round(trimmed_mean1, 3))), color = "green",
    size = 5) + theme_bw() + theme(panel.grid = element_blank()) +
    labs(title = "Histogram of the Box-Cox transformed data",
        x = "Price", y = "Frequency")
```

```
## Warning in is.na(x): is.na() applied to non-(list or vector) of type 'language'

## Warning in is.na(x): is.na() applied to non-(list or vector) of type 'language'

## Warning in is.na(x): is.na() applied to non-(list or vector) of type 'language'
```




This histogram has one center. Its shape is symmetric, with peak at center. Most of the data points appear to be close to the center instead of largely spread.

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.



Text

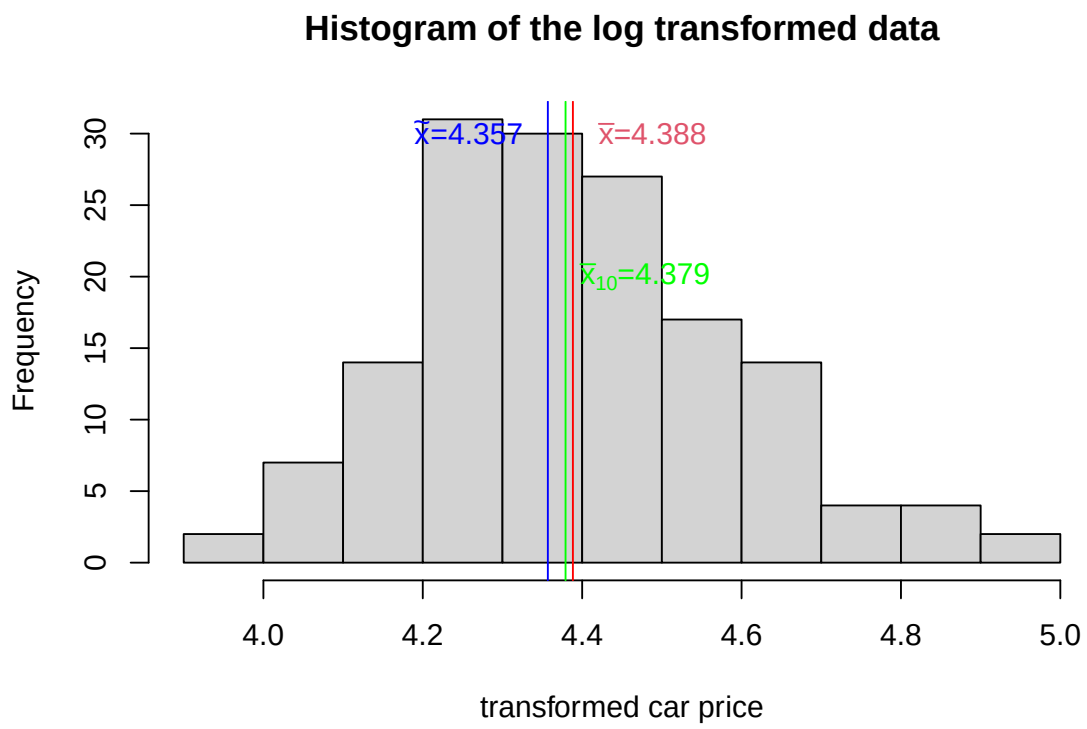
```
trans2 <- log10(df$price)
summary(trans2)
```

Instructor: This should be natural log but not log 10

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   3.965   4.253   4.357   4.388   4.504   4.932
```

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.

```
mean2 <- mean(trans2)
median2 <- median(trans2)
trimmed_mean2 <- mean(trans2, trim = 0.1)
hist(trans2, breaks = "Sturges", xlab = "transformed car price",
    main = "Histogram of the log transformed data")
abline(v = mean2, col = "red")
text(mean2 + 0.1, 30, substitute(paste(bar(x), "=",
    m), list(m = round(mean2, 3))), col = 2)
abline(v = median2, col = "blue")
text(median2 - 0.1, 30, substitute(paste(tilde(x),
    "=", m), list(m = round(median2, 3))), col = "blue")
abline(v = trimmed_mean2, col = "green")
text(trimmed_mean2 + 0.1, 20, substitute(paste(bar(x)[10],
    "=", m), list(m = round(trimmed_mean2, 3))), col = "green")
```

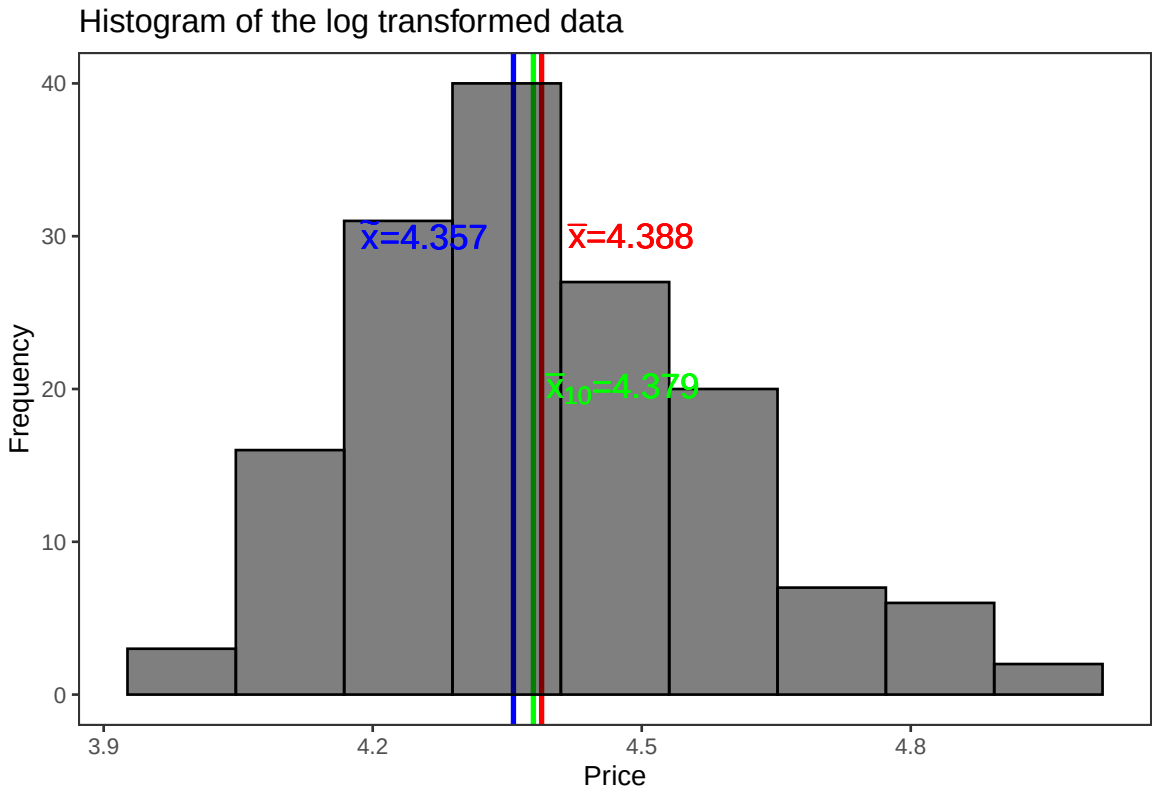



```
df2 <- df
df2[, "price"] <- trans2
ggplot(df2, aes(x = price)) + geom_vline(xintercept = c(mean(df2$price),
    median(df2$price), mean(df2$price, trim = 0.1)),
    color = c("red", "blue", "green"), size = 1) +
    geom_histogram(bins = 9, color = "black", fill = "black",
        alpha = 0.5) + annotate("text", x = mean2 +
    0.1, y = 30, label = substitute(paste(bar(x), "=",
    m), list(m = round(mean2, 3))), color = "red",
    size = 5) + annotate("text", x = median2 - 0.1,
    y = 30, label = substitute(paste(tilde(x), "=",
        m), list(m = round(median2, 3))), color = "blue",
    size = 5) + annotate("text", x = trimmed_mean2 +
    0.1, y = 20, label = substitute(paste(bar(x)[10],
    "=", m), list(m = round(trimmed_mean2, 3))), color = "green",
    size = 5) + theme_bw() + theme(panel.grid = element_blank()) +
    labs(title = "Histogram of the log transformed data",
        x = "Price", y = "Frequency")
```

```
## Warning in is.na(x): is.na() applied to non-(list or vector) of type 'language'

## Warning in is.na(x): is.na() applied to non-(list or vector) of type 'language'
```

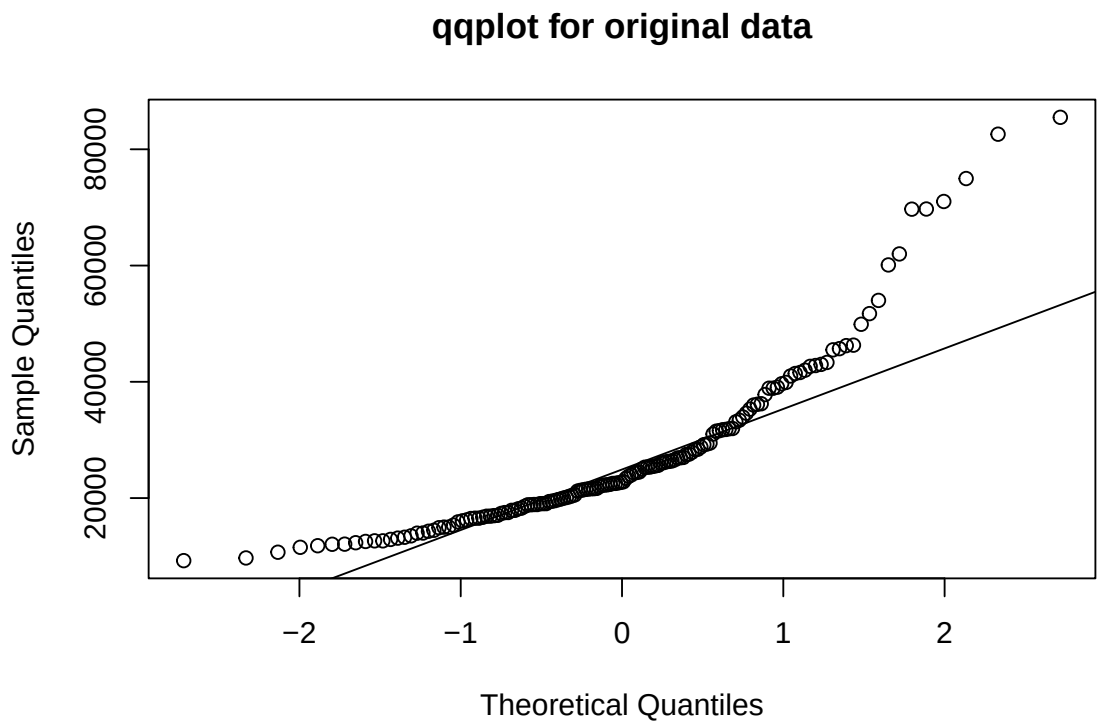
```
## Warning in is.na(x): is.na() applied to non-(list or vector) of type 'language'
```



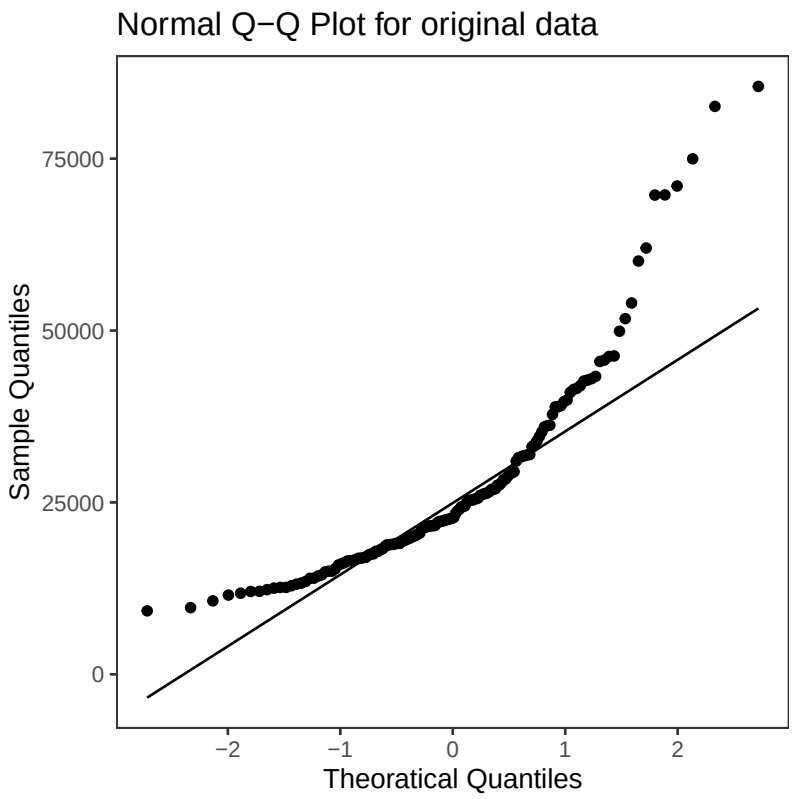

This histogram has one center. Its shape is asymmetric, with peak slightly closer to the left. Most of the data points appear to be close to the center instead of largely spread.

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.

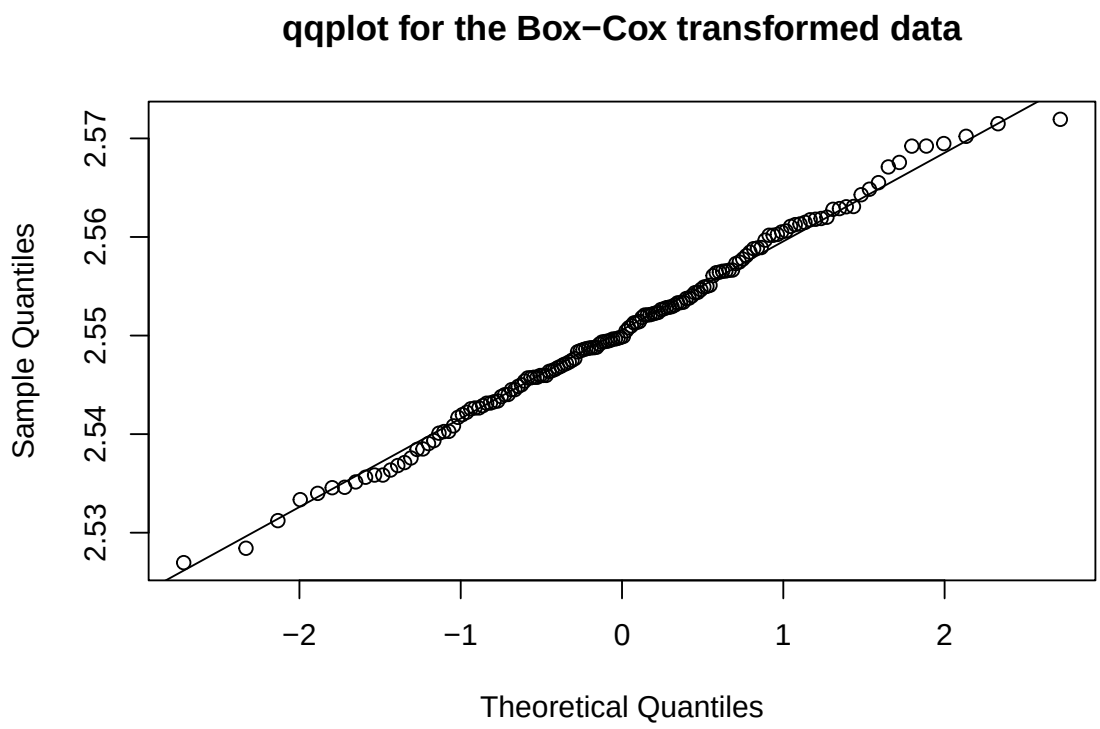
```
qqnorm(df$price, main = "qqplot for original data")
qqline(df$price)
```



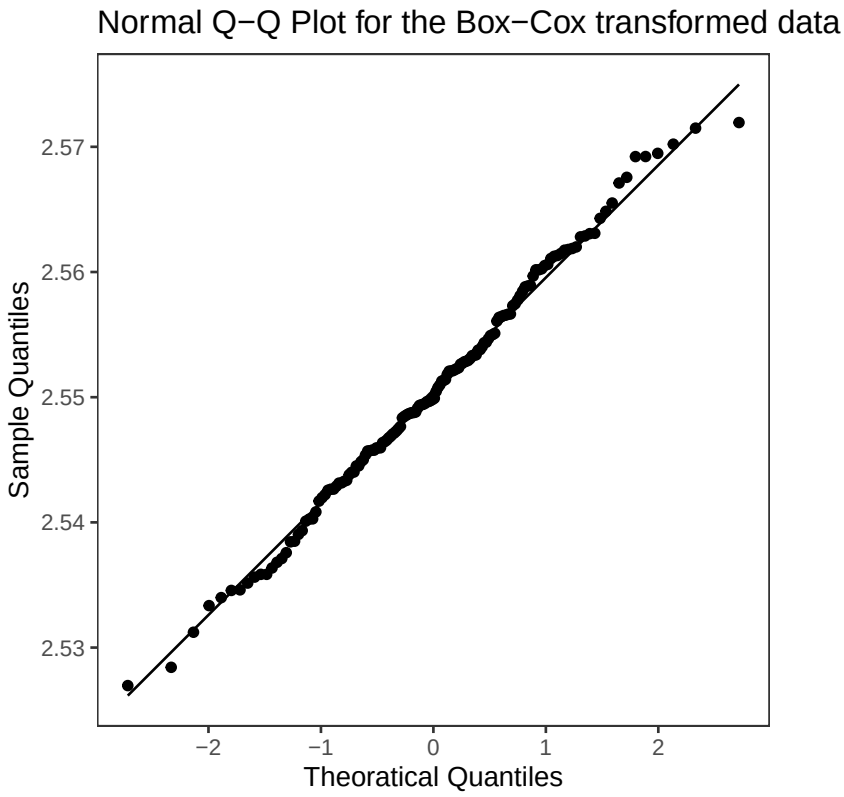

```
ggplot(df, aes(sample = price)) + geom_qq() + geom_qq_line() +
    labs(title = "Normal Q-Q Plot for original data",
        x = "Theoratical Quantiles", y = "Sample Quantiles") +
    theme_bw() + theme(panel.grid = element_blank(),
    aspect.ratio = 1)
```



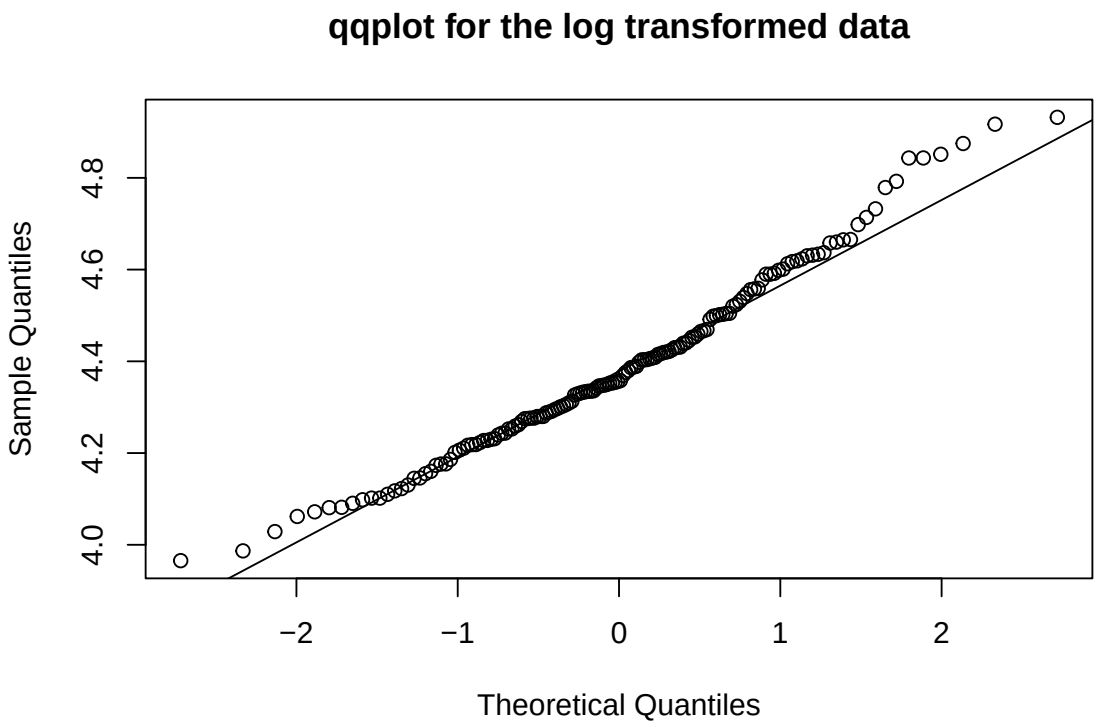

```
qqnorm(trans, main = "qqplot for the Box-Cox transformed data")
qqline(trans)
```



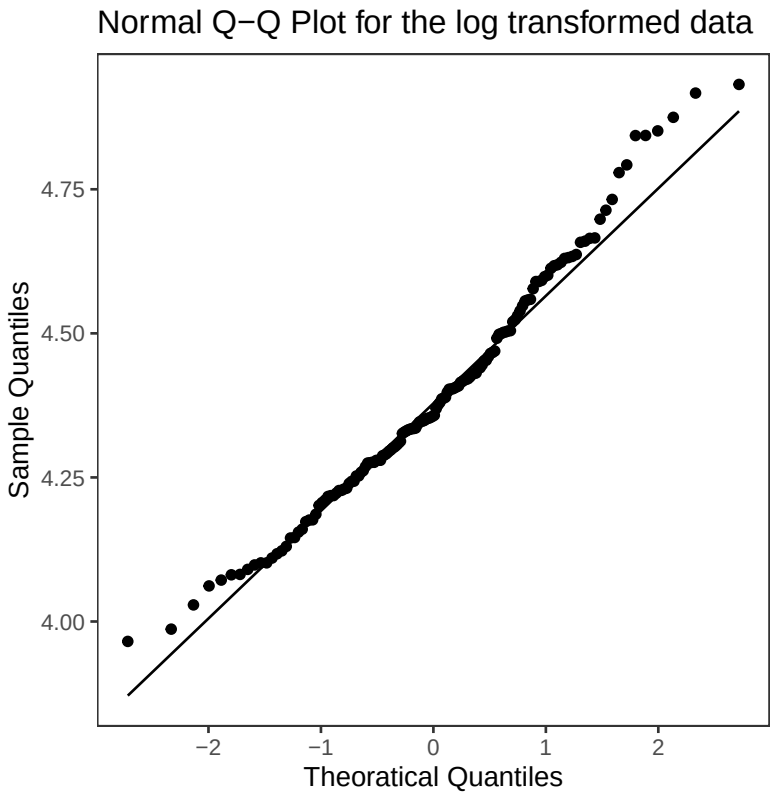

```
ggplot(df1, aes(sample = price)) + geom_qq() + geom_qq_line() +
    labs(title = "Normal Q-Q Plot for the Box-Cox transformed data",
        x = "Theoratical Quantiles", y = "Sample Quantiles") +
    theme_bw() + theme(panel.grid = element_blank(),
    aspect.ratio = 1)
```




```
qqnorm(trans2, main = "qqplot for the log transformed data")
qqline(trans2)
```




```
ggplot(df2, aes(sample = price)) + geom_qq() + geom_qq_line() +
    labs(title = "Normal Q-Q Plot for the log transformed data",
        x = "Theoratical Quantiles", y = "Sample Quantiles") +
    theme_bw() + theme(panel.grid = element_blank(),
    aspect.ratio = 1)
```




The Box-Cox transformed data appear to show distribution mostly close to normal distribution.

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:

```
### Create a matrix named 'mat' with 9 rows & 5
### columns
mat <- matrix(NA, nrow = 9, ncol = 5)

### Set row names and column names
rownames(mat) <- c("Original", "", "", "Box-Cox", "",
    "", "Log", "", "")
colnames(mat) <- c("x", "xbar-k*s", "xbar+k*s", "Theoretical %",
    "Actual %")

### Fill in known quantities
mat[, 1] <- c(1, 2, 3)
mat[, 4] <- c(68, 95, 99.7)
```

```r
### Fill in calculated values (I only give a
### preview of this and leave the remaining
### calculations for you).  I use 'orig' as the
### original data, 'bcdat' as the Box-Cox
### transformed data, and 'logdat' as the
### log-transformed data.  Name your variables
### anything you'd like.
orig <- df$price
for (i in c(1:3)) {
    mat[i, 2] <- mean(orig) - i * sd(orig)
    mat[i, 3] <- mean(orig) + i * sd(orig)
}

for (i in c(1:3)) {
    mat[i + 3, 2] <- mean(trans) - i * sd(trans)
    mat[i + 3, 3] <- mean(trans) + i * sd(trans)
}

for (i in c(1:3)) {
    mat[i + 6, 2] <- mean(trans2) - i * sd(trans2)
    mat[i + 6, 3] <- mean(trans2) + i * sd(trans2)
}

for (i in c(1:3)) {
    mat[i, 5] <- sum(orig >= mean(orig) - i * sd(orig) &
        orig <= mean(orig) + i * sd(orig))/length(orig) *
        100
}

for (i in c(1:3)) {
    mat[i + 3, 5] <- sum(trans >= mean(trans) - i *
        sd(trans) & trans <= mean(trans) + i * sd(trans))/length(trans) *
        100
}

for (i in c(1:3)) {
    mat[i + 6, 5] <- sum(trans2 >= mean(trans2) - i *
        sd(trans2) & trans2 <= mean(trans2) + i * sd(trans2))/length(trans2) *
        100
}


### Create a table
library(knitr)
```

```
kable(x = mat, digits = 2, row.names = T, format = "markdown")
```

| | x | xbar-k*s | xbar+k*s | Theoretical % | Actual % |
|---|---|---|---|---|---|
| Original | 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 | 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 | 1 | 4.19 | 4.59 | 68.0 | 66.45 |
| | 2 | 3.99 | 4.79 | 95.0 | 94.08 |
| | 3 | 3.79 | 4.98 | 99.7 | 100.00 |

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) A possible reason is that more people prefer relatively cheaper prices, so the distribution shape is right skewed.
2) A lot of statistical inference methods are based on the assumption that data show normal distribution, so I believe that transforming data can provide us more flexibility to utilize those methods.

Short Answers:

- About how long did this assignment take you? Did you feel it was too long, too short, or reasonable?

It took me about 4 hours, but that was because I registered the class late, so I had to go over the slides from the beginning. Also, I spent some time on figuring out why some of the plots did not show up (a markdown problem, not quite relevant with class). I think this assignment took me reasonable time.

- Who, if anyone, did you work with on this assignment?

Myself.

- What questions do you have relating to any of the material we have covered so far in class?

I am wondering how to calculate the theoretical quantiles.