# ECON103 Project 1

2024-10-29

## Contents



# 1 Question 1: Introduction of Dataset

The dataset contains information on various cars, including features such as price, car length, car width, curb weight, engine size, and more. It includes both numerical and categorical variables, allowing for analysis of the factors that influence car pricing. For this project, we are estimating the Ordinary Least Squares (OLS) model to estimate the relationship between variables that affect student performance. Source of the data: https://www.kaggle.com/datasets/goyalshalini93/car-data

The dataset provides a comprehensive list of variables; however, we have selected the following four for our analysis: - Car Length: A variable that reflects the vehicle's overall size and potential luxury or functionality, which may correlate with car price. - Car Width: This variable is considered as it often suggests greater stability and space, potentially impacting the vehicle's market value. - Curb Weight: The weight of a car, including all fluids and standard equipment, but excluding passengers or cargo. It is essential for understanding a vehicle's performance. Heavier cars may have a perceived higher value, but the relationship with price is to be explored through regression analysis. - Engine Size: A significant variable that may be associated with vehicle performance and fuel consumption, both of which could affect the car's market value. - Price: The dependent variable in the analysis, representing the market value of the car, which we aim to predict based on the selected independent variables.

```
# Setup and load packages
options(repos = c(CRAN = "https://cloud.r-project.org/"))  # Set the CRAN mirror
if (!requireNamespace("pastecs", quietly = TRUE)) {
    install.packages("pastecs")
}
library(pastecs)

# Importing Data
data <- read.csv("~saumyamahajan/Downloads/CarPrice_Assignment.csv")
#install.packages("pastecs")
#library(pastecs)
```

# 2 Question 2: Statistical Analysis of Variables

## 2.1 Statistical Summaries

```
stat.desc(data$carlength)
```

```
##      nbr.val     nbr.null       nbr.na          min          max        range
## 2.050000e+02 0.000000e+00 0.000000e+00 1.411000e+02 2.081000e+02 6.700000e+01
##          sum       median         mean      SE.mean CI.mean.0.95          var
## 3.568010e+04 1.732000e+02 1.740493e+02 8.616736e-01 1.698928e+00 1.522087e+02
##      std.dev     coef.var
## 1.233729e+01 7.088389e-02
```

Ranges from 141.1 to 208.1, with a mean of 174.0 and moderate variability (SD = 12.3), suggesting a fairly symmetric distribution.

```
stat.desc(data$carwidth)
```

```
##      nbr.val     nbr.null       nbr.na          min          max        range
## 2.050000e+02 0.000000e+00 0.000000e+00 6.030000e+01 7.230000e+01 1.200000e+01
##          sum       median         mean      SE.mean CI.mean.0.95          var
## 1.351110e+04 6.550000e+01 6.590780e+01 1.498275e-01 2.954091e-01 4.601900e+00
##      std.dev     coef.var
## 2.145204e+00 3.254856e-02
```

Ranges from 60.3 to 72.3, with a mean of 65.9 and low variability (SD = 2.1), indicating consistent widths among cars.

```
stat.desc(data$enginesize)
```

```
##      nbr.val     nbr.null       nbr.na          min          max        range
## 2.050000e+02 0.000000e+00 0.000000e+00 6.100000e+01 3.260000e+02 2.650000e+02
##          sum       median         mean      SE.mean CI.mean.0.95          var
```

```
## 2.601600e+04 1.200000e+02 1.269073e+02 2.908452e+00 5.734481e+00 1.734114e+03
##      std.dev     coef.var
## 4.164269e+01 3.281347e-01
```

Ranges from 61 to 326, with a mean of 126.9 and considerable variability (SD = 41.6), highlighting a right-skewed distribution.

```
stat.desc(data$curbweight)
```

```
##      nbr.val     nbr.null       nbr.na          min          max        range
## 2.050000e+02 0.000000e+00 0.000000e+00 1.488000e+03 4.066000e+03 2.578000e+03
##          sum       median         mean      SE.mean CI.mean.0.95          var
## 5.238910e+05 2.414000e+03 2.555566e+03 3.636588e+01 7.170119e+01 2.711079e+05
##      std.dev     coef.var
## 5.206802e+02 2.037436e-01
```

Ranges from 1,488 to 4,066, with a mean of 2,555.6 and high variability (SD = 520.7), reflecting a diverse range of vehicle types.

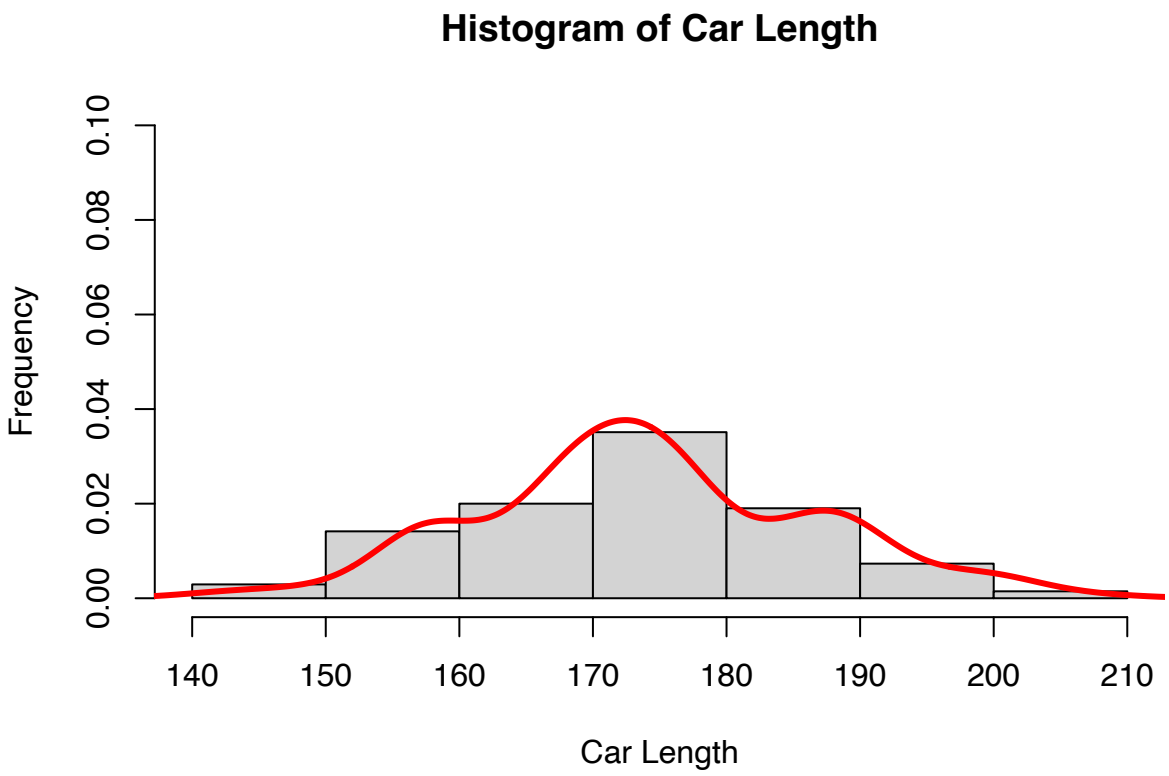
```
stat.desc(data$price)
```

```
##      nbr.val     nbr.null       nbr.na          min          max        range
## 2.050000e+02 0.000000e+00 0.000000e+00 5.118000e+03 4.540000e+04 4.028200e+04
##          sum       median         mean      SE.mean CI.mean.0.95          var
## 2.721726e+06 1.029500e+04 1.327671e+04 5.579656e+02 1.100119e+03 6.382176e+07
##      std.dev     coef.var
## 7.988852e+03 6.017193e-01
```

Ranges from $5,118 to $45,400, with a mean of approximately $13,277 and a median of $10,295.

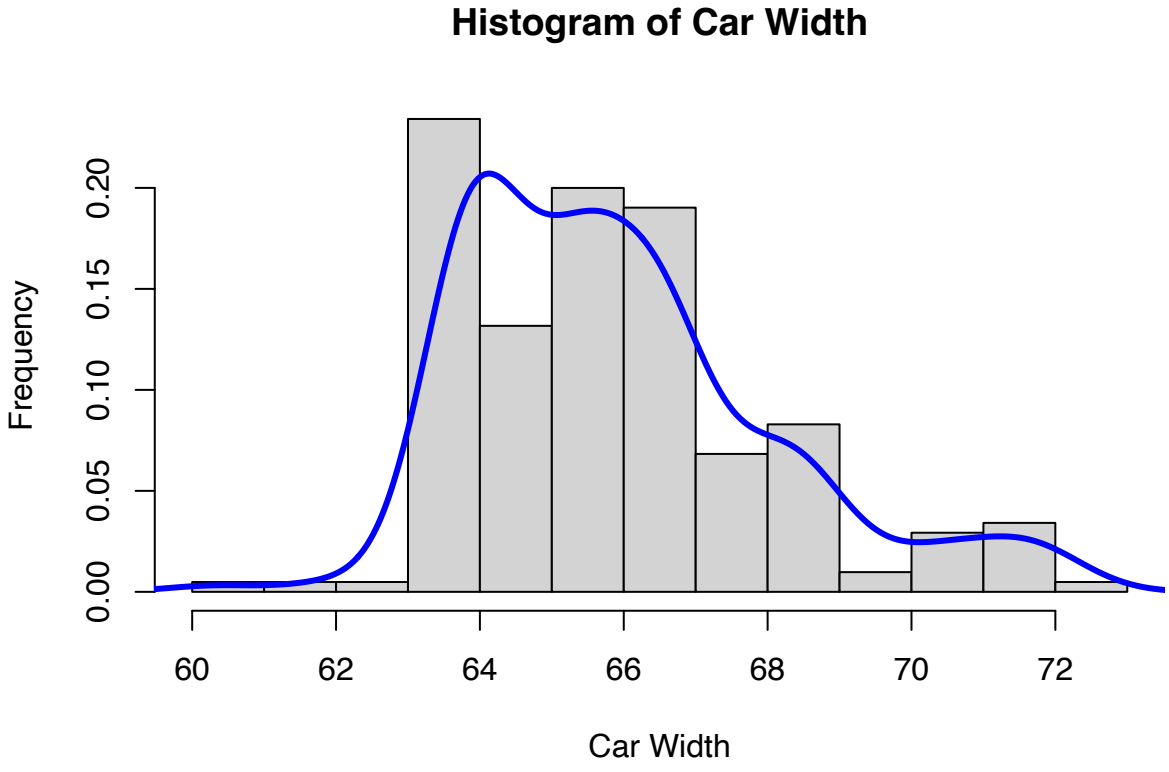
## 2.2 Histograms and Fitted Distributions

```
hist(data$carlength, main = "Histogram of Car Length", xlab = "Car Length", ylim = c(0,.1), ylab = "Fre
lines (density(data$carlength), col = "red", lwd = "3")
```



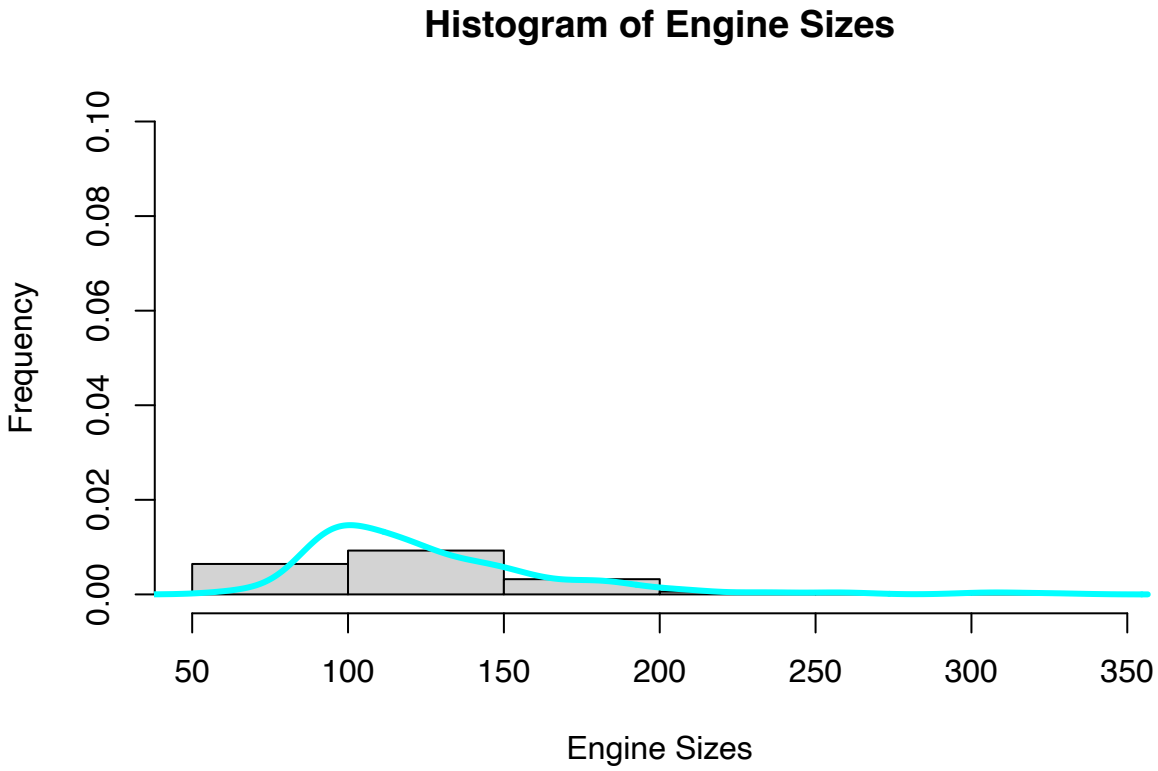

The histogram of car length shows a roughly normal distribution, with most cars concentrated around a central length.

```
hist(data$carwidth, main = "Histogram of Car Width", xlab = "Car Width", ylab = "Frequency", freq = FALS
lines (density(data$carwidth), col = "blue", lwd = "3")
```



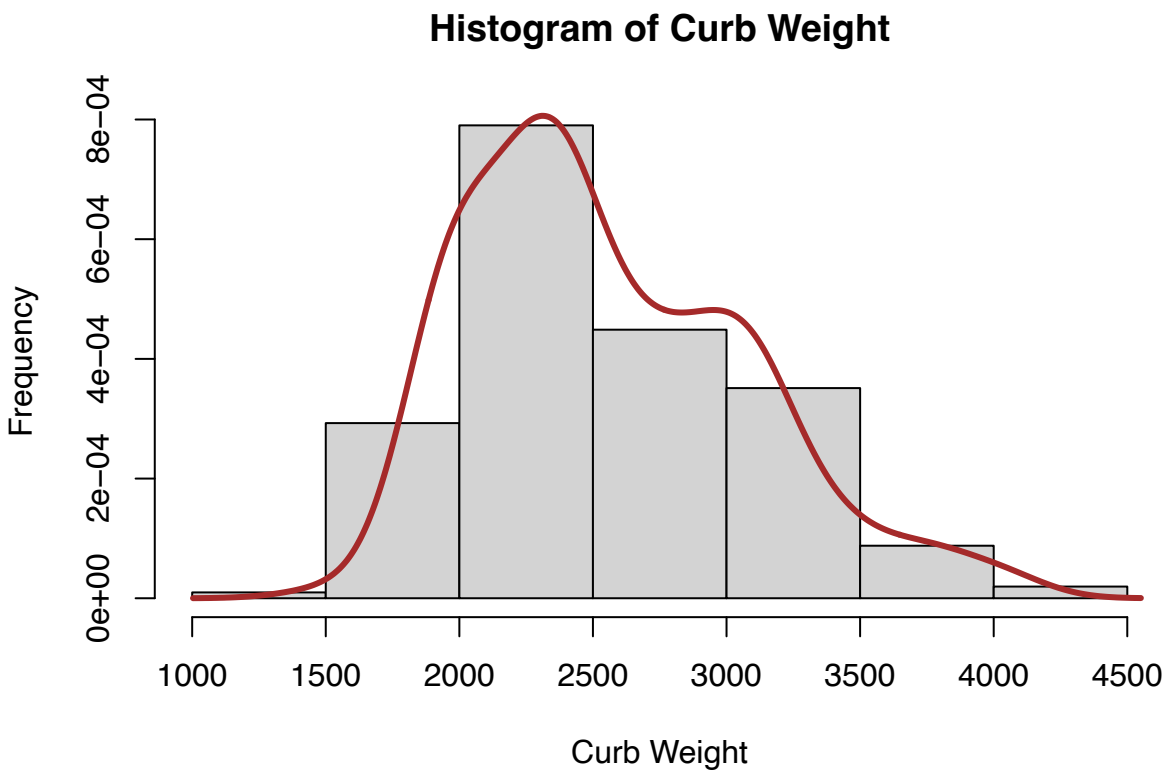
The histogram for car width displays a bimodal distribution, indicating the presence of two distinct groups of vehicles, possibly compact and larger models.

```
hist(data$enginesize, main = "Histogram of Engine Sizes", xlab = "Engine Sizes", ylim = c(0,.1), ylab =
lines (density(data$enginesize), col = "cyan", lwd = "3")
```



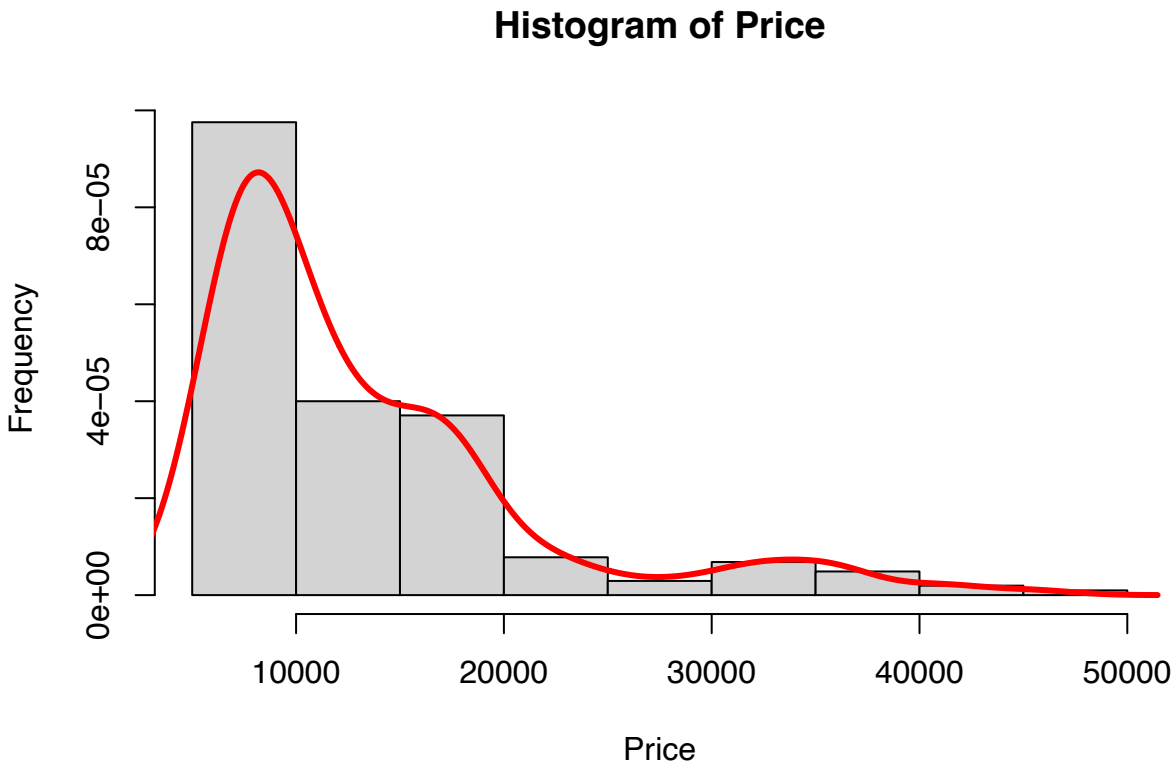

The histogram of engine sizes shows a right-skewed distribution, with a majority of cars having smaller engine sizes.

```
hist(data$curbweight, main = "Histogram of Curb Weight", xlab = "Curb Weight", ylab = "Frequency", freq
lines (density(data$curbweight), col = "brown", lwd = "3")
```




The curb weight histogram reveals a fairly uniform distribution, with a slight skew towards heavier cars.

```
hist(data$price, main = "Histogram of Price", xlab = "Price", ylab = "Frequency", freq = FALSE)
lines (density(data$price), col = "red", lwd = "3")
```




The histogram of car prices depicts a right-skewed distribution, with most prices clustering toward the lower

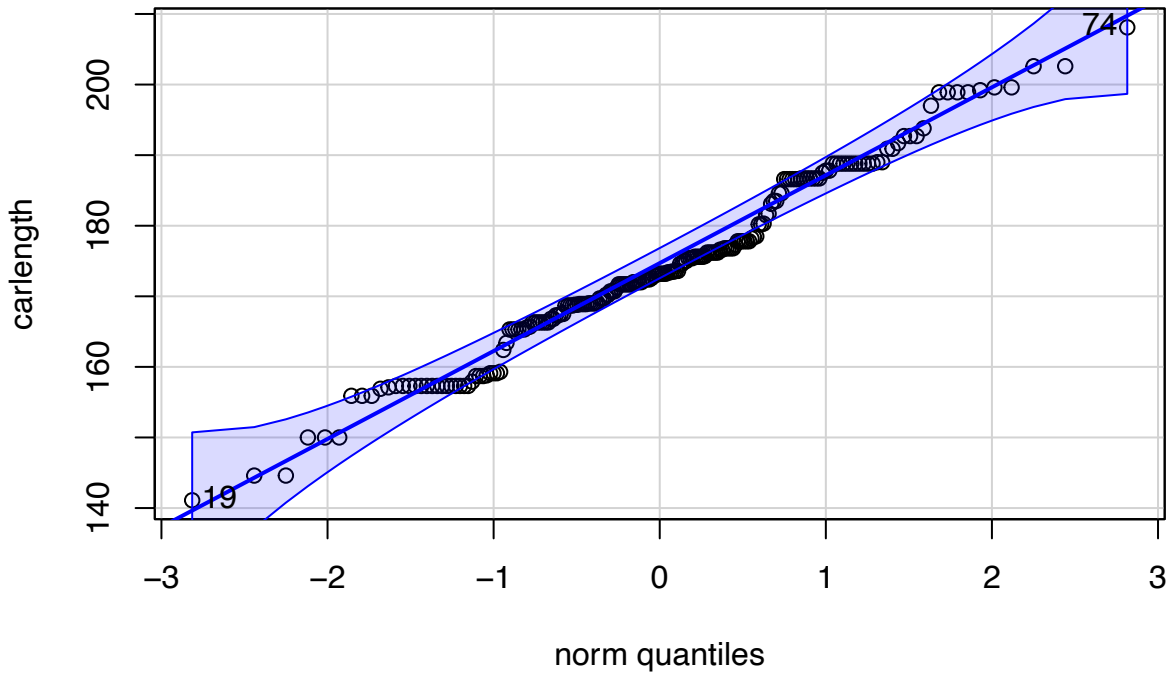
end, while a few high-priced outliers exist.

## 2.3 Quantile Plots

```
library(car)
```

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

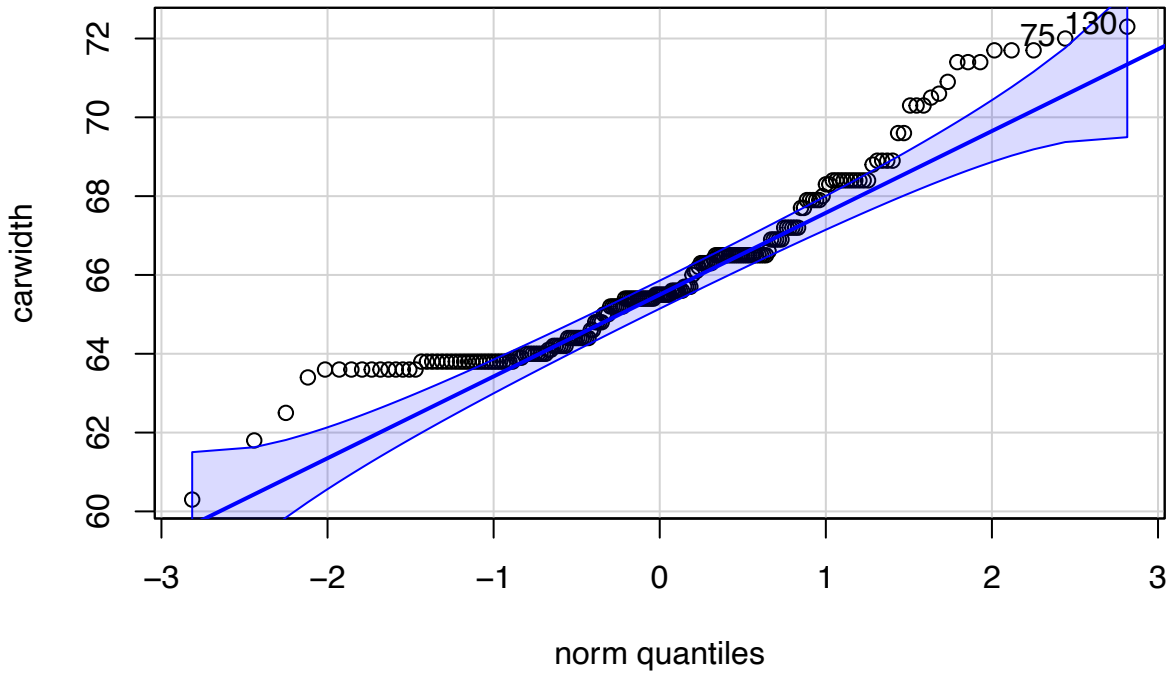
```
attach(data)
qqPlot(carlength)
```




```
## [1] 74 19
```

The Q-Q plot for car length indicates that the data closely follows a normal distribution, with most points aligning along the diagonal line.

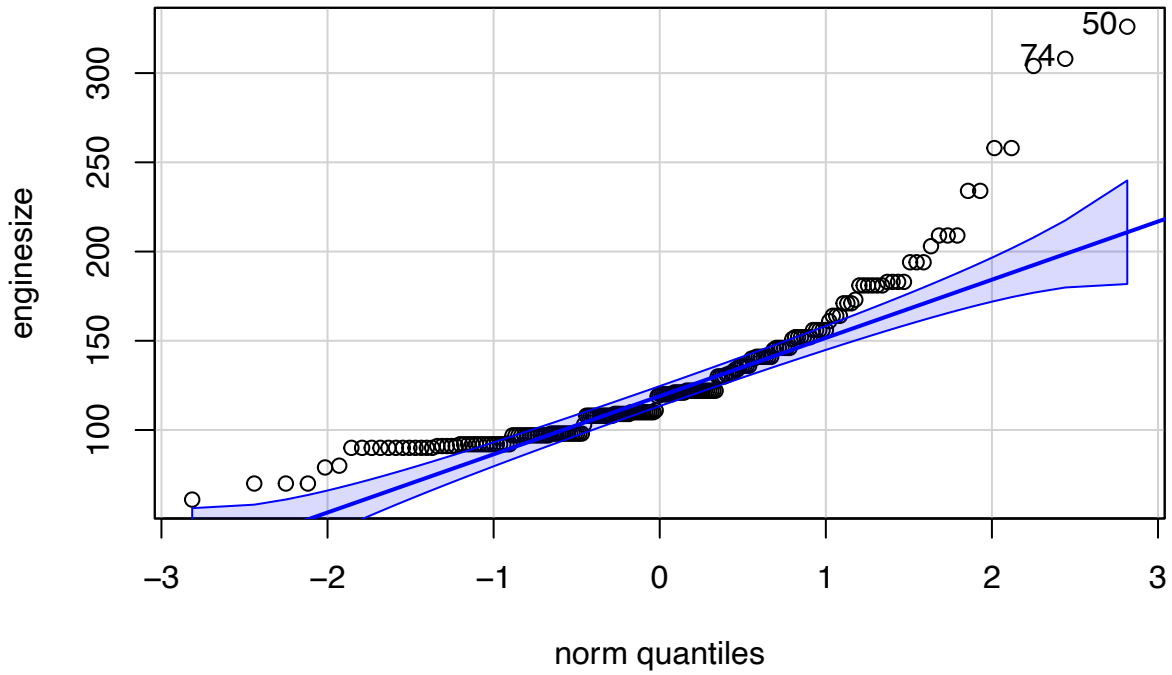
```
qqPlot(carwidth)
```




```
## [1] 130  75
```

The Q-Q plot for car width also shows a strong alignment with the diagonal, suggesting that this variable is normally distributed, with only minor deviations in the tails.

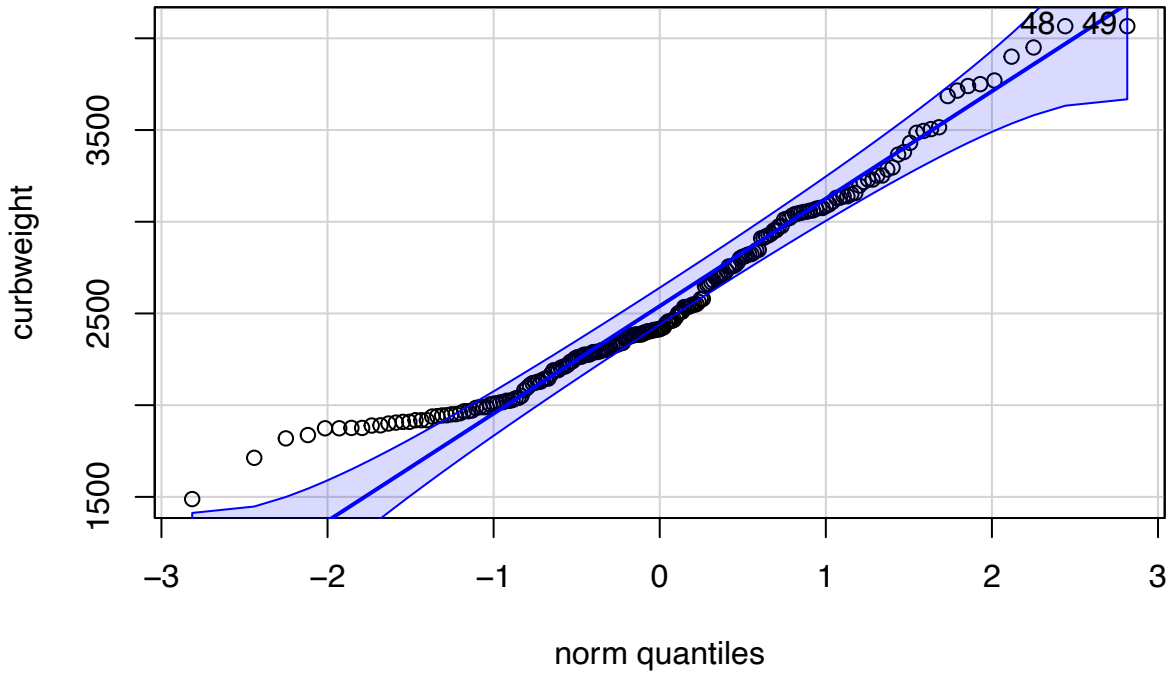
```
qqPlot(enginesize)
```




```
## [1] 50 74
```

The Q-Q plot for engine size reveals that the data does not conform to a normal distribution, as points deviate upward significantly at the higher values, indicating a concentration of smaller engine sizes.

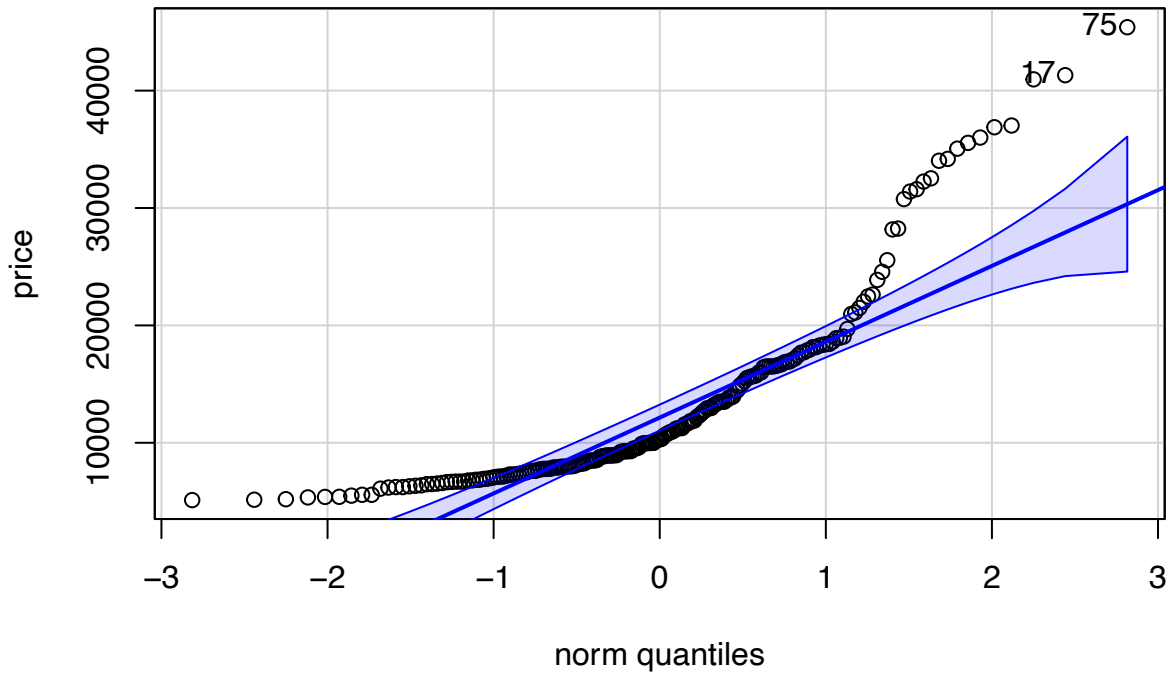
```
qqPlot(curbweight)
```




```
## [1] 48 49
```

The Q-Q plot for curb weight shows substantial divergence from the diagonal, especially in the upper region, indicating a distribution that is less normal and possibly influenced by a few heavier vehicles.

```
qqPlot(price)
```




```
## [1] 75 17
```

The Q-Q plot for price shows divergence from the diagonal, both in the lower and upper regions, indicating

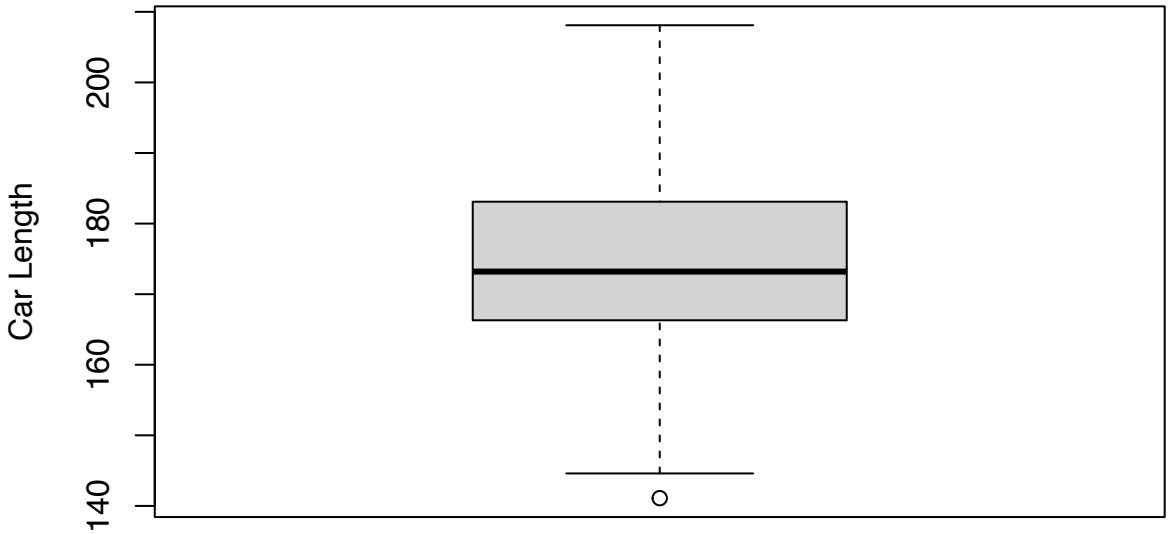
deviation from the normal distribution.

## 2.4 Boxplots: Displays median, quartiles, and potential outliers

```
attach(data)
```

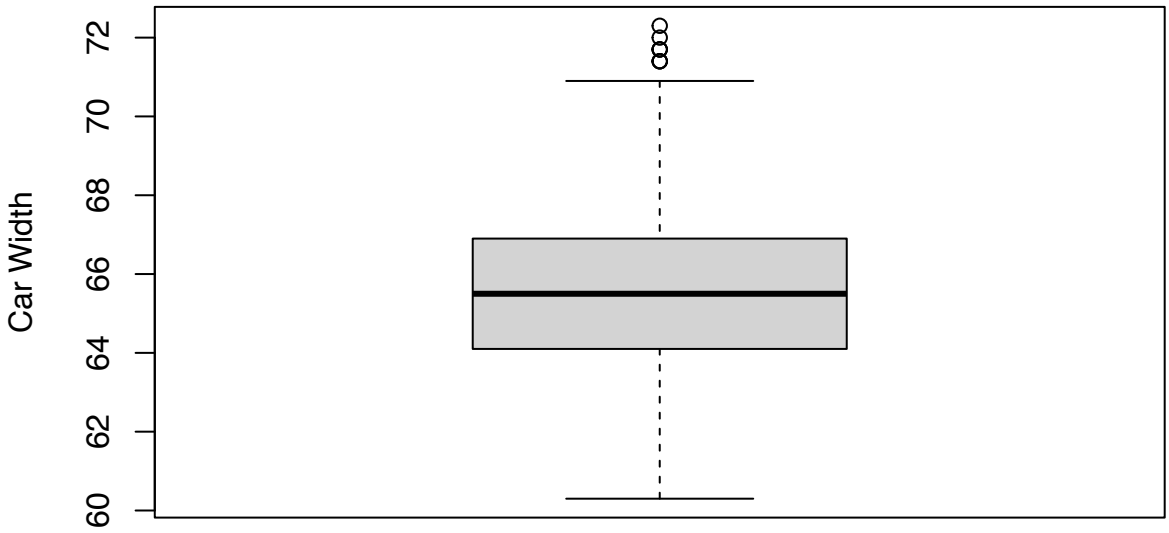
```
## The following objects are masked from data (pos = 3):
##
##     aspiration, boreratio, car_ID, carbody, carheight, carlength,
##     CarName, carwidth, citympg, compressionratio, curbweight,
##     cylindernumber, doornumber, drivewheel, enginelocation, enginesize,
##     enginetype, fuelsystem, fueltype, highwaympg, horsepower, peakrpm,
##     price, stroke, symboling, wheelbase
```

```
boxplot(carlength, ylab = "Car Length")
```



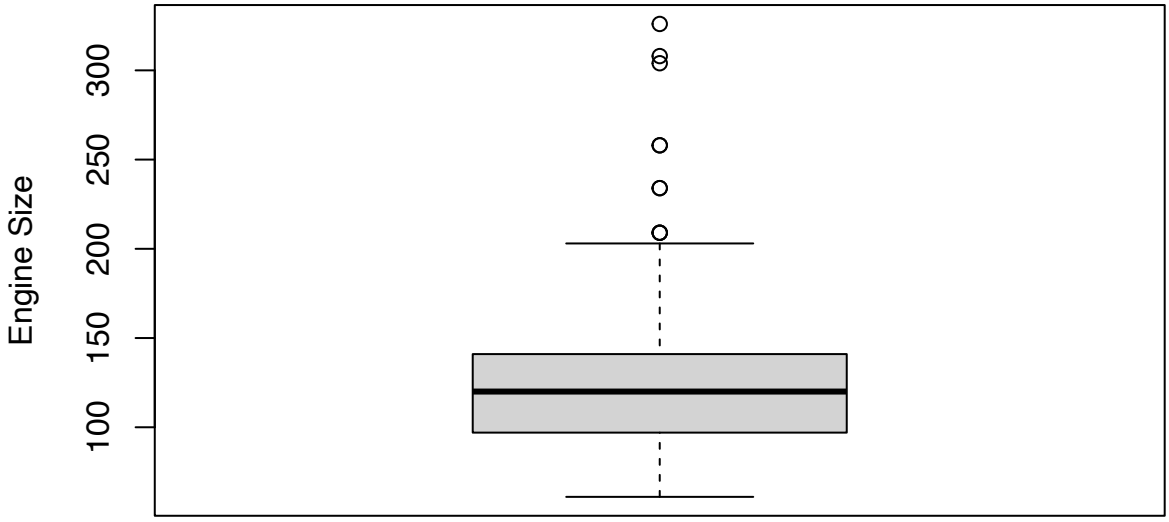

From this boxplot, it can be noticed that the median is around 173, with the first quartile as 168 and the second quartile as 185. There seem to be one outlier at 140.

```
boxplot(carwidth, ylab = "Car Width")
```



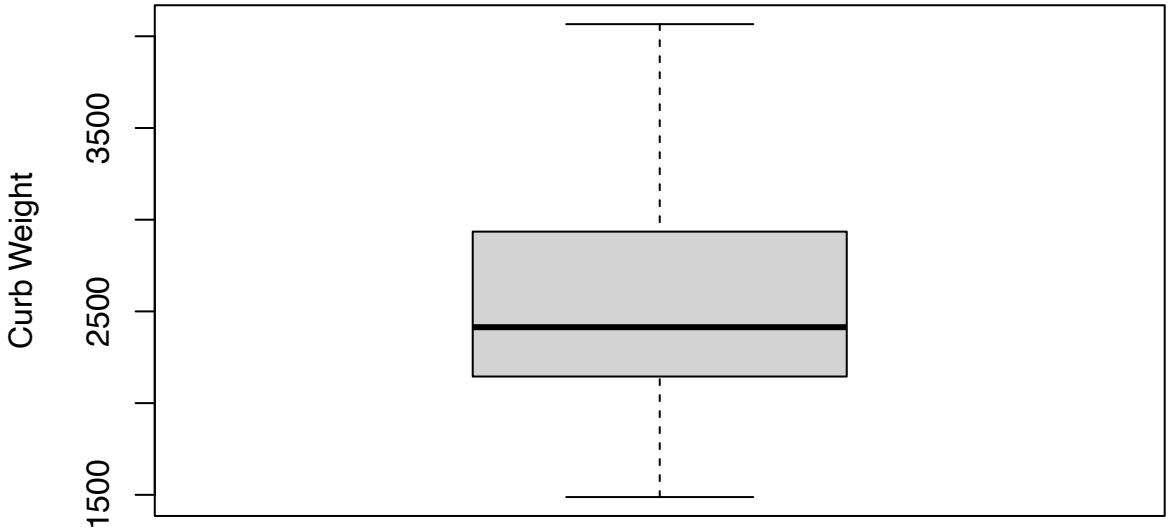
From this boxplot, it can be noticed that the median is around 66, with the first quartile as 64 and the second quartile as 67. There seem to be 4 outliers around 72.

```
boxplot(enginesize, ylab = "Engine Size")
```



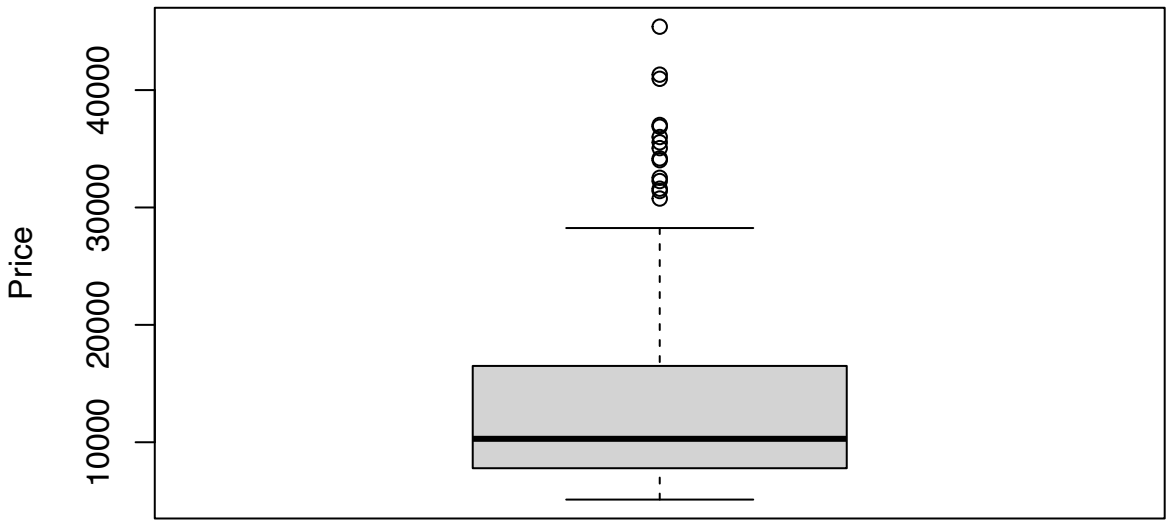

From this boxplot, it can be noticed that the median is around 150, with the first quartile as 100 and the second quartile as 140. There seem to be six outliers at approximately 200, 225, 250, 300, 300, and 350.

```
boxplot(curbweight, ylab = "Curb Weight")
```




From this boxplot, it can be noticed that the median is around 2500, with the first quartile as 2100 and the second quartile as 3000. There is no outlier.

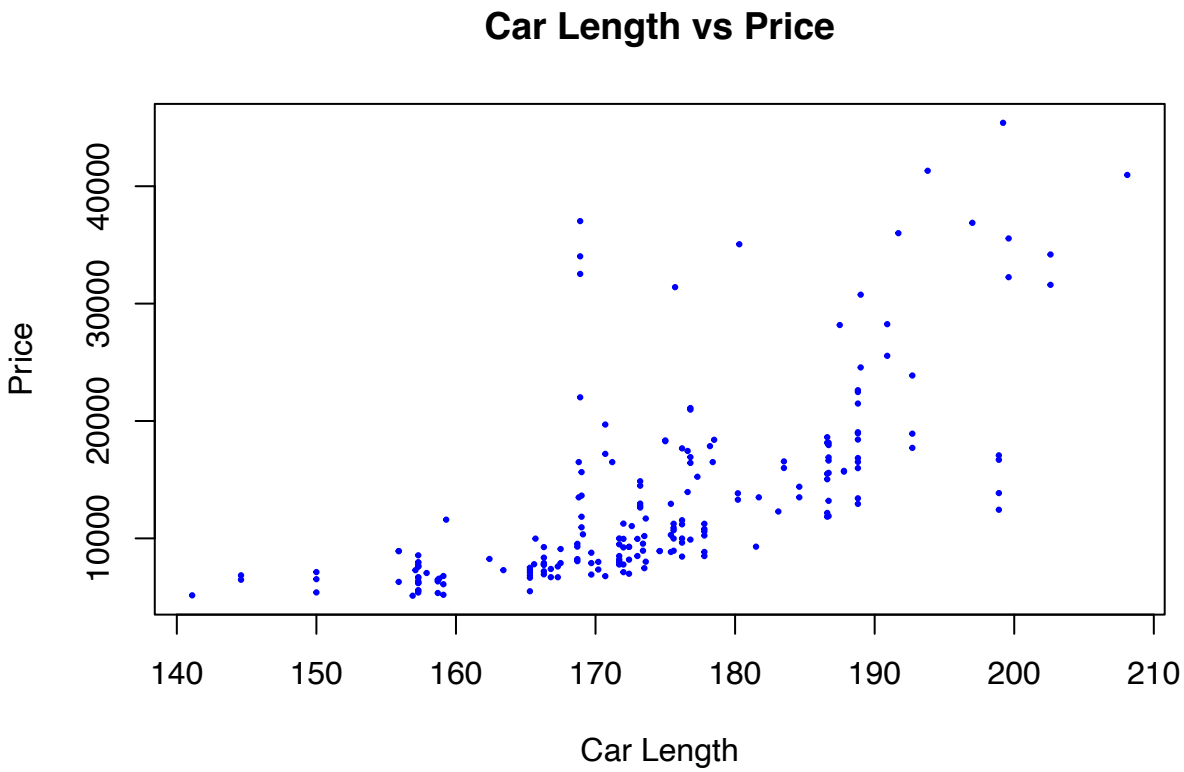
```
boxplot(price, ylab = "Price")
```




From this boxplot, it is apparent that the median is around 10000, with the first quartile at 2000 and the second quartile at 28000. There are several outliers ranging from approximately 30000 to 50000.

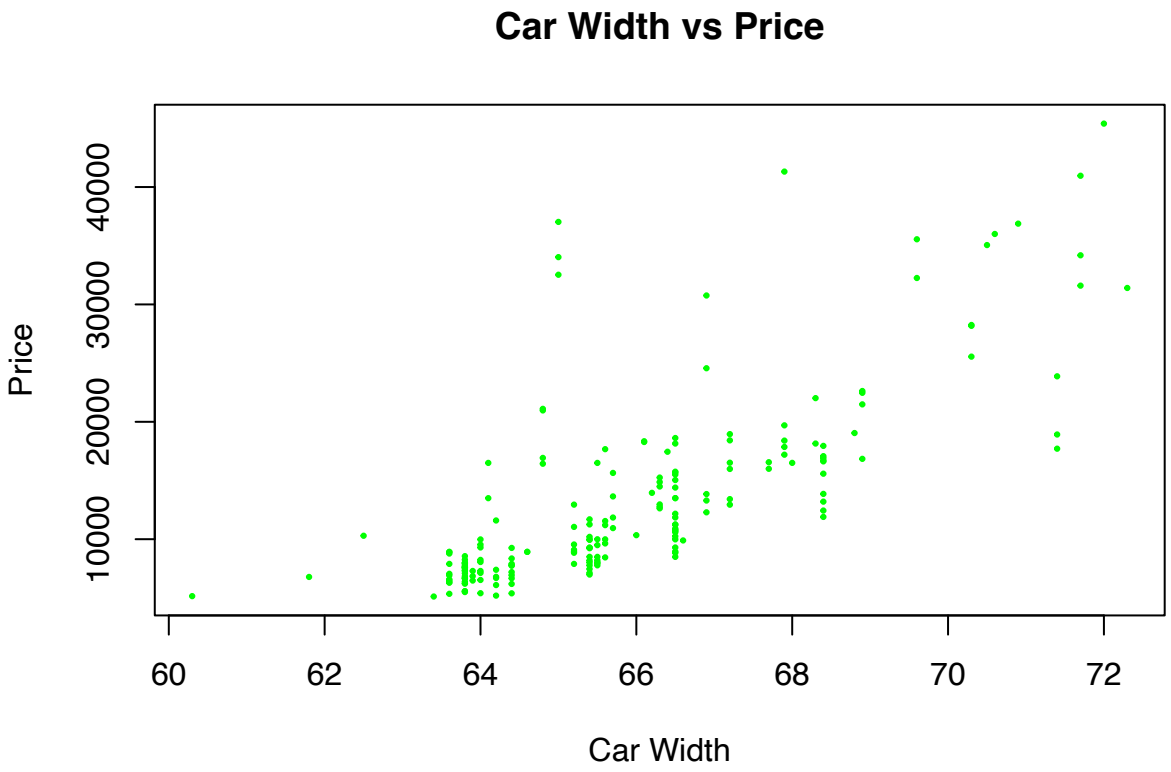
## 3 Question 3: Scatterplots

```
# Create scatter plots
plot(data$carlength, data$price,
     main = "Car Length vs Price",
     xlab = "Car Length", ylab = "Price", pch = 19, col = "blue", cex = 0.3)
```



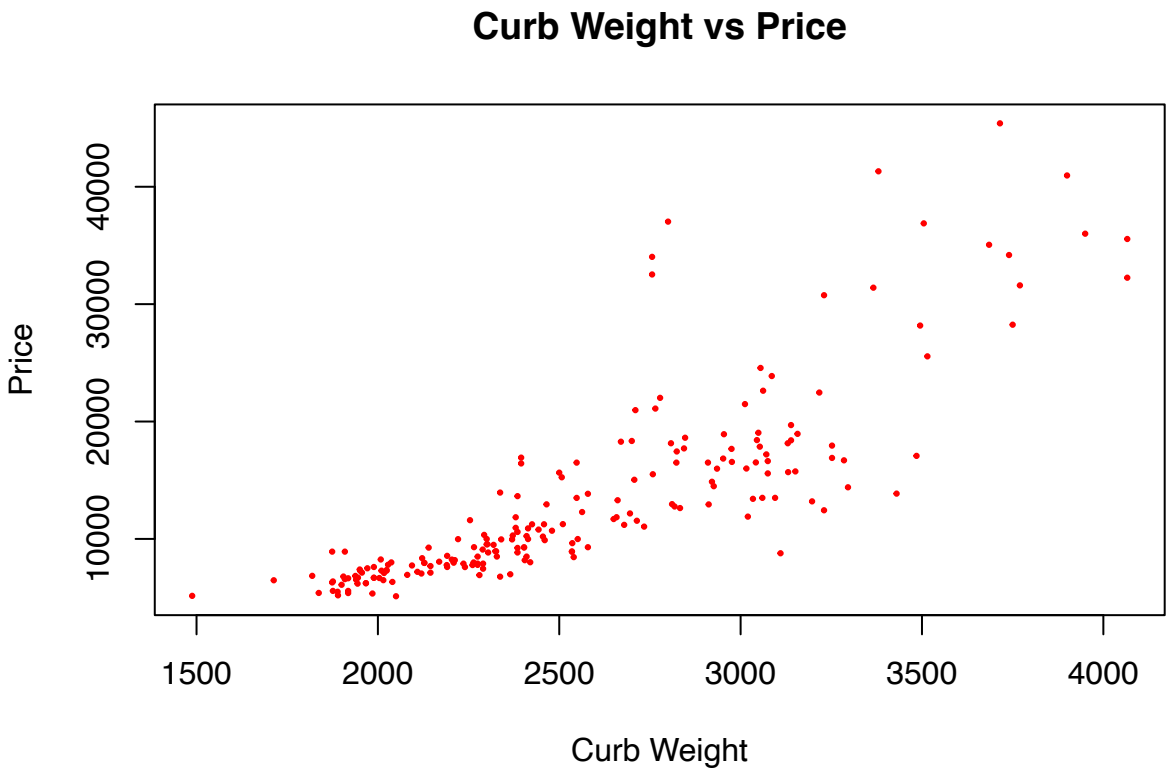

This scatterplot shows an upward trend, indicating a positive relationship between car length and price. It also seems to be non-linear (exponential), and might need a transformation to linearity.

```r
plot(data$carwidth, data$price,
     main = "Car Width vs Price",
     xlab = "Car Width", ylab = "Price", pch = 19, col = "green", cex = 0.3)
```



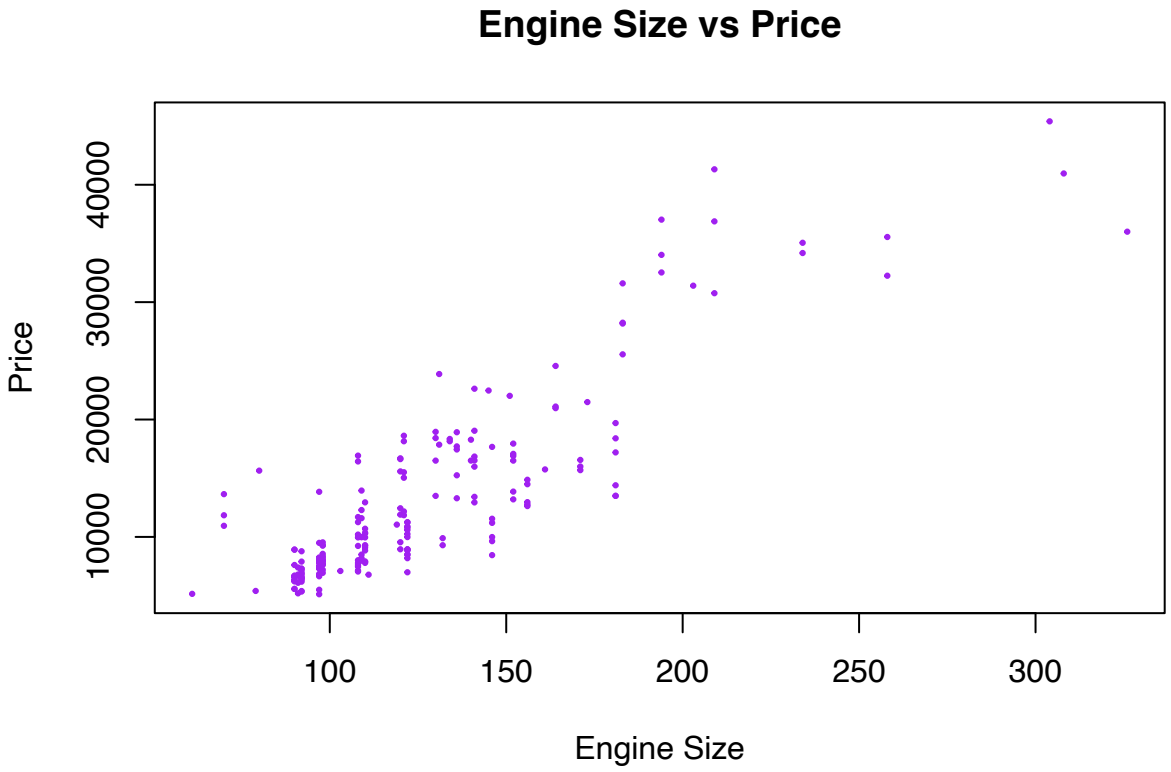

This scatterplot shows an upward trend, indicating a positive relationship between car width and price. It seems to be linear, however, the variance does not seem to be constant indicating that there might be a need for a transformation.

```r
plot(data$curbweight, data$price,
     main = "Curb Weight vs Price",
     xlab = "Curb Weight", ylab = "Price", pch = 19, col = "red", cex = 0.3)
```




This scatterplot shows an upward trend, indicating a positive relationship between curb width and price. It also seems to be non-linear (exponential), and might need a transformation to linearity.

```
plot(data$enginesize, data$price,
     main = "Engine Size vs Price",
     xlab = "Engine Size", ylab = "Price", pch = 19, col = "purple", cex = 0.3)
```




This scatterplot shows an upward trend, indicating a positive relationship between engine size and price. It also seems to be non-linear, and might need a transformation to linearity.

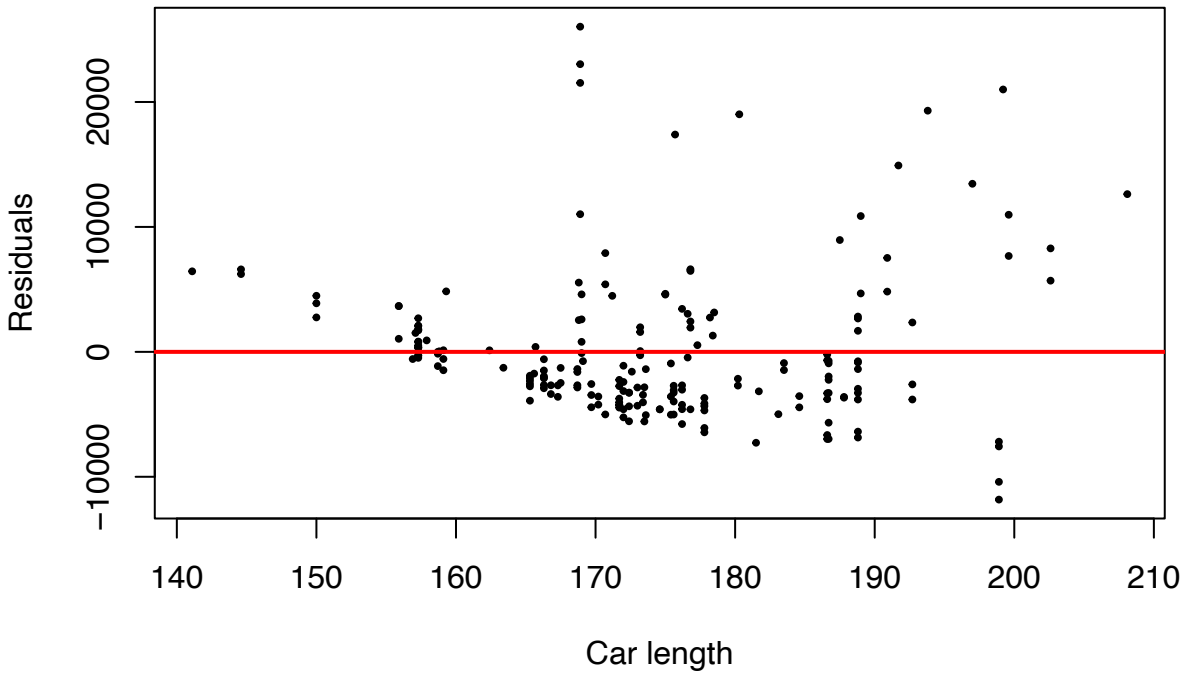
# 4 Question 4: Transformations

## 4.1 For each variable, let's explore the visualizations to determine which, if any, transformation we should apply.

## 4.2 Variable 1: Price ~ Car Length

### 4.2.1 Step 1: Residuals plot on untransformed data:

```
ols.mod1 <- lm(data$price~data$carlength, data=data)
plot(data$carlength, ols.mod1$residuals, pch=20, ylab="Residuals", xlab="Car length", cex=0.6)
abline(h=0,col="red", lwd=2)
```



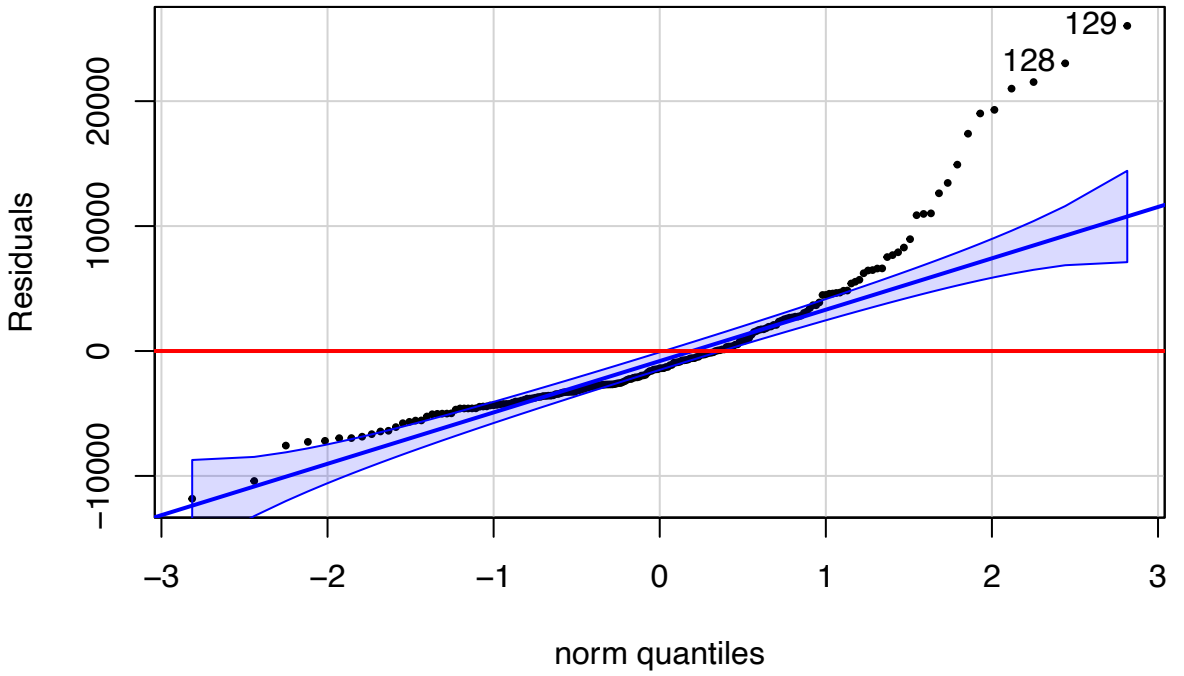

From this residuals plot, we can see that there is a slight increase in variation as car length increases, indicating that it might not fit the homoscedasticity assumption of linear regression. There is no apparent pattern to the residuals that is noticeable.

#### 4.2.2 Step 2: QQ Plot of residuals on untransformed data:

```
qqPlot(ols.mod1$residuals, pch=20, ylab="Residuals", cex=0.6)
```

```
## [1] 129 128
```

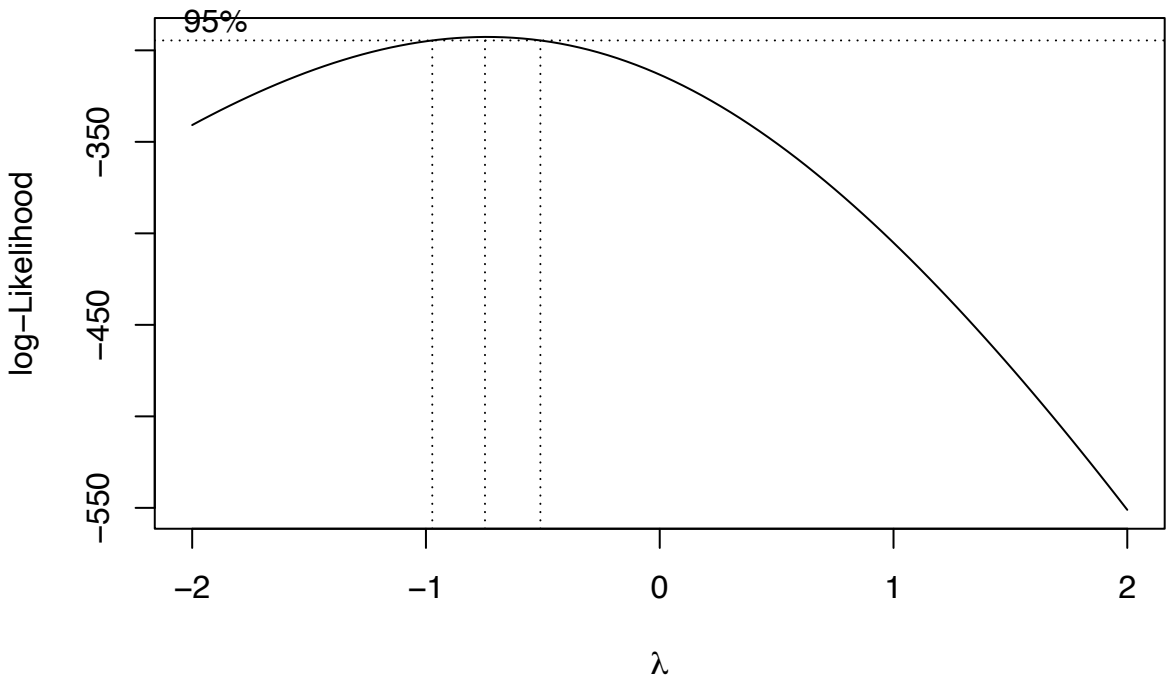
```
abline(h=0,col="red", lwd=2)
```



From this plot, it can be seen that the points deviate from the diagonal line in the upper right region quite drastically. Therefore, this is further evidence for a possible need for a transformation as it indicates that the data is not normally distributed.

#### 4.2.3 Step 3: Box-Cox

```
library(MASS)
boxcox(lm(data$price~data$carlength, data = data), lambda = seq(-2, 2, by = 0.1))
```




Box-cox indicates lambda of approximately -1,but we got economic result that does not make sense since an increase in the length of the car should not predict a decrease in the price when the original data has the opposite relationship (positive correlation).

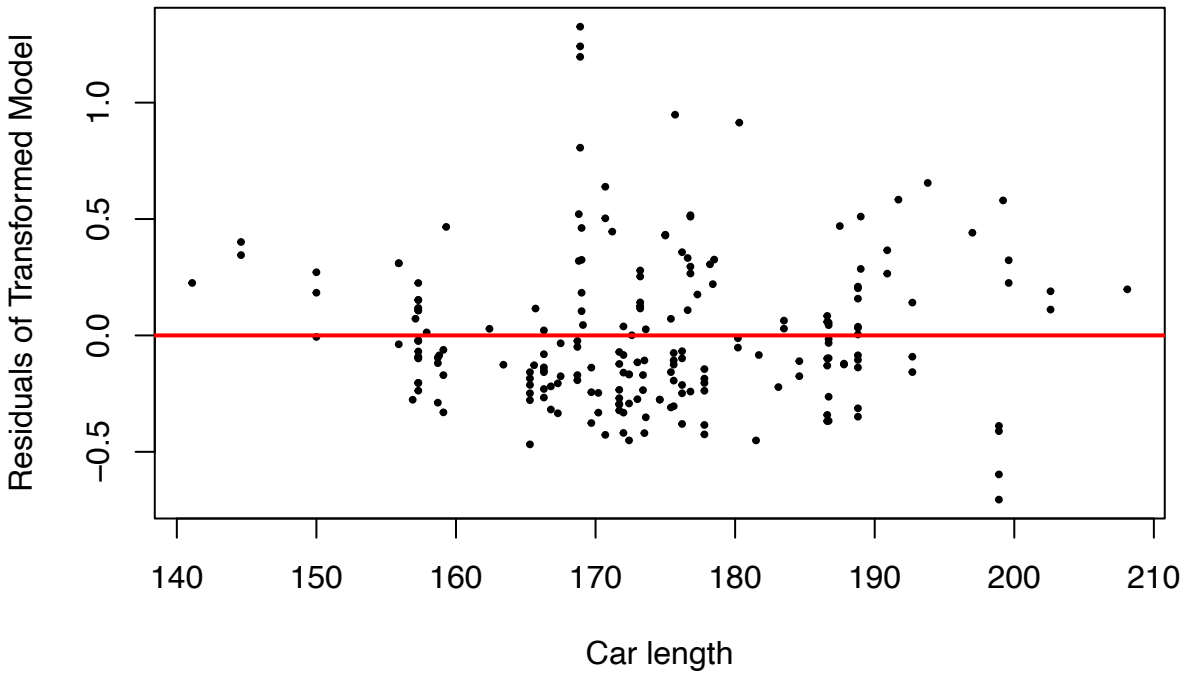
From these analyses and the histogram of price, which is heavily skewed to the right, a log transformation might be needed.

#### 4.2.4 Step 4: Exploring & Applying Transformations

Residuals plot of log-linear transformed data

```
data$log_price <- log(data$price)

ols.mod2 <- lm(data$log_price~data$carlength, data=data)
plot(data$carlength, ols.mod2$residuals, pch=20, ylab="Residuals of Transformed Model", xlab="Car lengt
abline(h=0,col="red", lwd=2)
```

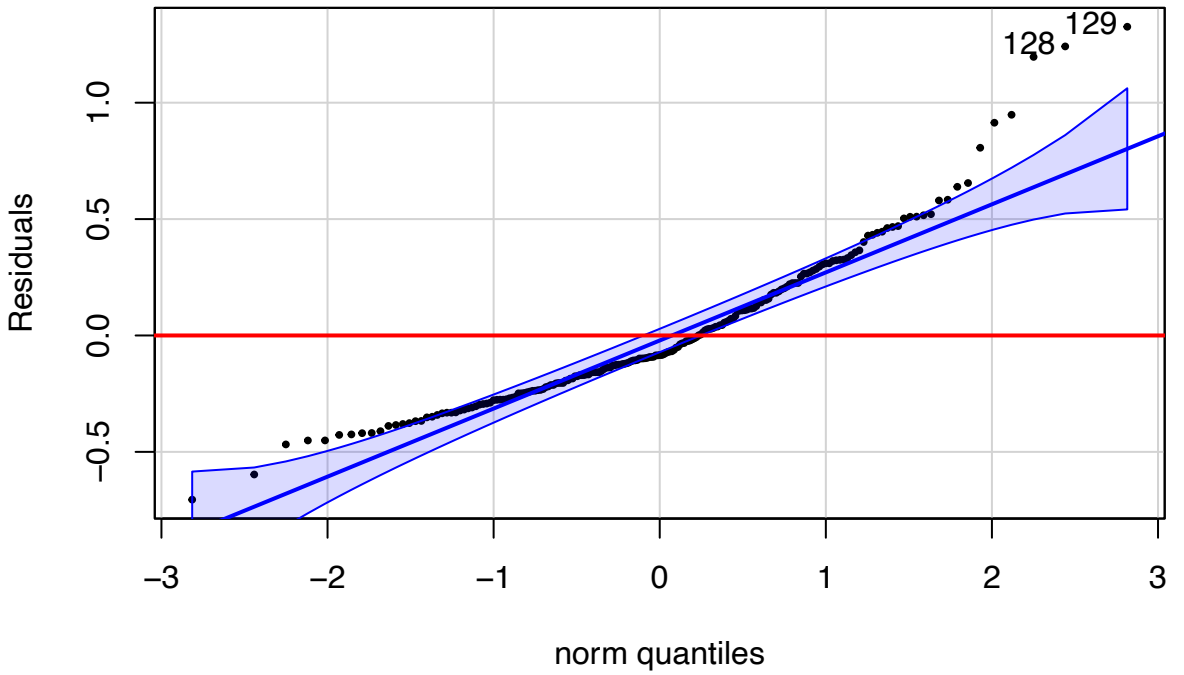



This seems to be more uniformly distributed than the untransformed data, and variance is relatively constant, which supports homoscedasticity. Additionally, no pattern is apparent.

```
qqPlot(ols.mod2$residuals, pch=20, ylab="Residuals", cex=0.6)
```

```
## [1] 129 128
```

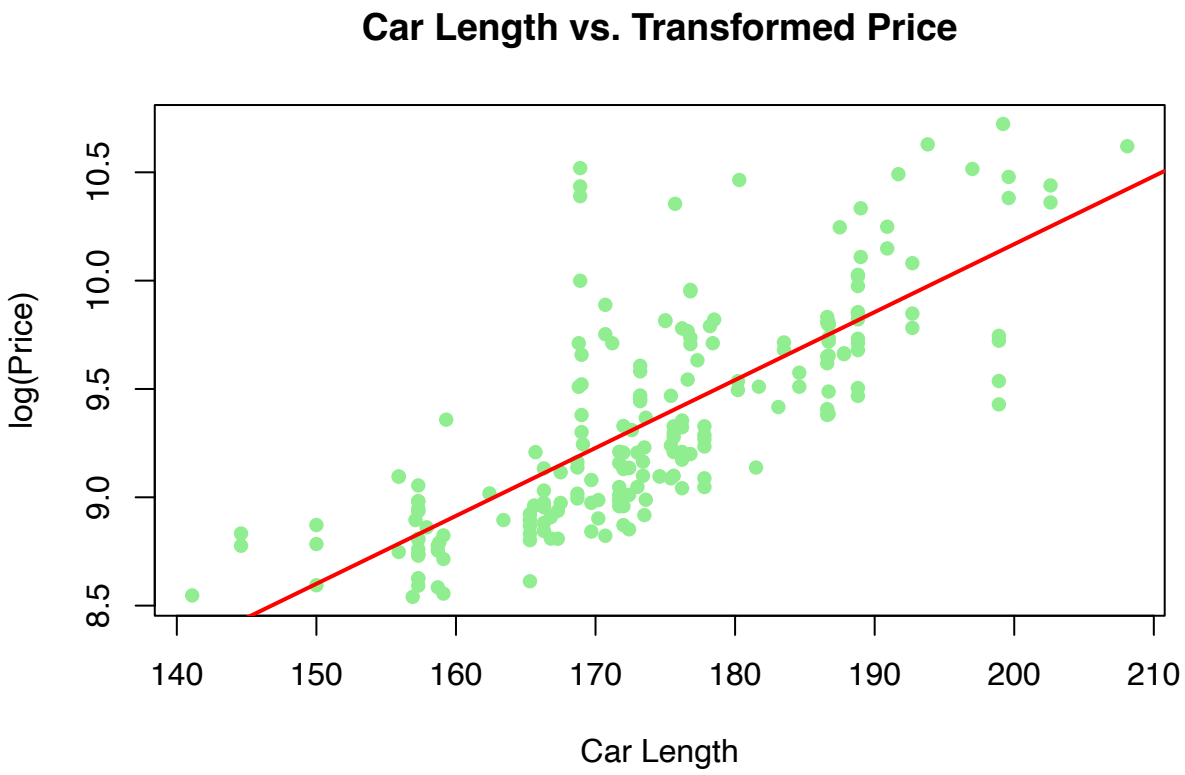
```
abline(h=0,col="red", lwd=2)
```




With the transformed data, the QQ plot of residuals seems to indicate a better fit for linear regression. This is because, excluding the outliers, the points are lying closer to the diagonal.

```r
plot(data$carlength, data$log_price,
     main = "Car Length vs. Transformed Price",
     xlab = "Car Length",
     ylab = "log(Price)",
     pch = 16, col = "lightgreen")

abline(ols.mod2, col="red", lwd=2)
```




From this scatterplot, we can see that the data points seem to show a linear relationship after the transformation of price.

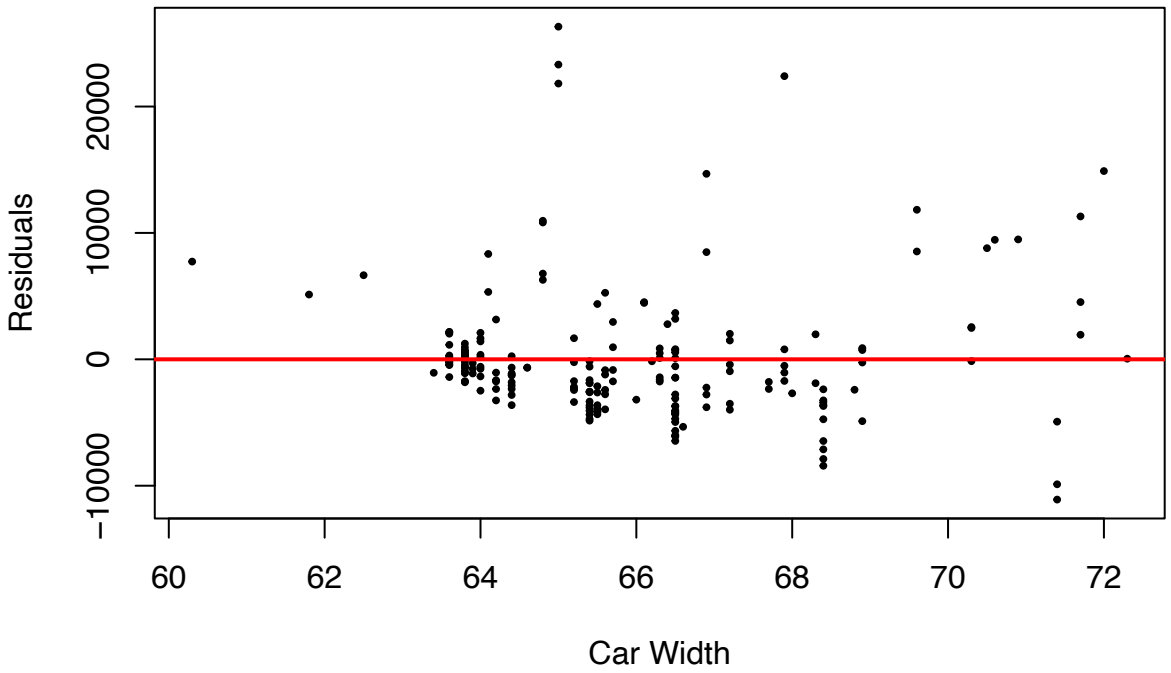
#### 4.2.5 Step 5: Final Transformation

A log-linear model provides percentage interpretation for coefficients, which is particularly intuitive in economics. While the Box-Cox method suggested a transformation closer to lambda = negative 1, which would imply an inverse transformation, the model becomes difficult to interpret or produces implausible results. Since $R^2$ indicates the proportion of variance explained by the model, it provides a measure of how well the log-linear model fits the data.

### 4.3 Variable 2: Price ~ Car Width

#### 4.3.1 Step 1: Residuals plot on untransformed data:

```r
ols.mod3 <- lm(data$price~data$carwidth, data=data)
plot(data$carwidth, ols.mod3$residuals, pch=20, ylab="Residuals", xlab="Car Width", cex=0.6)
abline(h=0,col="red", lwd=2)
```



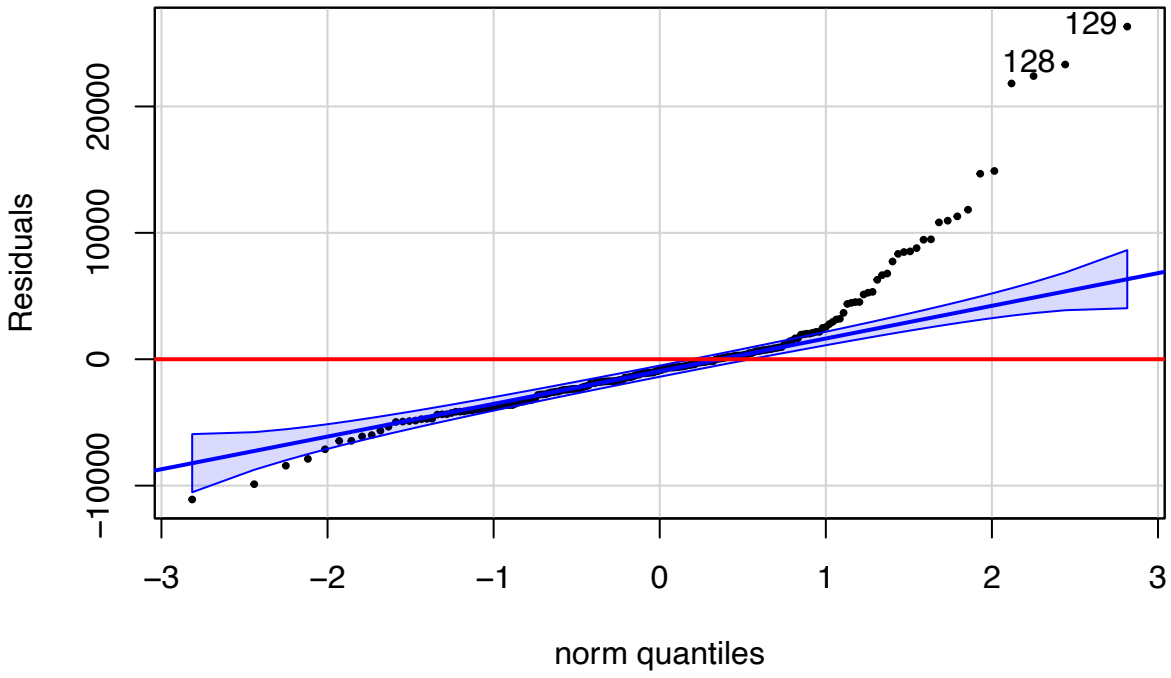

The residuals plot seems to fit our necessary assumptions, but we can do further visualizations to determine transformations, especially as price is a skewed variable.

#### 4.3.2 Step 2: QQ Plot of residuals on untransformed data:

```
qqPlot(ols.mod3$residuals, pch=20, ylab="Residuals", cex=0.6)
```

```
## [1] 129 128
```

```
abline(h=0,col="red", lwd=2)
```

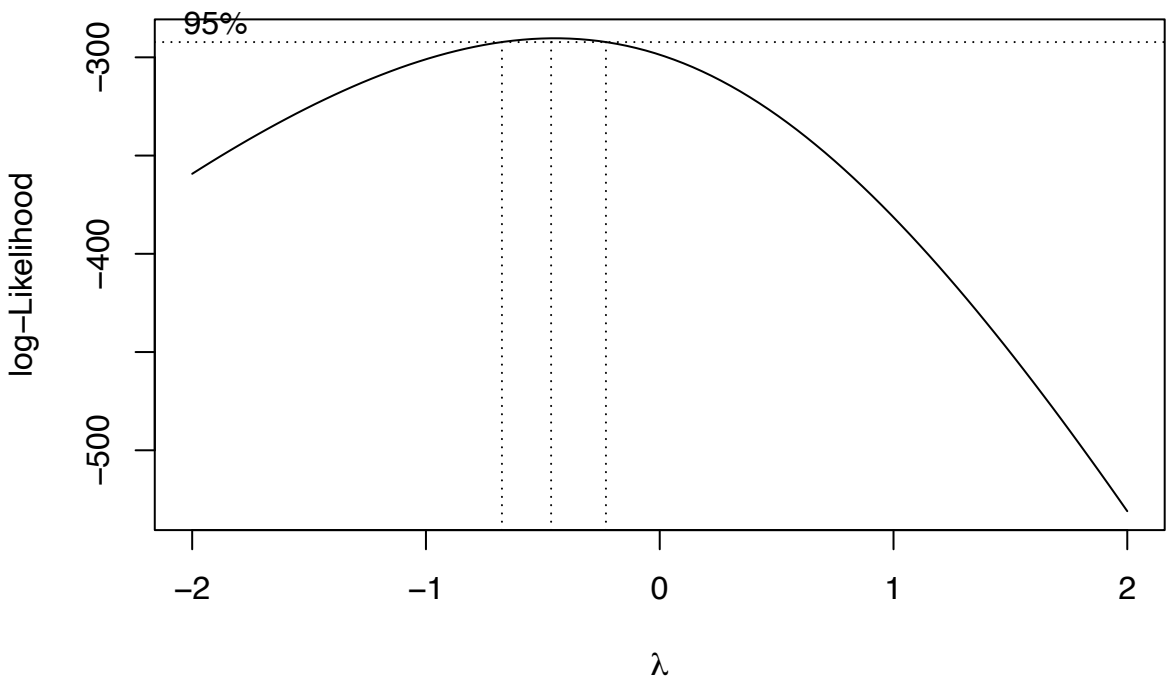



From the plot, we can see that there is a large deviation from the diagonal line in the upper region, suggesting

a transformation might be needed.

#### 4.3.3 Step 3: Box Cox

```
boxcox(lm(data$price~data$carwidth, data = data), lambda = seq(-2, 2, by = 0.1))
```

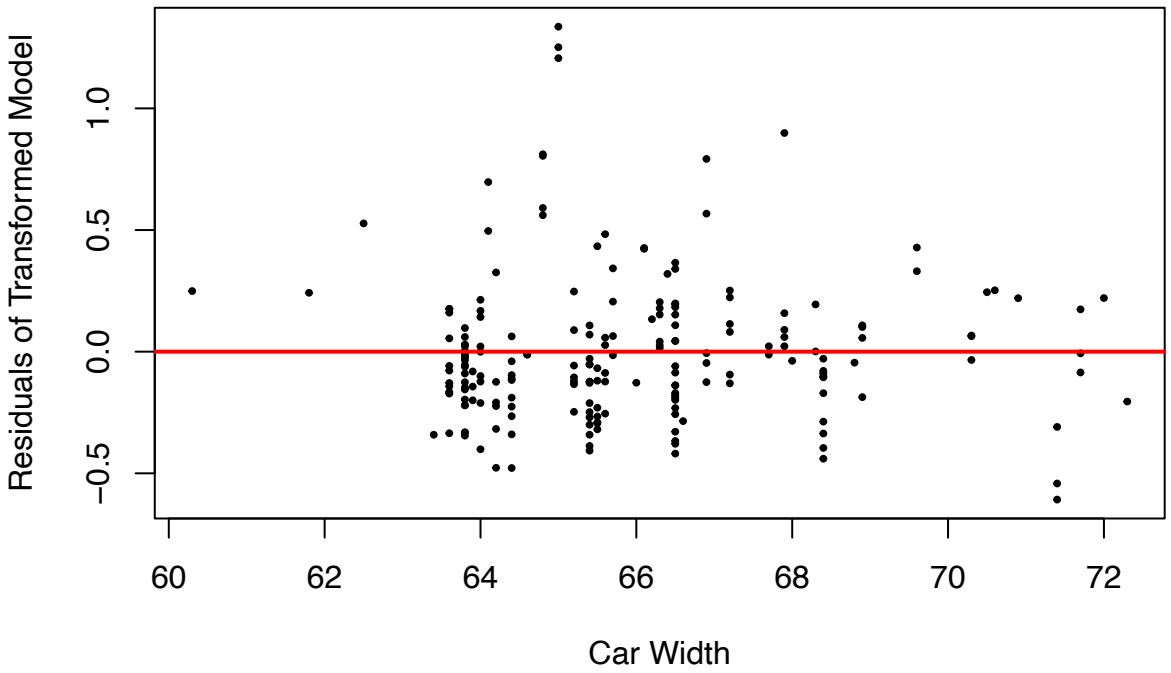



Box-cox indicates lambda of approximately -1,but we got economic result that does not make sense since an increase in the width of the car should not predict a decrease in the price when the original data has the opposite relationship (positive correlation).

From these analyses and the histogram of price, which is heavily skewed to the right, a log transformation might be needed.

#### 4.3.4 Step 4: Applying Transformations

```
ols.mod4 <- lm(data$log_price~data$carwidth, data=data)
plot(data$carwidth, ols.mod4$residuals, pch=20, ylab="Residuals of Transformed Model", xlab="Car Width"
abline(h=0,col="red", lwd=2)
```

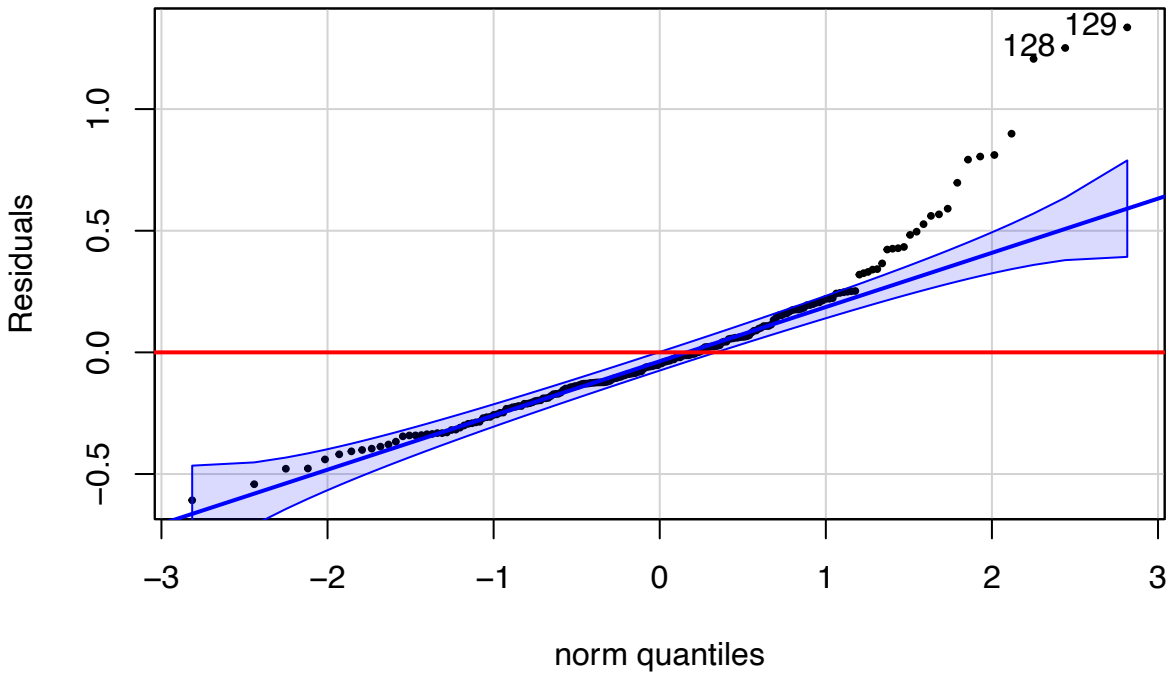



Residuals plot seems to be similar to the original but regardless, price is a variable that needs to be transformed due to skewness observed in histogram.

```
qqPlot(ols.mod4$residuals, pch=20, ylab="Residuals", cex=0.6)
```

```
## [1] 129 128
```

```
abline(h=0,col="red", lwd=2)
```

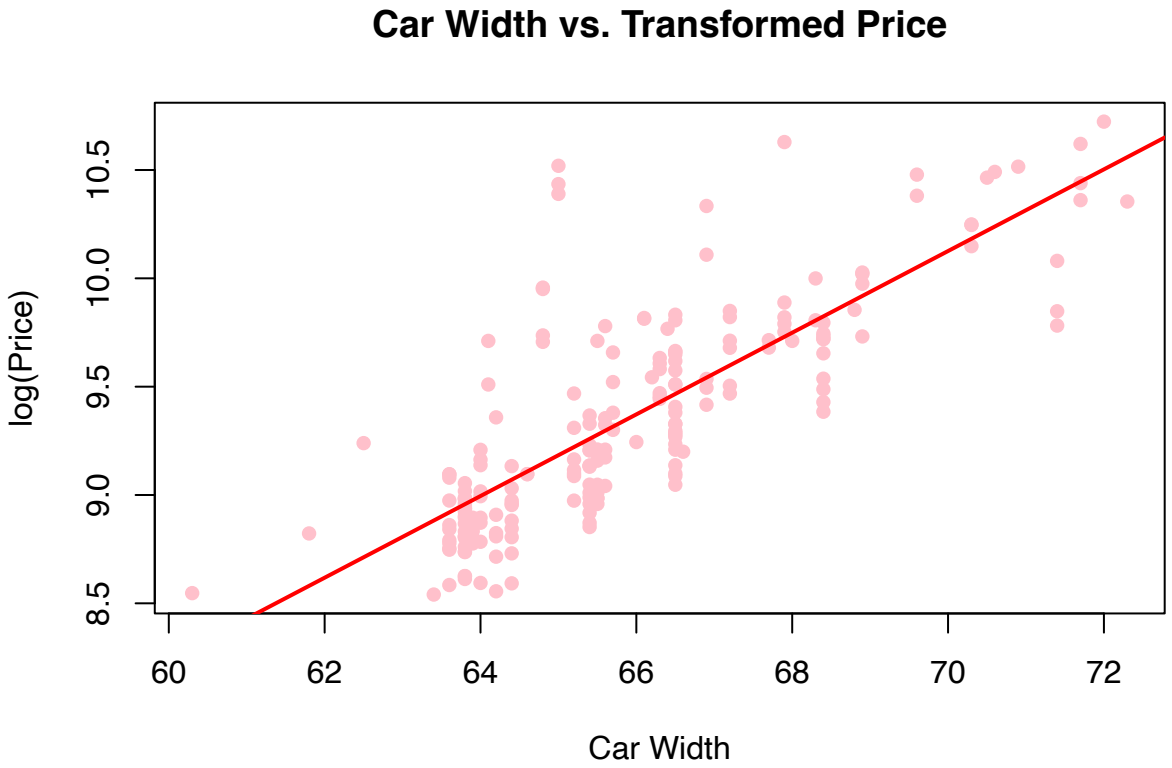



The points are closer to the diagonal line after the tranformation, indicating a more normal distribution.

```
plot(data$carwidth, data$log_price,
     main = "Car Width vs. Transformed Price",
```

```
     xlab = "Car Width",
     ylab = "log(Price)",
     pch = 16, col = "pink")

abline(ols.mod4, col="red", lwd=2)
```




From this scatterplot, we can see that the data points seem to show a linear relationship after the transformation of price.

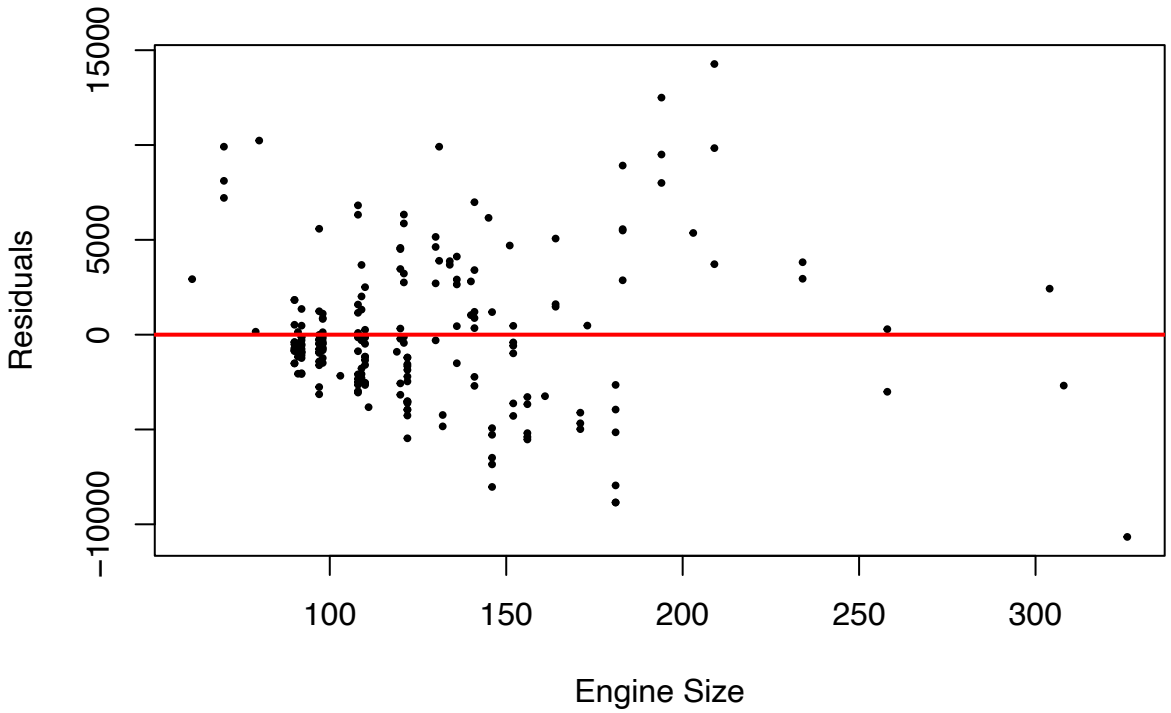
#### 4.3.5 Step 5: Final Transformation

A log-linear model provides percentage interpretation for coefficients, which is particularly intuitive in economics. While the Box-Cox method suggested a transformation closer to lambda = negative 1, which would imply an inverse transformation, the model becomes difficult to interpret or produces implausible results. Since $R^2$ indicates the proportion of variance explained by the model, it provides a measure of how well the log-linear model fits the data.

### 4.4 Variable 3: Price ~ Engine Size

#### 4.4.1 Step 1: Residuals plot on untransformed data:

```
ols.mod5 <- lm(data$price~data$enginesize, data=data)
plot(data$enginesize, ols.mod5$residuals, pch=20, ylab="Residuals", xlab="Engine Size", cex=0.6)
abline(h=0,col="red", lwd=2)
```



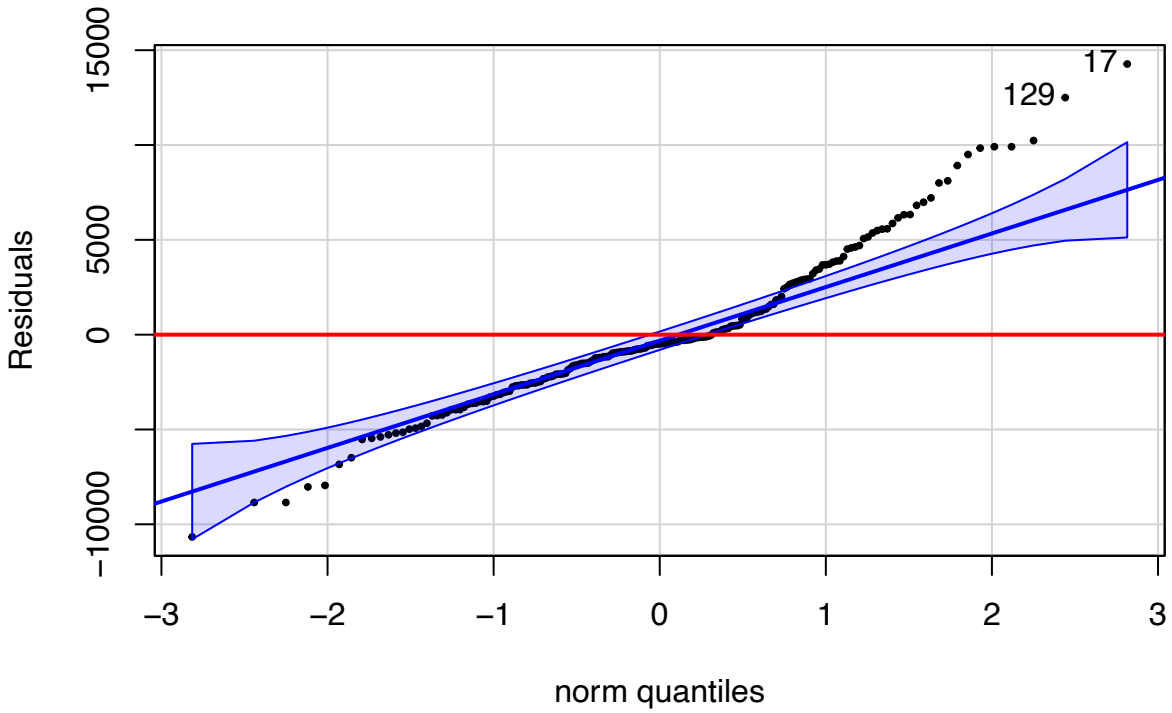

From the residuals plot, it can be seen that it the variance is not constant. Therefore, a transformation might be needed.

### 4.4.2 Step 2: QQ Plot of Residuals on untransformed data

```
qqPlot(ols.mod5$residuals, pch=20, ylab="Residuals", cex=0.6)
```

```
## [1]  17 129
```

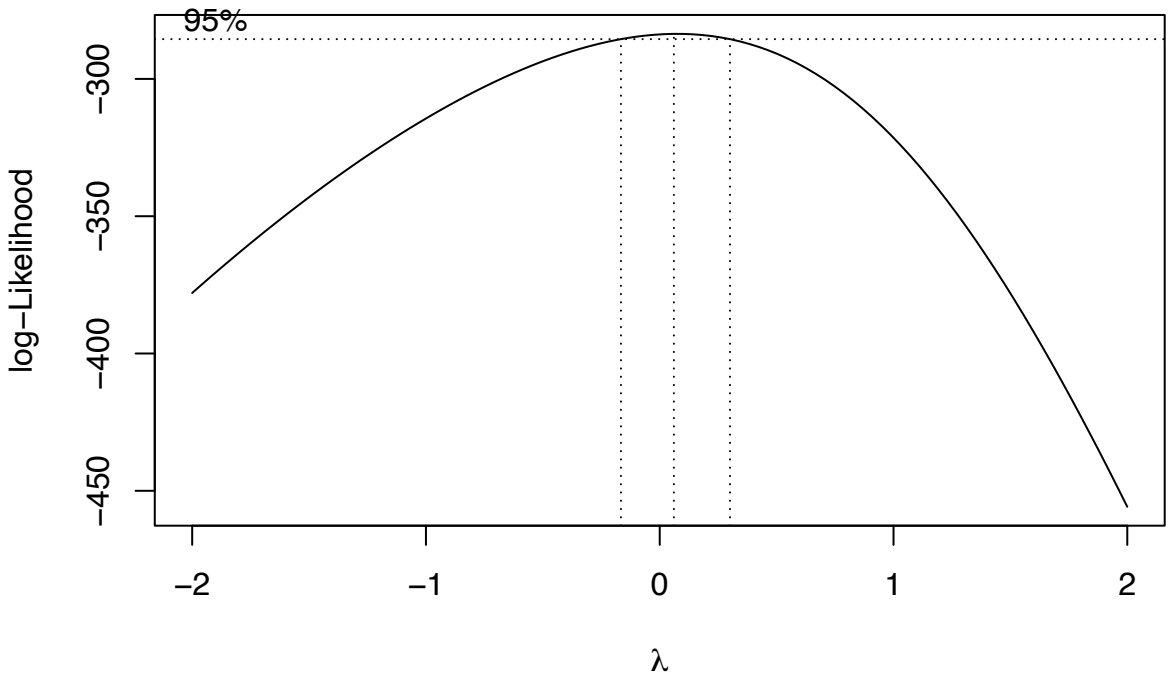
```
abline(h=0,col="red", lwd=2)
```




The points are closer to the diagonal after the transformation, indicating a more normal distribution.

#### 4.4.3 Step 3: Box Cox

```
boxcox(lm(data$price~data$enginesize, data = data), lambda = seq(-2, 2, by = 0.1))
```




Box-cox transformation suggests that lambda = 0, indicating that a log transformation of price is supported.

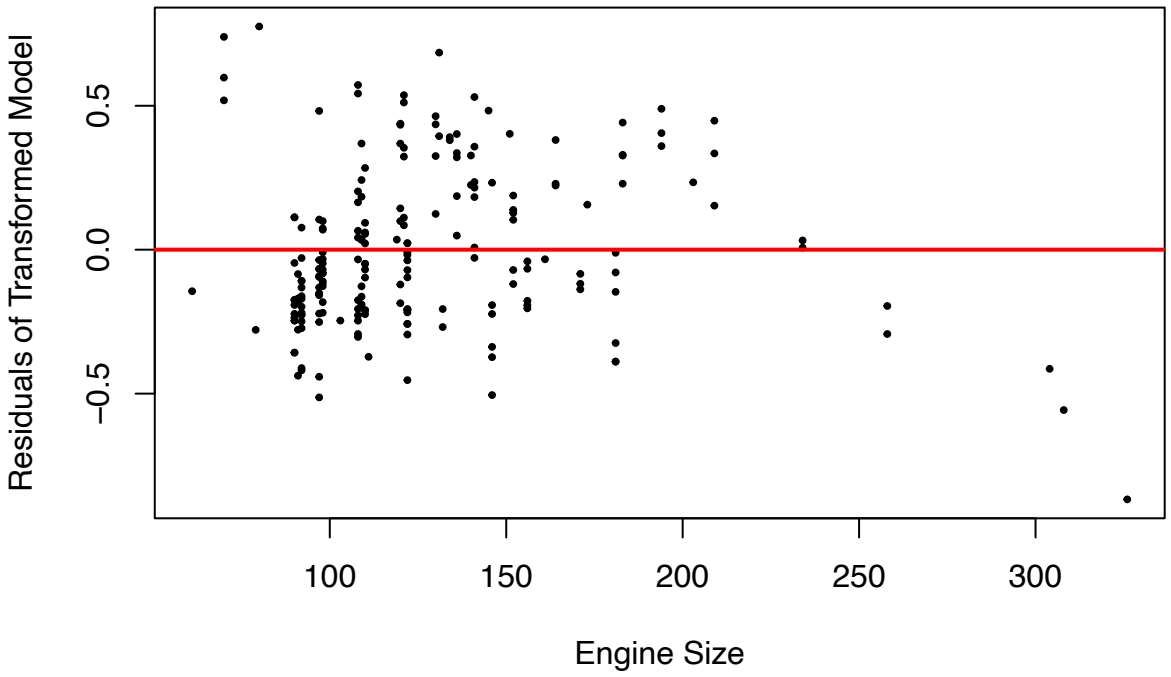
However, since histogram of engine size is right skewed, we might need to transform engine size as well. Therefore, we believe a log-log transformation might be ideal.

### 4.4.4 Step 4: Exploring & Applying Transformations

Let's explore and compare both log-linear and log-log transformations.

log linear

```
ols.mod6 <- lm(data$log_price~data$enginesize, data=data)
plot(data$enginesize, ols.mod6$residuals, pch=20, ylab="Residuals of Transformed Model", xlab="Engine S
abline(h=0,col="red", lwd=2)
```

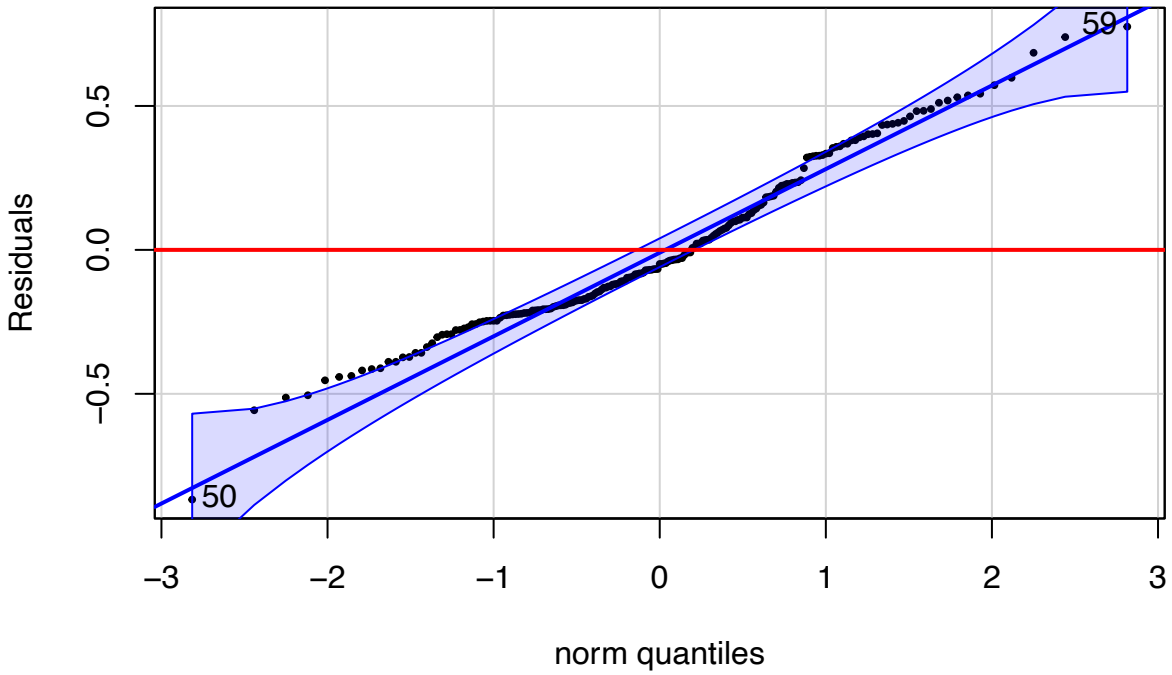



There seems to be a decrease in variance with an increase in engine size, which is not ideal.

```
qqPlot(ols.mod6$residuals, pch=20, ylab="Residuals", cex=0.6)
```

```
## [1] 50 59
```

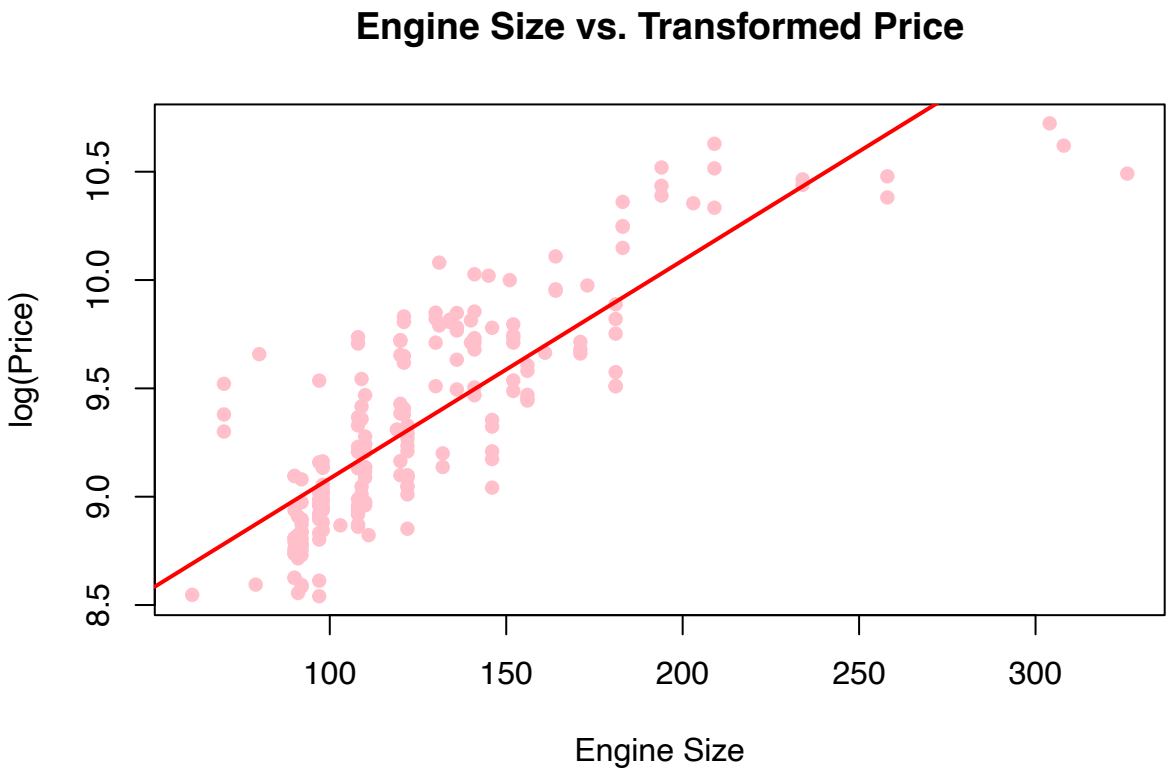
```
abline(h=0,col="red", lwd=2)
```




The points lie very close to the diagonal, suggesting a normal distribution of the transformed data.

```
plot(data$enginesize, data$log_price,
     main = "Engine Size vs. Transformed Price",
     xlab = "Engine Size",
     ylab = "log(Price)",
     pch = 16, col = "pink")

abline(ols.mod6, col="red", lwd=2)
```

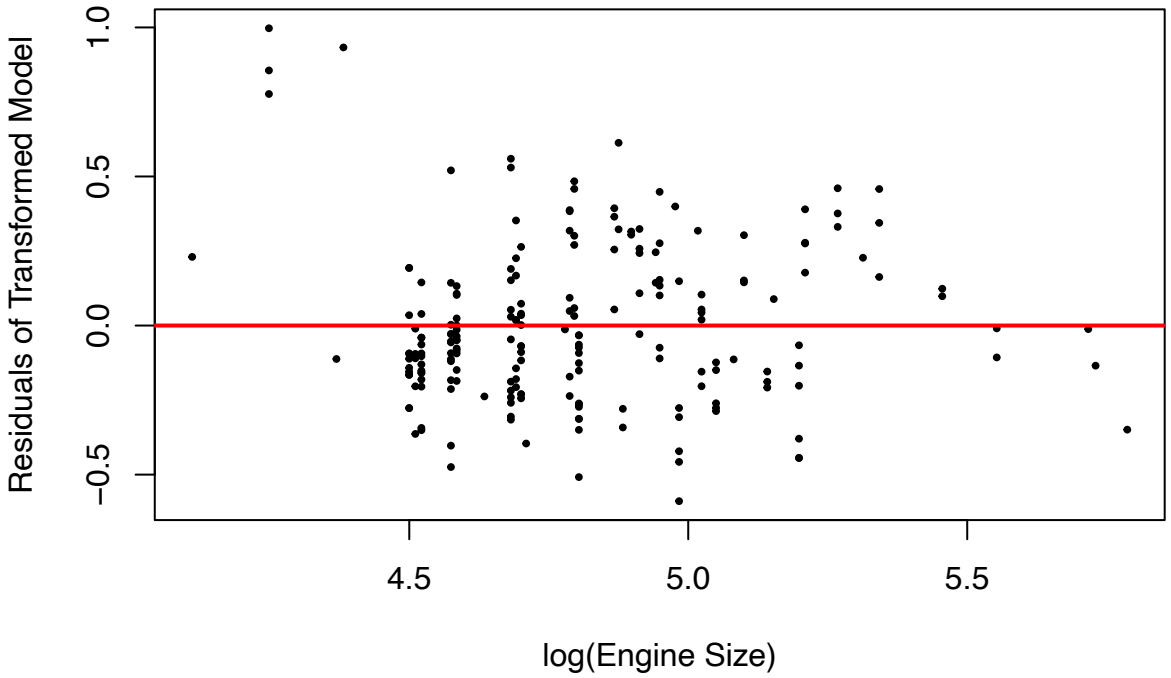



There seems to still be a slight nonlinear trend in the scatterplot despite the transformation.

log log

```
data$log_enginesize = log(data$enginesize)
```

```
ols.mod7 <- lm(data$log_price~data$log_enginesize, data=data)
plot(data$log_enginesize, ols.mod7$residuals, pch=20, ylab="Residuals of Transformed Model", xlab="log(
abline(h=0,col="red", lwd=2)
```

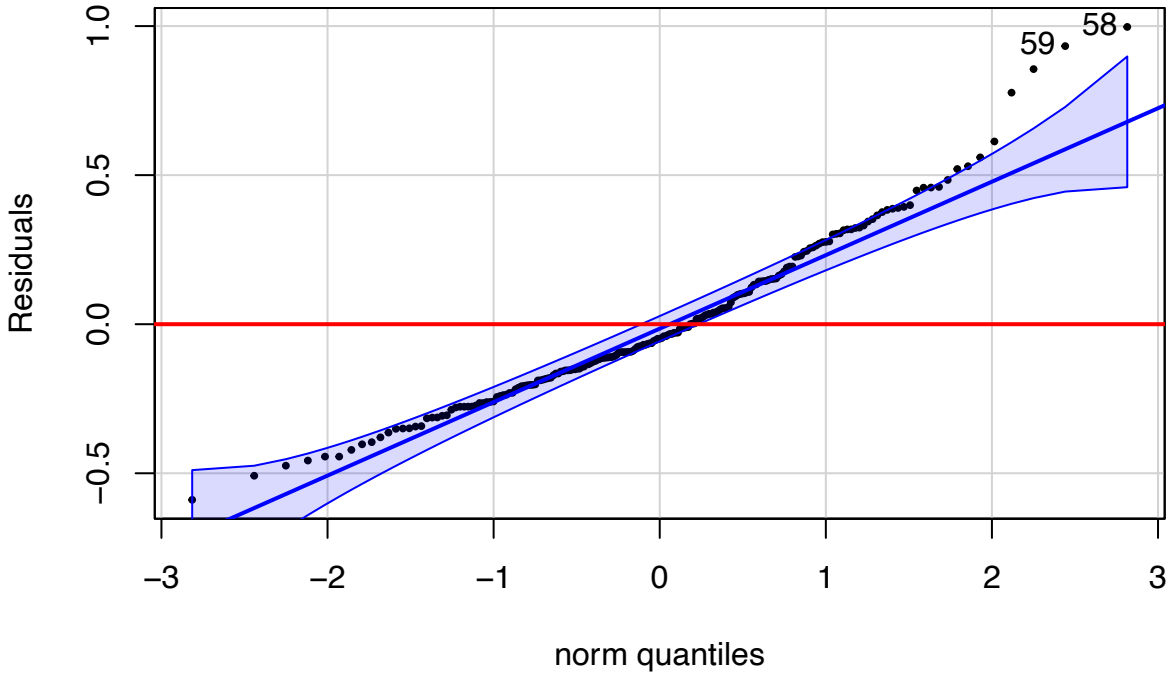


The residuals plot is randomly scattered, and supports our homoscedasticity assumptions.

```
qqPlot(ols.mod7$residuals, pch=20, ylab="Residuals", cex=0.6)
```

```
## [1] 58 59
```

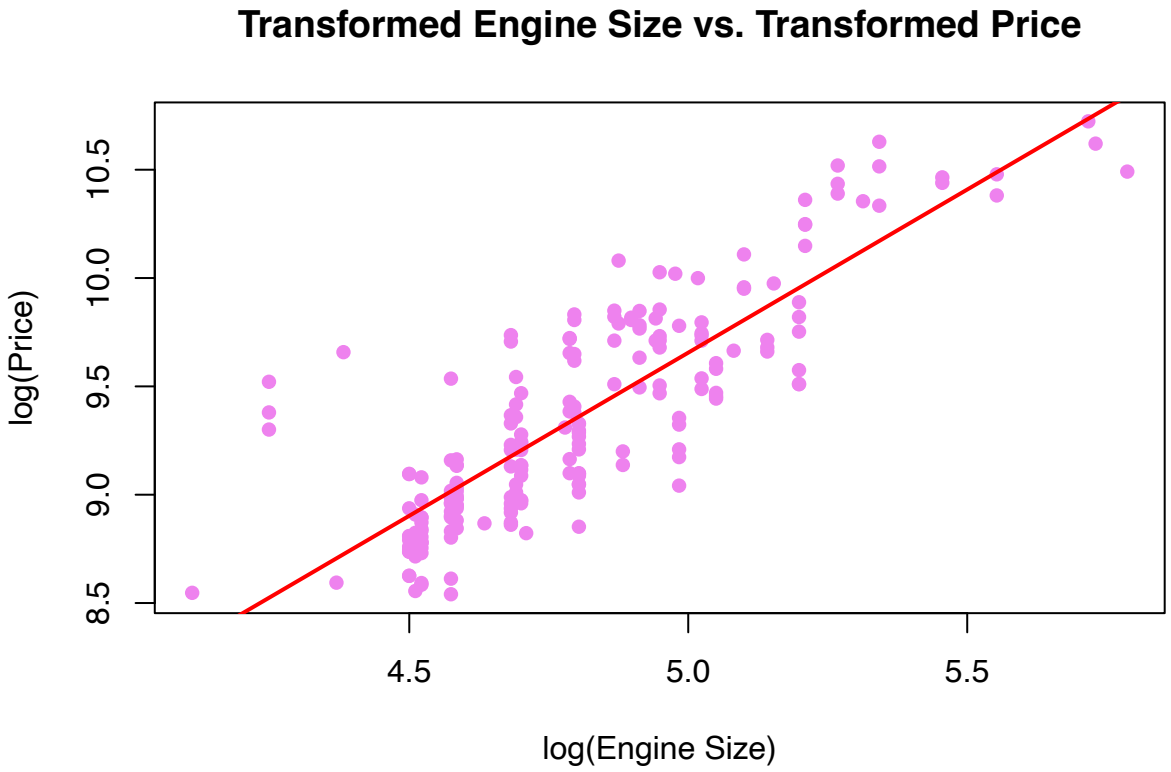
```
abline(h=0,col="red", lwd=2)
```




The points are closer to the diagonal line, barring certain outliers, indicating a more normal distribution post transformation.

```
plot(data$log_enginesize, data$log_price,
     main = "Transformed Engine Size vs. Transformed Price",
     xlab = "log(Engine Size)",
     ylab = "log(Price)",
     pch = 16, col = "violet")

abline(ols.mod7, col="red", lwd=2)
```




From the scatterplot, we can see that the data is linear after the log-log transformation

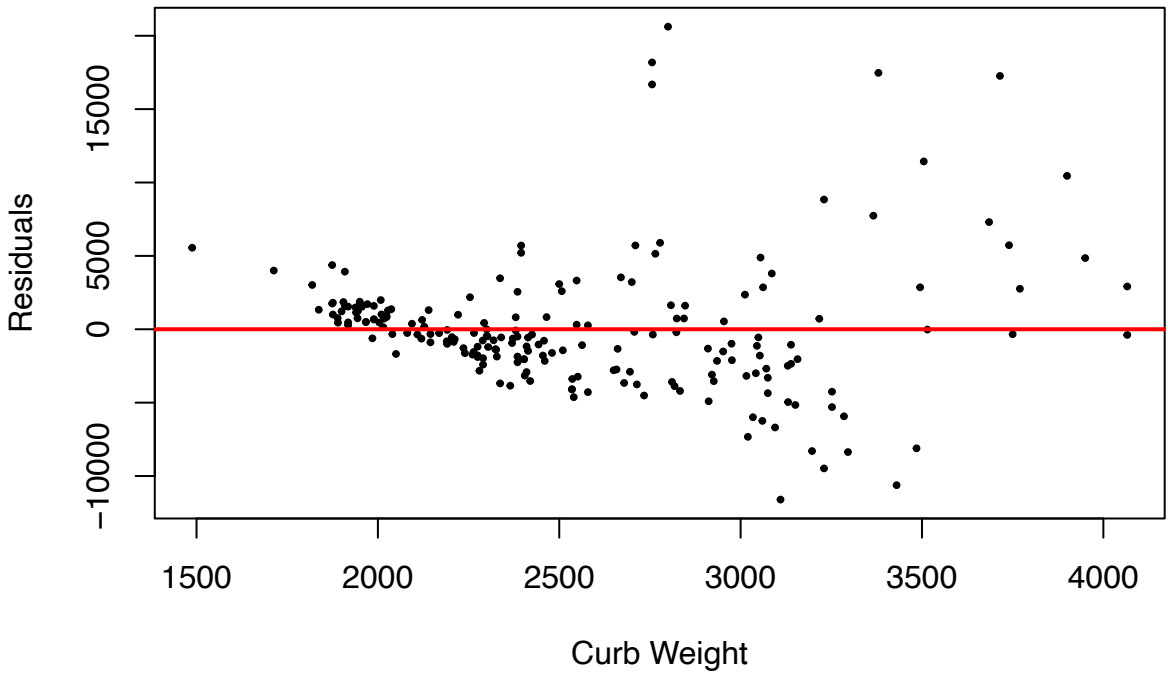
#### 4.4.5 Step 5: Final transformation

Log-log will be the chosen transformation because in comparing the scatterplots of the untransformed, log-linear and log-log data, it can be seen that the log-log is the most linear and therefore most likely suited for a linear regression.

### 4.5 Variable 4: Price ~ Curb Weight

#### 4.5.1 Step 1: Residuals plot on untransformed data:

```
ols.mod8 <- lm(data$price~data$curbweight, data=data)
plot(data$curbweight, ols.mod8$residuals, pch=20, ylab="Residuals", xlab="Curb Weight", cex=0.6)
abline(h=0,col="red", lwd=2)
```



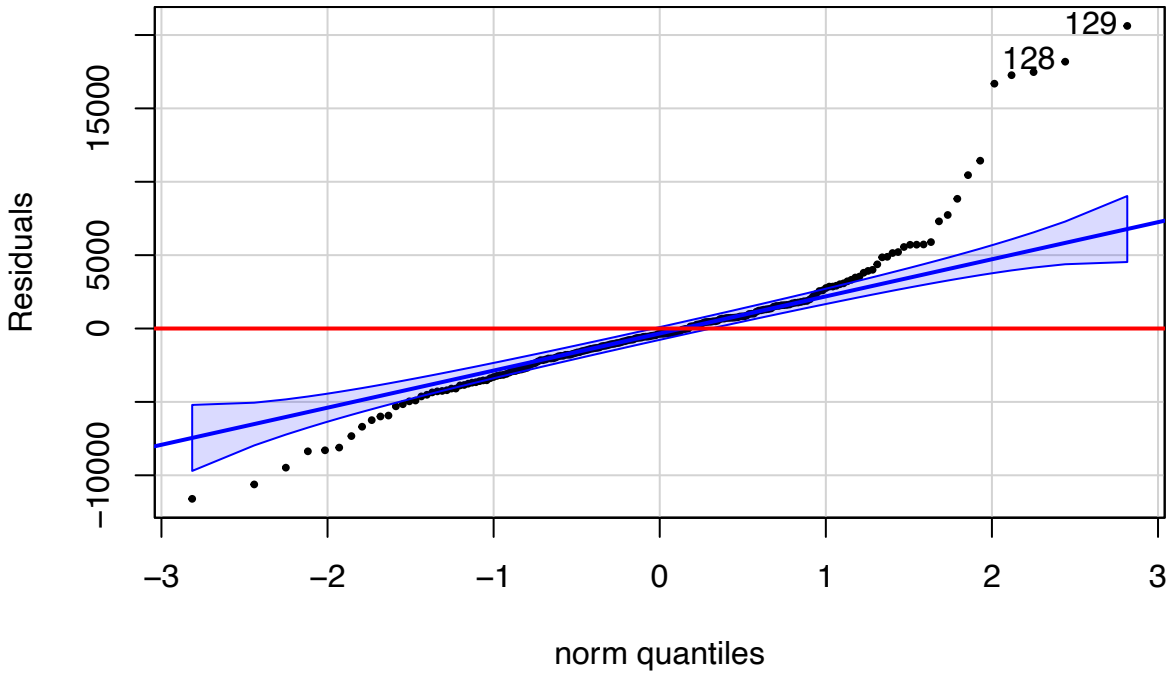

Variation is not constant and instead is increasing as curb weight increases, suggesting a need for a trasnformation.

#### 4.5.2 Step 2: QQ Plot of residuals on untransformed data:

```
qqPlot(ols.mod8$residuals, pch=20, ylab="Residuals", cex=0.6)
```

```
## [1] 129 128
```

```
abline(h=0,col="red", lwd=2)
```

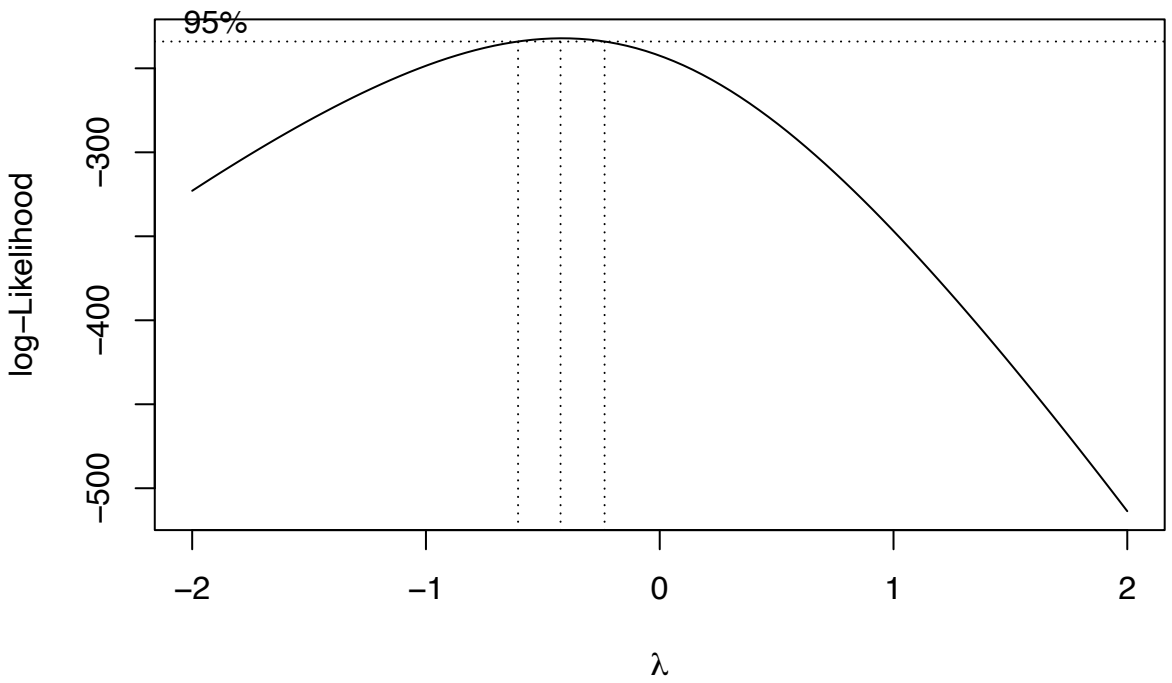



There seem to be large deviations from the diagonal line, suggesting that the untransformed data is not

normally distributed.

#### 4.5.3 Step 3: Box-Cox

```
boxcox(lm(data$price~data$curbweight, data = data), lambda = seq(-2, 2, by = 0.1))
```



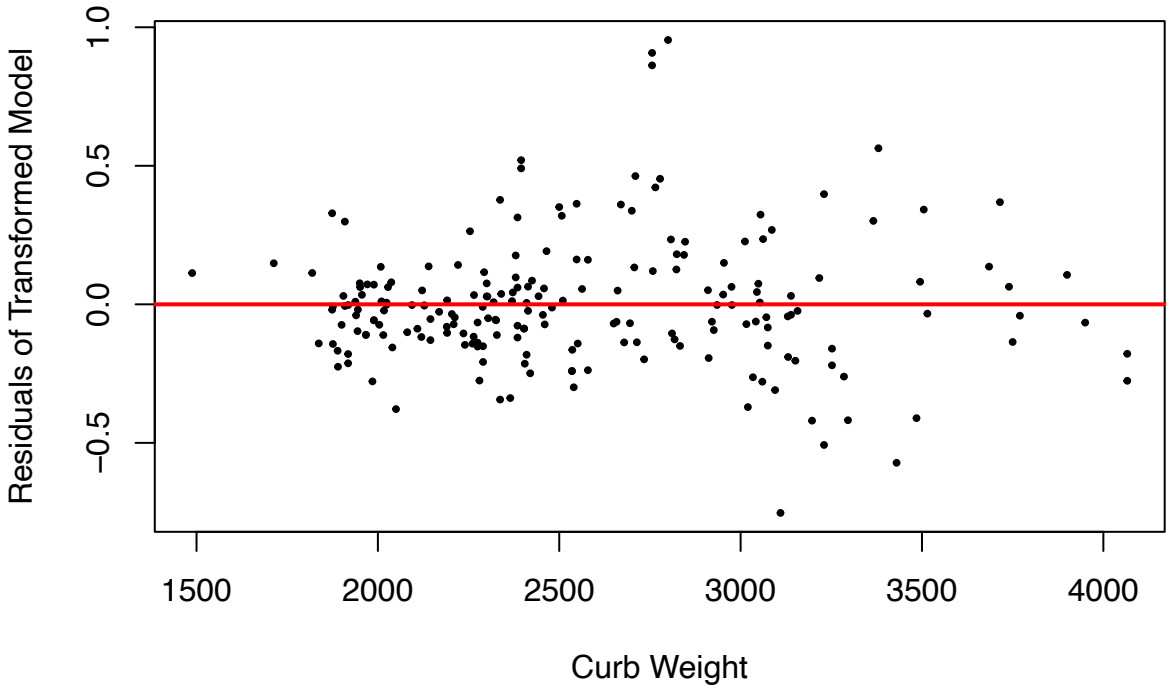

Once again, lambda is a negative value, which does not align with economic intuition for reasons previously explained.

Since histogram of curb weight is right skewed, might need log-log transformation

#### 4.5.4 Step 4: Exploring & Applying Transformations

log linear

```
ols.mod9 <- lm(data$log_price~data$curbweight, data=data)
plot(data$curbweight, ols.mod9$residuals, pch=20, ylab="Residuals of Transformed Model", xlab="Curb Wei
abline(h=0,col="red", lwd=2)
```

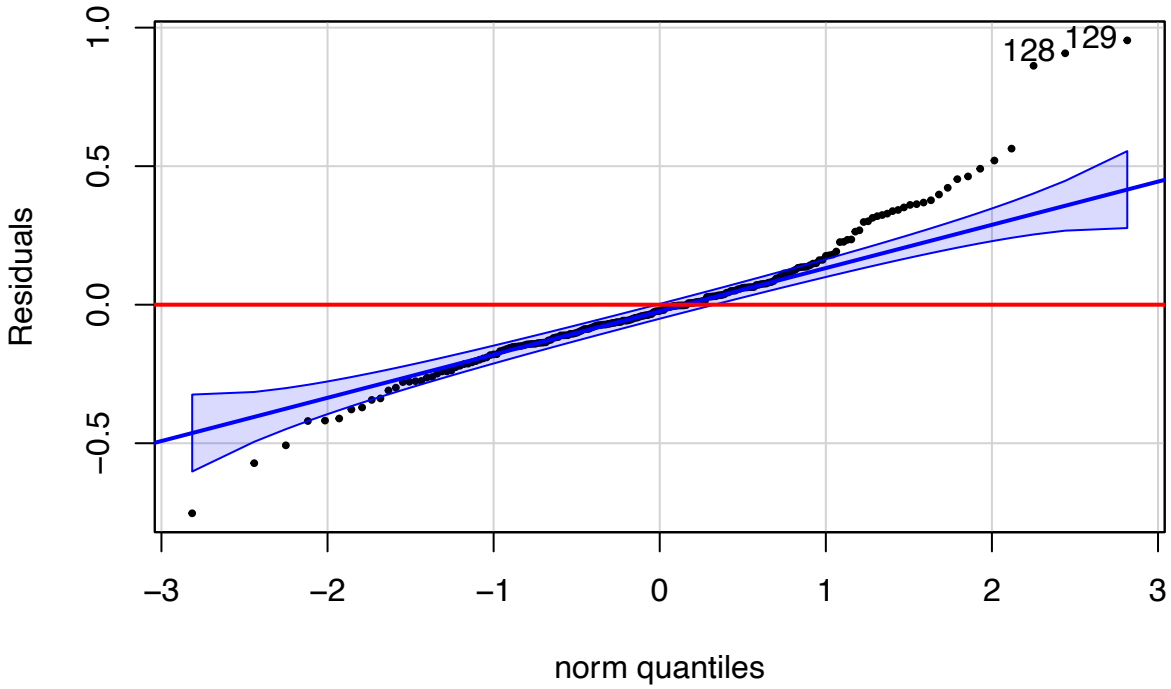



From the residuals plot, we can see that the assumption of homoscedasticity is met once the data is transformed as the variance is constant.

```
qqPlot(ols.mod9$residuals, pch=20, ylab="Residuals", cex=0.6)
```

```
## [1] 129 128
```

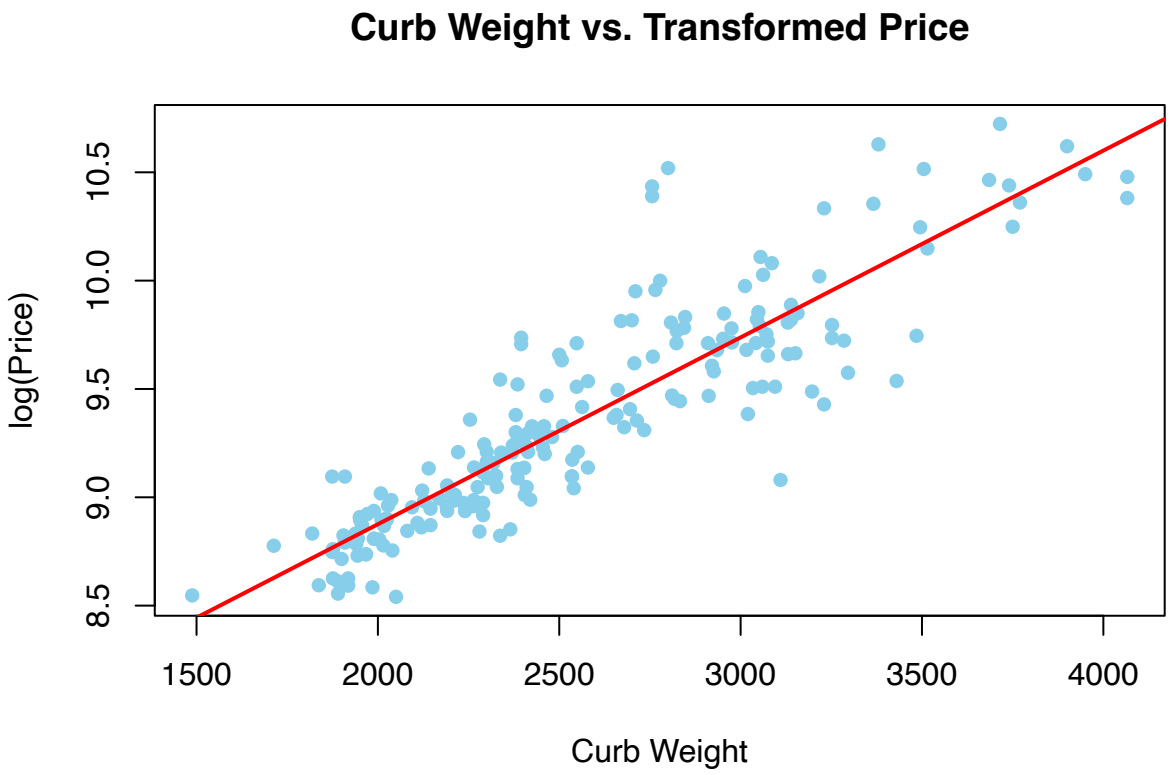
```
abline(h=0,col="red", lwd=2)
```




Excluding the outliers, the points seem closer to the diagonal than the untransformed data.

```
plot(data$curbweight, data$log_price,
     main = "Curb Weight vs. Transformed Price",
     xlab = "Curb Weight",
     ylab = "log(Price)",
     pch = 16, col = "skyblue")

abline(ols.mod9, col="red", lwd=2)
```

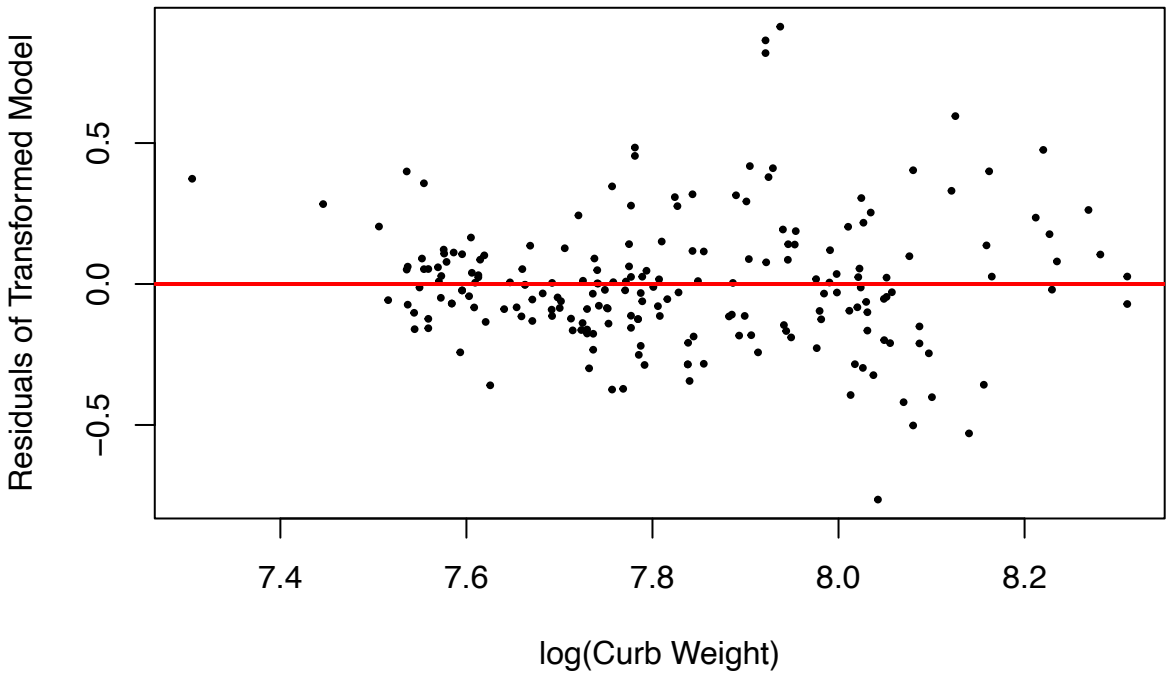



From the scatterplot, we can see that after the log-linear transformation has been applied, the data follows a more linear trend. However, let's try the log log model as well to compare and choose which transformation is ideal.

log log

```
data$log_curbweight = log(data$curbweight)
```

```
ols.mod10 <- lm(data$log_price~data$log_curbweight, data=data)
plot(data$log_curbweight, ols.mod10$residuals, pch=20, ylab="Residuals of Transformed Model", xlab="log
abline(h=0,col="red", lwd=2)
```

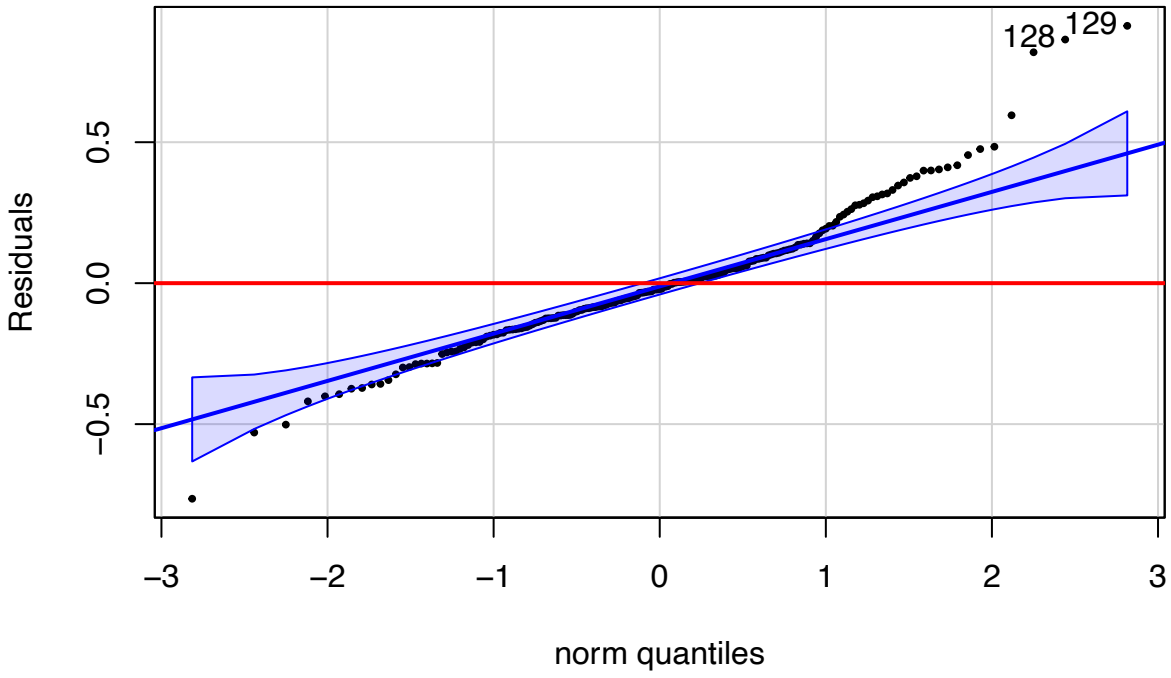



The residuals plot seems to be randomly scattered, which is desired.

```
qqPlot(ols.mod10$residuals, pch=20, ylab="Residuals", cex=0.6)
```

```
## [1] 129 128
```

```
abline(h=0,col="red", lwd=2)
```

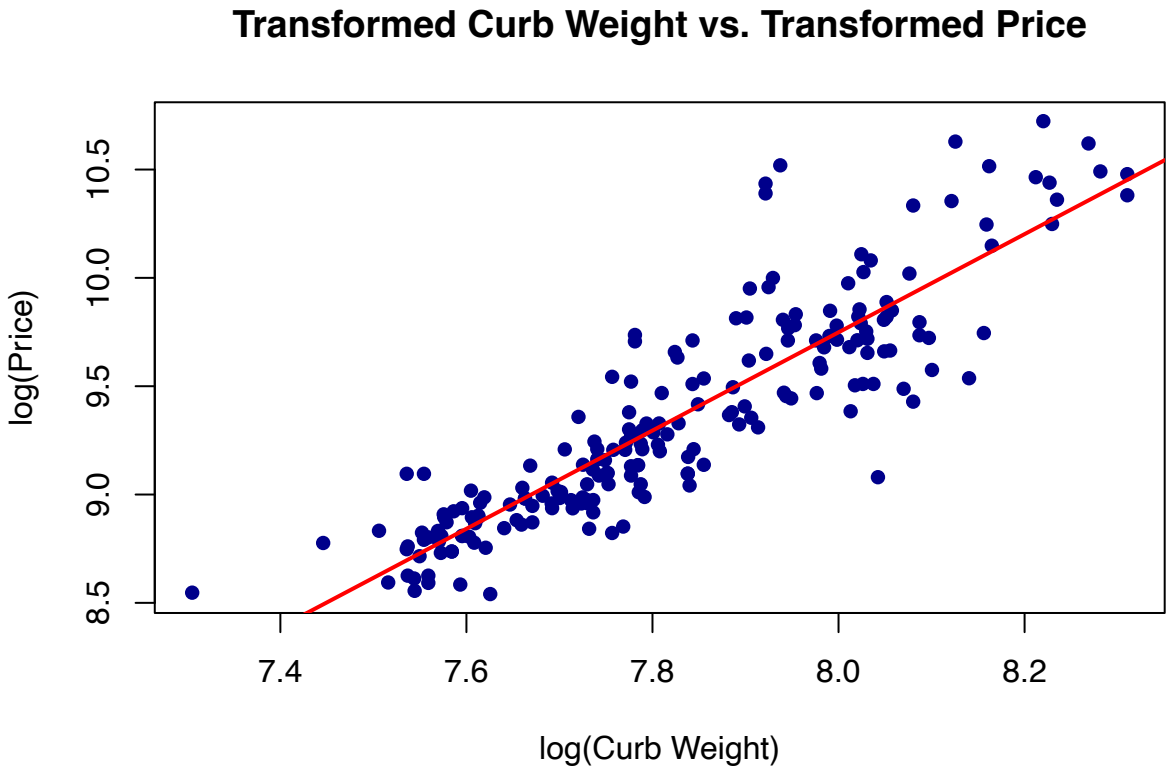



After the log-log transformation, it seems as though the points are a lot closer to the diagonal point, suggesting a distribution closer to the normal distribution.

```
plot(data$log_curbweight, data$log_price,
     main = "Transformed Curb Weight vs. Transformed Price",
```

```
     xlab = "log(Curb Weight)",
     ylab = "log(Price)",
     pch = 16, col = "darkblue")

abline(ols.mod10, col="red", lwd=2)
```




From the scatterplot, it can be seen that while the points are quite linear, the log-linear scatterplot is more clear in this trend.

#### 4.5.5 Step 5: Final Transformation

From the residuals and the scatterplot, the log-linear model is more ideal when compared to the log-log and untransformed data due to the linearity of the scatter plot.

# 5 Question 5: OLS Models

## 5.1 Model 1: Price ~ Car Length

```
model_length <- lm(log_price ~ carlength, data = data)
model_length
```

```
##
## Call:
## lm(formula = log_price ~ carlength, data = data)
##
## Coefficients:
## (Intercept)    carlength
##     3.89711      0.03136
```

```r
summary(model_length)
```

```
##
## Call:
## lm(formula = log_price ~ carlength, data = data)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.70524 -0.21881 -0.08421  0.17572  1.32621
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 3.897108   0.320366   12.16   <2e-16 ***
## carlength   0.031356   0.001836   17.08   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.3235 on 203 degrees of freedom
## Multiple R-squared:  0.5896, Adjusted R-squared:  0.5876
## F-statistic: 291.7 on 1 and 203 DF,  p-value: < 2.2e-16
```

Statistical Significance: The coefficient for carlength (0.03136) is highly statistically significant, with a p-value less than 0.001. This suggests strong evidence that carlength is positively associated with log_price.

Economic Significance: The coefficient 0.03136 for carlength implies that for each one-unit increase in carlength, the expected log of price increases by 0.03136. When transformed back to the original price scale, this translates to roughly a 3.1% increase in price for each additional unit of carlength.

With an R-squared of 0.5896, about 59% of the variation in log_price is explained by carlength, suggesting a moderately strong model fit for predicting log_price based on carlength.

Parameter Estimate Interpretation: When carlength is zero, the log price is approximately 3.8971 and each one-unit increase in carlength is associated with an approximate 3.1% increase in price.

## 5.2 Model 2: Price ~ Car Width

```r
model_width <- lm(log_price ~ carwidth, data = data)
summary(model_width)
```

```
##
## Call:
## lm(formula = log_price ~ carwidth, data = data)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.60796 -0.18684 -0.04654  0.11379  1.33585
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -3.067564   0.648469   -4.73 4.19e-06 ***
## carwidth     0.188479   0.009834   19.17  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.3013 on 203 degrees of freedom
## Multiple R-squared:  0.6441, Adjusted R-squared:  0.6423
```

```
## F-statistic: 367.3 on 1 and 203 DF,  p-value: < 2.2e-16
```

Statistical Significance: The coefficient for carwidth (0.1885) is highly statistically significant, with a p-value less than 0.001. This strong significance indicates that carwidth is a meaningful predictor of log_price.

Economic Significance: The coefficient of 0.1885 implies that for each additional unit increase in carwidth, the expected log_price increases by 0.1885. When converted back to the price scale, this represents an approximately 18.85% increase in price per unit increase in carwidth.

With an R-squared of 0.6441, around 64% of the variability in log_price is explained by carwidth, suggesting a strong model fit for predicting log_price based on car width.

Parameter Estimates Interpretation: When carwidth is zero, the log price would theoretically be -3.0676 and each one-unit increase in carwidth is associated with an approximately 18.85% increase in price.

## 5.3 Model 3: Price ~ Engine Size

```
model_engine <- lm(log_price ~ log_enginesize, data = data)
summary(model_engine)
```

```
##
## Call:
## lm(formula = log_price ~ log_enginesize, data = data)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -0.5891 -0.1812 -0.0488  0.1513  0.9972
##
## Coefficients:
##                Estimate Std. Error t value Pr(>|t|)
## (Intercept)     2.12661    0.32063   6.633 2.93e-10 ***
## log_enginesize  1.50579    0.06668  22.582  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2695 on 203 degrees of freedom
## Multiple R-squared:  0.7153, Adjusted R-squared:  0.7139
## F-statistic:   510 on 1 and 203 DF,  p-value: < 2.2e-16
```

Statistical Significance: The coefficient for enginesize (0.01007) is statistically significant, with a p-value less than 0.001. This strong significance suggests that enginesize is a meaningful predictor of log_price.

Economic Signficance: The coefficient of 0.01007 implies that for each additional unit increase in enginesize, the expected log_price increases by 0.01007. This translates to approximately a 1.01% increase in price per unit increase in enginesize, showing that larger engine sizes have a notable but moderate economic impact on price.

With an R-squared of 0.6922, around 69% of the variability in log_price is explained by enginesize, indicating a strong model fit for predicting log_price based on engine size.

Parameter Estimates Interpretation: When enginesize is zero, the log price would theoretically be 8.0773 and each one-unit increase in enginesize is associated with an approximately 1.01% increase in price.

## 5.4 Model 4: Price ~ Curb Weight

```
model_weight <- lm(log_price ~ curbweight, data = data)
summary(model_weight)
```

```
##
## Call:
## lm(formula = log_price ~ curbweight, data = data)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.75279 -0.12912 -0.02254  0.08142  0.95396
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 7.1508813  0.0803220   89.03   <2e-16 ***
## curbweight  0.0008624  0.0000308   28.00   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2291 on 203 degrees of freedom
## Multiple R-squared:  0.7943, Adjusted R-squared:  0.7933
## F-statistic: 783.9 on 1 and 203 DF,  p-value: < 2.2e-16
```

Statistical Significance: The coefficient for curbweight (0.0008624) is highly statistically significant, with a p-value less than 0.001. This indicates that curbweight is a significant predictor of log_price.

Economic Signiicance: The coefficient of 0.0008624 implies that for each one-unit increase in curbweight, the expected log_price increases by 0.0008624, translating to an approximately 0.086% increase in price. While the impact per unit is not apparently significant, it could be economically meaningful over larger ranges of curbweight, showing that heavier cars tend to have higher prices.

With an R-squared of 0.7943, around 79% of the variability in log_price is explained by curbweight, indicating a very strong model fit for predicting log_price based on car weight.

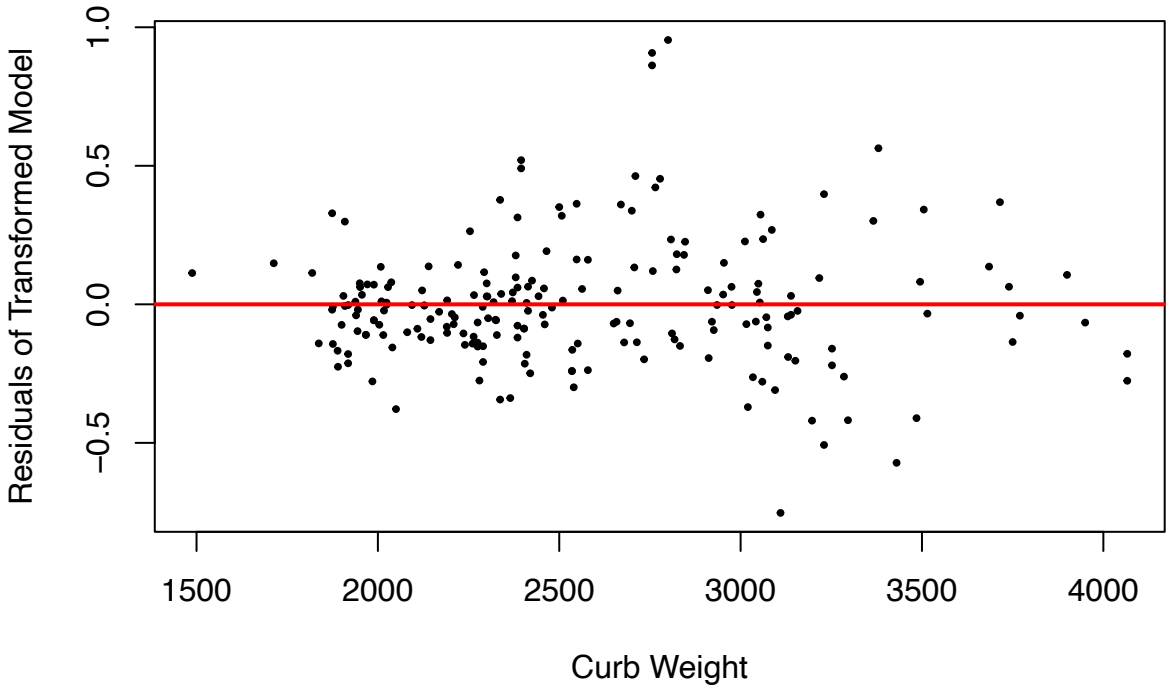
Parameter Estimates Interpretation: The intercept of 7.1509 represents the expected value of log_price when curbweight is zero and each one-unit increase in curbweight is associated with an approximately 0.086% increase in price.

# 6 Question 6: Identifying the Top Choice Model

Price Vs Curb Weight is the top choice model as it has the highest R^2 and also makes economic sense since one would expect heavier cars to be more expensive and weight would play a large role in determining the price of a car

Residuals of chosen model

```
plot(data$curbweight, model_weight$residuals, pch=20, ylab="Residuals of Transformed Model", xlab="Curb
abline(h=0,col="red", lwd=2)
```



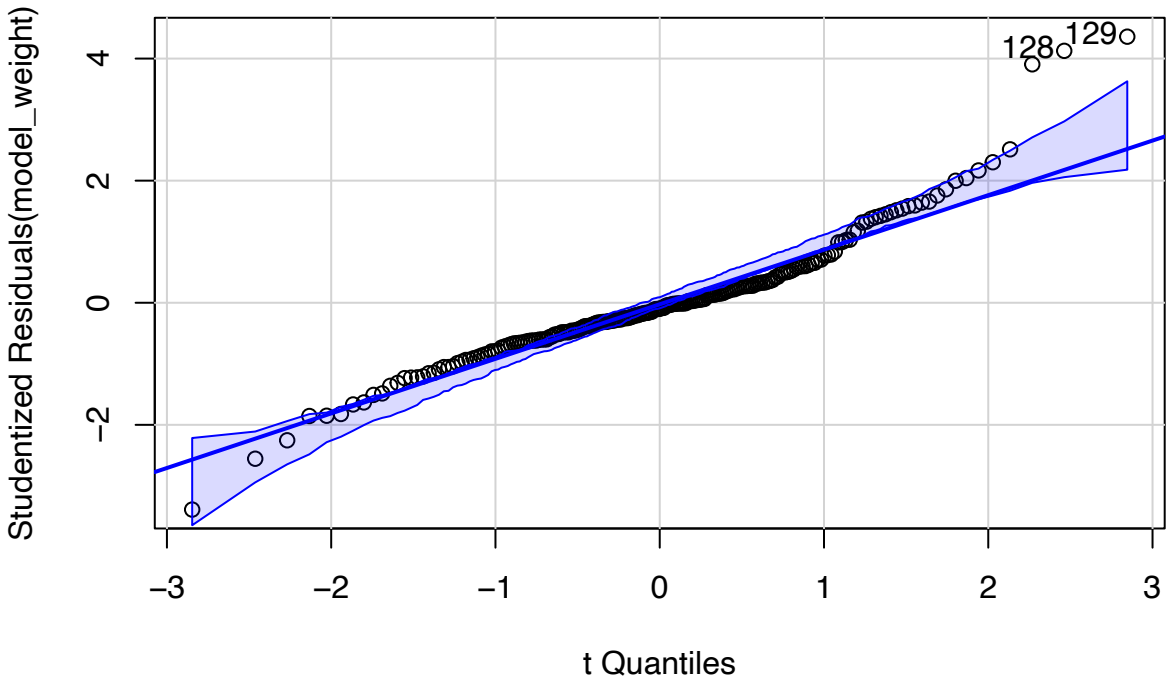

The residuals are plotted against curb weight, and we observe that they are randomly scattered around the horizontal line at zero. This randomness indicates that there is no systematic pattern in the residuals, which is a positive sign. In a well-fitted model, we want the residuals to be randomly distributed, suggesting that the model is capturing the underlying relationship between the variables appropriately.

In this plot, the spread of residuals remains relatively constant across the range of curb weights, supporting the assumption of homoscedasticity. This is important because heteroscedasticity can lead to inefficient estimates and affect the validity of hypothesis tests.

The red horizontal line at zero serves as a reference point. Ideally, the residuals should hover around this line without forming any specific shape, further confirming the appropriateness of the linear model.

## 6.1 QQ Plot of chosen model

```
qqPlot(model_weight)
```




```
## [1] 128 129
```

Mostly follows normal distribution since a lot of the points are close to the line. However, there are some outliers at the lowermost region and the uppermost region, with very heavy outliers in the uppermost region which can influence the distribution.

```
confint(model_weight, level = 0.95)
```

```
##                     2.5 %       97.5 %
## (Intercept) 6.9925088657 7.3092536432
## curbweight  0.0008016223 0.0009230821
```

Intercept (95% CI: 6.99 to 7.31): The intercept represents the expected value of log_price when curbweight is zero. While a curb weight of zero isn't practically meaningful in the context of a car, this value gives a baseline offset for the regression line. The narrow confidence interval suggests that the estimate for the intercept is precise, with a high level of confidence that the true value lies between 6.99 and 7.31.

Curbweight Coefficient (95% CI: 0.0008 to 0.0009): The coefficient of curbweight indicates the expected change in log_price for each additional unit of curb weight. Here, a positive value means that as the weight of a car increases, the log-transformed price is also expected to increase, confirming a positive association between car weight and price. The relatively tight interval from 0.0008 to 0.0009 suggests this estimate is precise and that curb weight has a consistently small yet statistically significant effect on log_price.

The narrow intervals for both parameters show that the model estimates are reliable and that curbweight is a meaningful predictor of log_price.

# 7 Question 7: Bootstrapping!

```
trans.mod = lm(log(price)~curbweight, data=data)
betahat.boot = Boot(trans.mod, R=1000)
usualEsts = summary(trans.mod)$coef[, 1:2]
summary(betahat.boot)
```

```
##
```

```
## Number of bootstrap replications R = 1000
##                original    bootBias     bootSE    bootMed
## (Intercept) 7.15088125 -2.4084e-03 0.07170412 7.14810093
## curbweight  0.00086235  1.0096e-06 0.00003048 0.00086298
```

The bootstrap bias indicates how much the bootstrap estimate deviates from the original estimate. A small bias close to zero suggests that the bootstrap distribution of the coefficient is centered around the original estimate, indicating that the original estimates are stable. The bias is quite small for both coefficients, suggesting that neither is significantly affected by bias from the bootstrap process.

The bootstrap median provides another point estimate for the coefficients based on the bootstrap samples. It's useful for understanding the central tendency of the bootstrapped coefficients. The bootstrap median for both coefficients is very close to the original estimates, suggesting that the original estimates are representative of the center of the bootstrap distribution.

Overall, this output suggests that: - The original coefficient estimates are stable with small biases. - The confidence in the relationship between curbweight and log_price is supported by the tight estimates in the bootstrap analysis. - The coefficient for curbweight is a reliable predictor of car prices, supporting the economic theory that heavier cars tend to be more expensive. - Given the small standard errors and biases, we can be more confident in using this model to make predictions or inform decisions related to car pricing based on curb weight.

```
confint(betahat.boot)
```

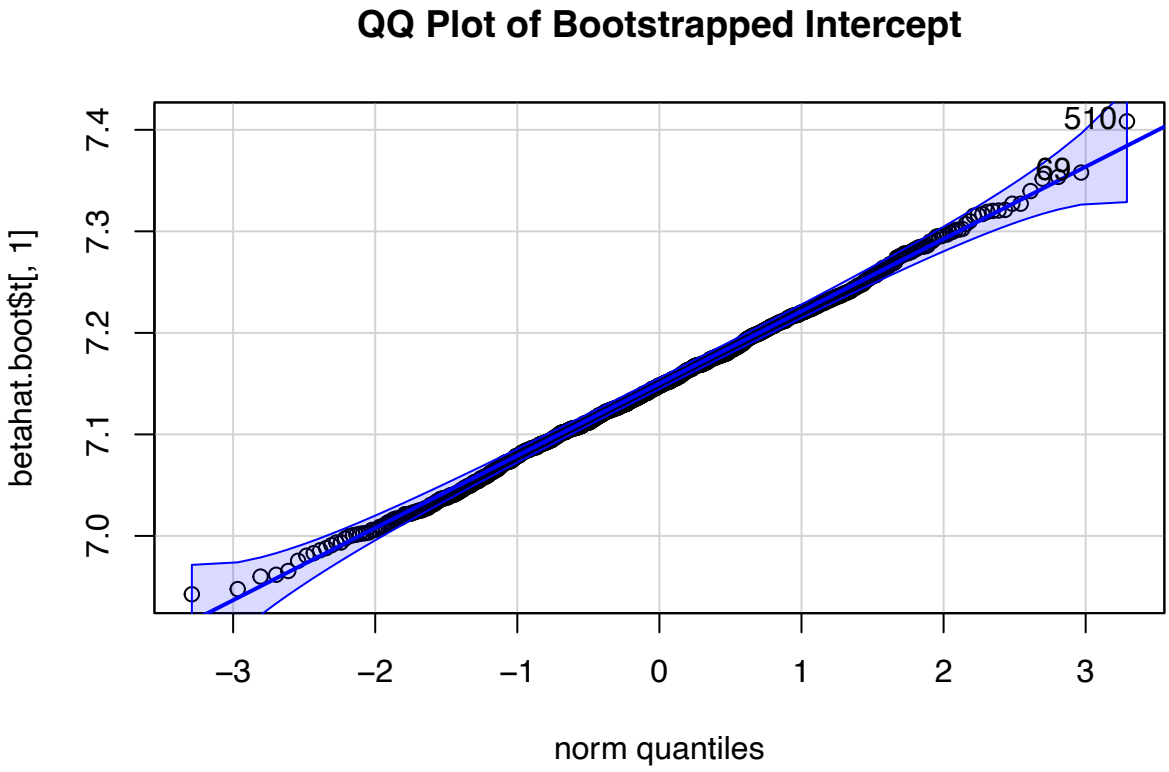
```
## Bootstrap bca confidence intervals
##
##                    2.5 %       97.5 %
## (Intercept) 7.0170212142 7.3013389582
## curbweight  0.0007984922 0.0009214433
```

Intercept: - Original CI: 6.99 to 7.31 - Bootstrapped range: 7.01 to 7.30 The bootstrapped range closely aligns with the original CI, indicating that the intercept estimate is stable across resampling.

Slope (Curbweight): - Original CI: 0.00080 to 0.00092 - Bootstrapped range: 0.00080 to 0.00092 The bootstrapped range for the slope mirrors the original confidence interval almost exactly, highlighting strong consistency. This reinforces that curbweight has a reliable, positive association with price and that the model's slope estimate is not sensitive to sample variations.

Therefore, the close alignment between the original and bootstrapped intervals for both the intercept and slope shows that the model's estimates are both stable and resilient to resampling.

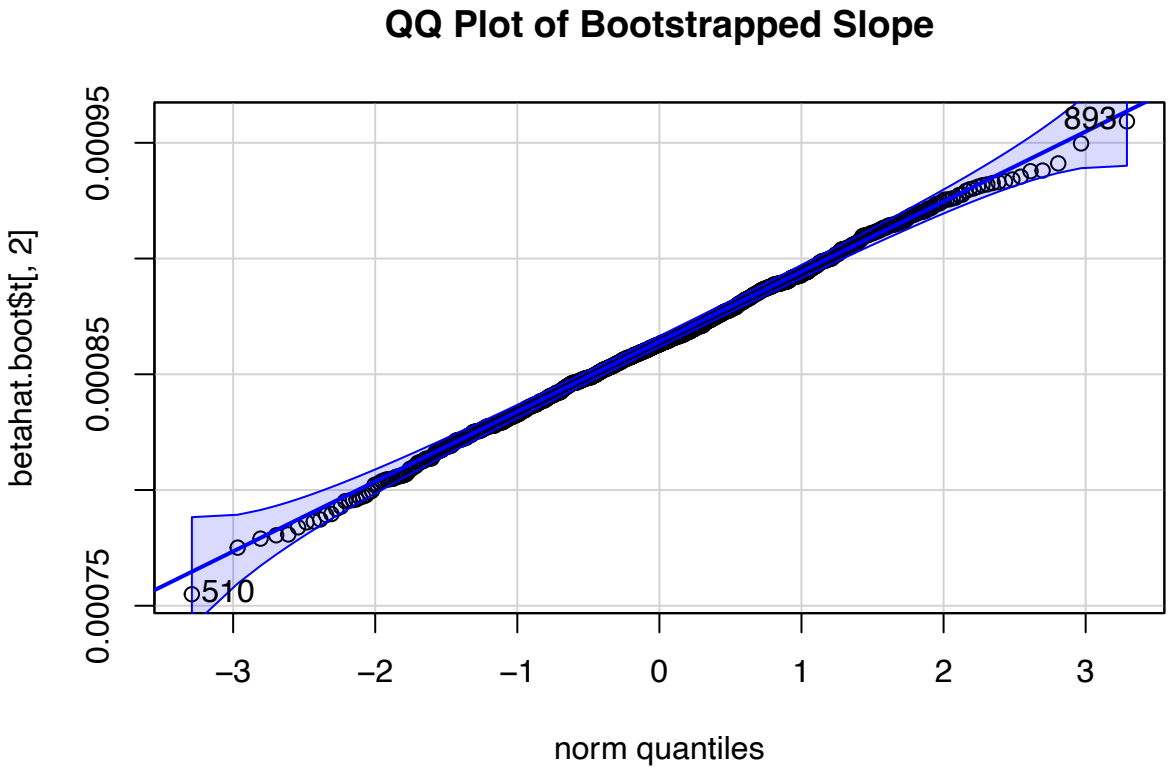
```
qqPlot(betahat.boot$t[, 1], main = "QQ Plot of Bootstrapped Intercept")
```




```
## [1] 510  69
```

This plot suggests that the distribution of bootstrapped intercepts is approximately normal as the points are close to the diagonal line, supporting the stability and reliability of the intercept estimate under the normality assumption.

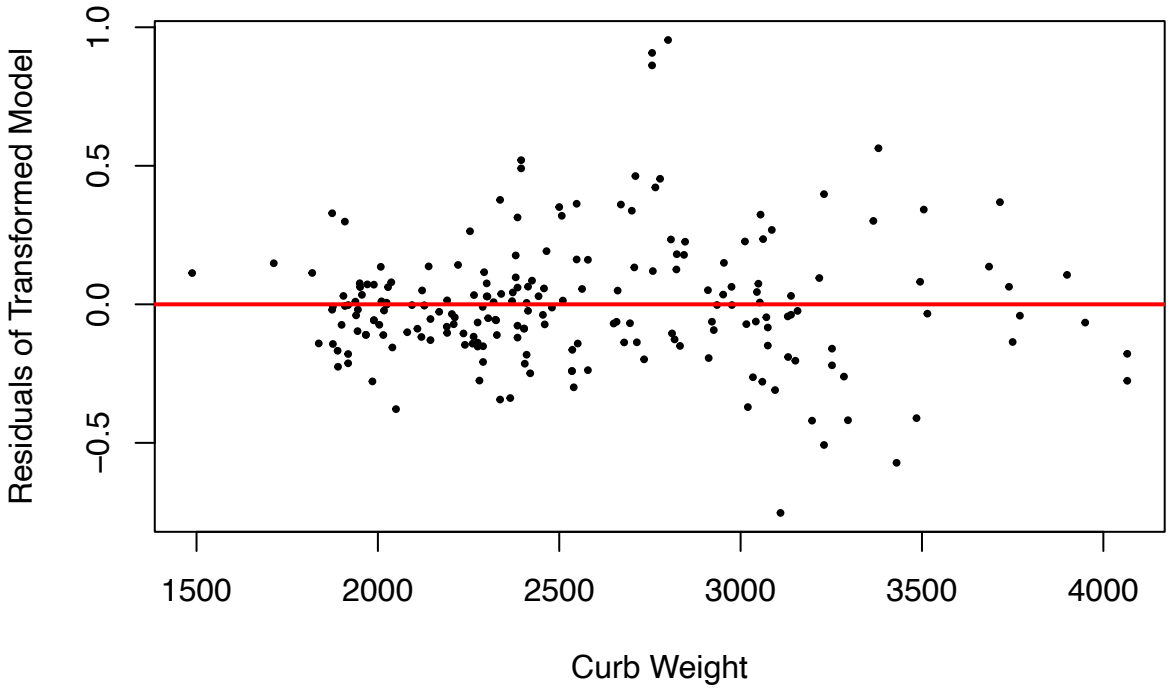
```
qqPlot(betahat.boot$t[, 2], main = "QQ Plot of Bootstrapped Slope")
```




```
## [1] 510 893
```

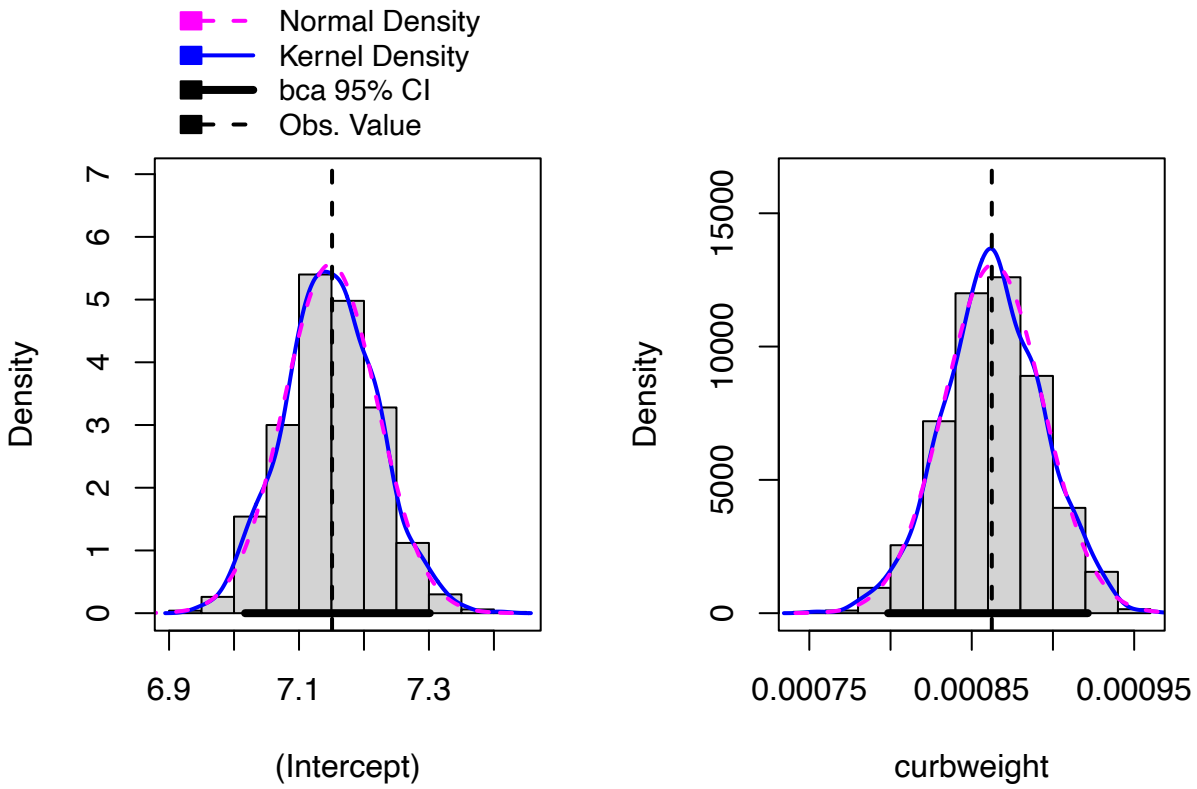
This plot suggests that the distribution of bootstrapped slopes is approximately normal as the points are close to the diagonal line, supporting the stability and reliability of the slope estimate under the normality assumption.

```
plot(data$curbweight, residuals(trans.mod), pch=20,
     ylab="Residuals of Transformed Model",
     xlab="Curb Weight", cex=0.6)
abline(h=0, col="red", lwd=2)
```




The points are randomly scattered, and there is no apparent trend. Additionally, the variance is relatively constant, supporting the homoscedasticity assumption.

```
hist(betahat.boot)
```




The bell-shaped histogram suggests that the bootstrapped estimates are normally distributed, which is desirable for statistical inference.

```r
if (!require(boot)) {
  install.packages("boot")
  library(boot)
}
```

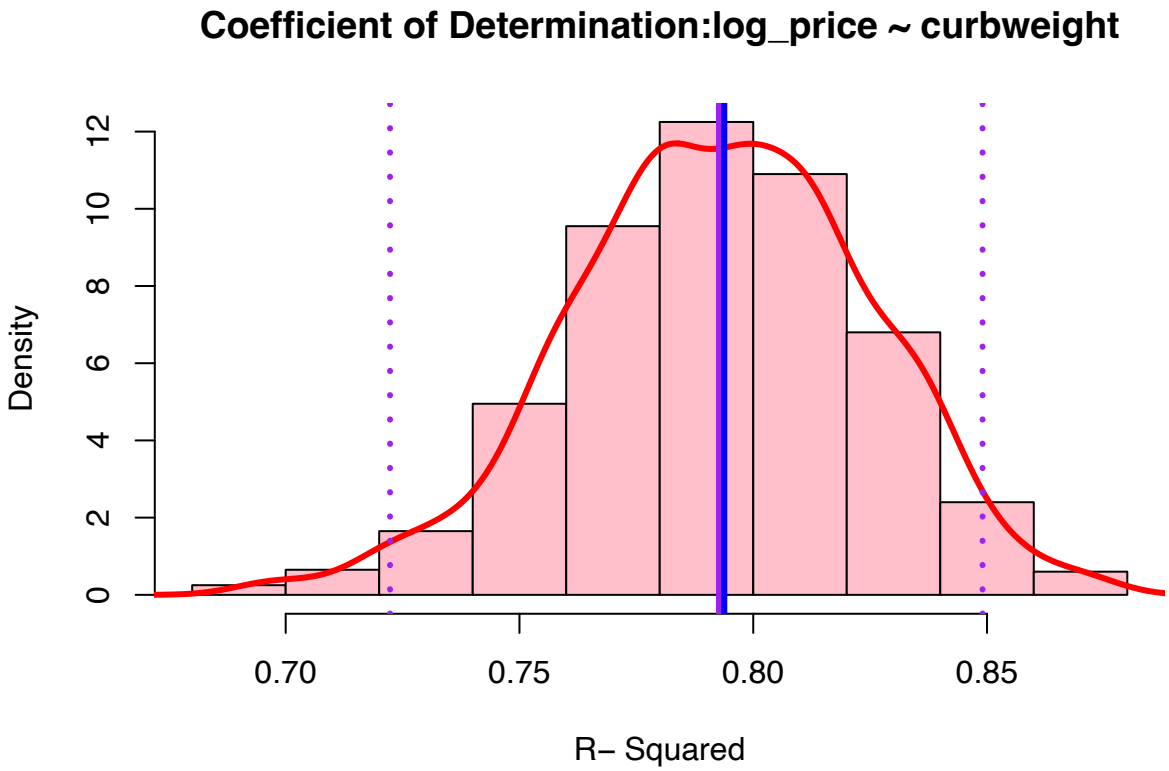
```
## Loading required package: boot
```

```
##
## Attaching package: 'boot'
```

```
## The following object is masked from 'package:car':
##
##     logit
```

```r
# function to obtain R-Squared from the data
rsq <- function(formula, data, indices) {
  d <- data[indices,] # allows boot to select sample
  fit <- lm(formula, data=d)
  return(summary(fit)$r.square)
}

results <- boot(data=data, statistic=rsq, R=1000, formula=log_price~curbweight)
ci1=boot.ci(results, type="bca", index=1,conf=0.95)
ci_rsq = ci1$bca[ , c(4, 5)]
hist(results$t[,1], main = 'Coefficient of Determination:log_price ~ curbweight', xlab = 'R- Squared', 
lines(density(results$t[,1]), col = 'red',lwd=3)
abline(v = ci_rsq, col = 'purple',lwd=3,lty=3)
abline(v=mean(results$t[,1]), col='purple',lwd=3,lty=1)
abline(v=median(results$t[,1]), col='blue',lwd=3,lty=1)
```




In analyzing the bootstrapped R-squared values for the model predicting log price from curb weight, we

observed a slight left skewness in the histogram of R-squared estimates. This suggests that while most bootstrap samples yield relatively high R-squared values, a few samples have significantly lower values. The mean R-squared (X) is slightly less than the median R-squared (Y), indicating that a small number of lower-performing samples are influencing the overall average.

# 8 Question 8: Overall conclusions/findings

This analysis investigated the relationship between car prices and selected vehicle characteristics, focusing on variables such as car length, width, engine size, and curb weight.

These attributes were chosen based on their theoretical and practical significance in influencing car pricing.

To address skewness observed in the price data, a log transformation was applied, which served to stabilize variance and enhance model accuracy.

This transformation allowed for more robust estimation of the relationships between predictors and price.

The results from the transformed models indicated that each of the selected characteristics had a significant and positive impact on car prices, collectively explaining a substantial proportion of the variance.

Among the models, the one using curb weight as a predictor was identified as the best fit, with an R-squared value of 79.43%, suggesting that curb weight alone accounts for nearly 80% of the variability in car price.

This aligns with economic expectations, as heavier cars generally incur greater material and production costs, which are reflected in higher pricing.

However, a limitation of our analysis is that we did not account for other categorical variables such as brand, which could heavily influence the price of the car. Future analysis could be done on all the variables to find an even better suited model.

In conclusion, curb weight emerged as the most influential predictor of car price, and log transformations proved beneficial for addressing skewed data and improving model interpretation.

This analysis highlights the prevalent role of physical characteristics, particularly weight, in determining car prices and highlights the practical utility of log transformations in predictive modeling.