BIGDATA HW 1

Nandini Ramanan

February 23, 2016

1. **What is your independent variable, what are your dependent variables given this analysis goal?** Quantitative dependent variable is automobile miles per gallon or MPG and multiple independent variables are attributes of the automobile and its engine.

```
Auto = read.csv("C:/Users/Nandini/Documents/Textbooks/Big data/Auto_MPG_data_
2.csv", header=T, na.strings="?")
Auto = na.omit(Auto)
dim(Auto)
```

```
## [1] 392   9
```

```
str(Auto)
```

```
## 'data.frame':    392 obs. of  9 variables:
##  $ mpg         : num  18 15 18 16 17 15 14 14 14 15 ...
##  $ cylinders   : int  8 8 8 8 8 8 8 8 8 8 ...
##  $ displacement: num  307 350 318 304 302 429 454 440 455 390 ...
##  $ horsepower  : int  130 165 150 150 140 198 220 215 225 190 ...
##  $ weight      : int  3504 3693 3436 3433 3449 4341 4354 4312 4425 3850 ..
.
##  $ acceleration: num  12 11.5 11 12 10.5 10 9 8.5 10 8.5 ...
##  $ year        : int  70 70 70 70 70 70 70 70 70 70 ...
##  $ origin      : int  1 1 1 1 1 1 1 1 1 1 ...
##  $ name        : Factor w/ 305 levels "amc ambassador brougham",..: 50 37
232 15 162 142 55 224 242 2 ...
##  - attr(*, "na.action")=Class 'omit'  Named int [1:6] 33 127 331 337 355 3
75
##   .. ..- attr(*, "names")= chr [1:6] "33" "127" "331" "337" ...
```

```
summary(Auto)
```

```
##       mpg          cylinders      displacement     horsepower
##  Min.   : 9.00   Min.   :3.000   Min.   : 68.0   Min.   : 46.0
##  1st Qu.:17.00   1st Qu.:4.000   1st Qu.:105.0   1st Qu.: 75.0
##  Median :22.75   Median :4.000   Median :151.0   Median : 93.5
##  Mean   :23.45   Mean   :5.472   Mean   :194.4   Mean   :104.5
##  3rd Qu.:29.00   3rd Qu.:8.000   3rd Qu.:275.8   3rd Qu.:126.0
##  Max.   :46.60   Max.   :8.000   Max.   :455.0   Max.   :230.0
##
##      weight      acceleration        year           origin
##  Min.   :1613   Min.   : 8.00   Min.   :70.00   Min.   :1.000
##  1st Qu.:2225   1st Qu.:13.78   1st Qu.:73.00   1st Qu.:1.000  
##  Median :2804   Median :15.50   Median :76.00   Median :1.000  
##  Mean   :2978   Mean   :15.54   Mean   :75.98   Mean   :1.577  
##  3rd Qu.:3615   3rd Qu.:17.02   3rd Qu.:79.00   3rd Qu.:2.000  
##  Max.   :5140   Max.   :24.80   Max.   :82.00   Max.   :3.000  
##                                                                
##                  name    
##  amc matador       :  5  
##  ford pinto        :  5  
##  toyota corolla    :  5  
##  amc gremlin       :  4  
##  amc hornet        :  4  
##  chevrolet chevette:  4  
##  (Other)           :365  
```

2. **Describe the data by reporting means and standard deviation of each variable; plot pairs of variables (in a plot matrix) and report observations from the plot.**

```
sapply(Auto[, 1:7], range)
```

```
##       mpg cylinders displacement horsepower weight acceleration year
## [1,]  9.0         3           68         46   1613          8.0   70
## [2,] 46.6         8          455        230   5140         24.8   82
```

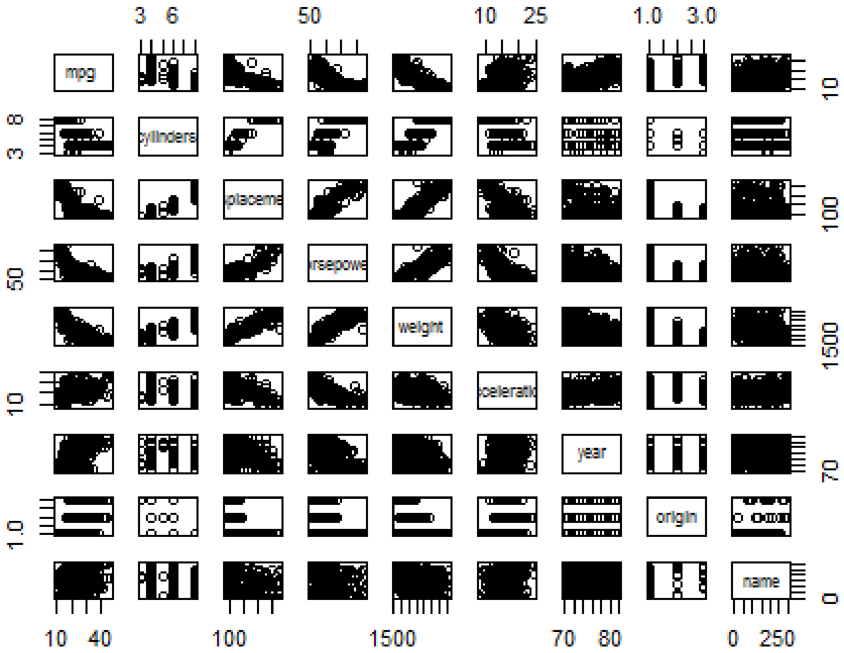
```
sapply(Auto[, 1:7], mean)
```

```
##          mpg    cylinders displacement   horsepower       weight 
##    23.445918     5.471939   194.411990   104.469388  2977.584184 
## acceleration         year 
##    15.541327    75.979592 
```

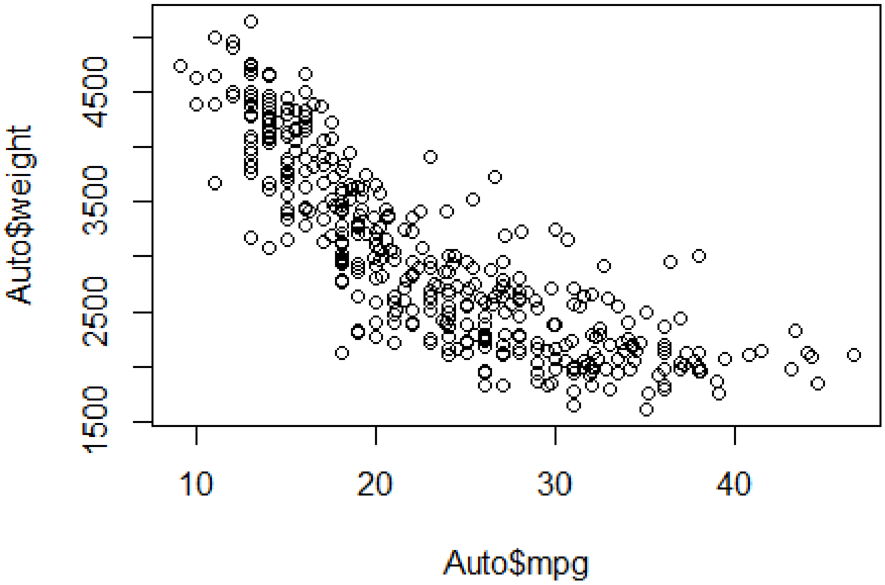
```
sapply(Auto[, 1:7], sd)
```

```
##          mpg    cylinders displacement   horsepower       weight 
##     7.805007     1.705783   104.644004    38.491160   849.402560 
## acceleration         year 
##     2.758864     3.683737 
```

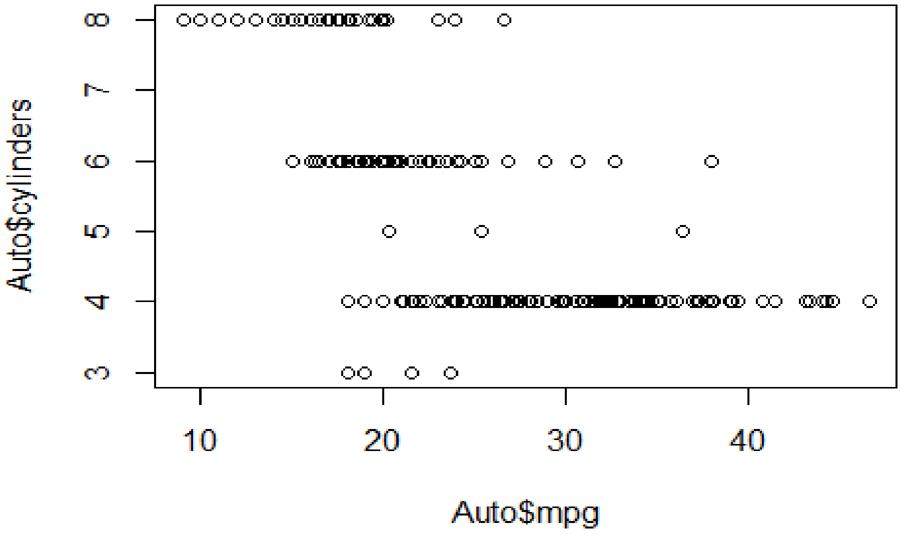
```
pairs(Auto)
```



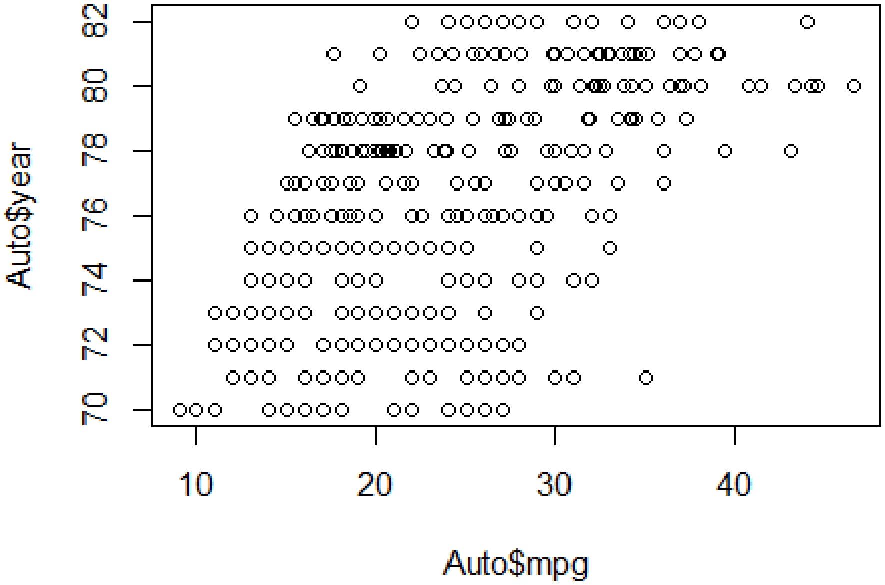
```r
# Heavier weight correlates with lower mpg.
plot(Auto$mpg, Auto$weight)
```



```r
# More cylinders, less mpg.
plot(Auto$mpg, Auto$cylinders)
```



```r
# Cars become more efficient over time.

plot(Auto$mpg, Auto$year)
```



```
#Weight, displacement and horsepower seem to have an inverse effect with mpg
#correlation matrix
cor(Auto[,1:7])
```

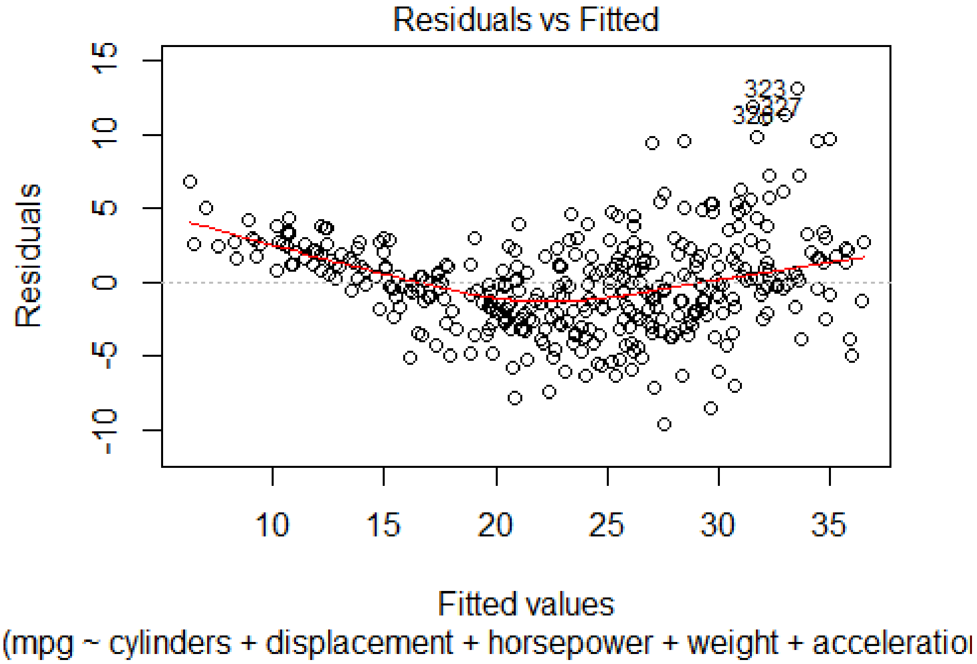
```
##                     mpg  cylinders displacement horsepower     weight
## mpg           1.0000000 -0.7776175   -0.8051269 -0.7784268 -0.8322442
## cylinders    -0.7776175  1.0000000    0.9508233  0.8429834  0.8975273
## displacement -0.8051269  0.9508233    1.0000000  0.8972570  0.9329944
## horsepower   -0.7784268  0.8429834    0.8972570  1.0000000  0.8645377
## weight       -0.8322442  0.8975273    0.9329944  0.8645377  1.0000000
## acceleration  0.4233285 -0.5046834   -0.5438005 -0.6891955 -0.4168392
## year          0.5805410 -0.3456474   -0.3698552 -0.4163615 -0.3091199
##              acceleration       year
## mpg             0.4233285  0.5805410
## cylinders      -0.5046834 -0.3456474
## displacement   -0.5438005 -0.3698552
## horsepower     -0.6891955 -0.4163615
## weight         -0.4168392 -0.3091199
## acceleration    1.0000000  0.2903161
## year            0.2903161  1.0000000
```

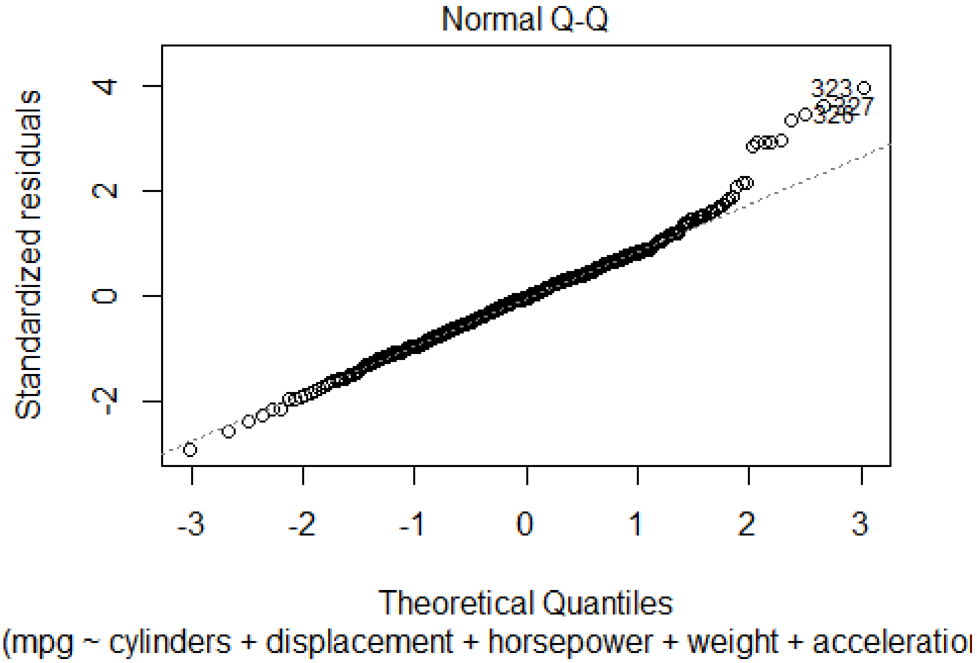
Observations are as follows:

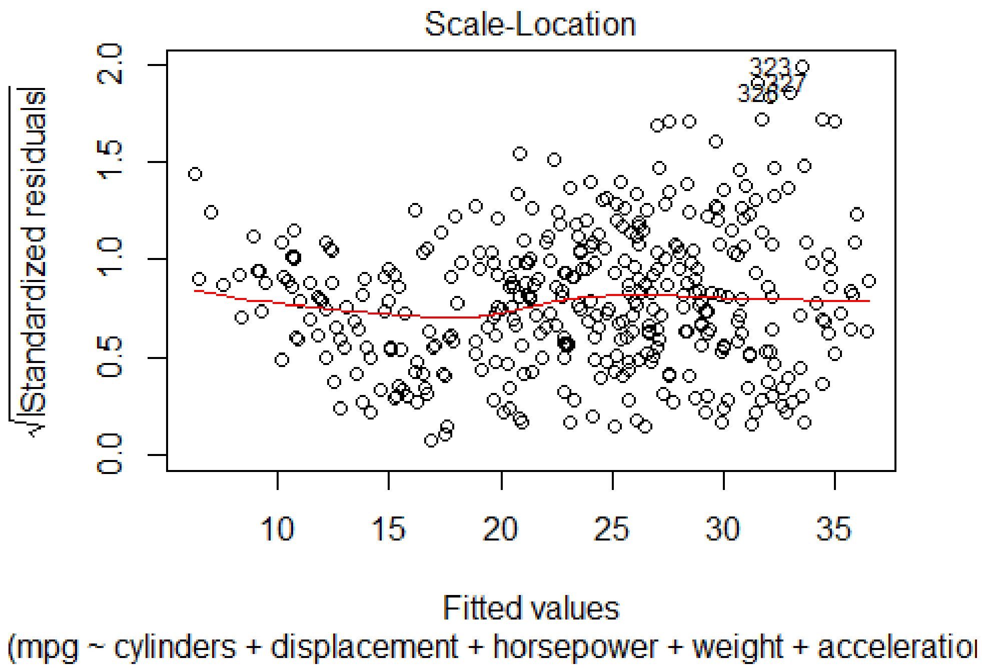
- Heavier weight correlates with lower mpg.
- More cylinders, less mpg.
- Cars become more efficient over time.
- Weight, displacement and horsepower seem to have an inverse effect with mpg

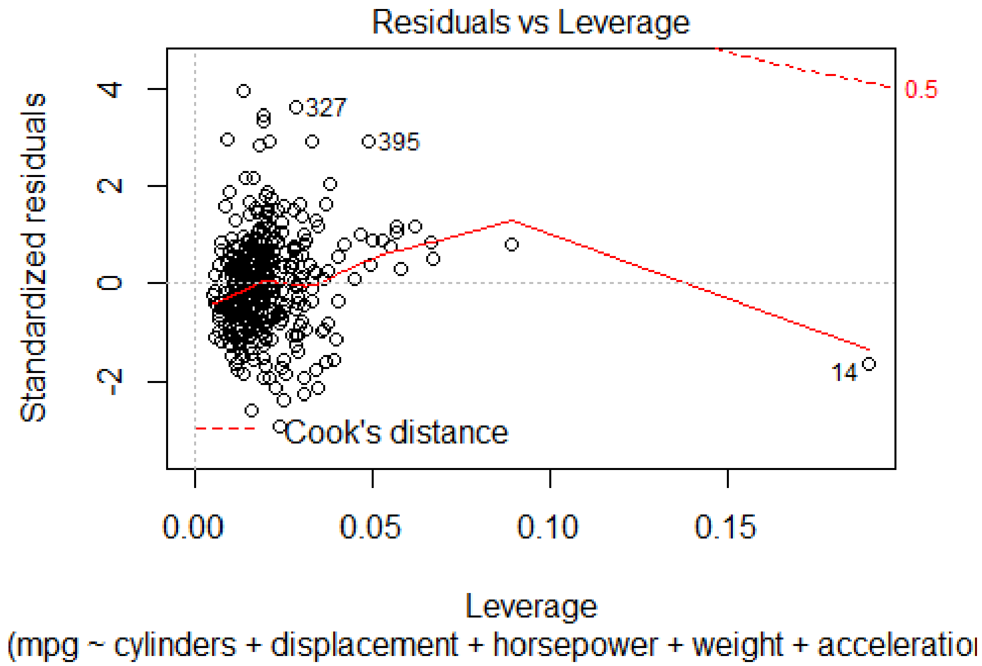
3. **Build a linear regression model, and report its summary.**

```
#throw all the predicates into LM
lm1<-lm(mpg ~ cylinders + displacement + horsepower + weight + acceleration +
year + origin,data = Auto)
plot(lm1)
```

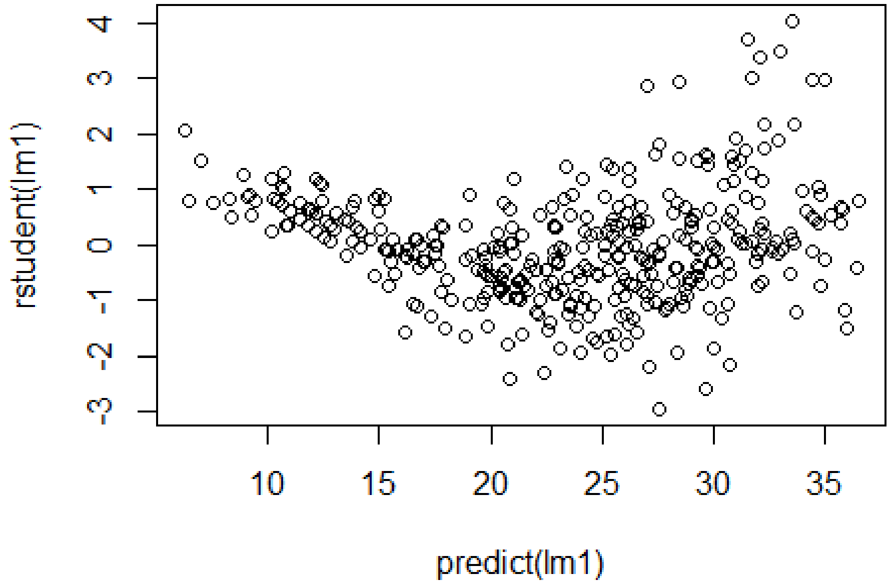

Residuals vs Fitted
323
327
326
Residuals
15
10
5
0
-5
-10
10
15
20
25
30
35
Fitted values
(mpg ~ cylinders + displacement + horsepower + weight + acceleratio


Normal Q-Q
323
327
326
Standardized residuals
4
2
0
-2
-3
-2
-1
0
1
2
3
Theoretical Quantiles
(mpg ~ cylinders + displacement + horsepower + weight + acceleratio





```r
plot(predict(lm1), rstudent(lm1))
```



```
summary(lm1)

##
## Call:
## lm(formula = mpg ~ cylinders + displacement + horsepower + weight +
##     acceleration + year + origin, data = Auto)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -9.5903 -2.1565 -0.1169  1.8690 13.0604
##
## Coefficients:
##                Estimate Std. Error t value Pr(>|t|)
## (Intercept)  -17.218435   4.644294  -3.707  0.00024 ***
## cylinders     -0.493376   0.323282  -1.526  0.12780
## displacement   0.019896   0.007515   2.647  0.00844 **
## horsepower    -0.016951   0.013787  -1.230  0.21963
## weight        -0.006474   0.000652  -9.929  < 2e-16 ***
## acceleration   0.080576   0.098845   0.815  0.41548
## year           0.750773   0.050973  14.729  < 2e-16 ***
## origin         1.426141   0.278136   5.127 4.67e-07 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.328 on 384 degrees of freedom
## Multiple R-squared:  0.8215, Adjusted R-squared:  0.8182
## F-statistic: 252.4 on 7 and 384 DF,  p-value: < 2.2e-16
```

```
Lm1 with all the independent variables in the model has an R square value of
.8182, which is not very bad but can be improved upon.

#seems like some interaction between cylinders, displacement and weight, whic
h isn’t very useful since for lm2 the r square value is still lower.


lm2 <- lm(mpg~ horsepower + acceleration + origin+ cylinders*displacement*wei
ght, data = Auto)
summary(lm2)
```

```
##
## Call:
## lm(formula = mpg ~ horsepower + acceleration + origin + cylinders *
##     displacement * weight, data = Auto)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -11.3465  -2.4054  -0.2855   1.9215  15.8545
##
## Coefficients:
##                                    Estimate Std. Error t value Pr(>|t|)
## (Intercept)                       2.358e+01  1.412e+01   1.670   0.0957 .
## horsepower                       -9.174e-02  1.719e-02  -5.335 1.64e-07 ***
## acceleration                     -1.208e-02  1.195e-01  -0.101   0.9195
## origin                            7.295e-01  3.615e-01   2.018   0.0443 *
## cylinders                         5.532e+00  2.842e+00   1.946   0.0524 .
## displacement                      9.429e-02  9.694e-02   0.973   0.3314
## weight                            4.423e-03  5.019e-03   0.881   0.3787
## cylinders:displacement           -2.298e-02  1.297e-02  -1.771   0.0773 .
## cylinders:weight                 -2.037e-03  9.072e-04  -2.245   0.0253 *
## displacement:weight              -4.708e-05  3.046e-05  -1.546   0.1230
## cylinders:displacement:weight     9.445e-06  4.052e-06   2.331   0.0203 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.922 on 381 degrees of freedom
## Multiple R-squared:  0.7539, Adjusted R-squared:  0.7474
## F-statistic: 116.7 on 10 and 381 DF,  p-value: < 2.2e-16
```

Can you reduce any independent variables to obtain a better model?

Yes

```
#do step evaluation to obtain the best model
base<- lm(mpg ~ 1 ,data = Auto)
summary(base)
```

```
##
## Call:
## lm(formula = mpg ~ 1, data = Auto)
##
```

```
## Residuals:
##      Min       1Q   Median       3Q      Max 
## -14.4459  -6.4459  -0.6959   5.5541  23.1541 
## 
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)    
## (Intercept)  23.4459     0.3942   59.48   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## Residual standard error: 7.805 on 391 degrees of freedom
```

```
base.forward <- step(base, scope =  ~cylinders + displacement + horsepower + 
weight + acceleration + year + origin, direction = "forward" )
```

```
## Start:  AIC=1611.93
## mpg ~ 1
## 
##                Df Sum of Sq     RSS    AIC
## + weight        1   16497.8  7321.2 1151.5
## + displacement  1   15440.2  8378.8 1204.4
## + horsepower    1   14433.1  9385.9 1248.9
## + cylinders     1   14403.1  9415.9 1250.1
## + year          1    8027.7 15791.3 1452.8
## + origin        1    7609.2 16209.8 1463.1
## + acceleration  1    4268.5 19550.5 1536.5
## <none>                      23819.0 1611.9
## 
## Step:  AIC=1151.49
## mpg ~ weight
## 
##                Df Sum of Sq    RSS     AIC
## + year          1   2752.28 4569.0  968.66
## + horsepower    1    327.39 6993.8 1135.56
## + origin        1    222.25 7099.0 1141.41
## + acceleration  1    168.34 7152.9 1144.37
## + displacement  1    150.93 7170.3 1145.33
## + cylinders     1    115.12 7206.1 1147.28
## <none>                      7321.2 1151.49
## 
## Step:  AIC=968.66
## mpg ~ weight + year
## 
##                Df Sum of Sq    RSS    AIC
## + origin        1   220.847 4348.1 951.24
## <none>                      4569.0 968.66
## + acceleration  1    10.450 4558.5 969.77
## + cylinders     1     4.958 4564.0 970.24
## + horsepower    1     3.302 4565.7 970.38
## + displacement  1     0.042 4568.9 970.66
```

```
##
## Step:  AIC=951.24
## mpg ~ weight + year + origin
##
##                Df Sum of Sq    RSS    AIC
## <none>                      4348.1 951.24
## + displacement  1   15.3765 4332.7 951.85
## + acceleration  1   15.0322 4333.1 951.89
## + horsepower    1   14.4048 4333.7 951.94
## + cylinders     1    0.1476 4348.0 953.23
```

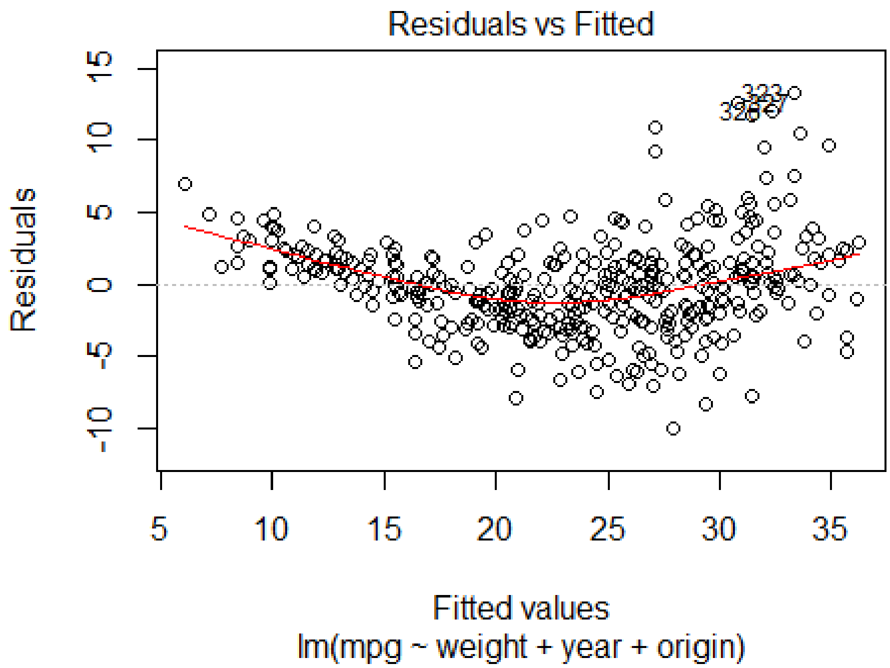
```
# we take model with low AIC , Step:  AIC=951.24, mpg ~ weight + year + origi
n
model=lm(mpg ~ weight+year+origin, data=Auto)
summary(model)
```

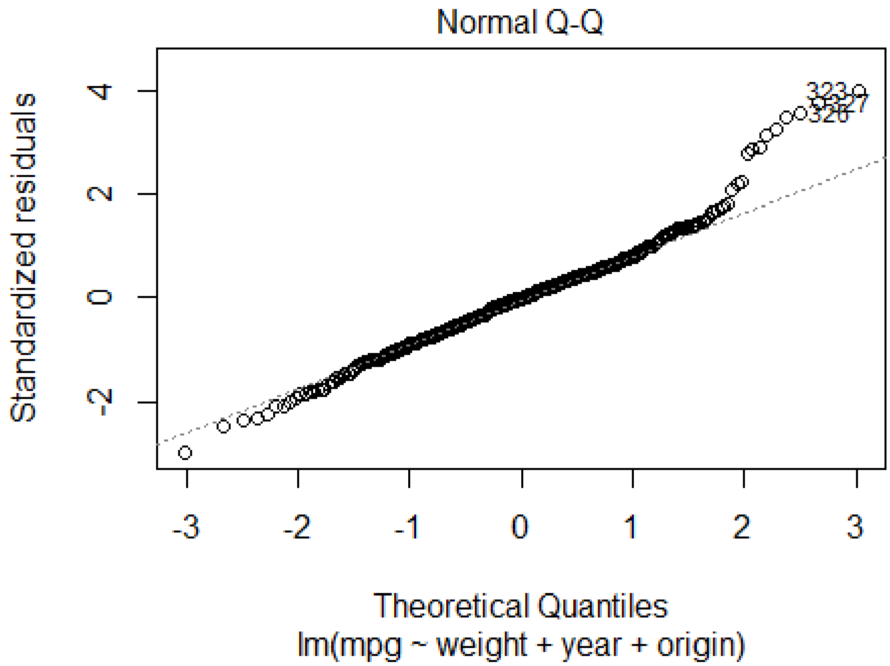
```
##
## Call:
## lm(formula = mpg ~ weight + year + origin, data = Auto)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -9.9440 -2.0948 -0.0389  1.7255 13.2722
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) -1.805e+01  4.001e+00  -4.510 8.60e-06 ***
## weight      -5.994e-03  2.541e-04 -23.588  < 2e-16 ***
## year         7.571e-01  4.832e-02  15.668  < 2e-16 ***
## origin       1.150e+00  2.591e-01   4.439 1.18e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.348 on 388 degrees of freedom
## Multiple R-squared:  0.8175, Adjusted R-squared:  0.816
## F-statistic: 579.2 on 3 and 388 DF,  p-value: < 2.2e-16
```

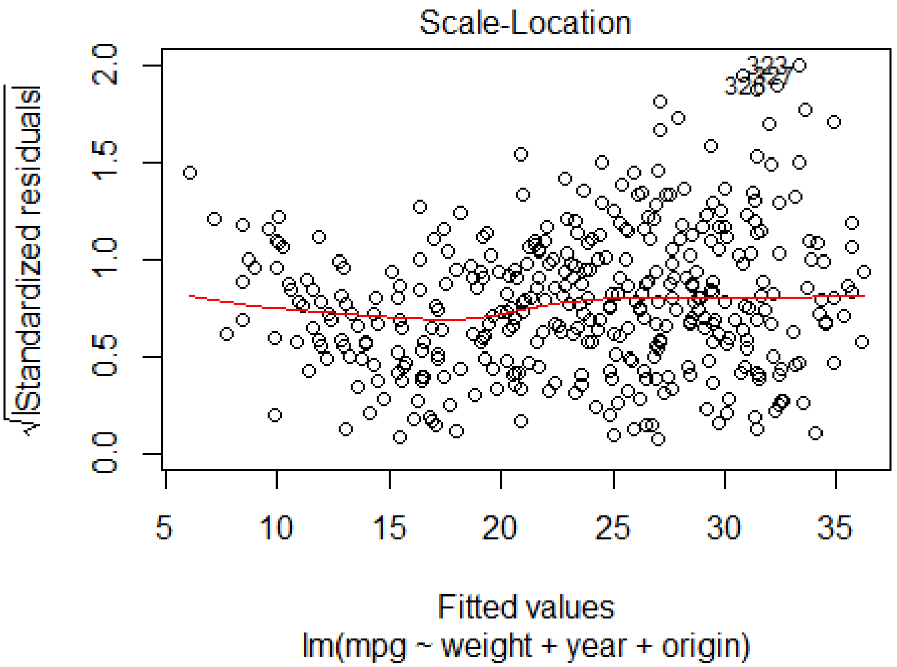
```
confint(model)
```

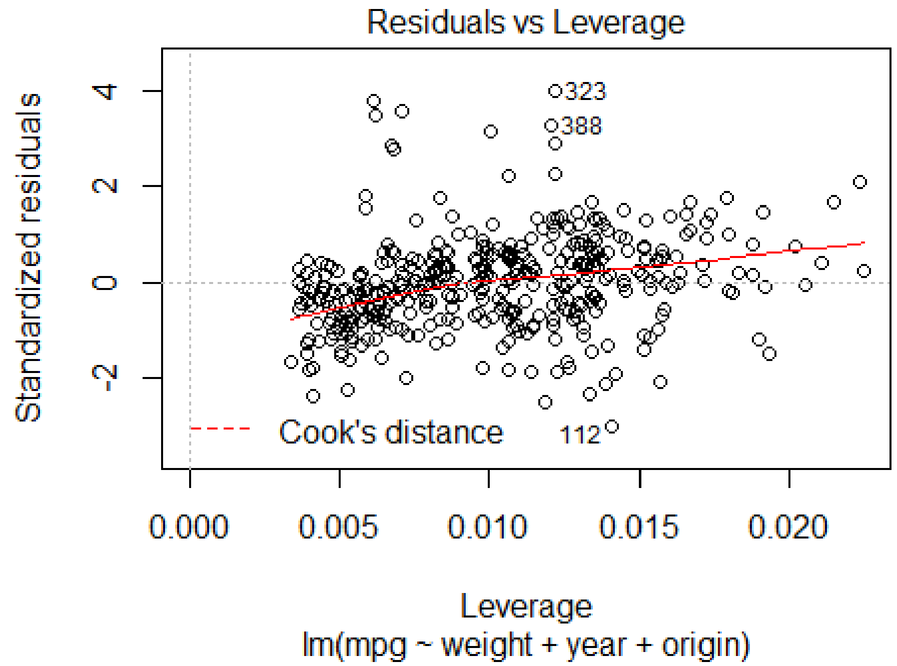
```
##                    2.5 %       97.5 %
## (Intercept) -25.912646751 -10.179053547
## weight       -0.006493731  -0.005494505
## year          0.662115688   0.852136534
## origin        0.640896984   1.659884594
```

```
plot(model)
```

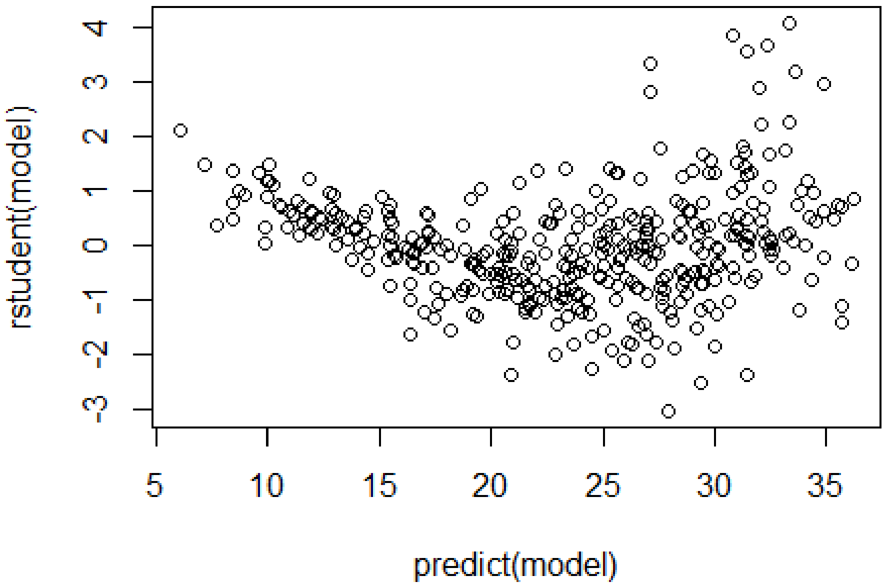

Residuals vs Fitted
Residuals
15
10
5
0
-5
-10
323
327
326
5
10
15
20
25
30
35
Fitted values
lm(mpg ~ weight + year + origin)


Normal Q-Q
Standardized residuals
4
2
0
-2
323
327
326
-3
-2
-1
0
1
2
3
Theoretical Quantiles
lm(mpg ~ weight + year + origin)





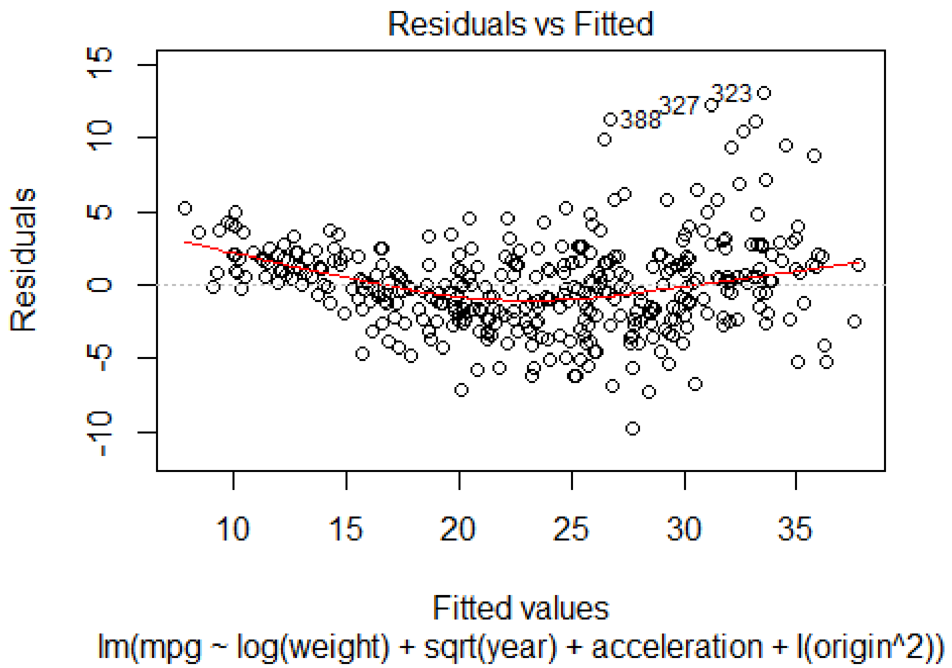
```r
plot(predict(model), rstudent(model))
```

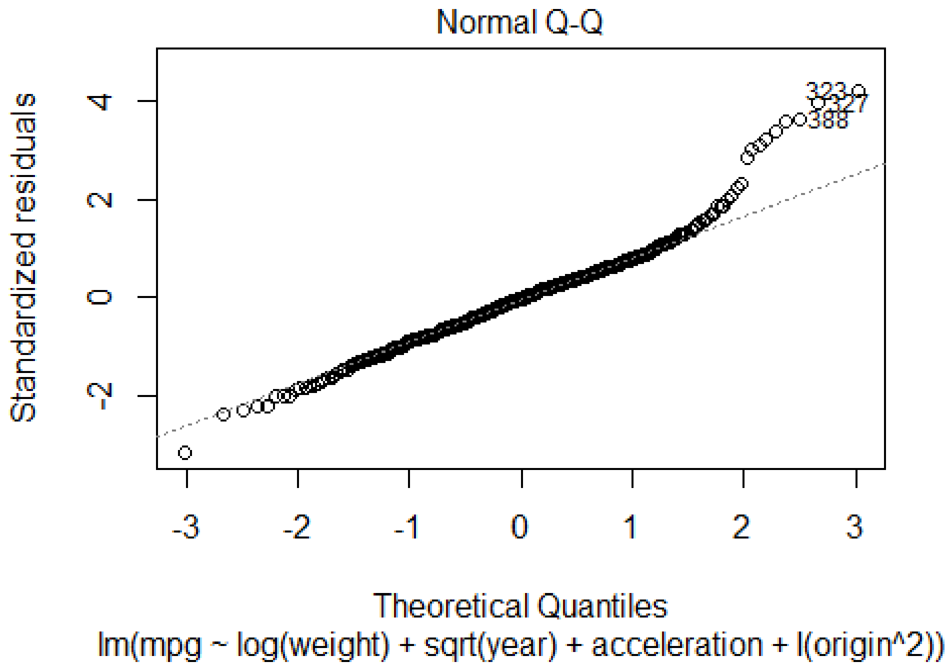


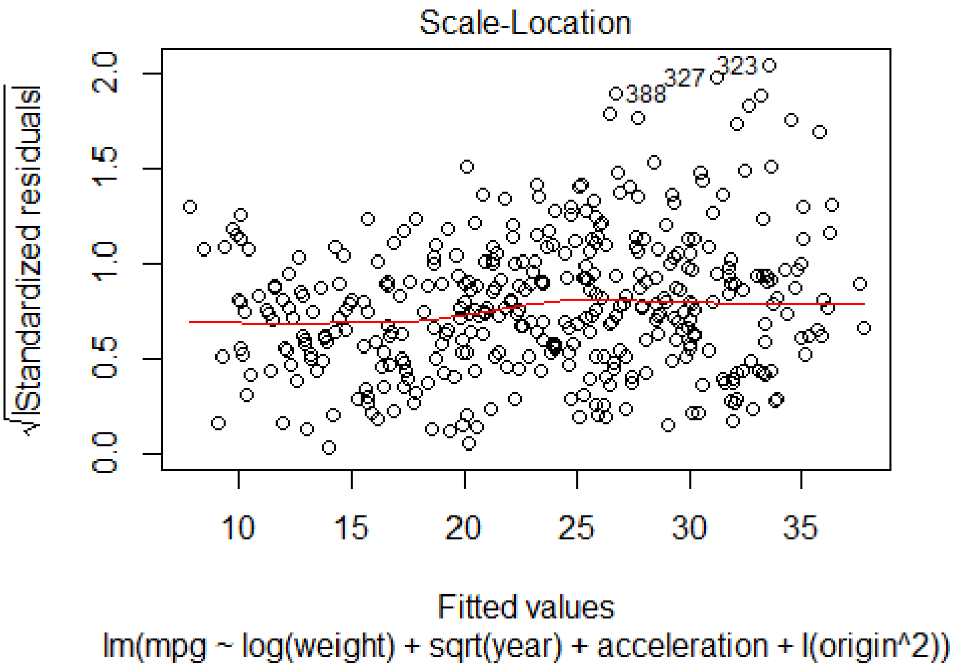
Can you create some other variable(s) to enhance the regression model? (Tip: You could transform some existing variable to a different, more useful, form.)

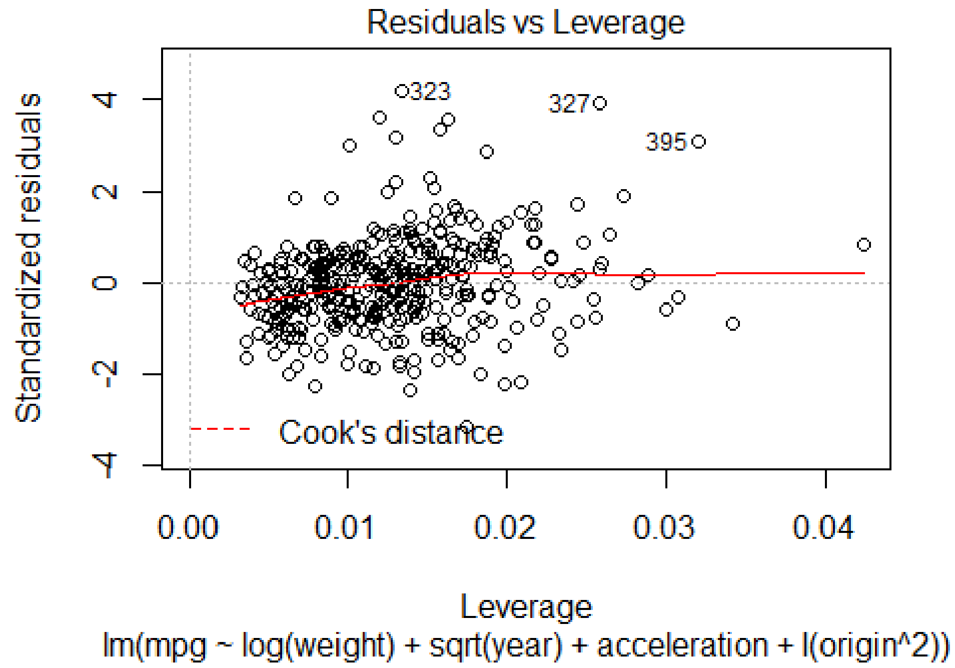
```
#  transformations; The residuals plot has less of a curve than the first reg
ression with all the terms.

lm3 <- lm(mpg~log(weight)+sqrt(year)+acceleration+I(origin^2), data = Auto)
summary(lm3)

##
## Call:
## lm(formula = mpg ~ log(weight) + sqrt(year) + acceleration +
##     I(origin^2), data = Auto)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -9.7436 -1.9269 -0.0651  1.6588 13.0555
##
## Coefficients:
##                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)     57.87752    9.97884   5.800 1.38e-08 ***
## log(weight)    -19.10647    0.75491 -25.310  < 2e-16 ***
## sqrt(year)      13.29388    0.80007  16.616  < 2e-16 ***
## acceleration     0.08033    0.06422   1.251  0.21174
## I(origin^2)      0.17493    0.06230   2.808  0.00524 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.136 on 387 degrees of freedom
## Multiple R-squared:  0.8402, Adjusted R-squared:  0.8385
## F-statistic: 508.6 on 4 and 387 DF,  p-value: < 2.2e-16
```

```
plot(lm3)
```

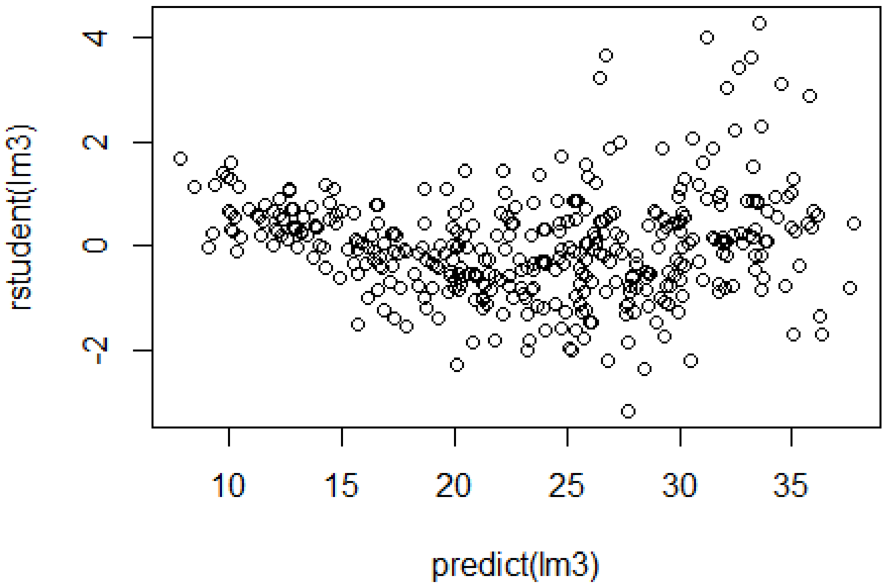

Residuals vs Fitted
Residuals
Fitted values
lm(mpg ~ log(weight) + sqrt(year) + acceleration + I(origin^2))
15
10
5
0
-5
-10
10
15
20
25
30
35
388
327
323


Normal Q-Q
Standardized residuals
Theoretical Quantiles
lm(mpg ~ log(weight) + sqrt(year) + acceleration + I(origin^2))
4
2
0
-2
-3
-2
-1
0
1
2
3
323
327
388





```
plot(predict(lm3), rstudent(lm3))
```



```
R squared for lm3 seems to have improved over other models as it is 0.8385

Hypothesis testing when results in a value lesser than .05 we reject the mode
l. Here the value we got is lesser than 2.2e-16, which means these models are
rejected.
```