Research Question

Do various car size-related attributes such as car length, car width, car height, and wheelbase significantly impact the price of the car?

Analyzing how car body types influence car prices is vital for understanding market dynamics in the automotive industry. Car body types play a crucial role in shaping consumer preferences, driving market demand, and influencing pricing strategies. Different consumers have varying preferences based on factors like lifestyle, perceived value, and prestige associated with specific body types. For instance, luxury sedans and SUVs are often priced higher due to their association with sophistication and status, while compact hatchbacks may be perceived as more affordable and practical options. Manufacturers utilize insights from this analysis to segment the market effectively, target specific consumer segments, differentiate their offerings from competitors, and allocate resources towards developing models that align with market demand and pricing expectations. Additionally, advancements in technology and innovation often lead to the introduction of new body types optimized for emerging trends such as electric drivetrains or autonomous driving features, further influencing pricing dynamics in the automotive market.

Proof of Why it is interesting:

https://www.sciencedirect.com/science/article/pii/S0965856416311478

First Model Selected:

Initial Regression Model $Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \beta_3 X_3 + \beta_4 X_4 + \epsilon i$

Potential Variables :

H_0: $\beta_1 = \beta_2 = \beta_3 = \beta_4 = 0$

Reduced Model: $Y_i = \beta_0 + \epsilon_i$

H_a: At least one of the Betas is not equal to 0.

Full Model $Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \beta_3 X_3 + \beta_4 X_4 + \epsilon i$

Variable	**Type of Variable**	**Column Name**
Car length	Continuous	X1
Car Width	Continuous	X2
Car Height	Continuous	X3
Wheelbase	Continuous	X4
Price	Continuous	Y

$\beta 1$: The coefficient associated with car length

$\beta 2$: The coefficient associated with car width

$\beta 3$: The coefficient associated with car height

$\beta 4$: Wheelbase

Though the assumptions are not met an initial test is done to verify.

```
> price.Fullmodel1 =lm(price~ carlength+carwidth+carheight
+                   +wheelbase,data = CarPriceData)
> summary(price.Fullmodel1)

Call:
lm(formula = price ~ carlength + carwidth + carheight + wheelbase,
    data = CarPriceData)

Residuals:
   Min     1Q Median     3Q    Max 
 -9932  -2902  -1028   1718  24364 

Coefficients:
              Estimate Std. Error t value Pr(>|t|)    
(Intercept) -129461.13   17398.76  -7.441 2.90e-12 ***
carlength       243.05      68.73   3.536 0.000504 ***
carwidth       2186.58     342.03   6.393 1.12e-09 ***
carheight      -505.06     196.18  -2.574 0.010761 *  
wheelbase      -167.52     140.81  -1.190 0.235583    
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 5034 on 200 degrees of freedom
Multiple R-squared:  0.6108,	Adjusted R-squared:  0.603 
F-statistic: 78.46 on 4 and 200 DF,  p-value: < 2.2e-16
```

From the model summary, we can see the F statistic is high, and the p value is less than 0.05, indicating the model is significant.

$$Y = -129461.13 + 243.05\,X_1 + 2186.58\,X_2 + -505.06\,X_3 + -167.52X_4 + \epsilon i$$

```
> anova(price.Fullmodel1)
Analysis of Variance Table

Response: price
           Df     Sum Sq    Mean Sq  F value    Pr(>F)    
carlength   1 6072096122 6072096122 239.6443 < 2.2e-16 ***
carwidth    1 1521803074 1521803074  60.0602 4.548e-13 ***
carheight   1  322287703  322287703  12.7196 0.0004528 ***
wheelbase   1   35861642   35861642   1.4153 0.2355835    
Residuals 200 5067590820   25337954                       
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

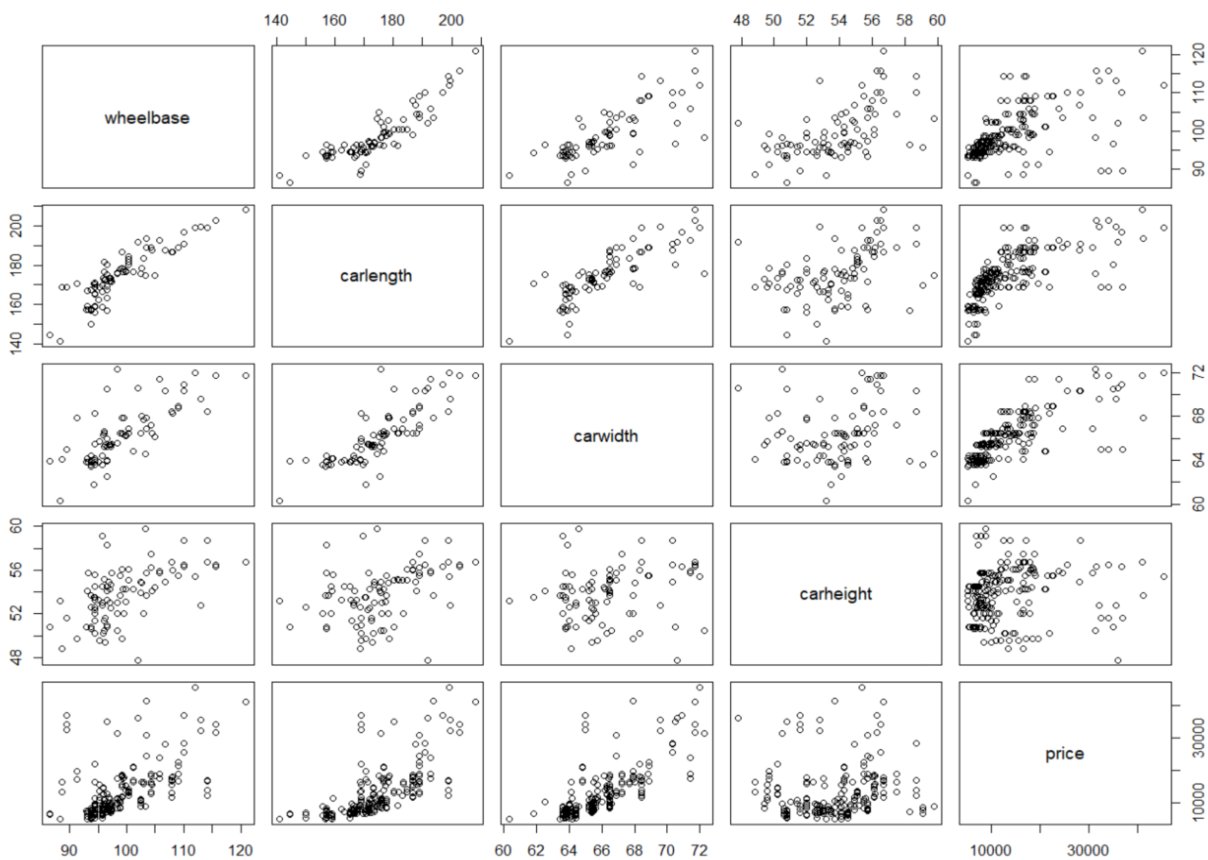
```
> #check for significance of variables by comparing it to full model
> anova(price.Reduced,price.Fullmodel1 )
Analysis of Variance Table

Model 1: price ~ 1
Model 2: price ~ carlength + carwidth + carheight + wheelbase
  Res.Df        RSS Df Sum of Sq     F    Pr(>F)
1    204 1.3020e+10
2    200 5.0676e+09  4 7.952e+09 78.46 < 2.2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

From this GLT test, we can see that it is a rejection without the proper assumptions.

Now we look at the model Assumptions:

1. Linear trend

Scatter plots



2. Outliers and influential points

Y outliers with Studentized deleted residual:

Residual :

```
> rStudentVals <- abs(rstudent(price.Fullmodel1))
> alpha <- 0.05
> # Calculate Bonferroni critical value
> Bonferroni_critical_value <- qt(1 - alpha / (2 * 205), df = 205 - 1 - 5)
> Bonferroni_critical_value
[1] 3.736313
> # Identify outliers based on Bonferroni critical value
> outliers <- abs(rStudentVals) > Bonferroni_critical_value
> # Print identified outliers
> print(rStudentVals[outliers])
      17      127      128      129
4.120000 4.181481 4.528528 5.246328
```

The possible outliers in Y are identified as:

```
      17      127      128      129
4.120000 4.181481 4.528528 5.246328
```

Bonferroni threshold is used here as assumptions of Normality are violated.

It can be seen that data is not normal and does not have constant variance based on the plot and the two tests.

X outliers with Hat Matrix leverage values

```
> residuals=lm.influence(price.Fullmodel1)$hat
> threshold=2*(5/205)
> outlier=abs(residuals)>threshold
> print(residuals[outlier])
         1          2          7          8          9         19         31         32 
0.05141024 0.05141024 0.05682391 0.05682391 0.05837457 0.05369769 0.05571200 0.055712
00
        37         41         44         48         49         50         69         71 
0.06432705 0.05548902 0.05668852 0.06459912 0.06459912 0.08256512 0.04973151 0.049695
93
        72         73         74        105        106        114        126        130 
0.04937345 0.08531066 0.08392683 0.06129423 0.06129423 0.05329273 0.05077230 0.158294
65
       154        155        156 
0.05383455 0.05383455 0.05383455
```

Influential Cases: (DFFITS, DFBETAS, Cook's Distance)

Influence on Single Fitted Values –DFFITS:

```
> thresh1<-2*sqrt(5/205)
> thresh1
[1] 0.3123475
> influentialPoints=dffits(price.Fullmodel1)>thresh1
> print(dffits(price.Fullmodel1)[influentialPoints])
       16        17        18        19        48        49        50        73
0.3705450 0.7141824 0.3894339 0.4976407 0.3574987 0.5391476 0.3565420 0.5343567
       74        75       127       128       129
0.7609479 0.7313398 0.8075064 0.8745263 1.0131442
```

Influence in all fitted value-Cook's distance

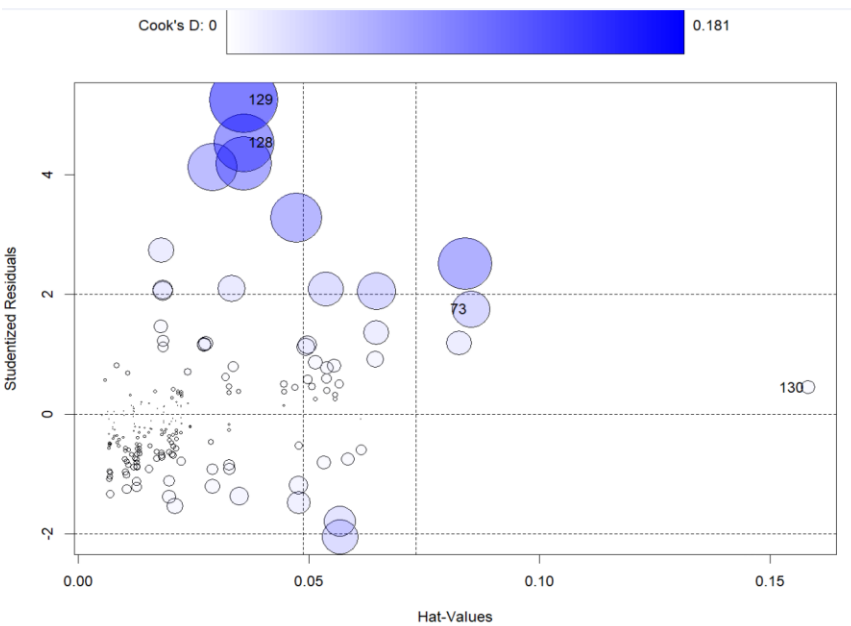
```
> cooksVals <- cooks.distance(price.Fullmodel1)
> max(cooksVals)
[1] 0.1812543
> # compute the critical F values to compare against cooksD
> qf(.2,5,200)
[1] 0.4677463
> thresh2=qf(.5,5,200)
> thresh2
[1] 0.873239
> influentialPoints=abs(cooksVals)>thresh2
> print(cooksVals[influentialPoints])
named numeric(0)
```

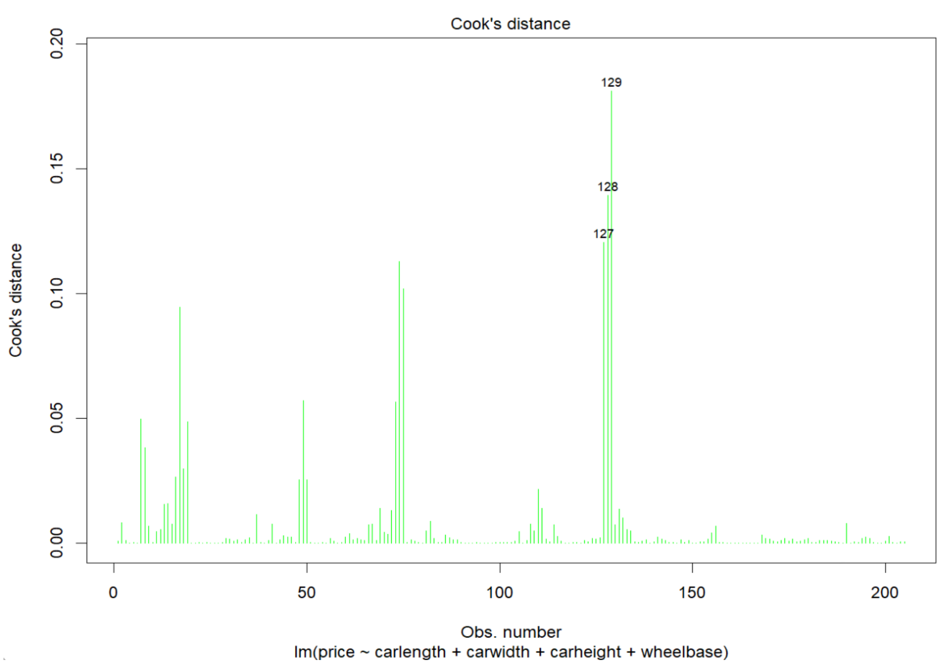
Influence on the Regression Coefficient DFBETAS:

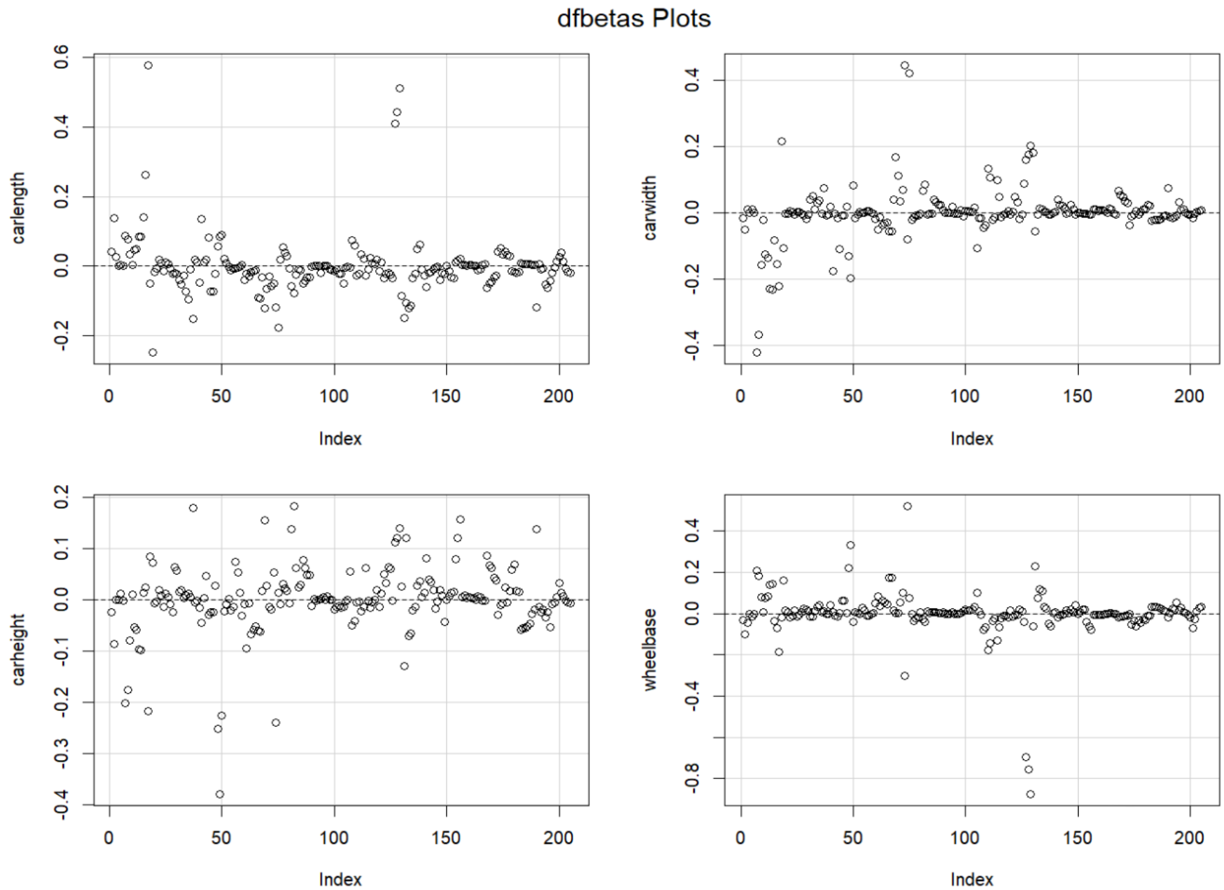
```
> thresh3<-2/sqrt(205)
> thresh3
[1] 0.1396861
> influentialPoints<-dfbetas(price.Fullmodel1, data = CarPriceData) >= thresh3
> influentialPoints<-dfbetas(price.Fullmodel1) >= thresh3
> print(dfbetas(price.Fullmodel1)[influentialPoints])
 [1] 0.4384107 0.3840720 0.1645156 0.1888017 0.1913678 0.1752689 0.1417181 0.1558764
 [9] 0.1624633 0.1402853 0.2631672 0.5784414 0.4085975 0.4425095 0.5126500 0.2162579
[17] 0.1679837 0.4469075 0.4213733 0.1615653 0.1749746 0.2027092 0.1816867 0.1794814
[25] 0.1541898 0.1819183 0.1559518 0.2097458 0.1837489 0.1417733 0.1437002 0.1597528
[33] 0.2215960 0.3341913 0.1735369 0.1754911 0.5206974 0.2302398
```

Diagnostic with the influence plot:

```
> influencePlot(price.Fullmodel1)
     StudRes        Hat       CookD
73  1.7497113 0.08531066 0.056524784
128 4.5285282 0.03595258 0.139365824
129 5.2463282 0.03595258 0.181254333
130 0.4445345 0.15829465 0.007462648
```



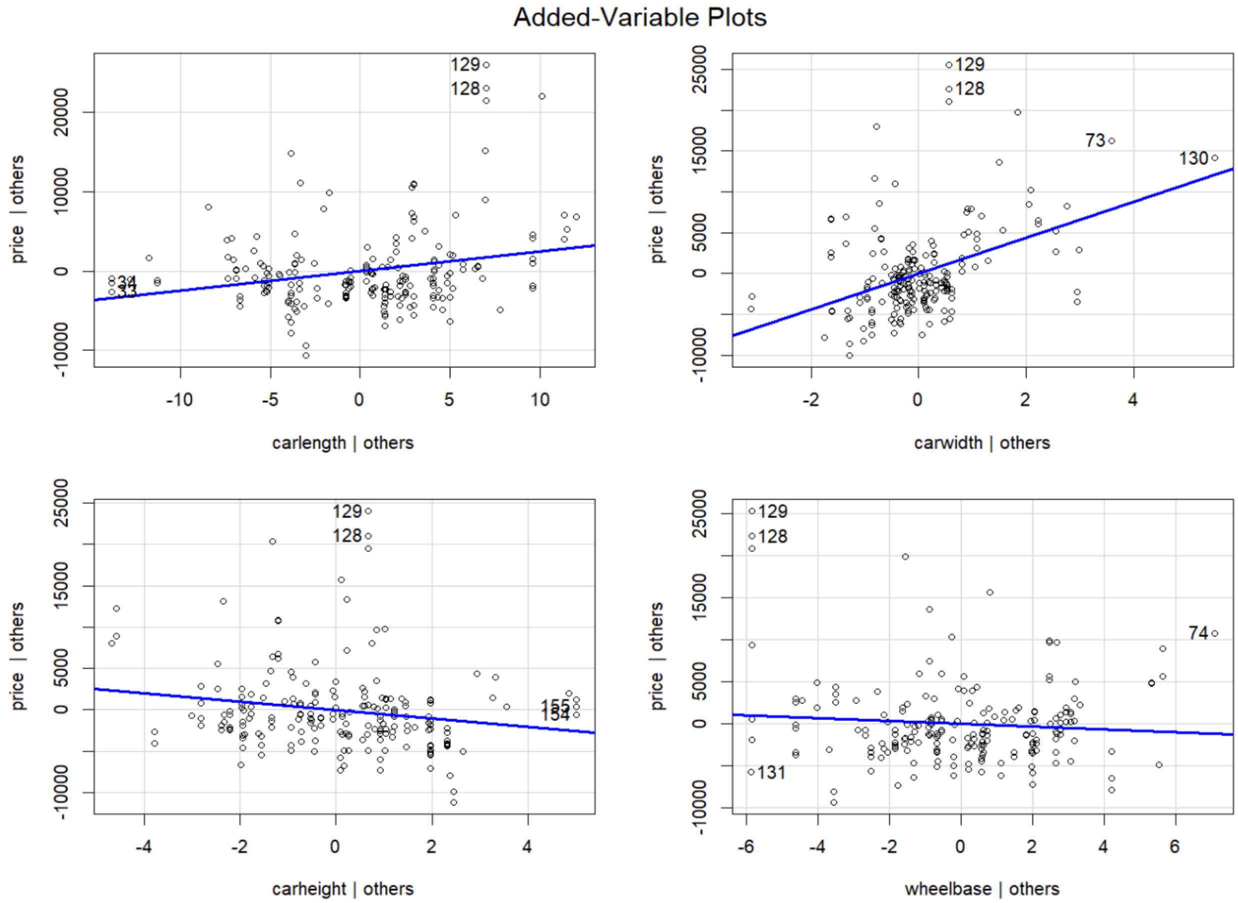




It can be seen that row 128 is a Y outlier whereas the hat_outliers have an X outlier at row 73 and row 130. No influential points were found. Given that there aren't many outliers we can move on for further examinations.

3. Marginal Effect of Predictor Variables

Added variable plots to check the effect of each variable



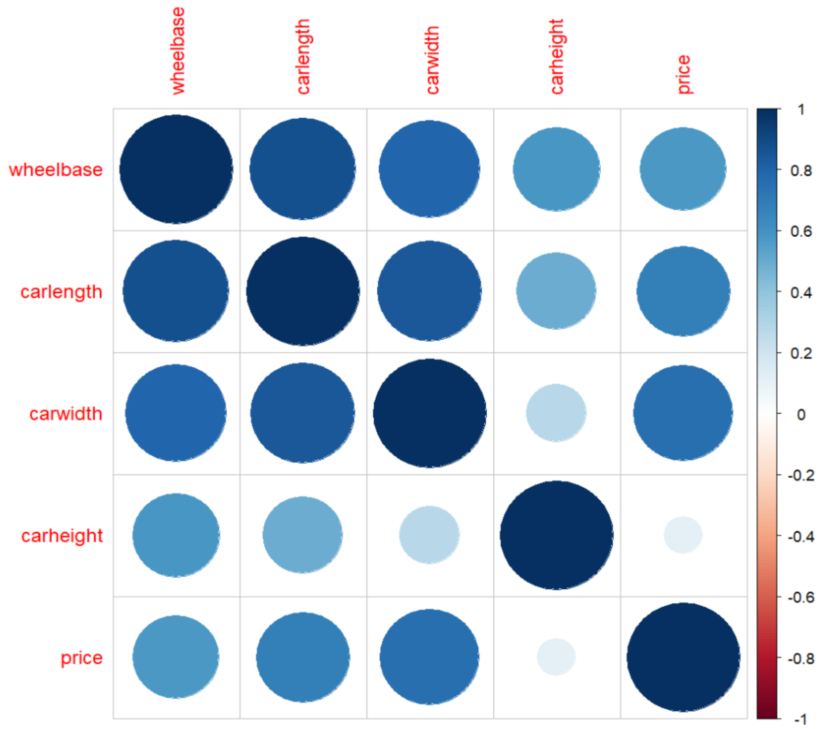
Carwidth shows a positive linear trend. Wheelbase doesn't show any added-on effect.

All the 4 variables seem to have non-constant variance as points are not scattered evenly across 0, instead are concentrated in certain areas.Also, from scatter plot, it is evident that there is increased correlation, particularly between wheelbase, carlength, and carwidth.

This means that all the Xs could benefit from transforming X.

4. Multicollinearity

```
> corMatrix
          wheelbase carlength  carwidth carheight     price
wheelbase 1.0000000 0.8745875 0.7951436 0.5894348 0.5778156
carlength 0.8745875 1.0000000 0.8411183 0.4910295 0.6829200
carwidth  0.7951436 0.8411183 1.0000000 0.2792103 0.7593253
carheight 0.5894348 0.4910295 0.2792103 1.0000000 0.1193362
price     0.5778156 0.6829200 0.7593253 0.1193362 1.0000000
```



From the above correlation results, it is observed that there is high correlation, particularly between wheelbase, carlength, and carwidth.

```
> vifFM <- vif(price.Fullmodel1)
> vifFM
carlength  carwidth carheight wheelbase
 5.788260  4.334334  1.850080  5.788509
```

As a thumb rule, VIF>=10 indicate excessive multicollinearity. Based on the above result, there is no VIF>10, hence there is no multicollinearity among variables.

5. Constant variance

studentized Breusch-Pagan test

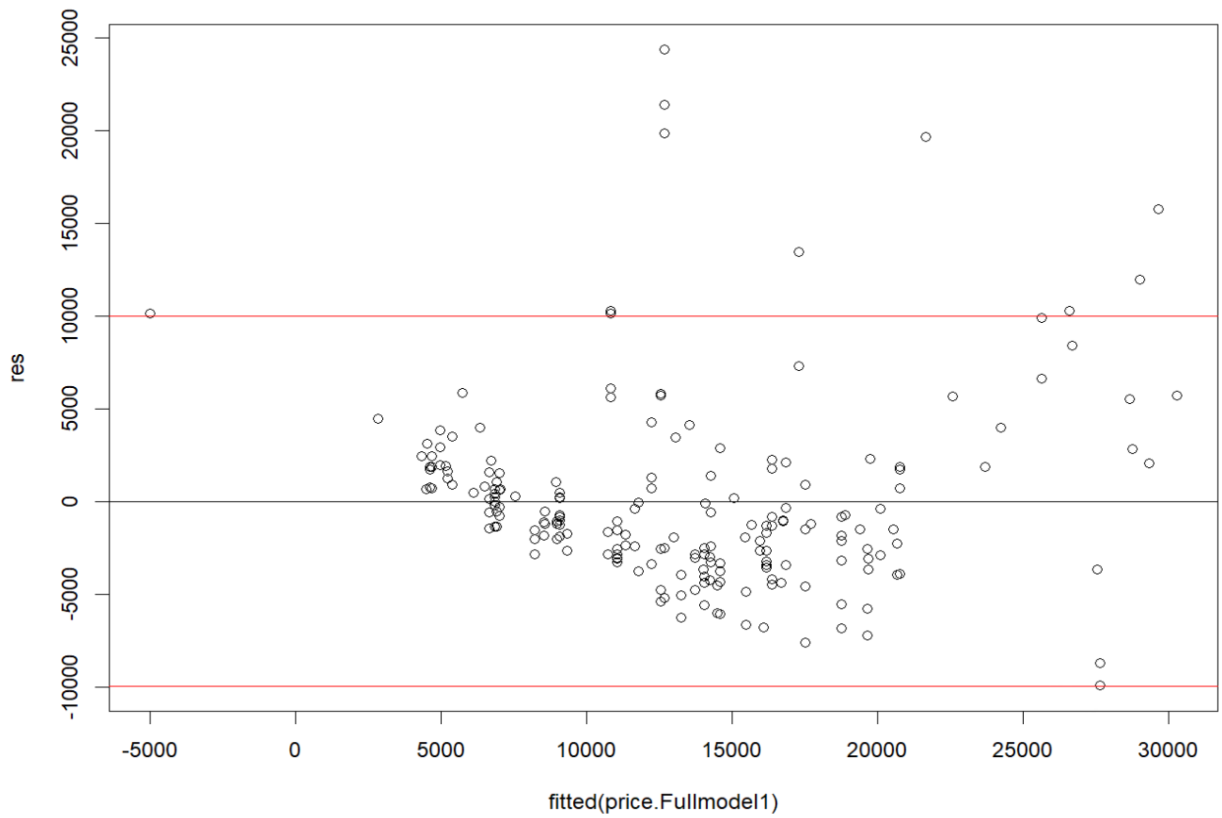
data: price.Fullmodel1
BP = 17.229, df = 4, p-value = 0.001745

```
> #Constant variance - Breusch Pagan Test (BP test)
> bptest(price.Fullmodel1)

	studentized Breusch-Pagan test

data:  price.Fullmodel1
BP = 17.229, df = 4, p-value = 0.001745
```

As the p-value is less than alpha, we reject the null hypothesis. This means the data has non-constant variance.

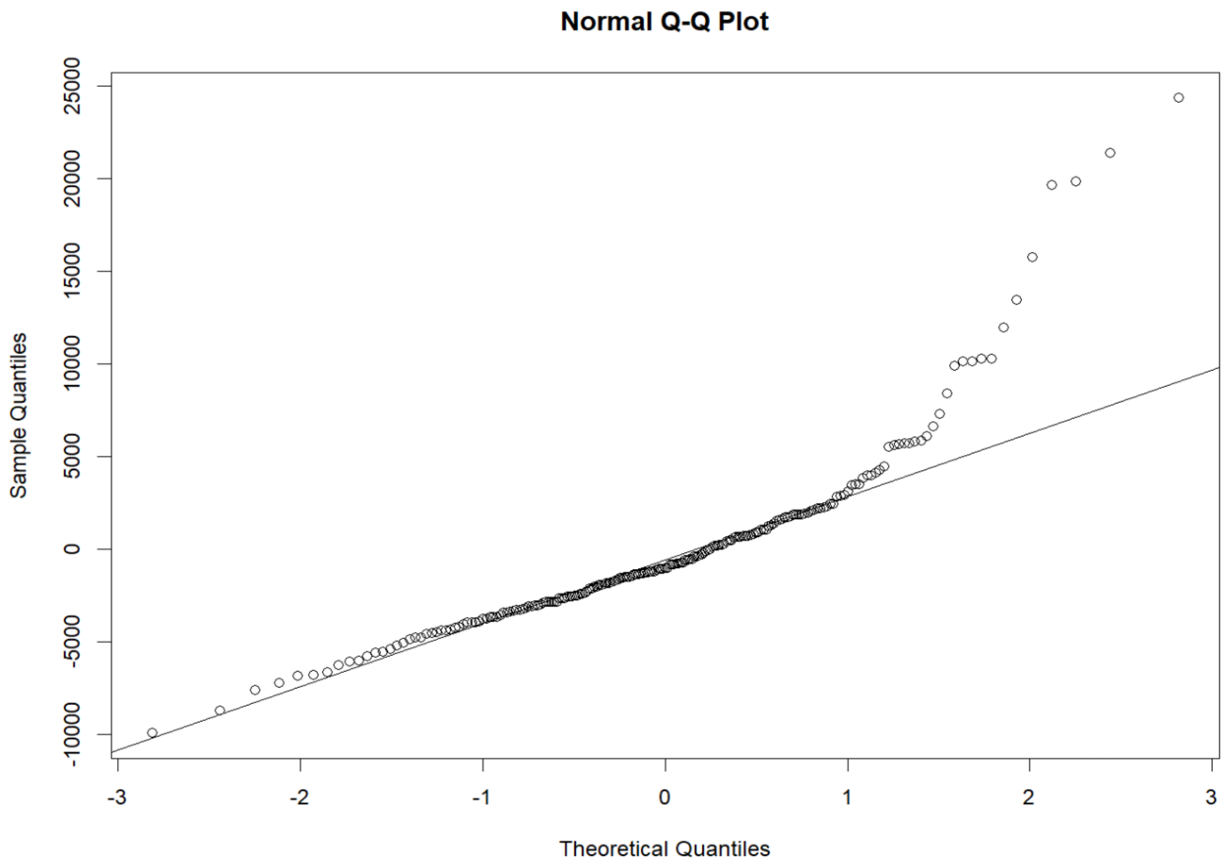
Residuals against fitted values:

```
> #Residuals against fitted values:
> res<- resid(price.Fullmodel1)
> plot(fitted(price.Fullmodel1), res)
> abline(0,0)
> residual_sd <- sd(resid(price.Fullmodel1))
> upper_bound <- 2 * residual_sd
> lower_bound <- -2 * residual_sd
> abline(h = upper_bound, col="red", linetype = "dashed")
> abline(h = lower_bound, col="red", linetype = "dashed")
```



5. Normality test

```
> shapiro.test(residuals(price.Fullmodel1))

	Shapiro-Wilk normality test

data:  residuals(price.Fullmodel1)
W = 0.84822, p-value = 2.295e-13
```



Based on the QQ plot and the Shapiro test, the data violates normality.

It can be seen that data is not normal and does not have constant variance based of the plot and the two tests.

Summary

Linear relationship:

There appears to be a decent linear relationship. The residual plot seems to have a random scatter, with no identifiable pattern. The R-squared value is decently high in the model summary.

Constant variance:

In terms of constant variance, the low p-value of the BP test means we reject the null hypothesis: the test indicates the residuals have non-constant variances.

Normal errors:

There appear to be non-normal errors. The low p-value from the Shapiro-Wilk test means we reject the null hypothesis: The test indicates non-normal errors. Furthermore, the points in the QQplot seem to deviate from the line especially at the tails, further indicating non-normality. Additionally, in the residual plot, there are many points that fall outside the 2 residual standard deviation distance from 0, indicating non-normality.

Outliers:

It can be seen that row 128 is a Y outlier whereas the hat_outliers have an X outlier at row 73 and row 130. No influential points were found.

Influential Points:

Cook's distance indicates no influential points on all fitted values.

Marginal Effect of Predictor Variables:

All the predictors seem to be accurately represented by the fitted regression function besides wheelbase.

Multicollinearity:

All VIF factors being under 10 indicates that there is no multicollinearity among variables.

To adjust for this we utilize the Boxcox transformation,

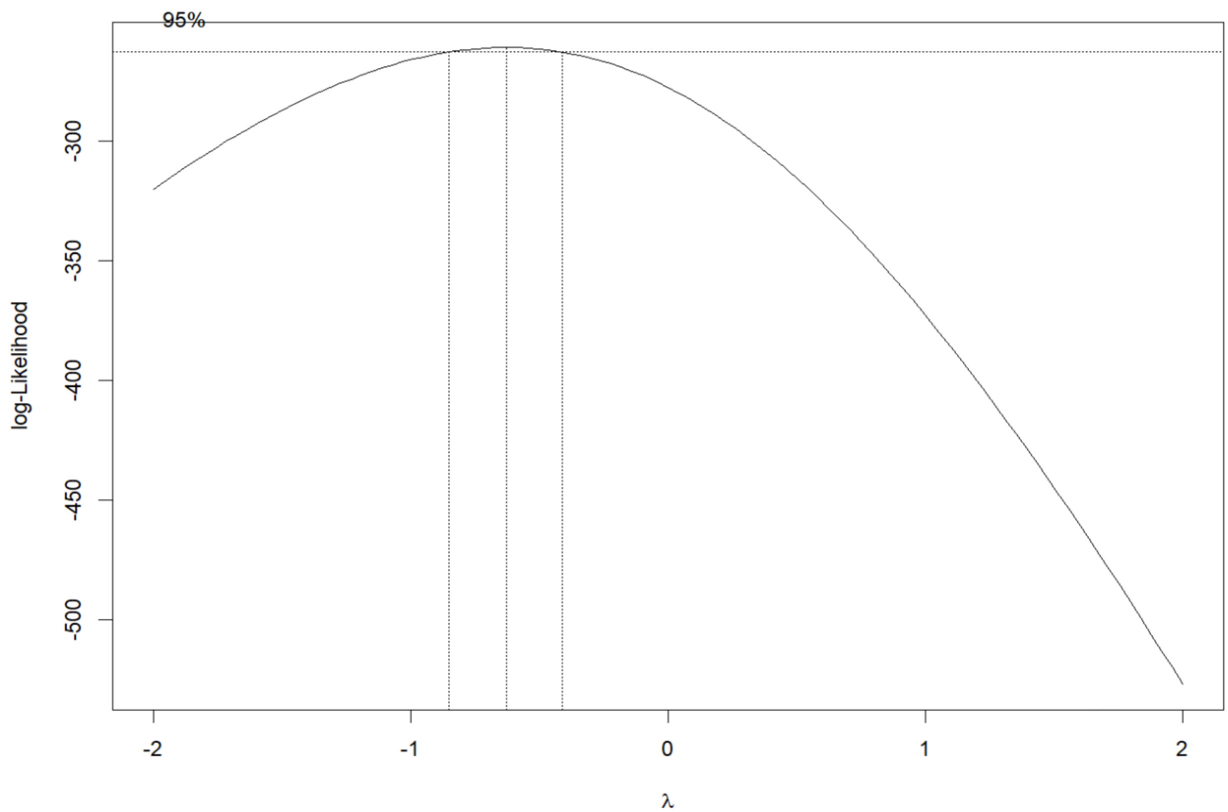
Remedies:

Box Cox transformation:

Since data is not normal, we try to transform Y through Box Cox transformation.

```
> bcmle <- boxcox(price.Fullmodel1, lambda = seq(-3,3, by=0.1))
> lambda <- bcmle$x[which.max(bcmle$y)]
> lambda
[1] -0.6363636
```

The best
λ=-0.6363636 $(biggest\ log-likelihood)$



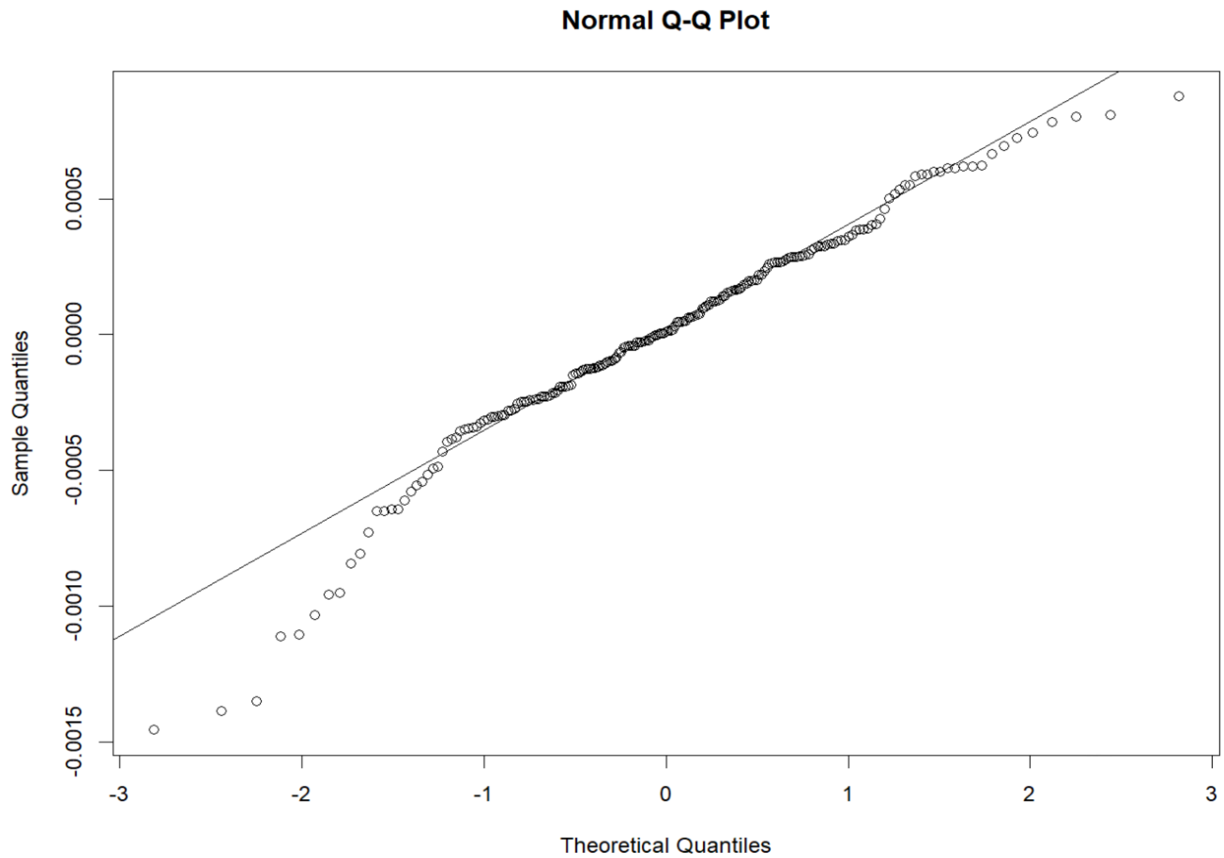

```
> CarPriceData$price_transformed <- (CarPriceData$price^lambda - 1) / lambda
> proj_model_bc = lm(formula = price_transformed ~ carlength+carwidth+carheig
ht+wheelbase, data=CarPriceData)
> summary(proj_model_bc)

Call:
lm(formula = price_transformed ~ carlength + carwidth + carheight +
    wheelbase, data = CarPriceData)

Residuals:
       Min         1Q     Median         3Q        Max
-1.379e-03 -4.445e-04 -1.914e-05  3.575e-04  2.286e-03

Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept)  1.547e+00  2.268e-03 682.071  < 2e-16 ***
carlength    7.353e-05  8.960e-06   8.206 2.76e-14 ***
carwidth     2.487e-04  4.459e-05   5.578 7.84e-08 ***
carheight   -7.248e-05  2.558e-05  -2.834  0.00507 **
wheelbase   -5.401e-05  1.836e-05  -2.942  0.00364 **
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.0006563 on 200 degrees of freedom
Multiple R-squared:  0.727,    Adjusted R-squared:  0.7215
F-statistic: 133.1 on 4 and 200 DF,  p-value: < 2.2e-16


> anova(proj_model_bc)
Analysis of Variance Table
```

```
Response: price_transformed
           Df     Sum Sq    Mean Sq  F value    Pr(>F)    
carlength   1 1.9771e-04 1.9771e-04 459.0598 < 2.2e-16 ***
carwidth    1 1.7946e-05 1.7946e-05  41.6695 8.010e-10 ***
carheight   1 9.9480e-06 9.9480e-06  23.0986 3.018e-06 ***
wheelbase   1 3.7280e-06 3.7280e-06   8.6569  0.003643 ** 
Residuals 200 8.6137e-05 4.3100e-07                       
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

From comparing the transformed model and the original model, we can see that the transformed model is a better fit based on R-squared and Adj. R-squared. Nevertheless, all the diagnostic tests must be performed to confirm that the model does not violate any assumptions.

```
> #Constant variance - Breusch Pagan Test (BP test)
> bptest(proj_model_bc)

	studentized Breusch-Pagan test

data:  proj_model_bc
BP = 9.2823, df = 4, p-value = 0.05442
```

```
> #Normality test
> shapiro.test(residuals(proj_model_bc))

	Shapiro-Wilk normality test

data:  residuals(proj_model_bc)
W = 0.9645, p-value = 5.014e-05

> qqnorm(residuals(proj_model_bc))
> qqline(residuals(proj_model_bc))
```



The boxcox transformation seems to have improved the constant variance but not normality issues as per the BP test and Shapiro test. Rechecked all the assumptions and found few outliers in X variables but not on Y. There are no influential points as well. Robust analysis can be attempted to try and resolve the normality issues.

Robust Analysis

- Attempting to resolve the normality issues

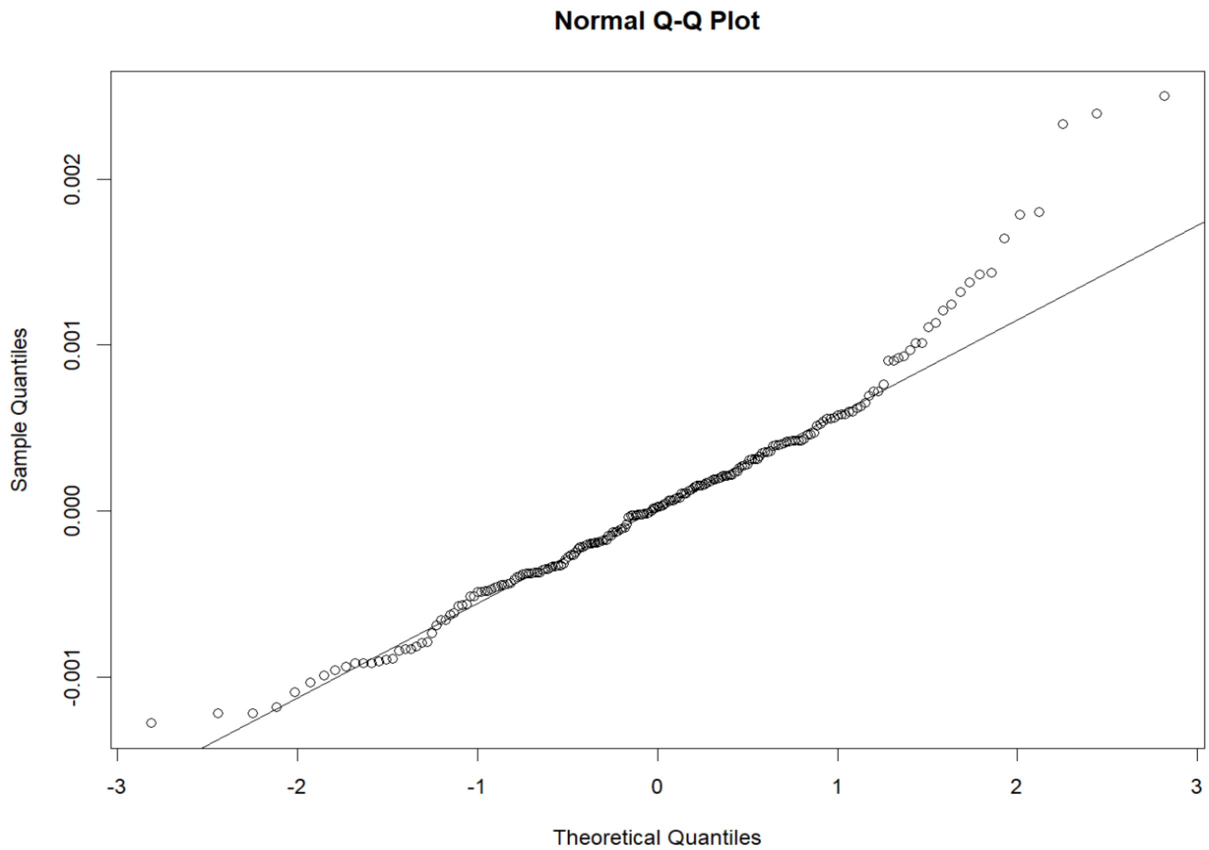
```
> Price.rob = rlm(price_transformed ~ carlength+carheight+carwidth+wheelbase,
data=CarPriceData, psi = psi.bisquare)
> summary(Price.rob)

Call: rlm(formula = price_transformed ~ carlength + carwidth + carheight +
    wheelbase, data = CarPriceData, psi = psi.bisquare)
Residuals:
       Min         1Q     Median         3Q        Max
-1.278e-03 -3.696e-04  2.597e-05  3.986e-04  2.501e-03

Coefficients:
            Value    Std. Error t value
(Intercept)   1.5457   0.0021   737.6155
carlength     0.0001   0.0000     7.5814
carwidth      0.0003   0.0000     6.4690
```

```
carheight    -0.0001   0.0000    -3.0405
wheelbase     0.0000   0.0000    -1.9344

Residual standard error: 0.0005588 on 200 degrees of freedom

> anova(Price.rob)
Analysis of Variance Table

Response: price_transformed
          Df     Sum Sq    Mean Sq F value Pr(>F)
carlength  1 1.8691e-04 1.8691e-04
carwidth   1 2.0622e-05 2.0622e-05
carheight  1 6.4510e-06 6.4510e-06
wheelbase  1 1.1280e-06 1.1280e-06
Residuals    8.7821e-05
```

Rechecking of Assumptions

Shapiro Test:

```
> #transformation Normality test
> shapiro.test(residuals(Price.rob))

        Shapiro-Wilk normality test

data:  residuals(Price.rob)
W = 0.9574, p-value = 8.179e-06
```



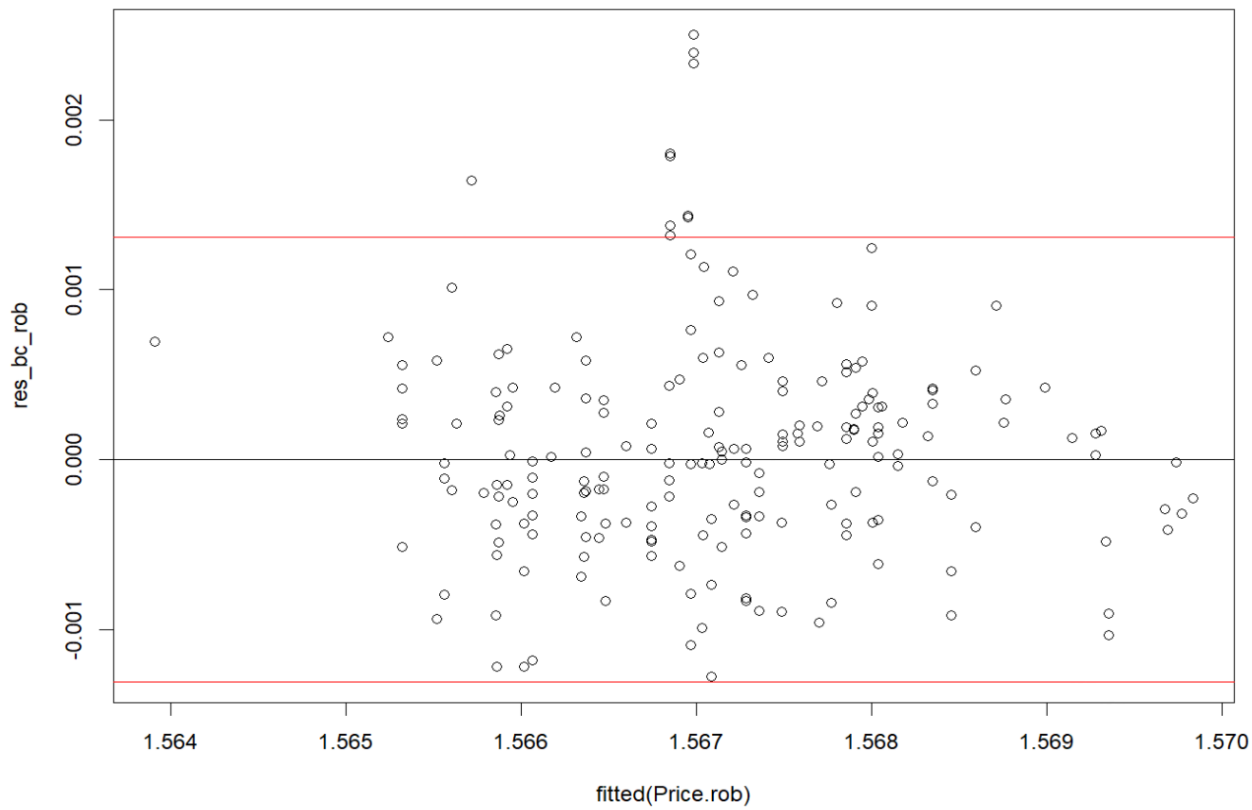
Since this did not work, we note that the normal data is not in this model. We approach with caution. Looking towards the marginal effect, constant variance and multicollinearity:

BP Test:

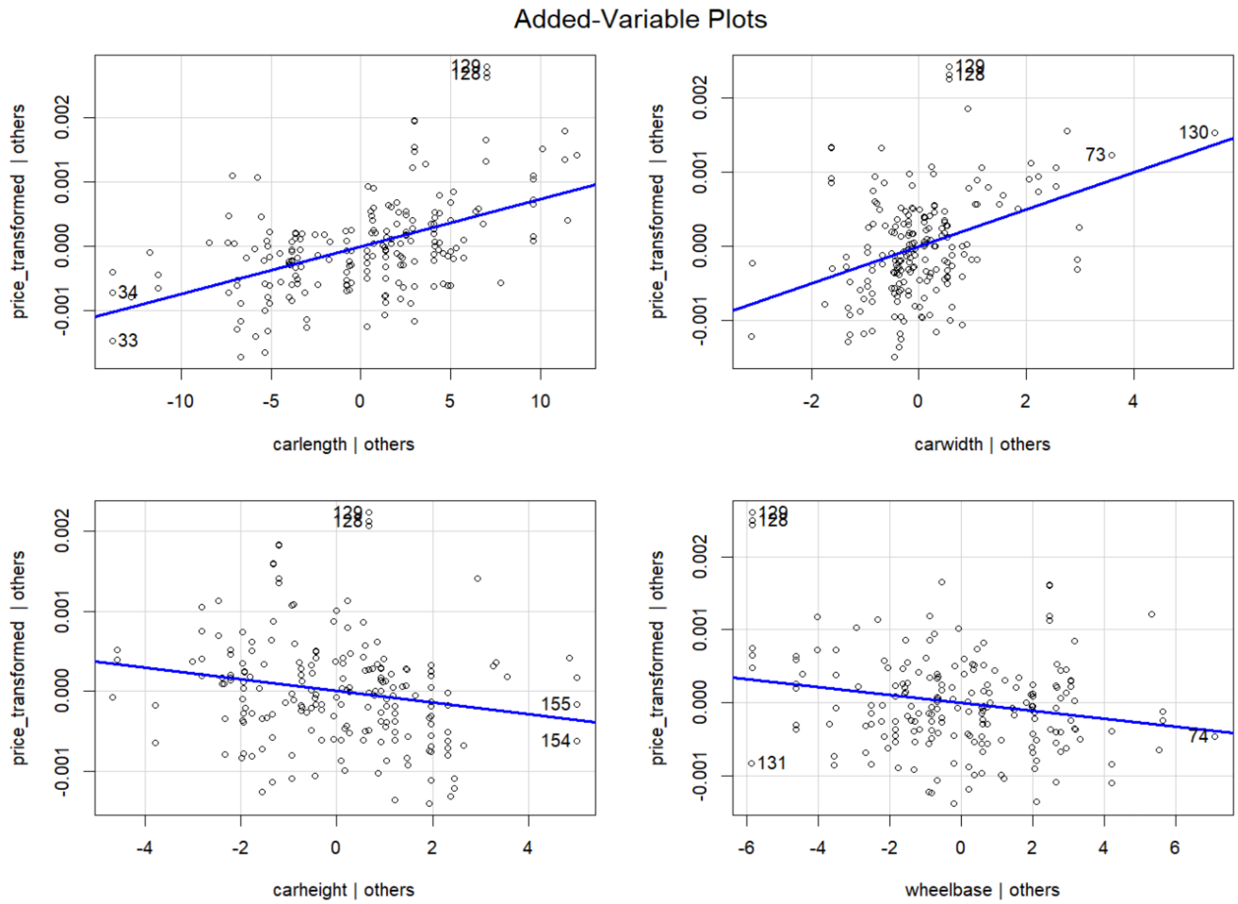
```
> bptest(Price.rob)

	studentized Breusch-Pagan test

data:  Price.rob
BP = 9.2823, df = 4, p-value = 0.05442
```

This shows that transformation has improved the constant variance.

```
#Residuals against fitted values:
> res_bc_rob<- resid(Price.rob)
> plot(fitted(Price.rob), res_bc_rob)
> abline(0,0)
> residual_sd <- sd(resid(Price.rob))
> upper_bound <- 2 * residual_sd
> lower_bound <- -2 * residual_sd
> abline(h = upper_bound, col="red", linetype = "dashed")
> abline(h = lower_bound, col="red", linetype = "dashed")
```



res_bc_rob
0.002
0.001
0.000
-0.001
1.564
1.565
1.566
1.567
1.568
1.569
1.570
fitted(Price.rob)



It can be seen that all predictors seem to have a add on effect given the others influence. This suggests that we should look into what are the interactions between them. This seems like all the predictors are relevant.

```
print(correlation_matrix)
           carlength  carwidth carheight wheelbase
carlength 1.0000000 0.8411183 0.4910295 0.8745875
carwidth  0.8411183 1.0000000 0.2792103 0.7951436
carheight 0.4910295 0.2792103 1.0000000 0.5894348
wheelbase 0.8745875 0.7951436 0.5894348 1.0000000

> vif <- vif(lm(price_transformed ~ carlength  + carwidth + carheight + wheel
base, data = CarPriceData))
> vif
carlength  carwidth carheight wheelbase
 5.788260  4.334334  1.850080  5.788509
```

Utilizing 10 as the base, it seems there is NO multicollinearity presence. This makes sense with the literature review as it seems that all of them do not interact with one another in impacting the body size of the car.

```
> predictor_variables <- CarAssign[c("carlength", "carwidth", "carheight",
"wheelbase")]
> correlation_matrix <- cor(predictor_variables)
```

> print(correlation_matrix)
carlength carwidth carheight wheelbase
carlength 1.0000000 0.8411183 0.4910295 0.8745875
carwidth 0.8411183 1.0000000 0.2792103 0.7951436
carheight 0.4910295 0.2792103 1.0000000 0.5894348

wheelbase 0.8745875 0.7951436 0.5894348 1.0000000

```
# Bootstrap the data
> transformedModel.boot <- boot(data = CarPriceData, statistic = boot.robustCoef, R = 100
maxit = 100)
There were 50 or more warnings (use warnings() to see the first 50)
> transformedModel.boot

ORDINARY NONPARAMETRIC BOOTSTRAP


Call:
boot(data = CarPriceData, statistic = boot.robustCoef, R = 100,
    maxit = 100)


Bootstrap Statistics :
         original        bias     std. error
t1*  1.547186e+00  9.199217e-05 2.282408e-03
t2*  7.352555e-05  4.600625e-07 9.100934e-06
t3*  2.487229e-04 -1.003272e-06 4.822244e-05
t4* -7.247894e-05 -2.384152e-06 2.162505e-05
t5* -5.401373e-05  1.798369e-07 2.408131e-05
> # 95% confidence intervals
> boot.ci(transformedModel.boot, type="perc", index=1)
BOOTSTRAP CONFIDENCE INTERVAL CALCULATIONS
Based on 100 bootstrap replicates

CALL :
boot.ci(boot.out = transformedModel.boot, type = "perc", index = 1)

Intervals :
Level     Percentile
95%   ( 1.543,  1.552 )
Calculations and Intervals on Original Scale
Some percentile intervals may be unstable
> boot.ci(transformedModel.boot, type="perc", index=2)
BOOTSTRAP CONFIDENCE INTERVAL CALCULATIONS
Based on 100 bootstrap replicates

CALL :
boot.ci(boot.out = transformedModel.boot, type = "perc", index = 2)

Intervals :
Level     Percentile
95%   ( 0.0001,  0.0001 )
Calculations and Intervals on Original Scale
Some percentile intervals may be unstable
> boot.ci(transformedModel.boot, type="perc", index=3)
BOOTSTRAP CONFIDENCE INTERVAL CALCULATIONS
Based on 100 bootstrap replicates

CALL :
boot.ci(boot.out = transformedModel.boot, type = "perc", index = 3)
```

```
Intervals :
Level     Percentile
95%   ( 0.0001,  0.0003 )
Calculations and Intervals on Original Scale
Some percentile intervals may be unstable
> boot.ci(transformedModel.boot, type="perc", index=4)
BOOTSTRAP CONFIDENCE INTERVAL CALCULATIONS
Based on 100 bootstrap replicates

CALL :
boot.ci(boot.out = transformedModel.boot, type = "perc", index = 4)

Intervals :
Level     Percentile
95%   (-0.0001,  0.0000 )
Calculations and Intervals on Original Scale
Some percentile intervals may be unstable
> boot.ci(transformedModel.boot, type="perc", index=5)
BOOTSTRAP CONFIDENCE INTERVAL CALCULATIONS
Based on 100 bootstrap replicates

CALL :
boot.ci(boot.out = transformedModel.boot, type = "perc", index = 5)

Intervals :
Level     Percentile
95%   (-0.0001,  0.0000 )
Calculations and Intervals on Original Scale
Some percentile intervals may be unstable
```

It can be seen that 0 is not included in any of the intervals, for the transformed price it can be said that the model is significant when analyzing the price.

If 0 falls outside of the confidence interval the model is significant.

However we further evaluate this by examining the actual price. When back transforming back to the actual price, we can see that

```
> c((1/1.543)^(1/lambda), (1/1.551)^(1/lambda))
[1] 1.976986 1.993117
> c((1/0.0001)^(1/lambda), (1/0.0001)^(1/lambda))
[1] 5.179475e-07 5.179475e-07
> c((1/0.0002)^(1/lambda), (1/0.0003)^(1/lambda))
[1] 1.539334e-06 2.911037e-06
> -1*c((1/0.0001)^(1/lambda), (1/0.0000)^(1/lambda))
[1] -5.179475e-07  0.000000e+00
> -1*c((1/0.0001)^(1/lambda), (1/0.0000)^(1/lambda))
[1] -5.179475e-07  0.000000e+00
```

Model Selection

Best Subset

```
X <- c(11, 12, 13, 10)
> PredCols <- CarPriceData[, X]
> bs <- BestSub(PredCols, CarPriceData$price_transformed, num=1)
> bs
```

```
  p 1 2 3 4         SSEp        r2     r2.adj       Cp      AICp      SBCp       PRESSp
1 2 0 1 0 0 1.161486e-04 0.6318246 0.6300109 68.68292 -2944.648 -2938.002 1.182143e-04
2 3 1 1 0 0 9.981376e-05 0.6836037 0.6804710 32.75549 -2973.719 -2963.750 1.024683e-04
3 4 1 1 0 1 8.959569e-05 0.7159936 0.7117547 11.03037 -2993.859 -2980.567 9.354338e-05
4 5 1 1 1 1 8.613712e-05 0.7269568 0.7214959  5.00000 -2999.929 -2983.314 9.069589e-05
```

The best model based on R^2 , R^2 adj , Cp, AICp, SBCp and PRESSp is the full model.

Stepwise Regression

```
step(proj_model_bc, method="both", trace=1)
Start:  AIC=-2999.93
price_transformed ~ carlength + carwidth + carheight + wheelbase

            Df  Sum of Sq        RSS     AIC
<none>                    8.6137e-05 -2999.9
- carheight  1 3.4586e-06 8.9596e-05 -2993.9
- wheelbase  1 3.7284e-06 8.9866e-05 -2993.2
- carwidth   1 1.3399e-05 9.9536e-05 -2972.3
- carlength  1 2.9000e-05 1.1514e-04 -2942.4

Call:
lm(formula = price_transformed ~ carlength + carwidth + carheight +
    wheelbase, data = CarPriceData)

Coefficients:
(Intercept)    carlength     carwidth    carheight    wheelbase
  1.547e+00    7.353e-05    2.487e-04   -7.248e-05   -5.401e-05
```

We undergo K-fold validation. Given that the stepwise and the best subset all indicate p = 5 is good. We still verify this by comparing p = 5 to p = 4 and p=3. This is done by dropping car height for the model.

K-fold Cross Validation

```
> #K-fold cross validation
> #p=4
> set.seed(123)
> train.control<-trainControl(method="cv", number=10)
> step.model <- train(price_transformed ~ carlength  + carwidth + wheelbase,
+                     data=CarPriceData,method="leapBackward",tuneGrid=
+                     data.frame(nvmax=5),trControl=train.control)
> step.model$results
  nvmax         RMSE  Rsquared          MAE       RMSESD RsquaredSD         MA
ESD
1     5 0.0006641033 0.7183195 0.0005096474 0.0001341656  0.1291041 8.614075e
-05
> #p=3
> set.seed(123)
> train.control<-trainControl(method="cv", number=10)
> step.model <- train(price_transformed ~ carlength  + carwidth ,
+                     data=CarPriceData,method="leapBackward",tuneGrid=
+                       data.frame(nvmax=5),trControl=train.control)
> step.model$results
  nvmax        RMSE Rsquared          MAE       RMSESD RsquaredSD        MAES
D
```

```
1     5 0.000691279 0.692981 0.0005289791 0.0001538415  0.1364719 9.632416e-0
5
> #p=5
> set.seed(123)
> train.control<-trainControl(method="cv", number=10)
> step.model <- train(price_transformed ~ carlength  + carwidth + carheight+w
heelbase,
+                      data=CarPriceData,method="leapBackward",tuneGrid=
+                        data.frame(nvmax=5),trControl=train.control)
> step.model$results
  nvmax         RMSE  Rsquared          MAE       RMSESD RsquaredSD        MA
ESD
1     5 0.0006514671 0.7244587 0.0004993885 0.0001425856  0.1281243 9.577558e
-05
```

Given that the design with all the current predictors is the best. We bootstrap to reveal the GLT and the T test in an non-normal environment.

```
> # Anova Bootstrap
> anova_test <- boot(data = transformedModel.boot$t, statistic = function(data, indices) {
+   model_full <- lm(price_transformed ~ 1, data = CarPriceData[indices, ])
+   model_reduced <- lm(price_transformed ~ carlength + carwidth + wheelbase, data = CarPrice
Data[indices, ])
+   return(anova(model_reduced, model_full)$"Pr(>F)"[2])
+ }, R = 100)
> print(anova_test)

ORDINARY NONPARAMETRIC BOOTSTRAP


Call:
boot(data = transformedModel.boot$t, statistic = function(data, 
    indices) {
    model_full <- lm(price_transformed ~ 1, data = CarPriceData[indices, 
        ])
    model_reduced <- lm(price_transformed ~ carlength + carwidth + 
        wheelbase, data = CarPriceData[indices, ])
    return(anova(model_reduced, model_full)$"Pr(>F)"[2])
}, R = 100)


Bootstrap Statistics :
        original       bias     std. error
t1* 5.391232e-30 4.097439e-25 2.611266e-24
```

This shows that the model is valid from the ANOVA. From the F test in ANOVA output, we see that at least one of the predictors has a non-zero coefficient, indicating a significant impact on the transformed price of a car.

If the MSE is very small you could still run a GLT test even with non-normal residuals.

```
> # Anova Bootstrap
> anova_test <- boot(data = transformedModel.boot$t, statistic = function(dat
a, indices) {
+   model_reduced<- lm(price_transformed ~ 1, data = CarPriceData[indices, ])
+ model_full <- lm(price_transformed ~ carlength + carwidth + carheight+ whee
lbase, data = CarPriceData[indices, ])
+   return(anova(model_reduced, model_full)$"Pr(>F)"[2])
+ }, R = 100)
> print(anova_test)

ORDINARY NONPARAMETRIC BOOTSTRAP


Call:
boot(data = transformedModel.boot$t, statistic = function(data, 
    indices) {
    model_reduced <- lm(price_transformed ~ 1, data = CarPriceData[indices, 
        ])
    model_full <- lm(price_transformed ~ carlength + carwidth + 
        carheight + wheelbase, data = CarPriceData[indices, ])
    return(anova(model_reduced, model_full)$"Pr(>F)"[2])
}, R = 100)


Bootstrap Statistics :
        original      bias     std. error
t1* 6.702779e-31 1.48298e-27 6.431244e-27
> #Bootstrapped GLT for individual p-values
> glt <- function(full_model, reduced_model) {
+   glt_result <- anova(full_model, reduced_model)
+   glt_result_p_value <- glt_result$"Pr(>F)"[2]
+   return(glt_result_p_value)
+ }
> full_model <- lm(price_transformed ~ carlength + carwidth + carheight + whe
elbase, data = CarPriceData)
> reduced_models <- lapply(1:4, function(i) {
+   formula <- as.formula(paste("price_transformed ~", paste(c("carlength", "
carwidth", "carheight", "wheelbase")[-i], collapse = "+")))
+   lm(formula, data = CarPriceData)
+ })
> glt_result_p_values <- sapply(reduced_models, function(reduced_model) glt(f
ull_model, reduced_model))
> print(glt_result_p_values)
[1] 2.764502e-14 7.841013e-08 5.071547e-03 3.643091e-03
```

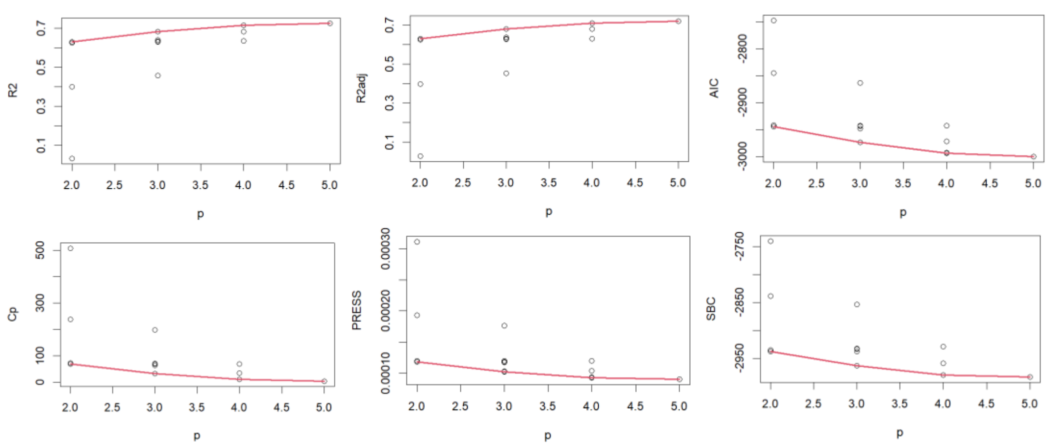
This code shows that all the variables are significant from the GLT.
From the low p-values, we reject the null hypothesis. At least one of the predictors in the final model has a significant impact on the transformed price of a car.

Conclusions

I have developed a model that accurately represents the transformed price of a car given its length, height, width, and wheelbase. The final GLT concludes that the model is significant. That means of the variables in the final model (carlength, carheight, carwidth and wheelbase), at least one of them has a significant impact on car price.

Note on Back Transformations

Since only Y was transformed, notably Y^-0.6363636, this final model predicts the price of a car to the negative 0. 6363636. To interpret the true price of a car, simply do (1/Y_transformed)^(1/0. 6363636) where Y_transformed would be the value of Y obtained from this model.



<u>SUMMARY</u>

Research question 4 aims to ascertain if a car's size-related attributes such as car length, car width, car height, and wheelbase significantly impact the price of the car.

EXPLANATORY VARIABLES	RESPONSE VARIABLE
X1: Car length (Continuous)	Y: Price (Continuous)
X2: Car Width (Continuous)	
X3: Car Height (Continuous)	
X4: Wheelbase (Continuous)	

Understanding how size-related attributes influence car prices is vital for understanding pricing dynamics in the automotive industry. Car size plays a crucial role in shaping consumer preferences, driving market demand, and influencing pricing strategies. Consumer preferences vary based on factors like lifestyle, perceived value, and prestige associated with different car sizes. Manufacturers leverage these insights to segment the market efficiently, target specific consumer segments, differentiate their offerings from competitors, and allocate resources towards developing models that match market demand and pricing expectations. Moreover, technological advancements and innovation often lead to the introduction of new car sizes optimized for emerging trends such as electric drivetrains or autonomous driving features, which further impact pricing dynamics in the automotive market.

HYPOTHESIS

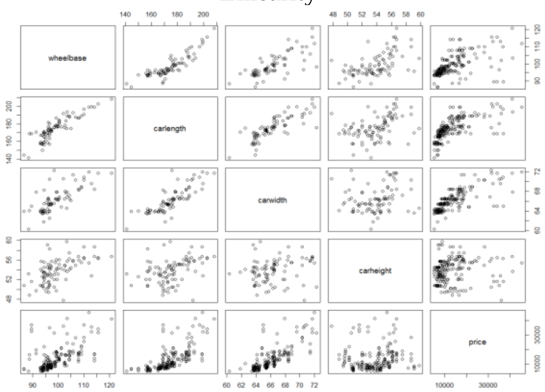
Do car length (X1), car width (X2), car height (X3), and wheelbase (X4) significantly impact the price of the car?

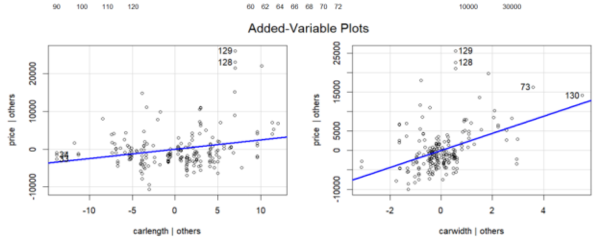
Reduced Model	*Full Model*
$H_o: \beta_1 = \beta_2 = \beta_3 = \beta_4 = 0$	$H_a: \beta_1 \neq \beta_2 \neq \beta_3 \neq \beta_4 \neq 0$
$Y = \beta_0 + \epsilon$	$Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \beta_3 X_3 + \beta_4 X_4 + \epsilon$
$dfE\ (Reduced) = n - p = 204$	$dfE\ (Full) = n - p = 200$

DIAGNOSTICS ON ORIGINAL MODEL

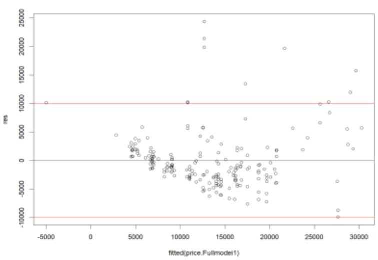
The basic assumptions for a multiple linear regression were tested and the results are shown in the table below along with the corresponding R ouput.

Linearity



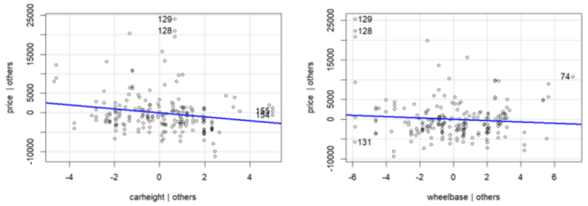


Error Variance



```
studentized Breusch-Pagan test

data:  price.Fullmodel1
BP = 17.229, df = 4, p-value = 0.001745
```

Breusch-Pagan test conducted shows that there is a **Violation of the Constant Error Variance assumption** In the residual plot, many points fall outside the 2 residual standard deviation distance from 0, indicating non-normality. From scatter plot, it's evident that there is increased correlation, particularly between wheelbase, carlength, and carwidth.



The added variable plots show a reasonable linear association for each of the factors. Wheelbase shows less add-on effect

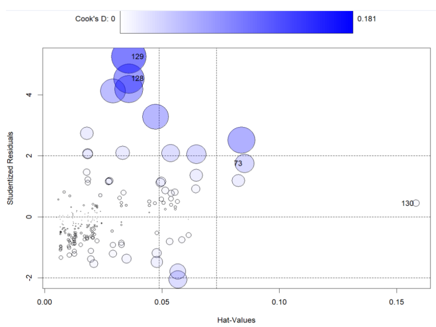
No Linearity issues

Variance Inflation Factors (VIF)

```
carlength  carwidth carheight wheelbase
 5.788260  4.334334  1.850080  5.788509
```

VIF values for each factor are below 10.

No multicollinearity issue.

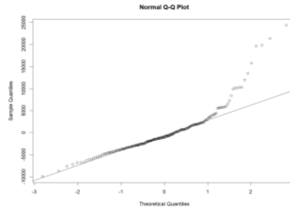


There are studentized residuals that are greater than the threshold, indicating that there are i.e. **Ouliers on Y**.

Few Hat values are greater than $2p/n = 0.04878049$ indicating that there are **Ouliers on X.**

```
> summary(price.Fullmodel1)

Call:
lm(formula = price ~ carlength + carwidth + carheight + wheelbase,
    data = CarPriceData)

Residuals:
   Min     1Q Median     3Q    Max
 -9932  -2902  -1028   1718  24364

Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept) -129461.13   17398.76  -7.441 2.90e-12 ***
carlength       243.05      68.73   3.536 0.000504 ***
carwidth       2186.58     342.03   6.393 1.12e-09 ***
carheight      -505.06     196.18  -2.574 0.010761 *
wheelbase      -167.52     140.81  -1.190 0.235583
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 5034 on 200 degrees of freedom
Multiple R-squared:  0.6108,    Adjusted R-squared:  0.603
F-statistic: 78.46 on 4 and 200 DF,  p-value: < 2.2e-16

> anova(price.Fullmodel1)
Analysis of Variance Table

Response: price
           Df     Sum Sq    Mean Sq  F value    Pr(>F)
carlength   1 6072096122 6072096122 239.6443 < 2.2e-16 ***
carwidth    1 1521803074 1521803074  60.0602 4.548e-13 ***
carheight   1  322287703  322287703  12.7196 0.0004528 ***
wheelbase   1   35861642   35861642   1.4153 0.2355835
Residuals 200 5067590820   25337954
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Normality



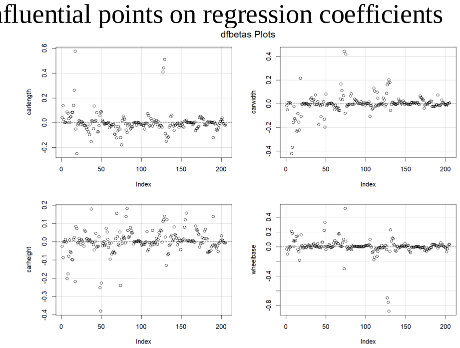
From the plot above we can see that the residuals are not normally distributed.

```
        Shapiro-Wilk normality test

data:  residuals(price.Fullmodel1)
W = 0.84822, p-value = 2.295e-13
```

Shapiro-Wilk test confirms that there is a

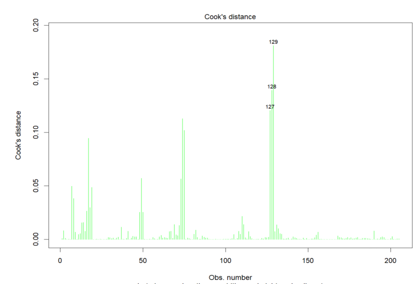
Violation of Normality assumption

Influential points on regression coefficients



$|\text{〖DFBETAS〗}_k| > 1$. **Influential points on β**

Influential points on fitted values – Cooks D



50th Percentile: $F((0.5; p, n-p)) = 0.873239$

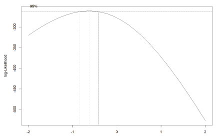
$D_i < "0.873239"$

No influential cases

Note: The points in the QQplot seem to deviate from the line especially at the tails, further indicating non-normality. The residual plot seems to have a random scatter, with no identifiable pattern.
The R-squared value is decently high in the model summary.

INITIAL REMEDIAL MEASURE

Box-cox transformation was carried out to deal with the issue of constant error variance and normality. As seen from the maximum likelihood graph the transformation is chosen to be carried out at $\lambda = -0.6363636$.



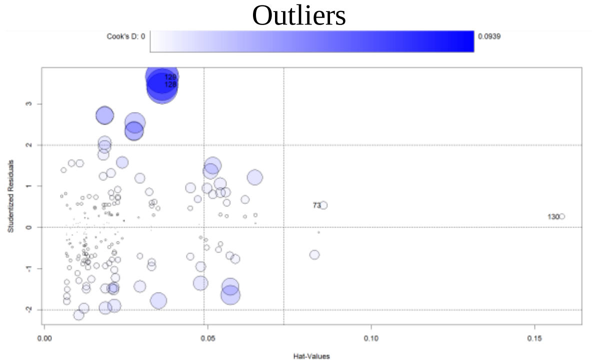
DIAGNOSTICS ON TRANSFORMED MODEL

Error Variance

```
	studentized Breusch-Pagan test

data:  proj_model_bc
BP = 9.2823, df = 4, p-value = 0.05442
```

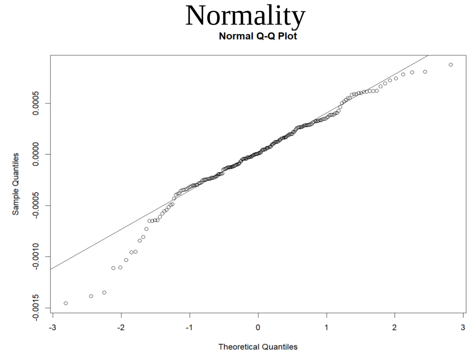
No violation of constant error variance in the transformed model.

Outliers



There are no studentized residuals that are greater than the threshold indicating that there are no outliers on Y. Few Hat values are greater than $2p/n = 0.04878049$. indicating that there are Outliers on X.

Normality




```
	Shapiro-Wilk normality test

data:  residuals(proj_model_bc)
W = 0.9645, p-value = 5.014e-05
```

The Shapiro-Wilkes test shows that there is still a Violation of Normality assumption

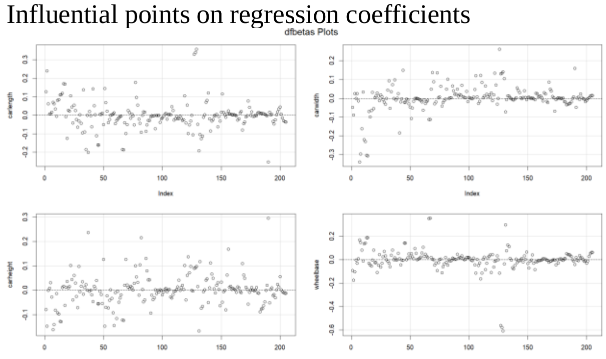
Variance Inflation Factors (VIF)

```
carlength  carwidth carheight wheelbase 
 5.788260  4.334334  1.850080  5.788509
```

VIF values for each factor are below 10.

No multicollinearity issue.

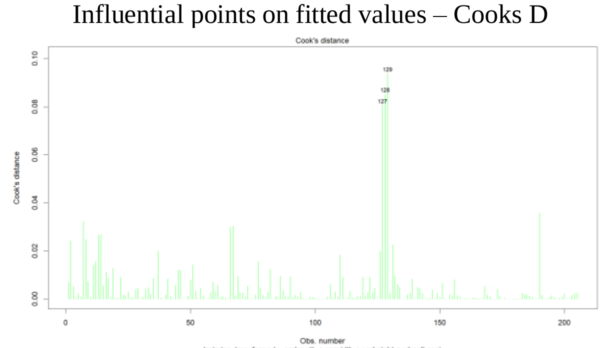
Influential points on regression coefficients



$|DFBETAS_k| > 1$. **Influential points on β**

Note: From comparing the transformed model and the original model, we can see that the transformed model is a better fit based on R-squared and Adj. R-squared.

Influential points on fitted values – Cooks D



Boundaries:

50th Percentile: $F((0.5; p, n-p)) = 0.873239$

$D_i < 0.873239$

No influential cases

ADVANCED REMEDIAL MEASURES

Robust regression and Bootstrap:

Since the transformed model had the presence of outliers and normality issues, robust regression was used to trim the influence of these extreme observations on estimations. The weight function chosen here is bisquare for dampening the influence of outliers based on their residuals. The R output is shown below:

```
Call: rlm(formula = price_transformed ~ carlength + carwidth + carheight +
    wheelbase, data = CarPriceData, psi = psi.bisquare)
Residuals:
       Min         1Q     Median         3Q        Max
-1.278e-03 -3.696e-04  2.597e-05  3.986e-04  2.501e-03

Coefficients:
            Value    Std. Error t value
(Intercept)   1.5457   0.0021   737.6155
carlength     0.0001   0.0000     7.5814
carwidth      0.0003   0.0000     6.4690
carheight    -0.0001   0.0000    -3.0405
wheelbase     0.0000   0.0000    -1.9344

Residual standard error: 0.0005588 on 200 degrees of freedom
Analysis of Variance Table

Response: price_transformed
           Df     Sum Sq    Mean Sq F value Pr(>F)
carlength   1 1.8691e-04 1.8691e-04
carwidth    1 2.0622e-05 2.0622e-05
carheight   1 6.4510e-06 6.4510e-06
wheelbase   1 1.1280e-06 1.1280e-06
Residuals     8.7821e-05
```

Robust regression could not resolve the normality issue. we note that the normal data is not in this model. We approach with caution.

```
        Shapiro-Wilk normality test

data:  residuals(Price.rob)
W = 0.9574, p-value = 8.179e-06
```

Confidence intervals for Betas from Bootstrap on robust
C.I. for β1: [1.543,1.552]
C.I. for β2: [0.0001,0.0001]
C.I. for β3: [0.0002,0.0003]
C.I. for β4: [-0.0001,0.0000]
C.I. for β5: [-0.0001,0.0000]
It can be seen that 0 is not included in any of the intervals and hence can be said that the model is significant when analyzing the price.

Confirmed the above interpretation by back transforming to the actual price, we can see that,

C.I. for β1: 1.976986, 1.993117
C.I. for β2: 5.179475e-07, 5.179475e-07
C.I. for β3:1.539334e-06, 2.911037e-06
C.I. for β4: 5.179475e-07 0.000000e+00
C.I. for β5: 5.179475e-07 0.000000e+00

MODEL SELECTION

Best Subset Algorithm:

```
  p 1 2 3 4         SSEp        r2    r2.adj       Cp      AICp      SBCp       PRESSp
1 2 0 1 0 0 1.161486e-04 0.6318246 0.6300109 68.68292 -2944.648 -2938.002 1.182143e-04
2 3 1 1 0 0 9.981376e-05 0.6836037 0.6804710 32.75549 -2973.719 -2963.750 1.024683e-04
3 4 1 1 0 1 8.959569e-05 0.7159936 0.7117547 11.03037 -2993.859 -2980.567 9.354338e-05
4 5 1 1 1 1 8.613712e-05 0.7269568 0.7214959  5.00000 -2999.929 -2983.314 9.069589e-05
```

K-fold cross-validation:

```
> step.model$results
  nvmax         RMSE  Rsquared          MAE       RMSESD RsquaredSD        MAESD
1     5 0.0006514671 0.7244587 0.0004993885 0.0001425856  0.1281243 9.577558e-05
```

Stepwise Regression:

```
Start:  AIC=-2999.93
price_transformed ~ carlength + carwidth + carheight + wheelbase

            Df  Sum of Sq        RSS     AIC
<none>                    8.6137e-05 -2999.9
- carheight  1 3.4586e-06 8.9596e-05 -2993.9
- wheelbase  1 3.7284e-06 8.9866e-05 -2993.2
- carwidth   1 1.3399e-05 9.9536e-05 -2972.3
- carlength  1 2.9000e-05 1.1514e-04 -2942.4

Call:
lm(formula = price_transformed ~ carlength + carwidth + carheight +
    wheelbase, data = CarPriceData)

Coefficients:
(Intercept)    carlength     carwidth    carheight    wheelbase
  1.547e+00    7.353e-05    2.487e-04   -7.248e-05   -5.401e-05
```

All three methods mentioned above suggested the full model with all 4 predictors as the best model.

We bootstrap to reveal the GLT and the F test in a non-normal environment. The global F-test result shows that the model is valid. In ANOVA output, we see that at least one of the predictors has a non-zero coefficient, indicating a significant impact on the transformed price of a car.

```
Bootstrap Statistics :
        original        bias     std. error
t1* 6.702779e-31 4.321673e-27 2.589884e-26
```

As MSE is very small, a GLT test is performed with non-normal residuals on the final model and confirmed that the model is significant with all 4 predictors.

```
> print(glt_result_p_values)
[1] 2.764502e-14 7.841013e-08 5.071547e-03 3.643091e-03
```

Note on Back Transformations: Since only Y was transformed, notably Y^-0.6363636, this final model predicts the price of a car to the negative 0. 6363636. To interpret the true price of a car, (1/Y_transformed)^(1/0. 6363636) is done where Y_transformed would be the value of Y obtained from this model.

```
> summary(proj_model_bc)

Call:
lm(formula = price_transformed ~ carlength + carwidth + carheight +
    wheelbase, data = CarPriceData)

Residuals:
       Min         1Q     Median         3Q        Max
-1.379e-03 -4.445e-04 -1.914e-05  3.575e-04  2.286e-03

Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept)  1.547e+00  2.268e-03 682.071  < 2e-16 ***
carlength    7.353e-05  8.960e-06   8.206 2.76e-14 ***
carwidth     2.487e-04  4.459e-05   5.578 7.84e-08 ***
carheight   -7.248e-05  2.558e-05  -2.834  0.00507 **
wheelbase   -5.401e-05  1.836e-05  -2.942  0.00364 **
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.0006563 on 200 degrees of freedom
Multiple R-squared:  0.727,	Adjusted R-squared:  0.7215
F-statistic: 133.1 on 4 and 200 DF,  p-value: < 2.2e-16

Analysis of Variance Table

Response: price_transformed
           Df     Sum Sq    Mean Sq  F value    Pr(>F)
carlength   1 1.9771e-04 1.9771e-04 459.0598 < 2.2e-16 ***
carwidth    1 1.7946e-05 1.7946e-05  41.6695 8.010e-10 ***
carheight   1 9.9480e-06 9.9480e-06  23.0986 3.018e-06 ***
wheelbase   1 3.7280e-06 3.7280e-06   8.6569  0.003643 **
Residuals 200 8.6137e-05 4.3100e-07
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

CONCLUSION:

I have developed a model that accurately represents the transformed price of a car given its length, height, width, and wheelbase. The final GLT concludes that the model is significant. That means at least one of the predictors in the final model (carlength, carheight, carwidth and wheelbase), has a significant impact on the price of a car.