Assignment

achyutganti

10/13/2017

```
options(warn=-1)
```

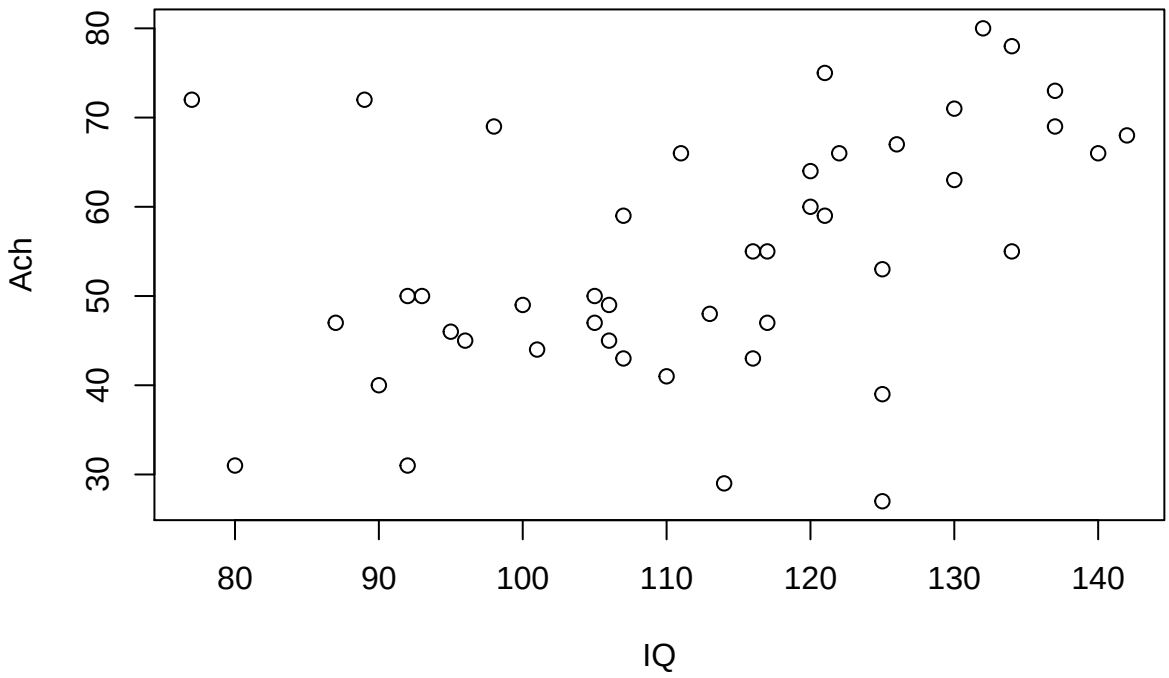
Preparing the dataset

```
IQ=c(100,117,98,87,106,134,77,107,125,105,89,96,105,95,126,111,121,106,134,
     125,140,137,142,130,92,125,120,107,121,90,132,116,137,113,110,114,122,130,116,101,92,120,80,117,93
Ach=c(49,47,69,47,45,55,72,59,27,50,72,45,47,46,67,66,59,49,78,39,66,
      69,68,71,31,53,64,43,75,40,80,55,73,48,41,29,66,63,43,44,50,60,31,55,50)
IQ2=IQ^2

scores<- data.frame(IQ,Ach)

head(scores)
```

```
##    IQ Ach
## 1 100  49
## 2 117  47
## 3  98  69
## 4  87  47
## 5 106  45
## 6 134  55
```

Plotting a scatter plot for the dataset

```
plot(scores)
```



The plot says that as the IQ increases, the scores on the acheivement tests increases.

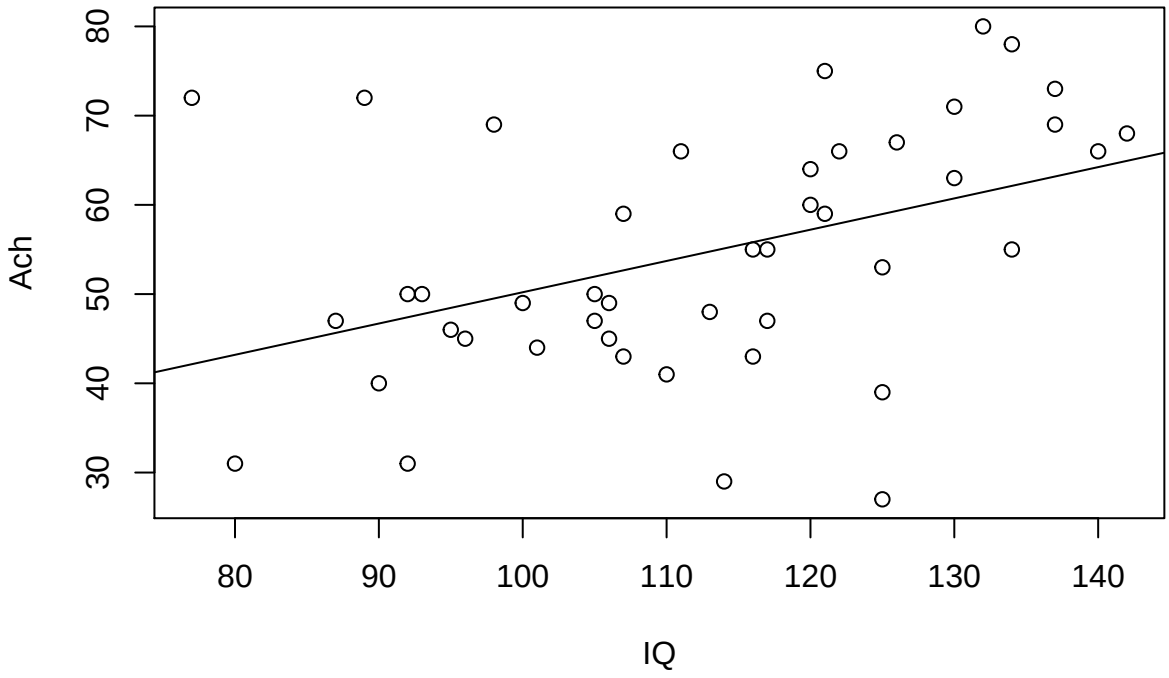
Fitting a linear model on acheivement on IQ

```
iq<- lm(Ach~IQ, data = scores)
summary(iq)
```

```
##
## Call:
## lm(formula = Ach ~ IQ, data = scores)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -31.972  -6.765  -0.816   6.781  29.856
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  15.1496    12.8587   1.178  0.24521
## IQ            0.3506     0.1131   3.099  0.00341 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 12.59 on 43 degrees of freedom
## Multiple R-squared:  0.1826, Adjusted R-squared:  0.1636
## F-statistic: 9.607 on 1 and 43 DF,  p-value: 0.003413
```

Plot an ab-line through the data

```
plot(scores)
abline(lm(Ach~IQ, data = scores))
```



A quadratic model for IQ and IQ squared.

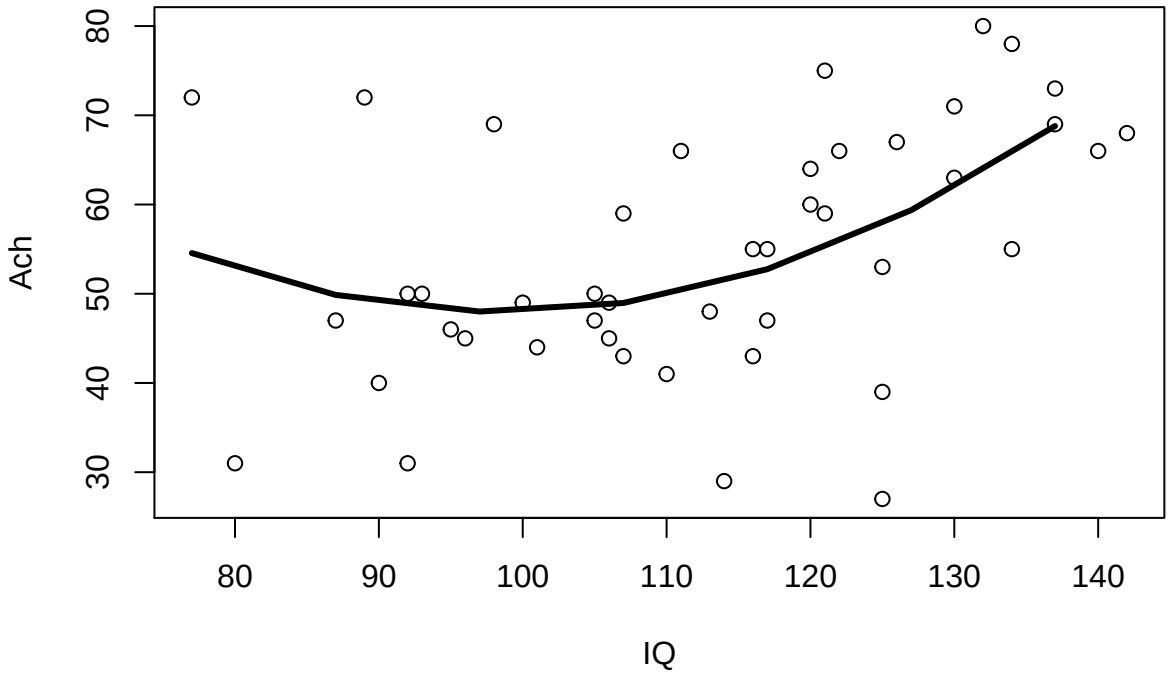
```
quad_iq <- lm(Ach~IQ+I(IQ^2))
summary(quad_iq)
```

```
##
## Call:
## lm(formula = Ach ~ IQ + I(IQ^2))
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -30.8147  -5.9676   0.2556   8.4255  22.7294
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) 185.239568  74.533636   2.485   0.0170 *
## IQ           -2.784475   1.359371  -2.048   0.0468 *
## I(IQ^2)       0.014121   0.006103   2.314   0.0257 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
##
## Residual standard error: 12 on 42 degrees of freedom
## Multiple R-squared:  0.275,  Adjusted R-squared:  0.2405
## F-statistic: 7.966 on 2 and 42 DF,  p-value: 0.001167
```

Plotting the quadratic model

```
IQvalues <- seq(77, 142, 10)
predictedcounts <- predict(quad_iq,list(IQ=IQvalues, IQ2=IQvalues^2))
plot(scores)
lines(IQvalues, predictedcounts, col = "black", lwd = 3)
```



The quadratic model seems to be a better fit to the data because the Adjusted R.sq for that model is 0.2405 whereas the adjusted R.sq for the linear model is only 0.1636. Well, I think the quadratic model could be made more stable and accurate by holding out the outliers and then plotting the model once again.

2. Fitting a Simple Regression Model

Preparing the STAT2.cafeteria dataset in R

```
cafeteria<-c(1,2,3,4,5,6,7,8,9,10,11,12,13,14)
dispensers<-c(0,0,1,1,2,2,3,3,4,4,5,5,6,6)
sales<-c(507.9,498,568.3,575.5,651.6,657.1,713.8,699.6,758.4,765.7,797.7,814.3,836.8,825.1)
```

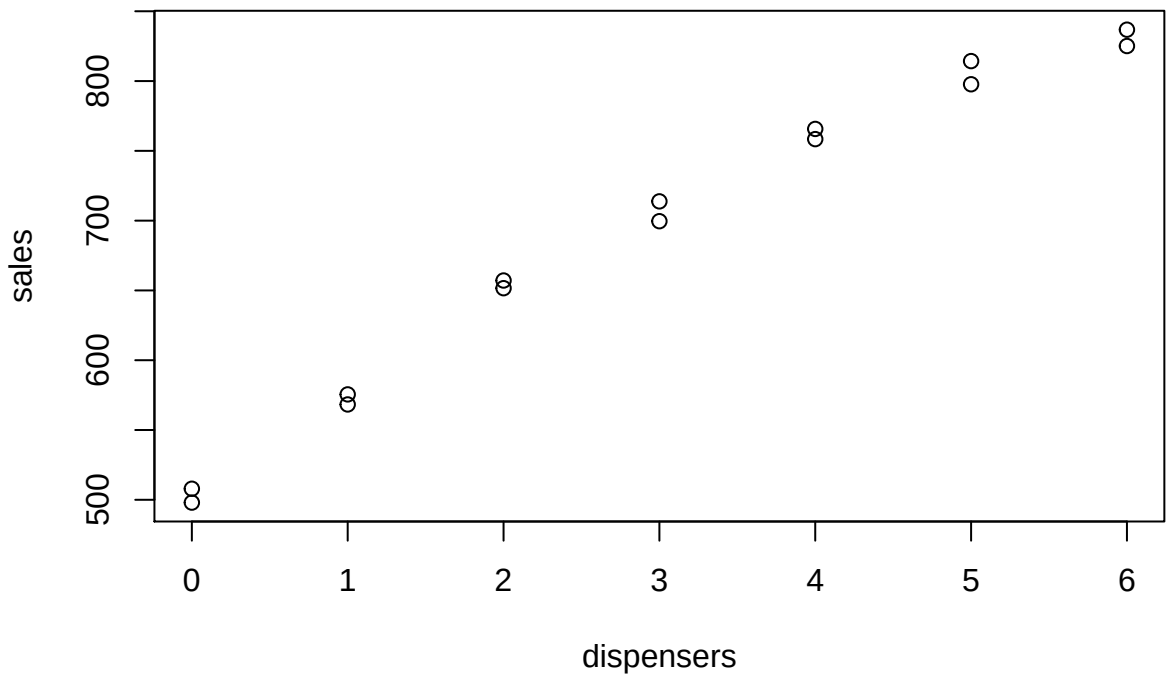
```
cafe = data.frame(cafeteria,dispensers,sales)
head(cafe)
```

```
##   cafeteria dispensers sales
## 1         1          0 507.9
## 2         2          0 498.0
## 3         3          1 568.3
## 4         4          1 575.5
## 5         5          2 651.6
## 6         6          2 657.1
```

The dataframe is ready. Let us plot and see how the response and the predictor are related.

Scatterplot of Sales-vs-Dispensers.

```
plot(dispensers,sales)
```



As the number of dispensers increases, sales in each cafeteria increases. The relation looks a bit curved in nature. So a a non-linear model might fit the data well. Let us seee.

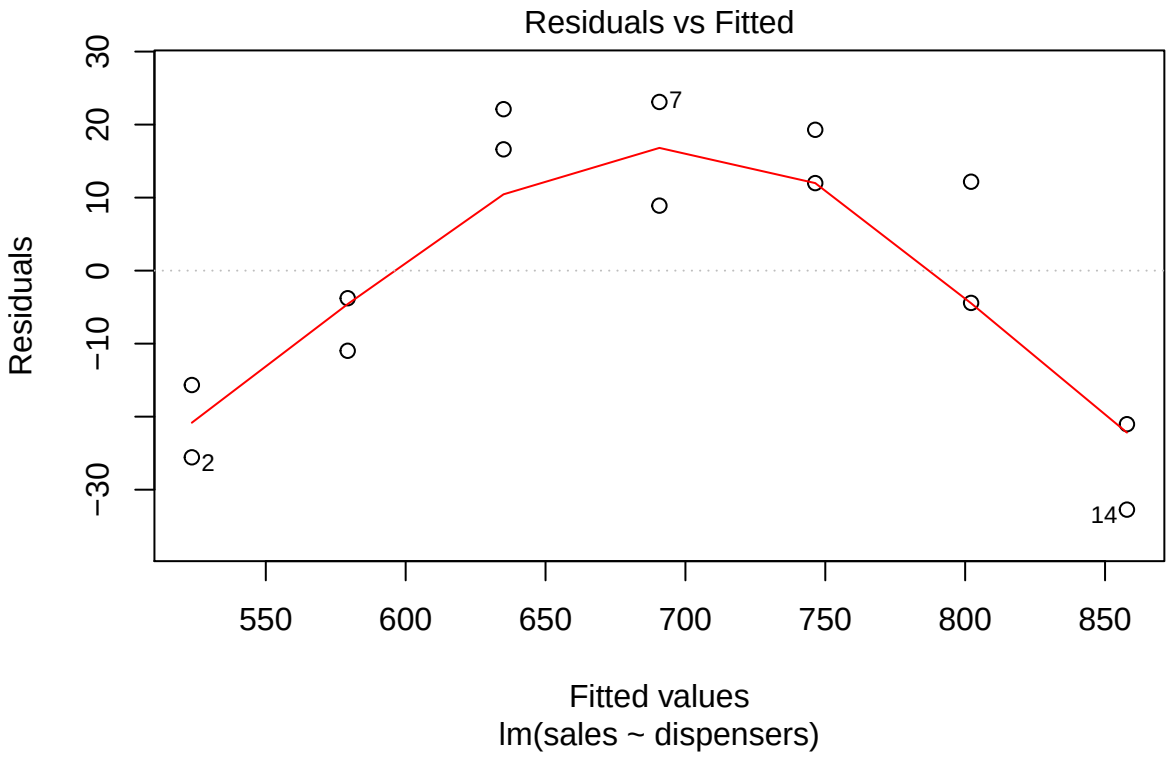
Regression model for the data and observing the summary statistics of that model.

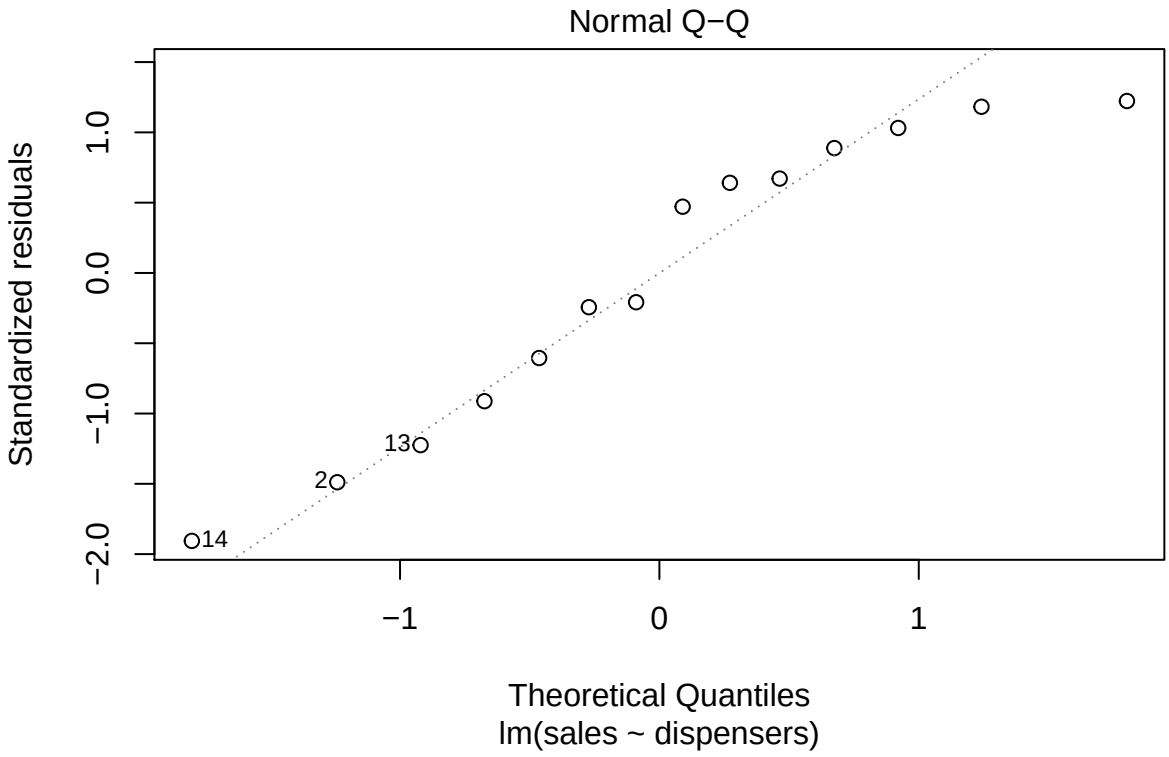
```
sales_lm <- lm(sales~dispensers, data=cafe)
summary(sales_lm)
```

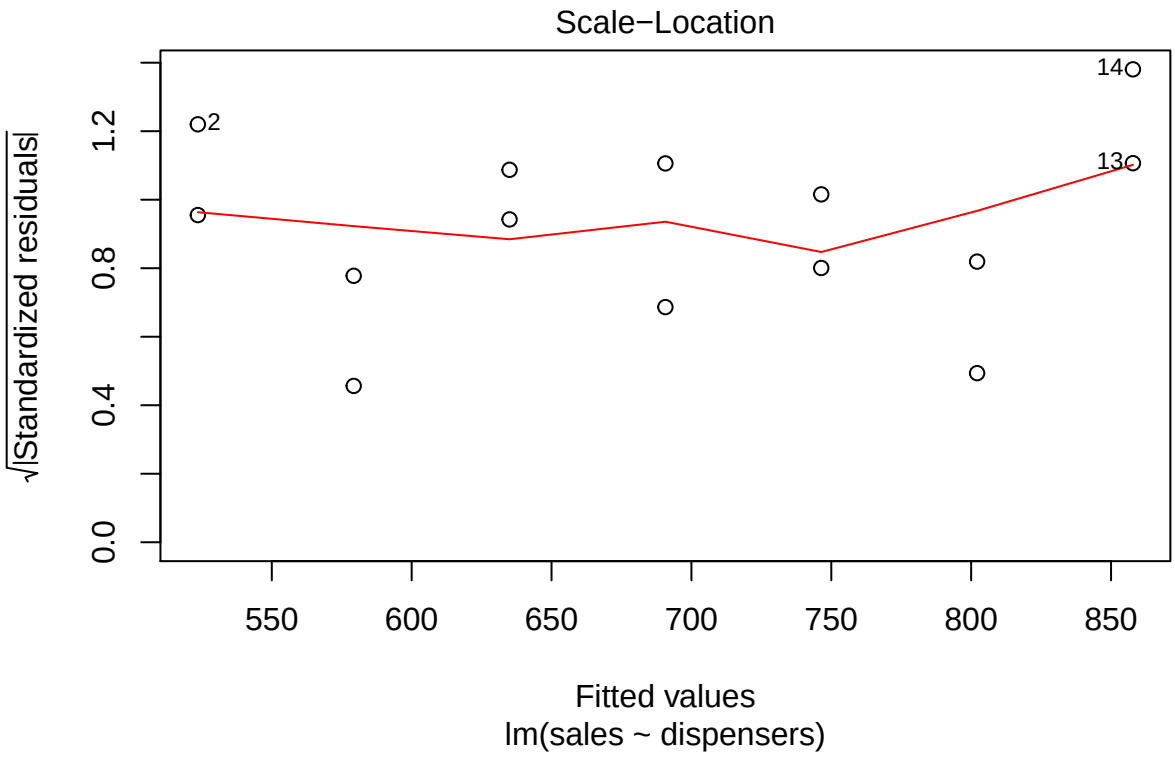
```
##
## Call:
## lm(formula = sales ~ dispensers, data = cafe)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -32.732 -14.496   2.561  15.503  23.100
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  523.568      9.443   55.45 7.81e-16 ***
## dispensers    55.711      2.619   21.27 6.78e-11 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 19.6 on 12 degrees of freedom
## Multiple R-squared:  0.9742, Adjusted R-squared:  0.972
## F-statistic: 452.5 on 1 and 12 DF,  p-value: 6.78e-11
```

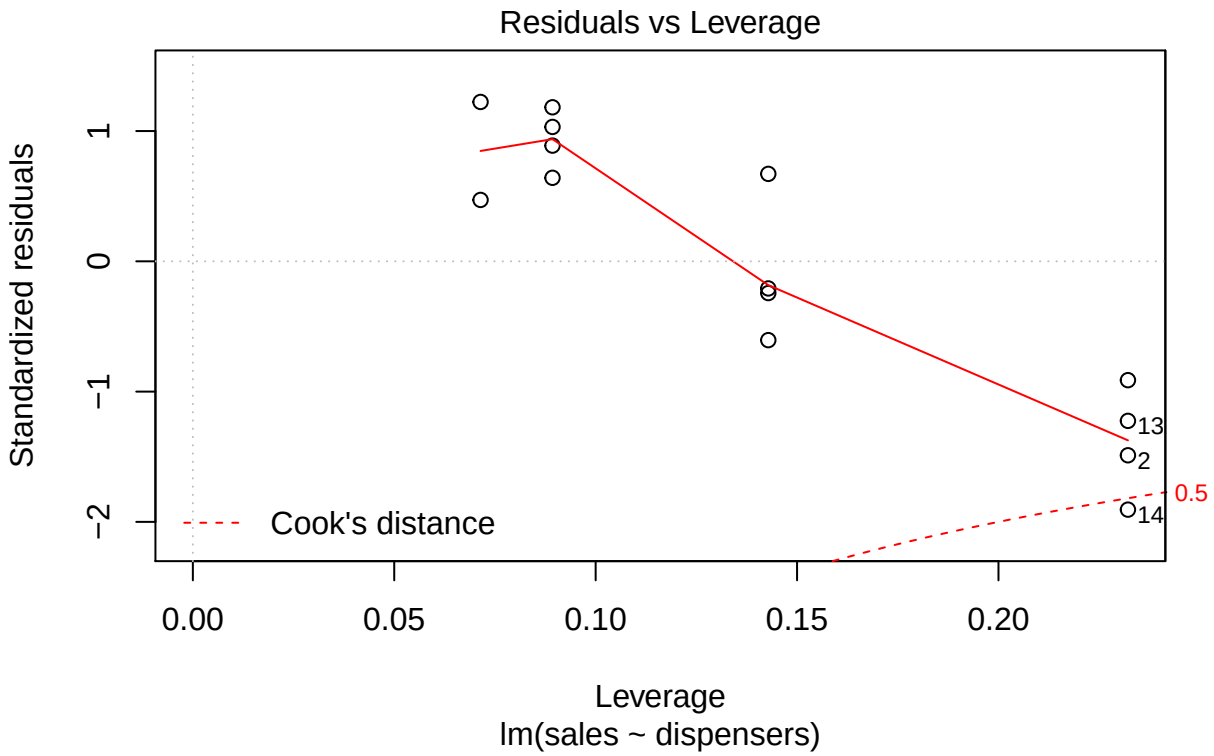
Diagonostic panel of plots.

```
plot(sales_lm)
```


Residuals vs Fitted
Residuals
30
20
10
0
−10
−30
7
2
14
550
600
650
700
750
800
850
Fitted values
lm(sales ~ dispensers)


Normal Q−Q
Standardized residuals
1.0
0.0
−1.0
−2.0
13
2
14
−1
0
1
Theoretical Quantiles
lm(sales ~ dispensers)


Scale−Location
√|Standardized residuals|
0.0
0.4
0.8
1.2
2
14
13
550
600
650
700
750
800
850
Fitted values
lm(sales ~ dispensers)



After examining the residual plots obtained by the linear regression, we can say that the model did not fit the data that well because there seems to exist a discernable pattern in those plots. So a polynomial regression model might do well on the data.

Fitting quadratic model for the sales-vs-dispensers

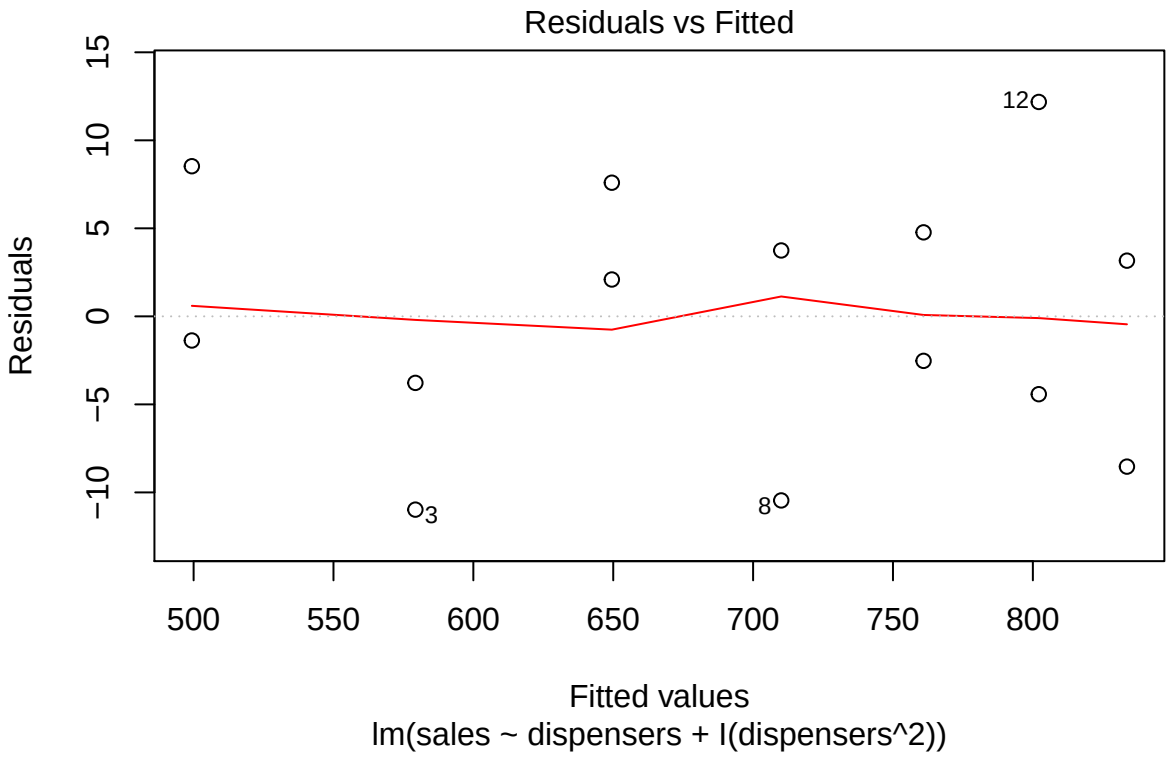
```
sales_qm <- lm(sales~dispensers+I(dispensers^2), data=cafe)
summary(sales_qm)
```

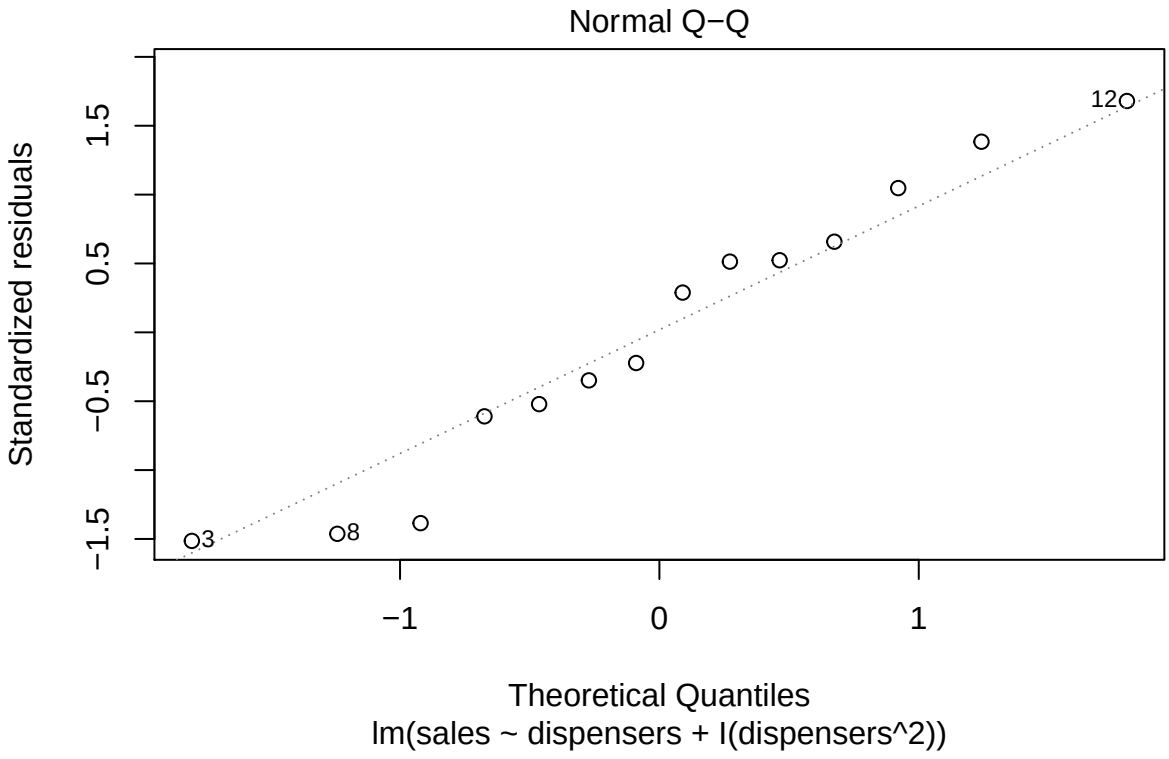
```
##
## Call:
## lm(formula = sales ~ dispensers + I(dispensers^2), data = cafe)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -10.9786  -4.2607   0.3607   4.5143  12.1786
##
## Coefficients:
##                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)     499.3714     4.8339 103.306  < 2e-16 ***
## dispensers       84.7464     3.7735  22.458 1.54e-10 ***
## I(dispensers^2)  -4.8393     0.6042  -8.009 6.46e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
```

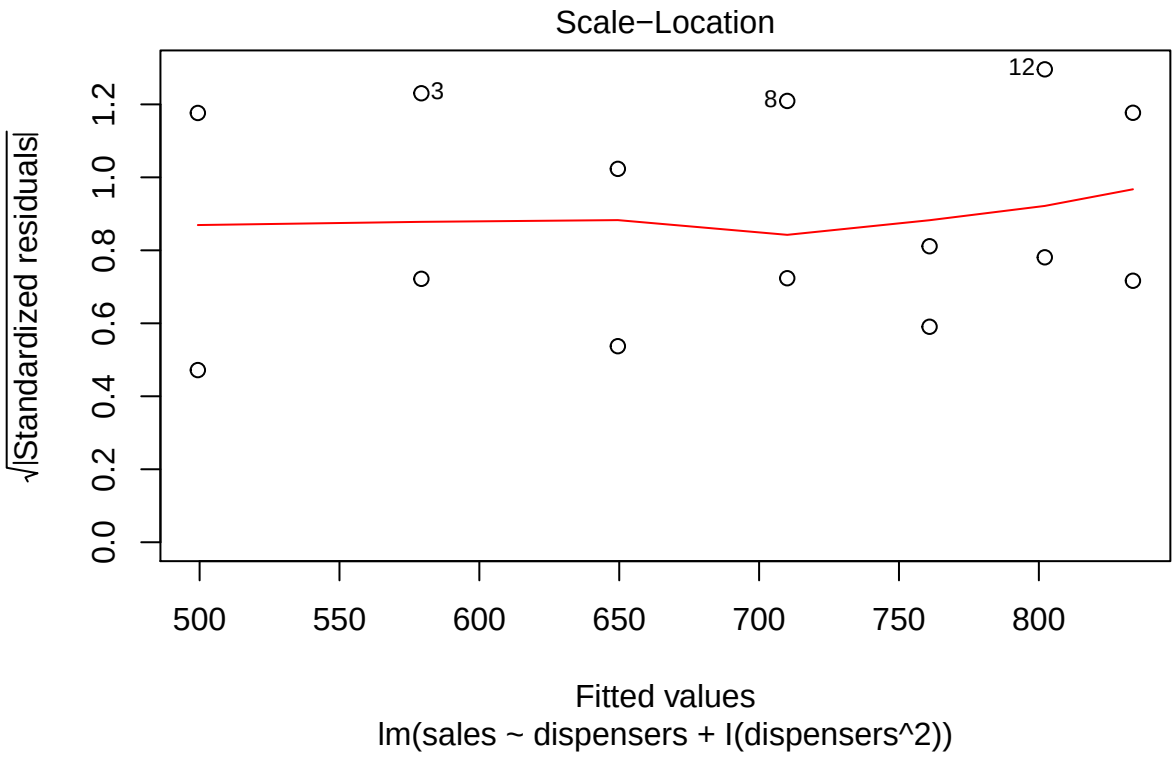
```
## Residual standard error: 7.832 on 11 degrees of freedom
## Multiple R-squared:  0.9962, Adjusted R-squared:  0.9955
## F-statistic:  1449 on 2 and 11 DF,  p-value: 4.756e-14
```

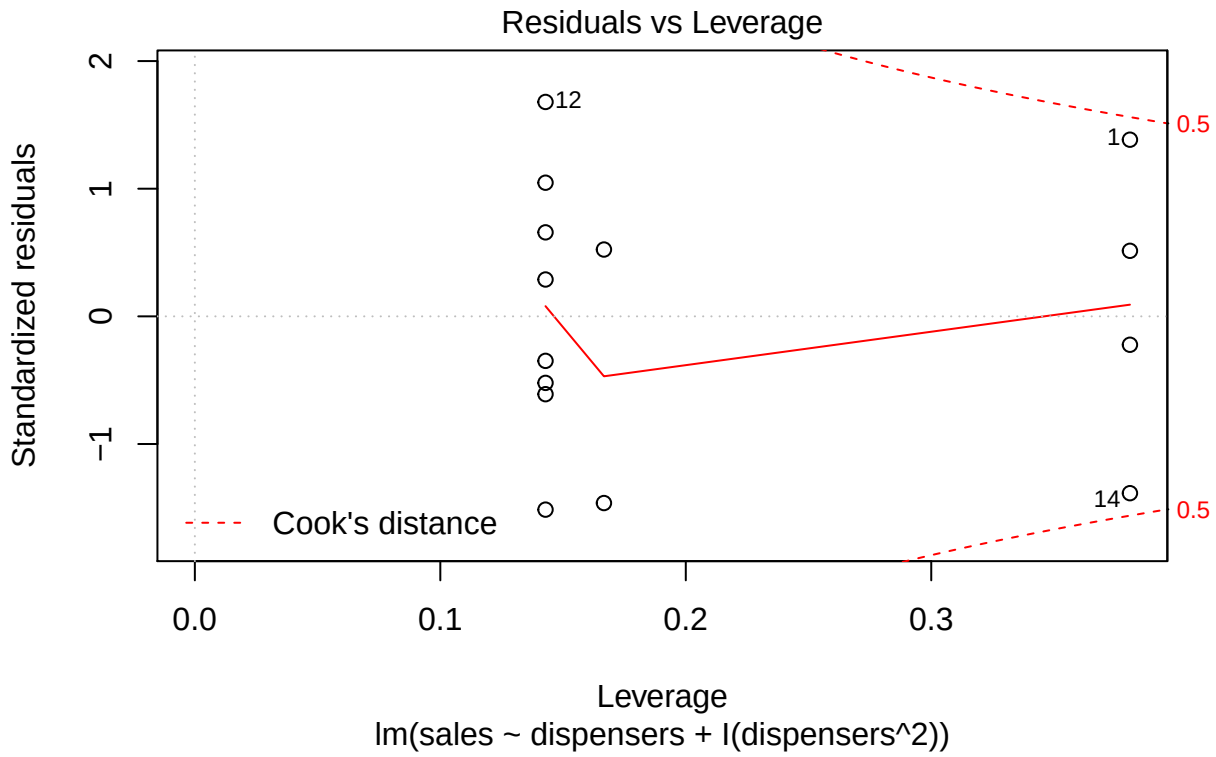
Requesting the diagonostic plots

```
plot(sales_qm)
```

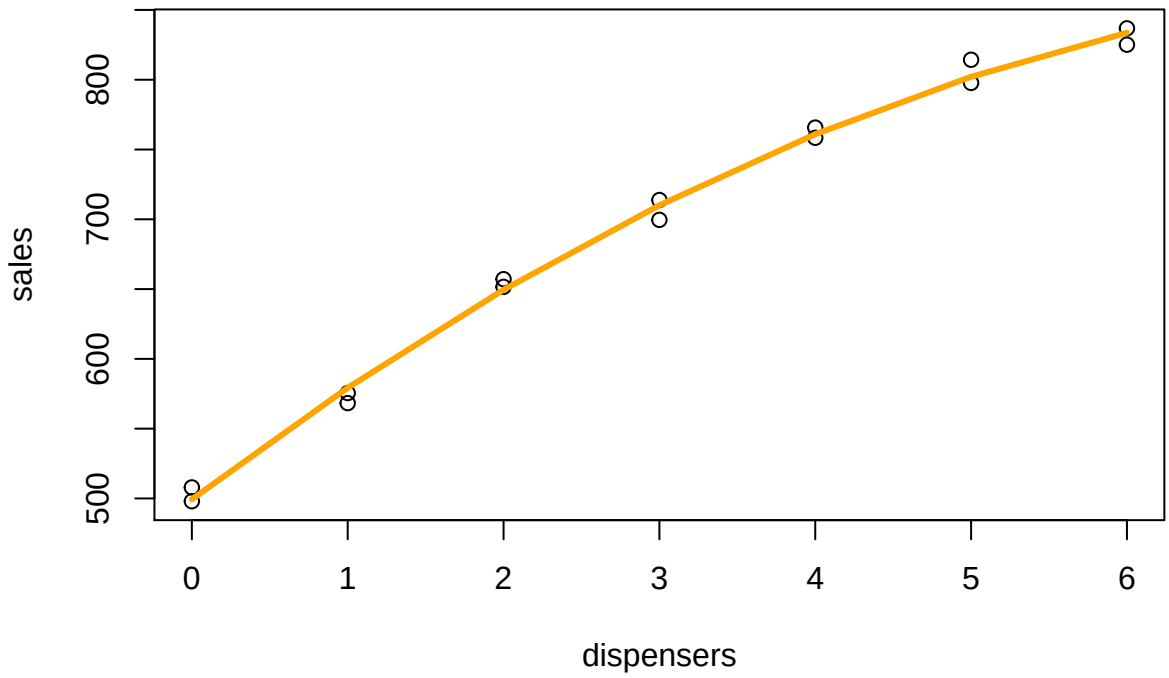

Residuals vs Fitted
Residuals
15
10
5
0
−5
−10
12
3
8
500
550
600
650
700
750
800
Fitted values
lm(sales ~ dispensers + I(dispensers^2))


Normal Q−Q
Standardized residuals
1.5
0.5
−0.5
−1.5
12
3
8
−1
0
1
Theoretical Quantiles
lm(sales ~ dispensers + I(dispensers^2))


Scale−Location
√|Standardized residuals|
0.0
0.2
0.4
0.6
0.8
1.0
1.2
3
8
12
500
550
600
650
700
750
800
Fitted values
lm(sales ~ dispensers + I(dispensers^2))




```
dispensersvalues=seq(0,6,1)
predictedcounts <- predict(sales_qm,list(dispensers=dispensersvalues, dispensers2=dispensersvalues^2))
plot(dispensers,sales)
lines(dispensersvalues, predictedcounts, col = "orange", lwd = 3)
```



Yes, from the above plots, we can say that the quadratic model fits the data well than compared to the linear regression model in the previous example. The residual-vs-predictor plot in the diagonostic plots shows a random scatter about a reference line almost at zero. The q-q plot also looks better than that of the linear regression plot. So, we can say for sure that our quadratic model fits well than compared to the linear model.

Co-relation between dispensers and dispensers squared.

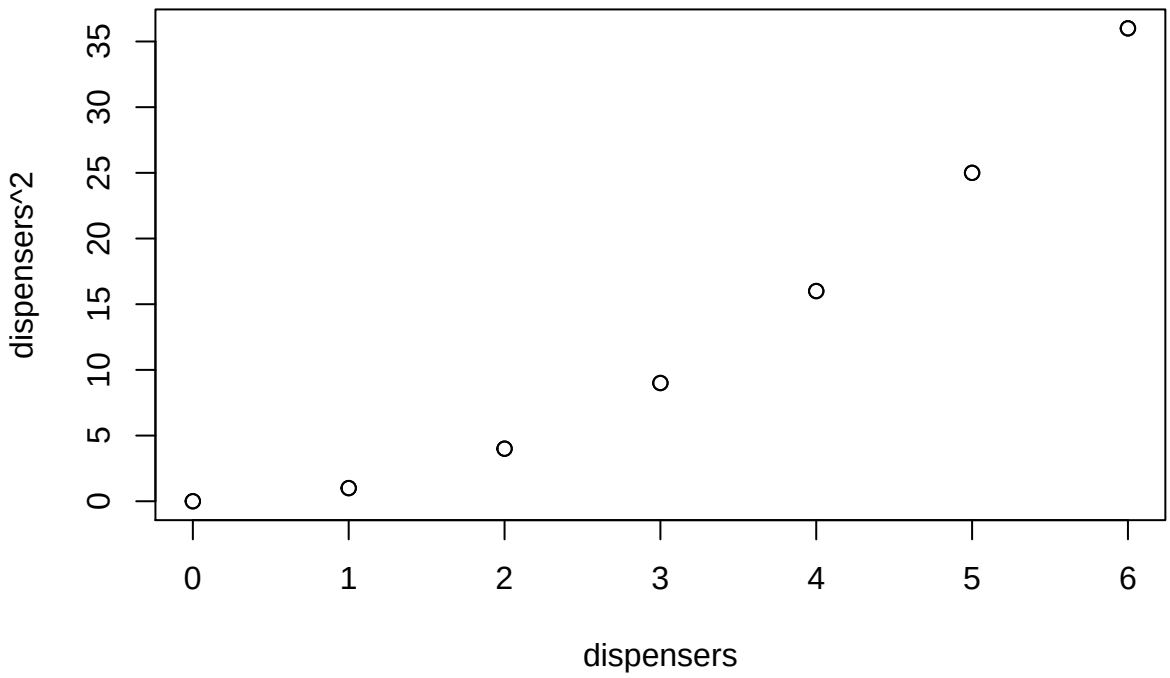
Tabular output

```
cor(dispensers,dispensers^2)
```

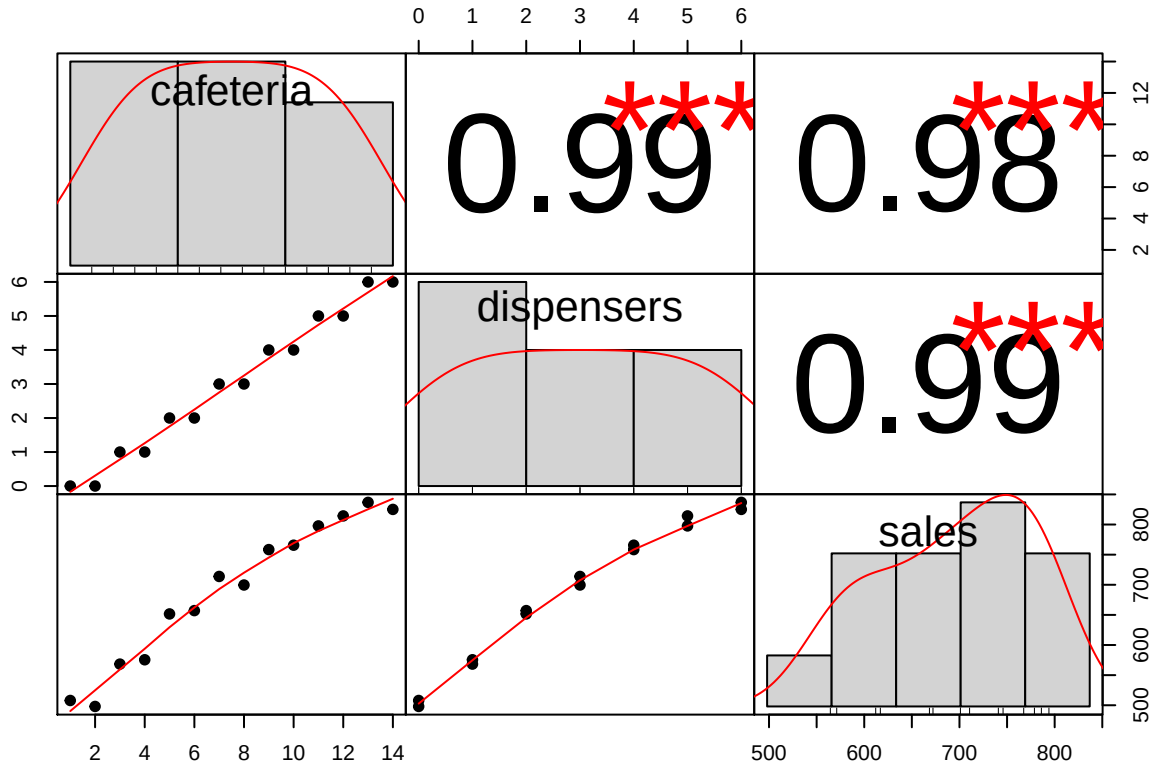
```
## [1] 0.9607689
```

Graphical output for the co-relation

```
plot(dispensers,dispensers^2)
```




```
chart.Correlation(cafe, histogram=TRUE, pch=19)
```



By observing the above two plots we can say that there exixts high correlation between dispensers and dispensers squared. (0.96).

Computing the variance Inflation factor and the Collinearity diagonostics statistics.

```
ols_coll_diag(sales_qm)
```

```
## Tolerance and Variance Inflation Factor
## ----------------------------------------
## # A tibble: 2 x 3
##         Variables  Tolerance   VIF
##             <chr>      <dbl> <dbl>
## 1      dispensers 0.07692308    13
## 2 I(dispensers^2) 0.07692308    13
##
##
## Eigenvalue and Condition Index
## ------------------------------
##   Eigenvalue Condition Index  intercept  dispensers I(dispensers^2)
## 1 2.68572598        1.000000 0.02083752 0.003210318      0.00463641
## 2 0.30018217        2.991151 0.41154113 0.003495386      0.03650748
## 3 0.01409185       13.805333 0.56762136 0.993294296      0.95885611
```

```
ols_vif_tol(sales_qm)
```

```
## # A tibble: 2 x 3
##         Variables  Tolerance   VIF
##             <chr>      <dbl> <dbl>
## 1      dispensers 0.07692308    13
## 2 I(dispensers^2) 0.07692308    13
```

Standardizing the predictors

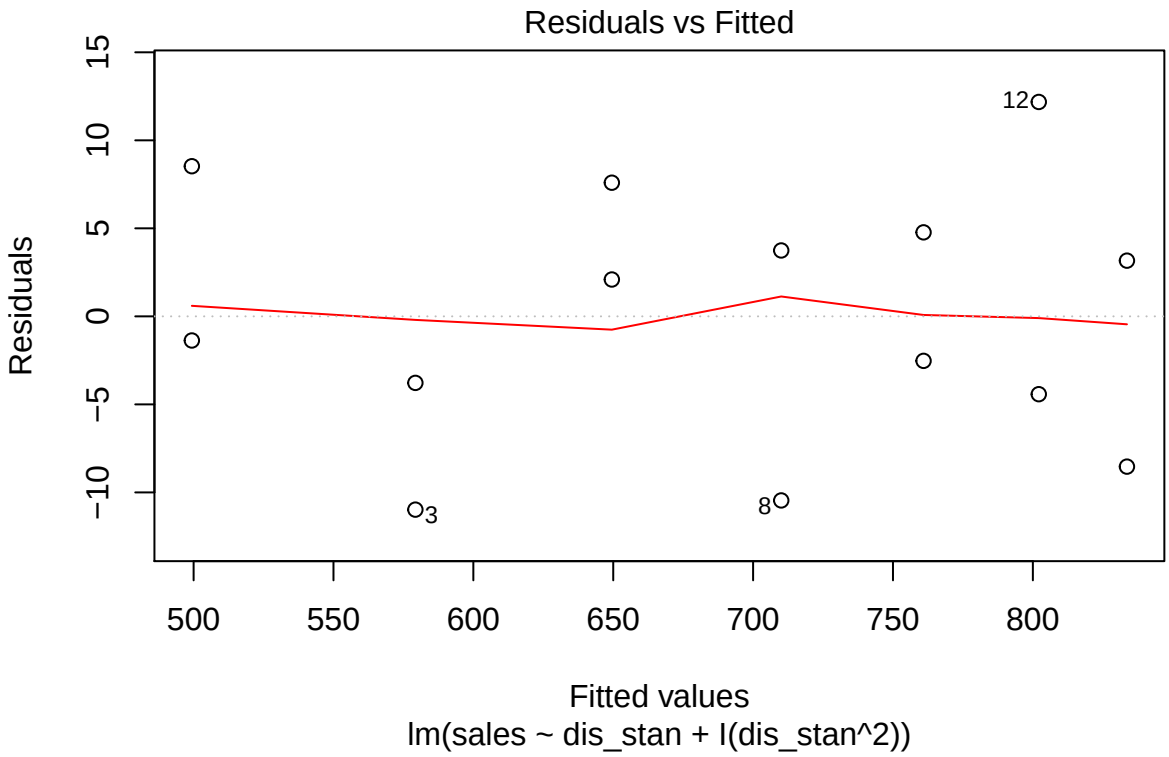
```
dis_stan<-scale(dispensers)
dis2_stan<-scale(dispensers^2)
```

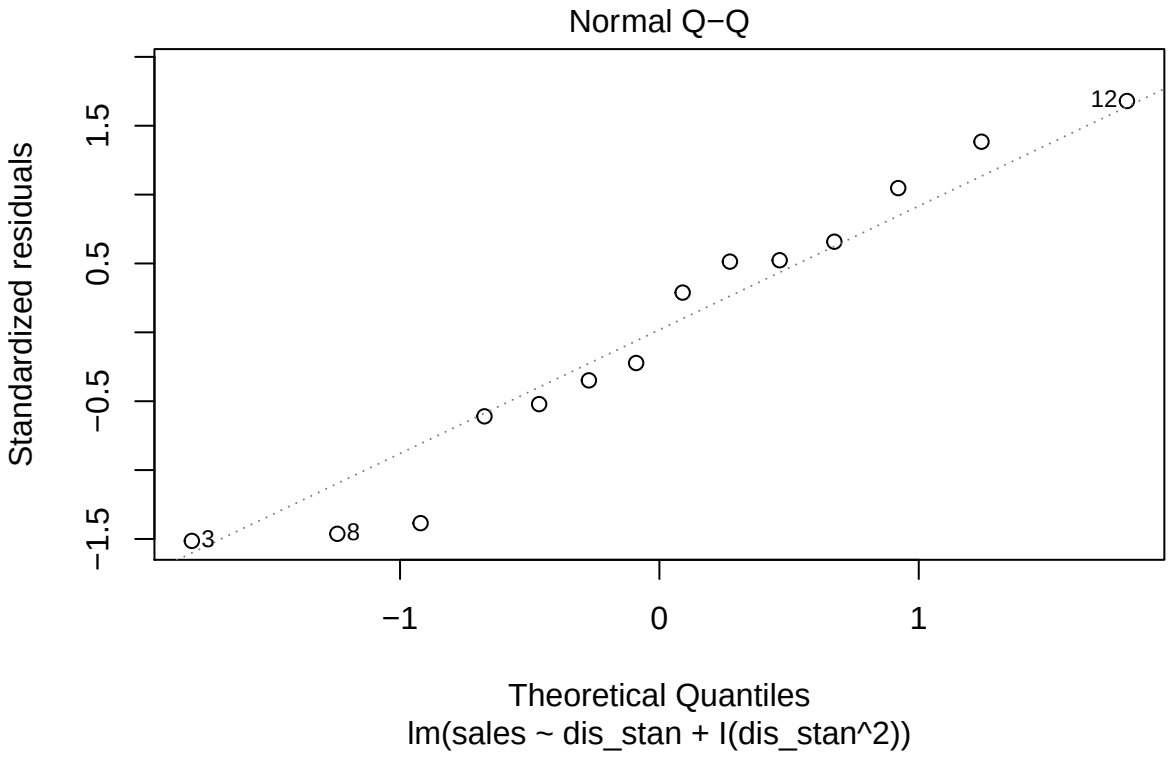
Regression model for the centered Variables.

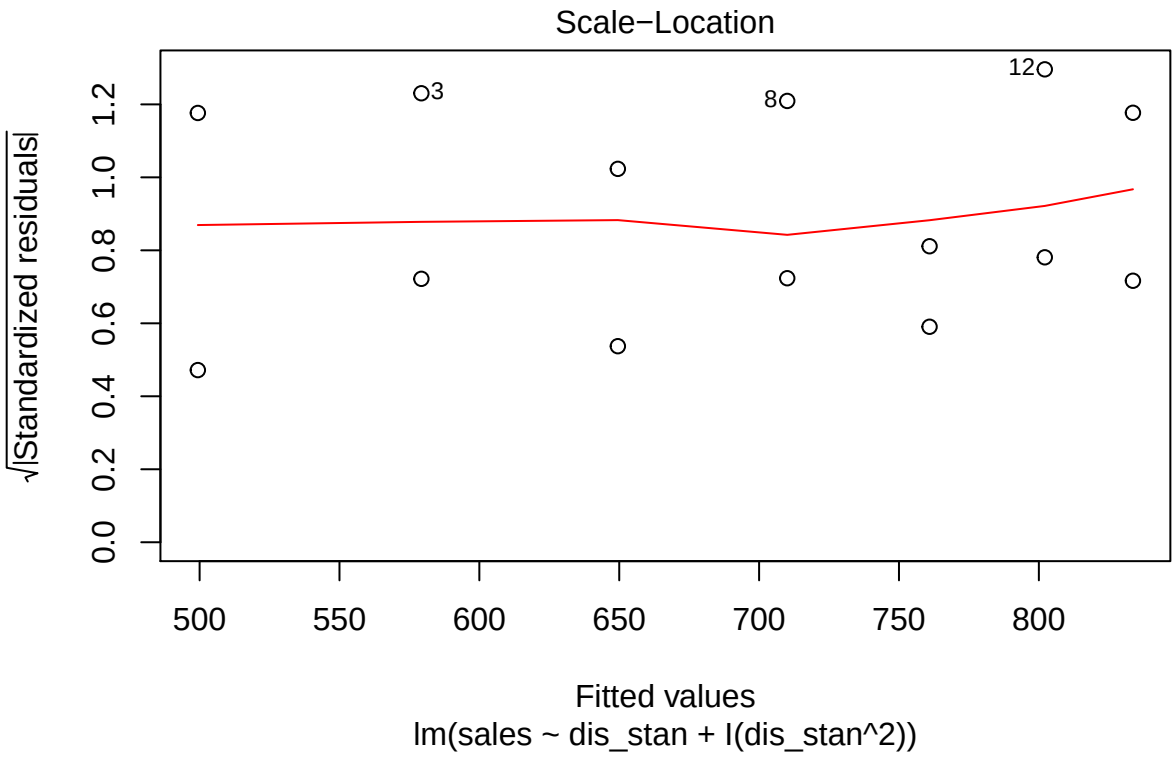
```
st_qm<- lm(sales~dis_stan+I(dis_stan^2))
summary(st_qm)
```

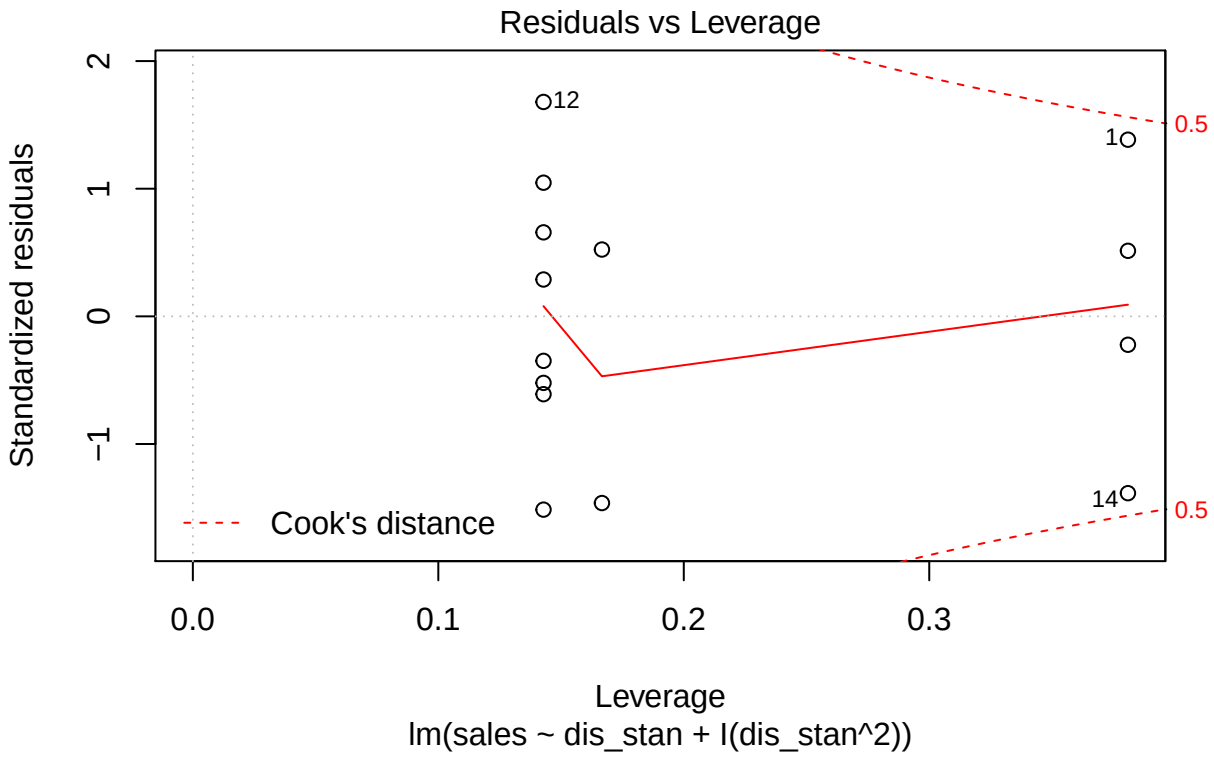
```
##
## Call:
## lm(formula = sales ~ dis_stan + I(dis_stan^2))
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -10.9786  -4.2607   0.3607   4.5143  12.1786
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)    710.057      3.197 222.078  < 2e-16 ***
## dis_stan       115.627      2.172  53.232 1.27e-14 ***
## I(dis_stan^2)  -20.846      2.603  -8.009 6.46e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 7.832 on 11 degrees of freedom
## Multiple R-squared:  0.9962, Adjusted R-squared:  0.9955
## F-statistic:  1449 on 2 and 11 DF,  p-value: 4.756e-14
```

```
plot(st_qm)
```


Residuals vs Fitted
Residuals
15
10
5
0
−5
−10
12
3
8
500
550
600
650
700
750
800
Fitted values
lm(sales ~ dis_stan + I(dis_stan^2))


Normal Q−Q
Standardized residuals
1.5
0.5
−0.5
−1.5
12
3
8
−1
0
1
Theoretical Quantiles
lm(sales ~ dis_stan + I(dis_stan^2))


Scale−Location
√|Standardized residuals|
0.0
0.2
0.4
0.6
0.8
1.0
1.2
3
8
12
500
550
600
650
700
750
800
Fitted values
lm(sales ~ dis_stan + I(dis_stan^2))



Collinearity diagonostics to check if the independent variables are still correlated.

```
ols_vif_tol(st_qm)
```

```
## # A tibble: 2 x 3
##       Variables Tolerance   VIF
##           <chr>     <dbl> <dbl>
## 1      dis_stan         1     1
## 2 I(dis_stan^2)         1     1
```

It is good to find that the variance inflation factor has decreased significantly(VIC= 1) which means that the model with the centered variables are now more stable than the previous one.

2. Generating Candidate models.

Preparing the dataset for STAT2.cars in R

```
cars4 <- read_csv("~/R-class-518/cars4.csv")
```

```
## Parsed with column specification:
## cols(
##   Manufacturer = col_character(),
```

```
##   Model = col_character(),
##   Type = col_character(),
##   Price = col_double(),
##   Citympg = col_integer(),
##   Hwympg = col_integer(),
##   Cylinders = col_integer(),
##   EngineSize = col_double(),
##   Horsepower = col_integer(),
##   FuelTank = col_double(),
##   Passengers = col_integer(),
##   Luggage = col_integer(),
##   Weight = col_integer(),
##   Origin = col_character(),
##   logprice = col_double()
## )
```

```
head(cars4)
```

```
## # A tibble: 6 x 15
##   Manufacturer   Model    Type Price Citympg Hwympg Cylinders EngineSize
##          <chr>   <chr>   <chr> <dbl>   <int>  <int>     <int>      <dbl>
## 1        Acura Integra   Small  15.9      25     31         4        1.8
## 2        Acura  Legend Midsize  33.9      18     25         6        3.2
## 3         Audi     100 Midsize  37.7      19     26         6        2.8
## 4         Audi      90 Compact  29.1      20     26         6        2.8
## 5          BMW    535i Midsize  30.0      22     30         4        3.5
## 6        Buick Century Midsize  15.7      22     31         4        2.2
## # ... with 7 more variables: Horsepower <int>, FuelTank <dbl>,
## #   Passengers <int>, Luggage <int>, Weight <int>, Origin <chr>,
## #   logprice <dbl>
```

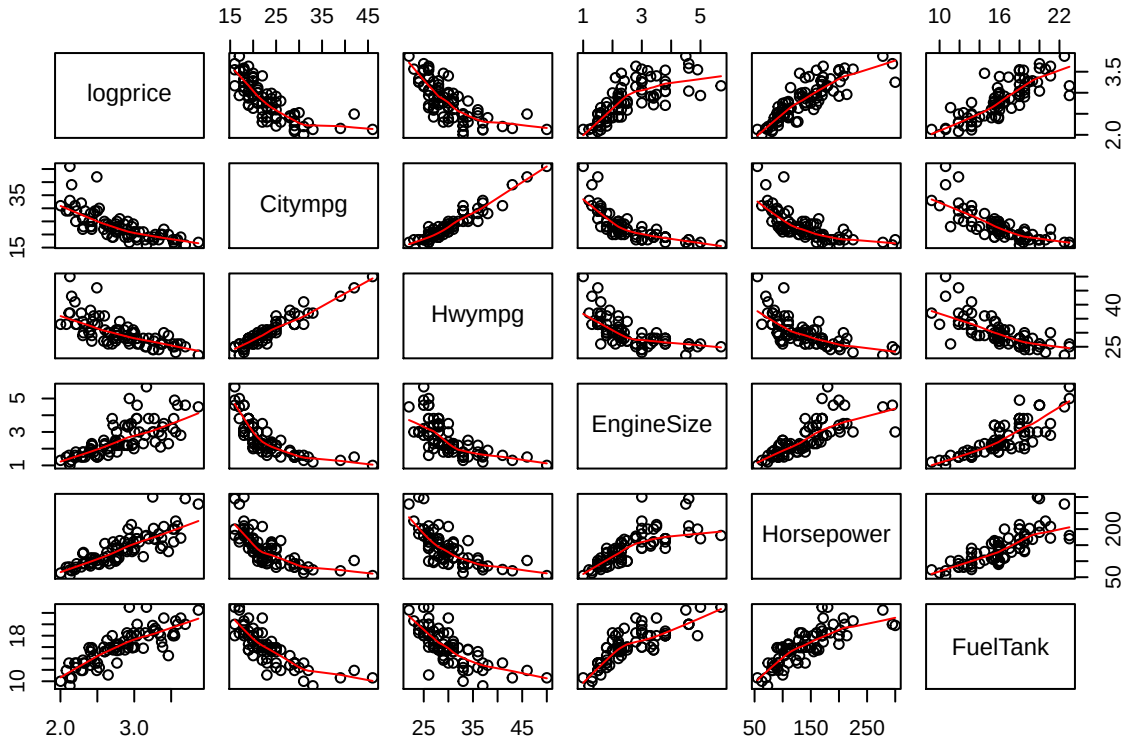
Observing the scatter plots for logprice vs all the numerical predictors.

First let us separate non-numerical variables in the dataset.

```
cardata<- cars4[,-c(1,2,3,14)]
```

I have separated variables of interest which seemed to havea a curvilinear relationship with the response variable. Now let us get an idea of how the relationship of logprice is with the variables.

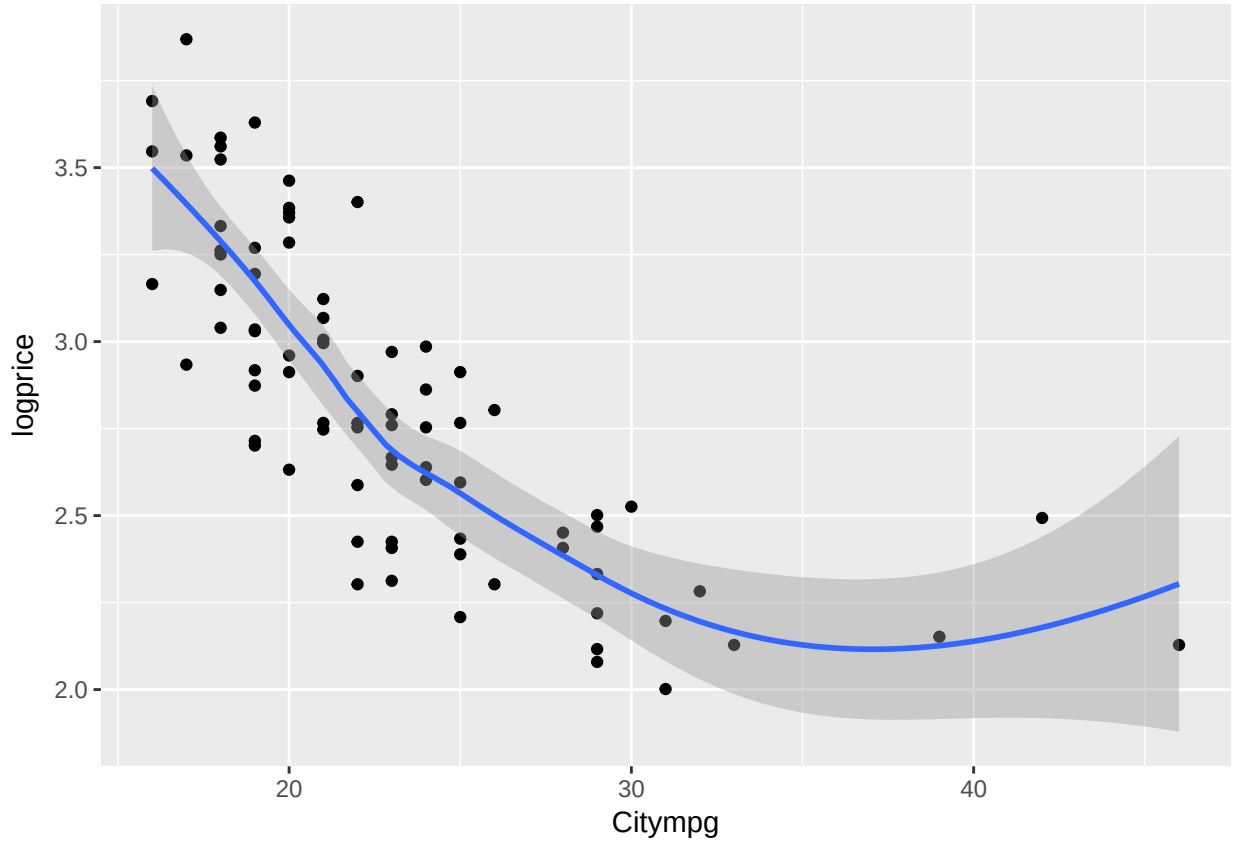
```
sub_data <- subset(cardata,select = c('logprice','Citympg','Hwympg','EngineSize','Horsepower','FuelTank
pairs(sub_data,panel=panel.smooth)
```



Now, let us plot some individual plots for better understanding. I have used the ggplot function in the R to plot the smoothing splines for the data which uses **loess** method to fit the spline.

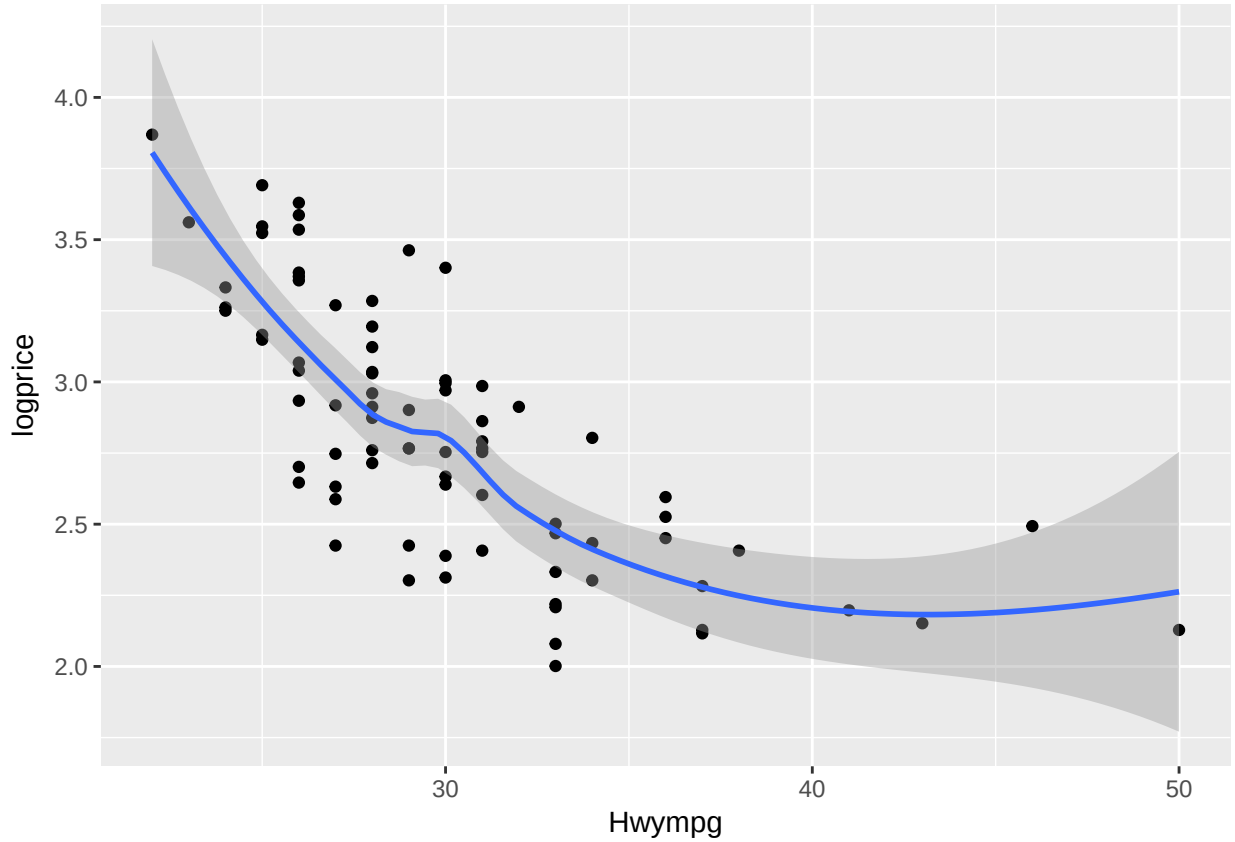
```
ggplot(sub_data,mapping = aes(y=logprice,x=Citympg))+geom_point()+geom_smooth()
```

```
## `geom_smooth()` using method = 'loess'
```



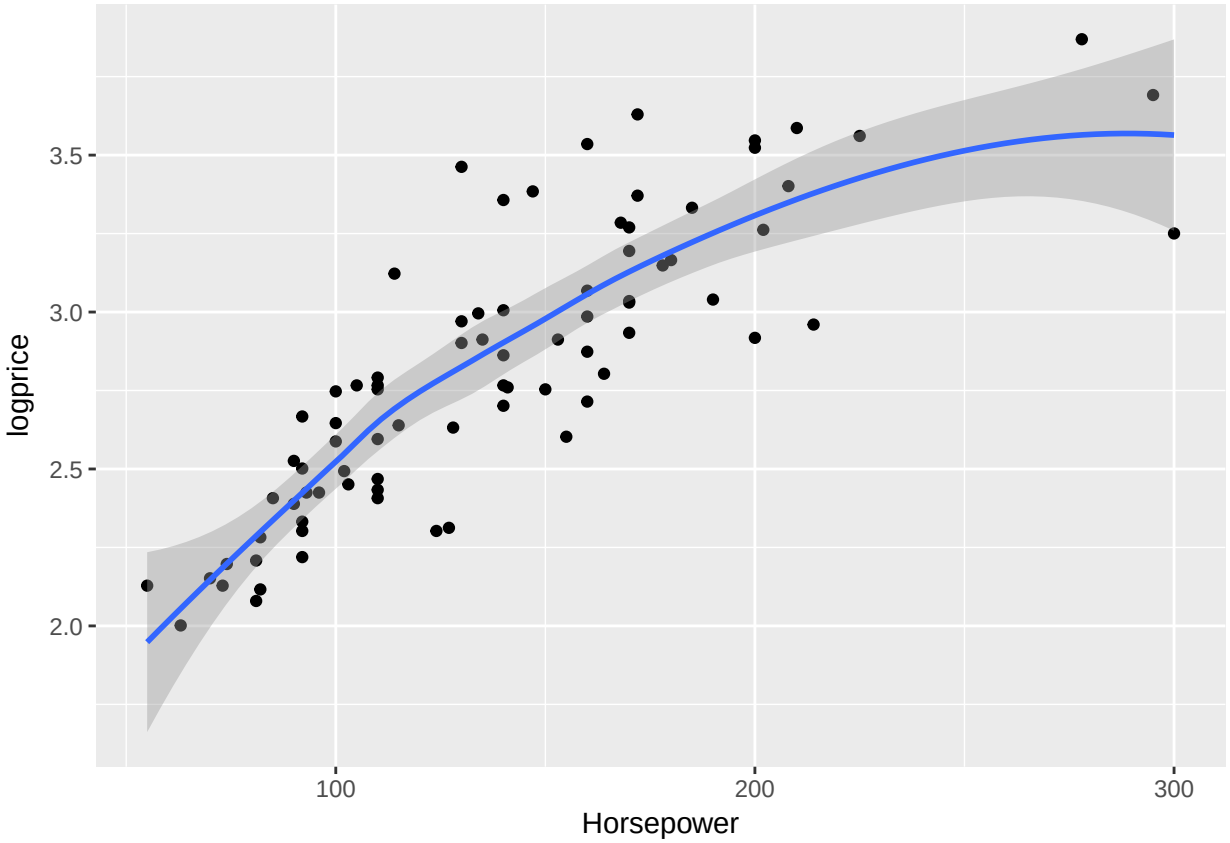

```
ggplot(sub_data,mapping = aes(y=logprice,x=Hwympg))+geom_point()+geom_smooth()
```

```
## `geom_smooth()` using method = 'loess'
```



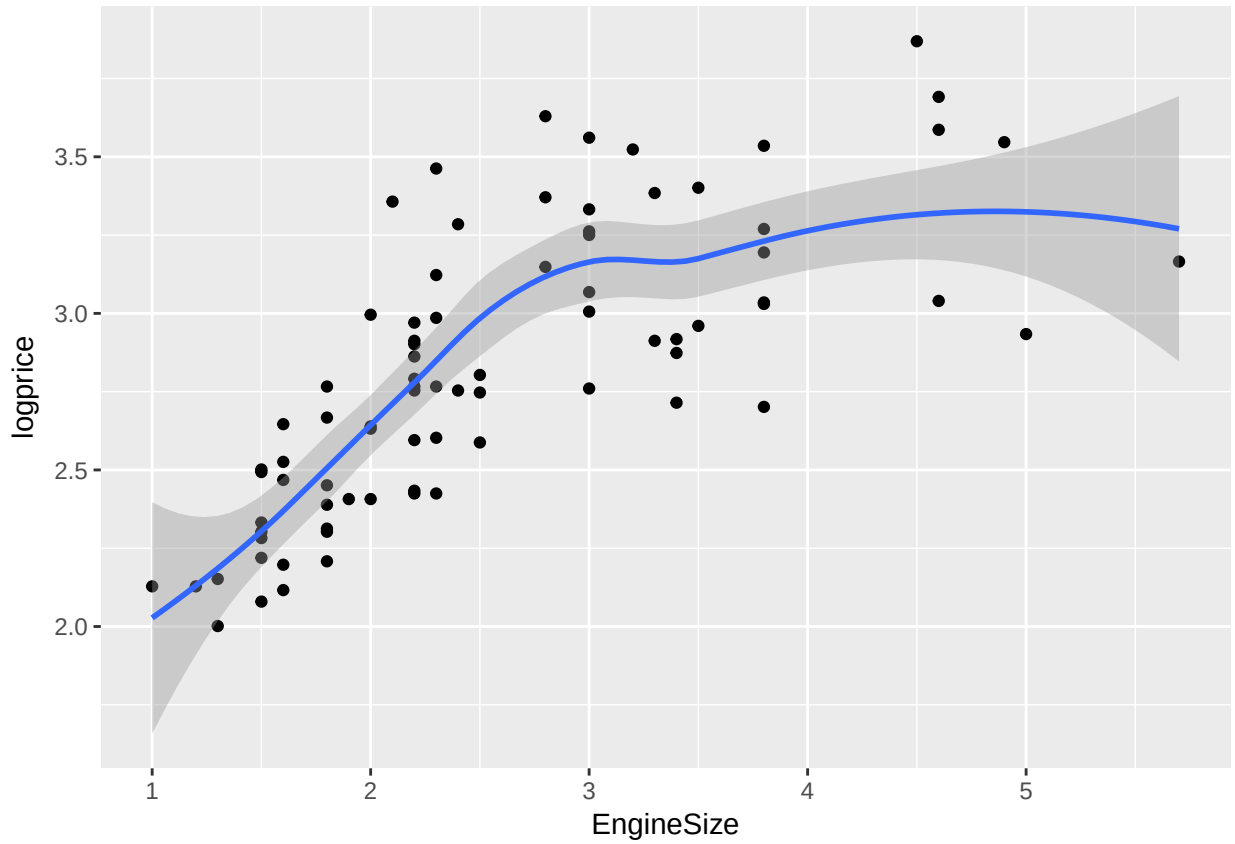

```
ggplot(sub_data,mapping = aes(y=logprice,x=Horsepower))+geom_point()+geom_smooth()
```

```
## `geom_smooth()` using method = 'loess'
```



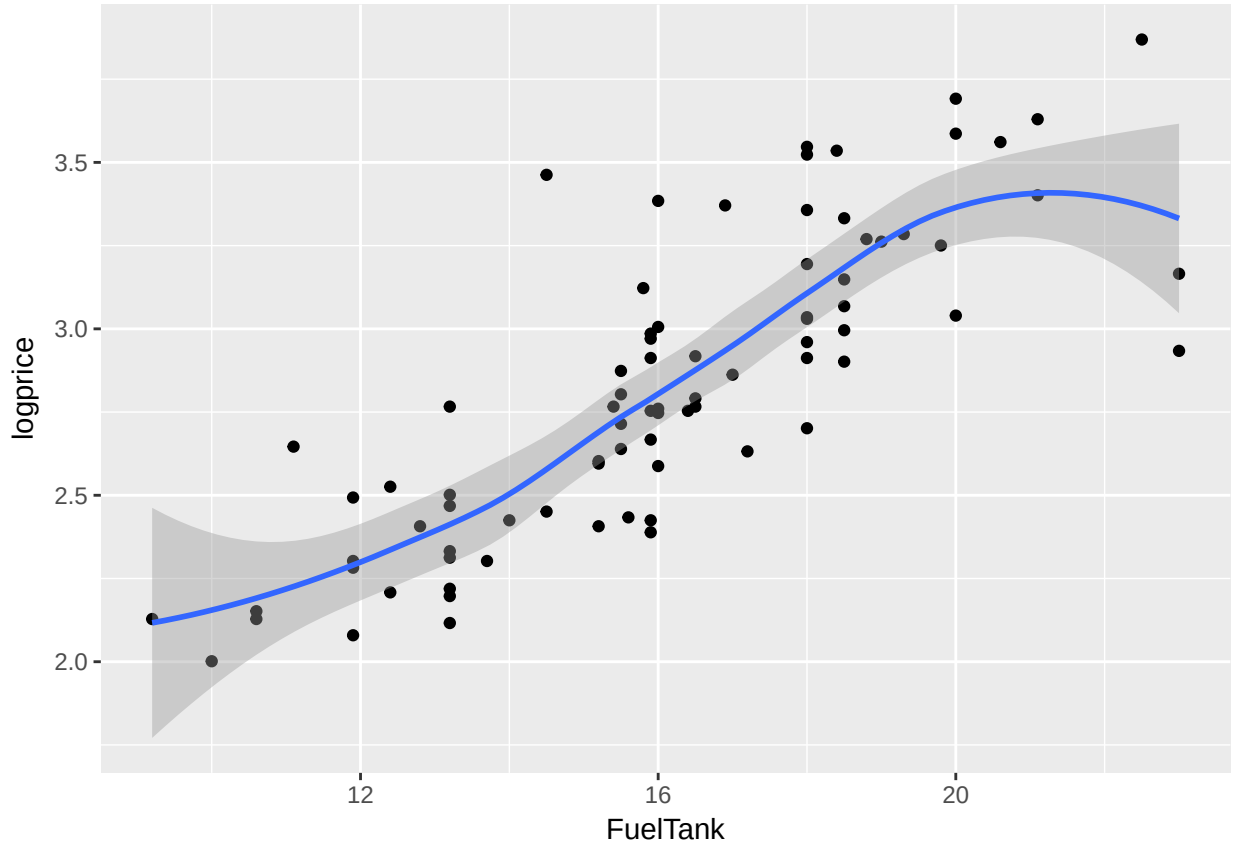

```
ggplot(sub_data,mapping = aes(y=logprice,x=EngineSize))+geom_point()+geom_smooth()
```

```
## `geom_smooth()` using method = 'loess'
```




```
ggplot(sub_data,mapping = aes(y=logprice,x=FuelTank))+geom_point()+geom_smooth()
```

```
## `geom_smooth()` using method = 'loess'
```



Now we know which variables to square. I would square `Citympg`, `Hwympg`, `Horsepower`,`FuelTank`. The variable `EngineSize` appears to have some complex shape than the normal curve with a degree 2.

Here are the squared variables.

```
cardata<- cars4[,-c(1,2,3,14)]
cardata$Citympg2<- cardata$Citympg^2
cardata$Hwympg2<- cardata$Hwympg^2
cardata$EngineSize2<- cardata$EngineSize^2
cardata$Horsepower2<- cardata$Horsepower^2
cardata$fueltank2<- cardata$FuelTank^2
cardata$EngineSize3<- cardata$EngineSize^3
cardata=cardata[-1]
head(cardata)
```

```
## # A tibble: 6 x 16
##   Citympg Hwympg Cylinders EngineSize Horsepower FuelTank Passengers
##     <int>  <int>     <int>      <dbl>      <int>    <dbl>      <int>
## 1      25     31         4        1.8        140     13.2          5
## 2      18     25         6        3.2        200     18.0          5
## 3      19     26         6        2.8        172     21.1          6
## 4      20     26         6        2.8        172     16.9          5
## 5      22     30         4        3.5        208     21.1          4
## 6      22     31         4        2.2        110     16.4          6
## # ... with 9 more variables: Luggage <int>, Weight <int>, logprice <dbl>,
## #   Citympg2 <dbl>, Hwympg2 <dbl>, EngineSize2 <dbl>, Horsepower2 <dbl>,
## #   fueltank2 <dbl>, EngineSize3 <dbl>
```

Standardize the variables with the robustHD package in R

```
ccc=standardize(cardata[,-c(3,7,8,10)], centerFun = mean, scaleFun = sd)
```

```
cardataa=cbind(ccc,cardata[,c(3,7,8,10)])
```

Let us perform a Backward Elimination method on the dataset

```
model <- lm(logprice ~ ., data = cardataa)
ols_step_backward(model)
```

```
## We are eliminating variables based on p value...
```

```
## No more variables satisfy the condition of prem: 0.3
```

```
## Backward Elimination Method
##
## Candidate Terms:
##
## 1 . Citympg
## 2 . Hwympg
## 3 . EngineSize
## 4 . Horsepower
## 5 . FuelTank
## 6 . Weight
## 7 . Citympg2
## 8 . Hwympg2
## 9 . EngineSize2
## 10 . Horsepower2
## 11 . fueltank2
## 12 . EngineSize3
## 13 . Cylinders
## 14 . Passengers
## 15 . Luggage
##
## ---------------------------------------------------------------------------
##                             Elimination Summary
## ---------------------------------------------------------------------------
##         Variable                    Adj.
## Step      Removed      R-Square    R-Square     C(p)        AIC        RMSE
## ---------------------------------------------------------------------------
##    1    EngineSize2      0.8183      0.7797    14.0035     -6.7040    0.2111
##    2    FuelTank         0.8181      0.7828    12.0542     -8.6409    0.2096
##    3    Passengers        0.818      0.7859    10.1022    -10.5811    0.2081
##    4    Hwympg2          0.8177      0.7887     8.1960    -12.4645    0.2068
##    5    Hwympg           0.8177      0.7916     6.2152    -14.4407    0.2053
##    6    EngineSize3      0.8171      0.7939     4.4157    -16.1920    0.2042
##    7    Luggage          0.8165      0.7961     2.6184    -17.9413    0.2031
##    8    Cylinders        0.8161      0.7985     0.7779    -19.7447    0.2019
## ---------------------------------------------------------------------------
```

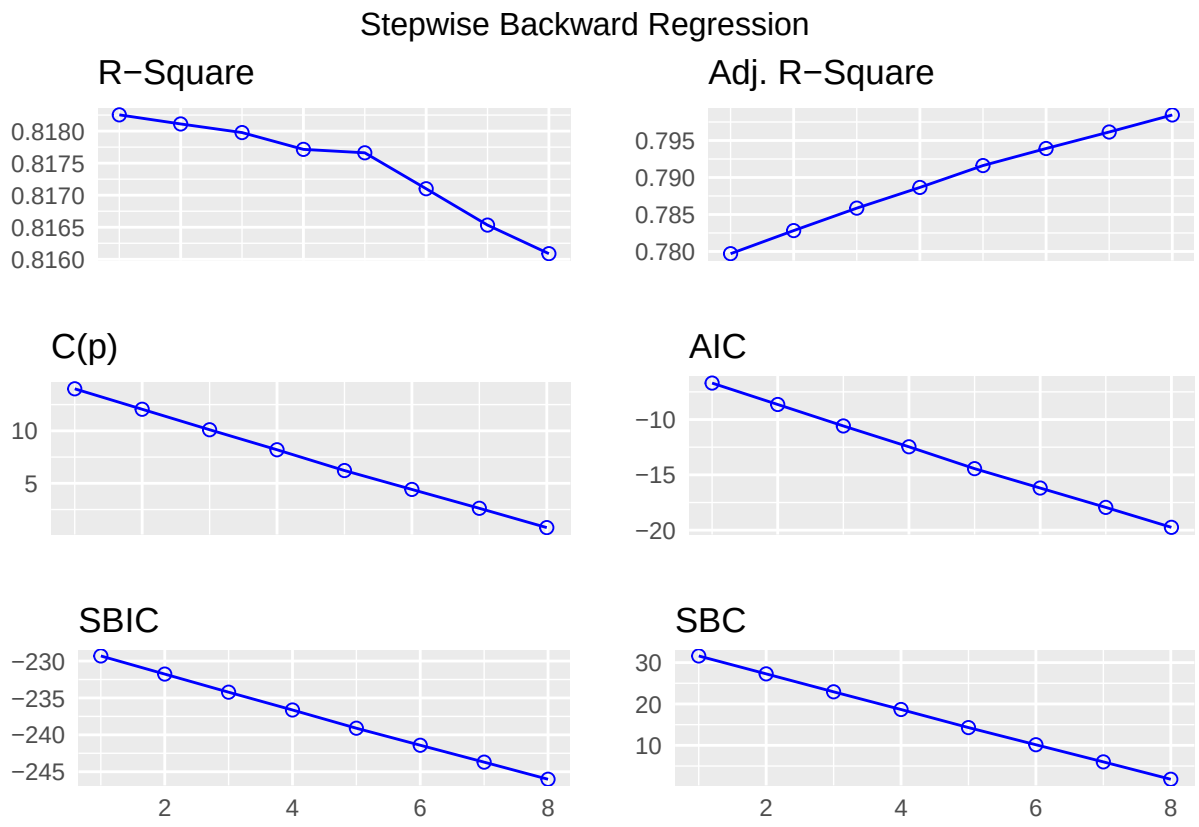
We now plot it.

```
k <- ols_step_backward(model)
```

```
## We are eliminating variables based on p value...
```

```
## No more variables satisfy the condition of prem: 0.3
```

```
plot(k)
```



Forward Selection on the dataset

```
model2 <- lm(logprice ~ ., data = cardataa)
ols_step_forward(model2,details = TRUE)
```

```
## We are selecting variables based on p value...
```

```
## 1 variable(s) added....
```

```
## Variable Selection Procedure
##  Dependent Variable: logprice
##
##  Forward Selection: Step 1
##
##  Variable Weight Entered
##
##                          Model Summary
## ----------------------------------------------------------------
```

```
## R                       0.850       RMSE                 0.239
## R-Squared               0.722       Coef. Var            8.453
## Adj. R-Squared          0.718       MSE                  0.057
## Pred R-Squared          0.707       MAE                  0.183
## ----------------------------------------------------------------
##  RMSE: Root Mean Square Error
##  MSE: Mean Square Error
##  MAE: Mean Absolute Error
##
##                                ANOVA
## -------------------------------------------------------------------------
##                Sum of
##               Squares        DF    Mean Square       F         Sig.
## -------------------------------------------------------------------------
## Regression     11.681         1         11.681    205.018    0.0000
## Residual        4.501        79          0.057
## Total          16.182        80
## -------------------------------------------------------------------------
##
##                                 Parameter Estimates
## ----------------------------------------------------------------------------------------------
##       model     Beta    Std. Error    Std. Beta       t        Sig     lower    upper
## ----------------------------------------------------------------------------------------------
## (Intercept)    2.824         0.027                 106.475    0.000    2.771    2.877
##      Weight    0.382         0.027        0.850     14.318    0.000    0.329    0.435
## ----------------------------------------------------------------------------------------------

## 1 variable(s) added...

## Forward Selection: Step 2
##
##  Variable Horsepower Entered
##
##                          Model Summary
## ----------------------------------------------------------------
## R                       0.872       RMSE                 0.223
## R-Squared               0.761       Coef. Var            7.888
## Adj. R-Squared          0.755       MSE                  0.050
## Pred R-Squared          0.736       MAE                  0.171
## ----------------------------------------------------------------
##  RMSE: Root Mean Square Error
##  MSE: Mean Square Error
##  MAE: Mean Absolute Error
##
##                                ANOVA
## -------------------------------------------------------------------------
##                Sum of
##               Squares        DF    Mean Square       F         Sig.
## -------------------------------------------------------------------------
## Regression     12.312         2           6.156    124.064    0.0000
## Residual        3.870        78           0.050
## Total          16.182        80
## -------------------------------------------------------------------------
##
##                                 Parameter Estimates
```

```
## ---------------------------------------------------------------------------------------
##       model     Beta    Std. Error    Std. Beta       t        Sig     lower    upper
## ---------------------------------------------------------------------------------------
## (Intercept)    2.824         0.025                114.096    0.000    2.775    2.873
##      Weight    0.233         0.049        0.519     4.800    0.000    0.137    0.330
##  Horsepower    0.173         0.049        0.385     3.566    0.001    0.077    0.270
## ---------------------------------------------------------------------------------------

## 1 variable(s) added...

## Forward Selection: Step 3
##
##  Variable EngineSize2 Entered
##
##                         Model Summary
## ----------------------------------------------------------------
## R                       0.880       RMSE                0.218
## R-Squared               0.774       Coef. Var           7.719
## Adj. R-Squared          0.765       MSE                 0.048
## Pred R-Squared          0.738       MAE                 0.164
## ----------------------------------------------------------------
##  RMSE: Root Mean Square Error
##  MSE: Mean Square Error
##  MAE: Mean Absolute Error
##
##                                ANOVA
## -------------------------------------------------------------------
##                Sum of
##               Squares        DF    Mean Square      F         Sig.
## -------------------------------------------------------------------
## Regression     12.523         3          4.174    87.851    0.0000
## Residual        3.659        77          0.048
## Total          16.182        80
## -------------------------------------------------------------------
##
##                                   Parameter Estimates
## ---------------------------------------------------------------------------------------------
##       model      Beta    Std. Error    Std. Beta       t        Sig      lower     upper
## ---------------------------------------------------------------------------------------------
## (Intercept)     2.824         0.024                116.591    0.000     2.776     2.872
##      Weight     0.322         0.064        0.716     5.071    0.000     0.196     0.449
##  Horsepower     0.166         0.048        0.369     3.482    0.001     0.071     0.261
## EngineSize2    -0.097         0.046       -0.216    -2.109    0.038    -0.189    -0.005
## ---------------------------------------------------------------------------------------------

## 1 variable(s) added...

## Forward Selection: Step 4
##
##  Variable FuelTank Entered
##
##                         Model Summary
## ----------------------------------------------------------------
## R                       0.886       RMSE                0.214
## R-Squared               0.784       Coef. Var           7.591
## Adj. R-Squared          0.773       MSE                 0.046
```

```
## Pred R-Squared          0.738       MAE                 0.160
## ----------------------------------------------------------------
##  RMSE: Root Mean Square Error
##  MSE: Mean Square Error
##  MAE: Mean Absolute Error
##
##                               ANOVA
## ---------------------------------------------------------------------
##                Sum of
##               Squares        DF    Mean Square      F         Sig.
## ---------------------------------------------------------------------
## Regression     12.690         4          3.172    69.048    0.0000
## Residual        3.492        76          0.046
## Total          16.182        80
## ---------------------------------------------------------------------
##
##                                  Parameter Estimates
## ----------------------------------------------------------------------------------------------
##       model      Beta    Std. Error    Std. Beta       t        Sig      lower     upper
## ----------------------------------------------------------------------------------------------
## (Intercept)     2.824         0.024                 118.567    0.000     2.776     2.871
##      Weight     0.235         0.077        0.523      3.045    0.003     0.081     0.389
##  Horsepower     0.157         0.047        0.349      3.330    0.001     0.063     0.251
## EngineSize2    -0.097         0.045       -0.216     -2.141    0.035    -0.188    -0.007
##    FuelTank     0.105         0.055        0.233      1.906    0.060    -0.005     0.215
## ----------------------------------------------------------------------------------------------

## 1 variable(s) added...

## Forward Selection: Step 5
##
##  Variable Horsepower2 Entered
##
##                          Model Summary
## ----------------------------------------------------------------
## R                       0.889       RMSE                0.213
## R-Squared               0.790       Coef. Var           7.530
## Adj. R-Squared          0.776       MSE                 0.045
## Pred R-Squared          0.732       MAE                 0.160
## ----------------------------------------------------------------
##  RMSE: Root Mean Square Error
##  MSE: Mean Square Error
##  MAE: Mean Absolute Error
##
##                               ANOVA
## ---------------------------------------------------------------------
##                Sum of
##               Squares        DF    Mean Square      F         Sig.
## ---------------------------------------------------------------------
## Regression     12.791         5          2.558    56.576    0.0000
## Residual        3.391        75          0.045
## Total          16.182        80
## ---------------------------------------------------------------------
##
##                                  Parameter Estimates
```

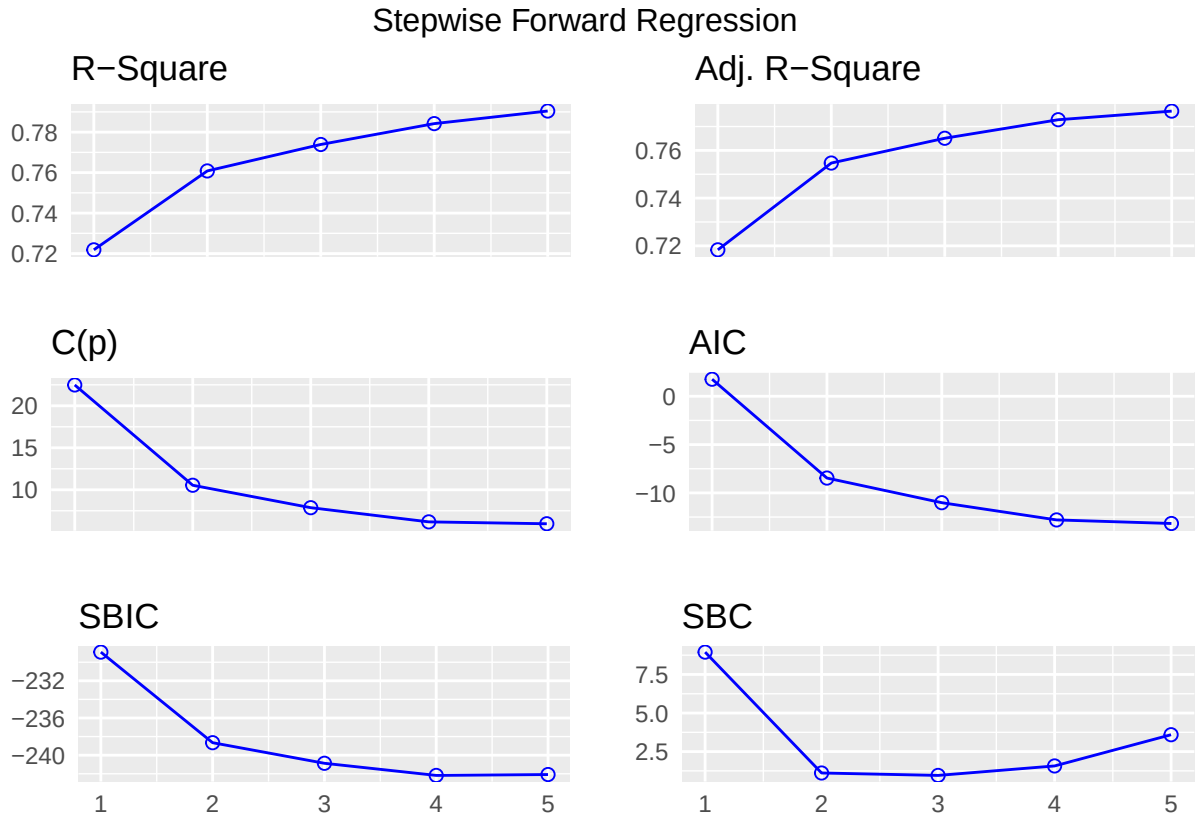
```
## ----------------------------------------------------------------------------------------
##       model      Beta    Std. Error    Std. Beta       t        Sig      lower    upper
## ----------------------------------------------------------------------------------------
## (Intercept)     2.824         0.024                 119.521    0.000     2.777    2.871
##      Weight     0.181         0.085         0.402     2.128    0.037     0.012    0.350
##  Horsepower     0.390         0.163         0.866     2.394    0.019     0.065    0.714
## EngineSize2    -0.081         0.046        -0.180    -1.753    0.084    -0.173    0.011
##    FuelTank     0.097         0.055         0.217     1.774    0.080    -0.012    0.207
## Horsepower2    -0.196         0.131        -0.436    -1.492    0.140    -0.457    0.066
## ----------------------------------------------------------------------------------------

## No more variables satisfy the condition of penter: 0.3

## Forward Selection Method
##
## Candidate Terms:
##
## 1 . Citympg
## 2 . Hwympg
## 3 . EngineSize
## 4 . Horsepower
## 5 . FuelTank
## 6 . Weight
## 7 . Citympg2
## 8 . Hwympg2
## 9 . EngineSize2
## 10 . Horsepower2
## 11 . fueltank2
## 12 . EngineSize3
## 13 . Cylinders
## 14 . Passengers
## 15 . Luggage
##
## ---------------------------------------------------------------------------
##                            Selection Summary
## ---------------------------------------------------------------------------
##         Variable                    Adj.
## Step      Entered      R-Square    R-Square     C(p)        AIC        RMSE
## ---------------------------------------------------------------------------
##    1    Weight           0.7218      0.7183    22.4837     1.7658    0.2387
##    2    Horsepower       0.7608      0.7547    10.5415    -8.4646    0.2228
##    3    EngineSize2      0.7739      0.7651     7.8682   -11.0153    0.2180
##    4    FuelTank         0.7842      0.7729     6.1802   -12.7962    0.2144
##    5    Horsepower2      0.7904      0.7765     5.9540   -13.1669    0.2126
## ---------------------------------------------------------------------------
```

```
l<- ols_step_forward(model2)
```

```
## We are selecting variables based on p value...

## 1 variable(s) added....

## 1 variable(s) added...
## 1 variable(s) added...
## 1 variable(s) added...
## 1 variable(s) added...
```

```
## No more variables satisfy the condition of penter: 0.3
```

```
plot(l)
```



Stepwise Selection

```
model3 <- lm(logprice ~ ., data = cardataa)
ols_stepwise(model3)
```

```
## We are selecting variables based on p value...

## 1 variable(s) added....

## 1 variable(s) added...
## 1 variable(s) added...
## 1 variable(s) added...

## No more variables to be added or removed.

## Stepwise Selection Method
##
## Candidate Terms:
##
## 1 . Citympg
## 2 . Hwympg
## 3 . EngineSize
## 4 . Horsepower
```

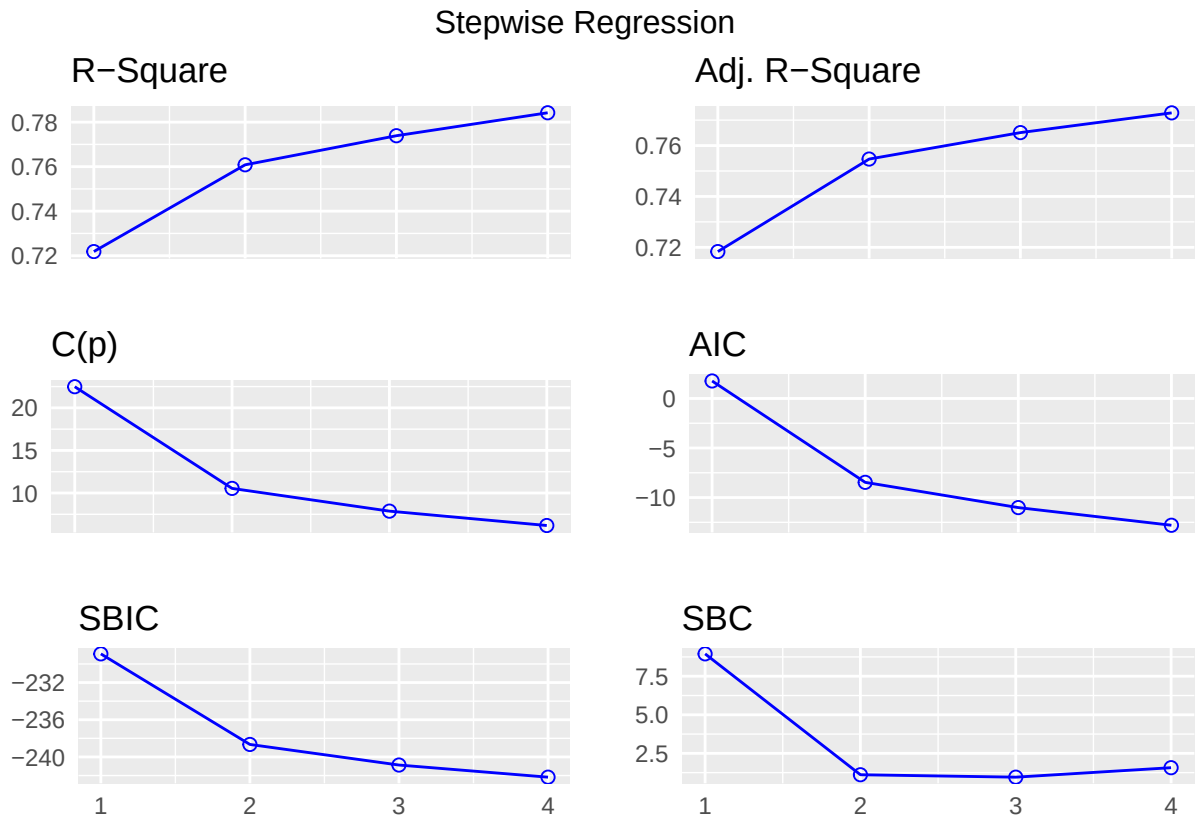
```
## 5 . FuelTank
## 6 . Weight
## 7 . Citympg2
## 8 . Hwympg2
## 9 . EngineSize2
## 10 . Horsepower2
## 11 . fueltank2
## 12 . EngineSize3
## 13 . Cylinders
## 14 . Passengers
## 15 . Luggage
##
## ----------------------------------------------------------------------------------------
##                                Stepwise Selection Summary
## ----------------------------------------------------------------------------------------
##                         Added/                   Adj.
## Step     Variable      Removed     R-Square    R-Square     C(p)        AIC        RMSE
## ----------------------------------------------------------------------------------------
##    1      Weight       addition       0.722       0.718    22.4840     1.7658    0.2387
##    2    Horsepower     addition       0.761       0.755    10.5410     -8.4646    0.2228
##    3    EngineSize2    addition       0.774       0.765     7.8680    -11.0153    0.2180
##    4     FuelTank      addition       0.784       0.773     6.1800    -12.7962    0.2144
## ----------------------------------------------------------------------------------------
```

Let us plot the stepwise model.

```
m<-ols_stepwise(model3)
```

```
## We are selecting variables based on p value...

## 1 variable(s) added....

## 1 variable(s) added...
## 1 variable(s) added...
## 1 variable(s) added...

## No more variables to be added or removed.
```

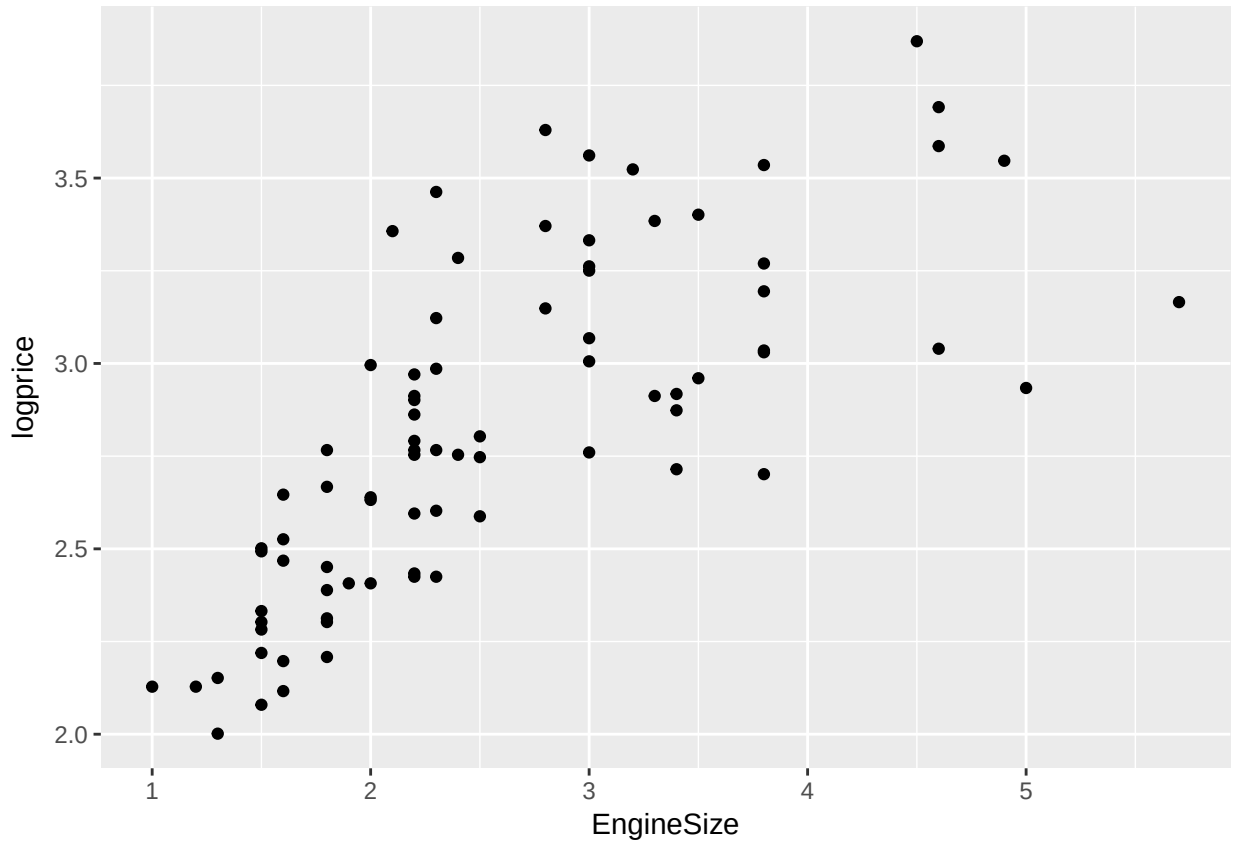
```
plot(m)
```



After looking at the three methods of variable selection/Elimination I came to a conclusion that Backward Elimination model was the best with the Adjusted R.sq value of 0.7985, whereas Forward Selection had an Adj R.sq of 0.7765 and the Stepwise model had 0.773 Adj R.sq value. So the important variables that can be placed in the model according to the backward selection are Citympg, Citympg^2, EngineSize,Weight,Horssepower,Horsepower^2. Ofcourse the intercept.

Engine Size plot (Checking whether it overfits the data)

```
ggplot(sub_data, aes(x=EngineSize, y=logprice))+geom_point()+
  stat_smooth(method=lm, formula=logprice~ns(x,5))
```



Plot of EngineSize and logprice wih nknots=5 does not overfit the data and looks more smoother now.