Assignment 2

Viviana Romero Alarcon

04 octubre 2023

Question 1:

Consider the simple linear regression problem $y = \beta_0 + \beta_1 x + \epsilon$, where $\epsilon \sim \mathcal{N}(0, \sigma^2)$. Assume we are given training data $\{(x_i, y_i)\}_{i=1}^n$ and let $\hat{y}_i = \hat{\beta}_0 + \hat{\beta}_1 x_i$ be the fitted model. Define $e_i = y_i - \hat{y}_i$. Show the following:

1. $\sum_{i=1}^n e_i = 0$.

$$
\begin{aligned}
\sum_{i=1}^{n} e_i &= \sum_{i=1}^{n} (y_i - \hat{y}_i) = 0 \\
&= \sum_{i=1}^{n} y_i - \sum_{i=1}^{n} \hat{y}_i \\
&= \sum_{i=1}^{n} y_i - \sum_{i=1}^{n} (\hat{\beta}_0 + \hat{\beta}_1 x_i) \\
&= \sum_{i=1}^{n} y_i - \sum_{i=1}^{n} [(\bar{y} - \hat{\beta}_1 x_i) + \hat{\beta}_1 x_i] \\
&= \sum_{i=1}^{n} y_i - \sum_{i=1}^{n} (\bar{y} - \hat{\beta}_1 x_i) + \sum_{i=1}^{n} \hat{\beta}_1 x_i \\
&= \sum_{i=1}^{n} y_i - \sum_{i=1}^{n} \bar{y} - \sum_{i=1}^{n} \hat{\beta}_1 x_i + \sum_{i=1}^{n} \hat{\beta}_1 x_i \\
&= \sum_{i=1}^{n} y_i - \bar{y}n - \hat{\beta}_1 \sum_{i=1}^{n} x_i + \hat{\beta}_1 \sum_{i=1}^{n} x_i \\
&= \bar{y}n - \bar{y}n - \hat{\beta}_1 \bar{x}n + \hat{\beta}_1 \bar{x}n \\
\sum_{i=1}^{n} e_i &= 0
\end{aligned}
$$

2. The regression line always goes through the point $(\bar{x}, \bar{y})$.

Considering:

1. $\hat{y}_i = \hat{\beta}_0 + \hat{\beta}_1 x_i$.
2. $RSS = \sum_{i=1}^n (y_i - \bar{y}_i)^2 = \sum_{i=1}^n (y_i - \hat{\beta}_0 - \hat{\beta}_1 x_i)^2$.
3. $\frac{\partial (c-x)^n}{\partial x} = n(c-x)^{n-1} \frac{\partial x}{\partial x}$.

Then, we should minimize the slope

$$
\begin{aligned}
min\hat{\beta}_0 = 0 &= \frac{\partial \sum_{i=1}^{n}(y_i - \hat{\beta}_0 - \hat{\beta}_1 x_i)^2}{\partial \hat{\beta}_0} \\
0 &= \sum_{i=1}^{n} \frac{\partial (y_i - \hat{\beta}_0 - \hat{\beta}_1 x_i)^2}{\partial \hat{\beta}_0} \\
0 &= \sum_{i=1}^{n} -2(y_i - \hat{\beta}_0 - \hat{\beta}_1 x_i)\frac{\partial \hat{\beta}_0}{\partial \hat{\beta}_0} \\
0 &= \sum_{i=1}^{n} -2(y_i - \hat{\beta}_0 - \hat{\beta}_1 x_i).1 \\
0 &= -2\sum_{i=1}^{n}(y_i - \hat{\beta}_0 - \hat{\beta}_1 x_i) \\
0 &= -2\sum_{i=1}^{n} y_i - \sum_{i=1}^{n}\hat{\beta}_0 - \sum_{i=1}^{n}\hat{\beta}_1 x_i \\
\frac{0}{-2} &= \bar{y}n - \hat{\beta}_0 n - \hat{\beta}_1 \bar{x} n \\
0 &= \bar{y}n - \hat{\beta}_0 n - \hat{\beta}_1 \bar{x} n \\
0 &= n(\bar{y} - \hat{\beta}_0 - \hat{\beta}_1 \bar{x}) \\
\frac{0}{n} &= \bar{y} - \hat{\beta}_0 - \hat{\beta}_1 \bar{x} \\
0 &= \bar{y} - \hat{\beta}_0 - \hat{\beta}_1 \bar{x} \\
\hat{\beta}_0 &= \bar{y} - \hat{\beta}_1 \bar{x} \\
\hat{\beta}_0 + \hat{\beta}_1 \bar{x} = \bar{y} &= point(\bar{x}\bar{y})
\end{aligned}
$$

Question 2:

This question involves the use of multiple linear regression on `Auto` data set. For most of the analysis, you will need to remove the name variable.

```
library(pacman)
pacman::p_load(ggplot2, ISLR, ggcorrplot)

#load data
data(Auto)

#structure of Auto
str(Auto)
```

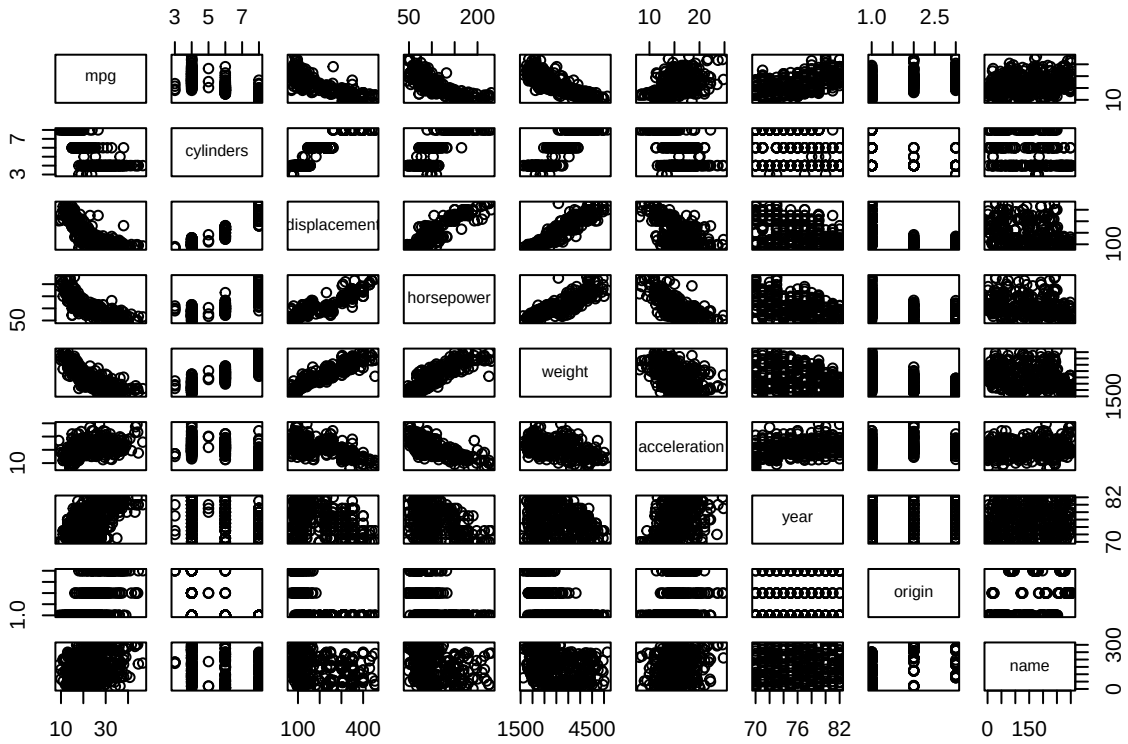
```
## 'data.frame':    392 obs. of  9 variables:
##  $ mpg         : num  18 15 18 16 17 15 14 14 14 15 ...
##  $ cylinders   : num  8 8 8 8 8 8 8 8 8 8 ...
##  $ displacement: num  307 350 318 304 302 429 454 440 455 390 ...
##  $ horsepower  : num  130 165 150 150 140 198 220 215 225 190 ...
##  $ weight      : num  3504 3693 3436 3433 3449 ...
##  $ acceleration: num  12 11.5 11 12 10.5 10 9 8.5 10 8.5 ...
##  $ year        : num  70 70 70 70 70 70 70 70 70 70 ...
##  $ origin      : num  1 1 1 1 1 1 1 1 1 1 ...
##  $ name        : Factor w/ 304 levels "amc ambassador brougham",..: 49 36 231 14 161 141 54 223 241 1
```

```
colnames(Auto)
```

```
## [1] "mpg"          "cylinders"    "displacement" "horsepower"   "weight"
## [6] "acceleration" "year"         "origin"       "name"
```

1. Produce a scatterplot matrix of all of the variables (you can use the `pair()` command for this.

```
# using pairs() for scatterplot matrix
pairs(Auto)
```



2. Use the `lm()` function to perform a multiple linear regression with mpg as the response variable. Use the `summary()` command to print the results. Comment on your findings.

```
models <- list()

# Creating additive models

models$model1 <- lm(data = Auto,formula = mpg ~ cylinders + displacement )

models$model2 <- lm(data = Auto,formula = mpg ~ cylinders + displacement + horsepower  )

models$model3 <- lm(data = Auto,formula = mpg ~ cylinders + displacement + horsepower + acceleration )


models$model4 <- lm(data = Auto,formula = mpg ~ cylinders + displacement + horsepower + acceleration + w

models$model5 <- lm(data = Auto,formula = mpg ~ cylinders  + weight +  displacement + year)

models$model6 <- lm(data = Auto,formula = mpg ~ cylinders +  displacement + horsepower + acceleration +

# See summary
lapply(models, summary)
```

```
## $model1
##
## Call:
## lm(formula = mpg ~ cylinders + displacement, data = Auto)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -13.2304  -3.0383  -0.5243   2.4307  18.3134
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  36.537707   1.196611  30.534  < 2e-16 ***
## cylinders    -0.576348   0.443276  -1.300    0.194
## displacement -0.051118   0.007226  -7.074 7.02e-12 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 4.631 on 389 degrees of freedom
## Multiple R-squared:  0.6498, Adjusted R-squared:  0.648
## F-statistic: 360.8 on 2 and 389 DF,  p-value: < 2.2e-16
##
##
## $model2
##
## Call:
## lm(formula = mpg ~ cylinders + displacement + horsepower, data = Auto)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -11.7144  -3.1391  -0.3149   2.3481  16.5726
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  39.305268   1.324633  29.673  < 2e-16 ***
## cylinders    -0.719431   0.434180  -1.657 0.098331 .
## displacement -0.029120   0.008623  -3.377 0.000807 ***
## horsepower   -0.059935   0.013498  -4.440 1.17e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 4.523 on 388 degrees of freedom
## Multiple R-squared:  0.6667, Adjusted R-squared:  0.6641
## F-statistic: 258.7 on 3 and 388 DF,  p-value: < 2.2e-16
##
##
## $model3
##
## Call:
## lm(formula = mpg ~ cylinders + displacement + horsepower + acceleration,
##     data = Auto)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -12.4705  -3.2232  -0.4872   2.2327  16.7881
```

```
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  48.02130    2.78672  17.232  < 2e-16 ***
## cylinders    -0.70624    0.42788  -1.651 0.099644 .
## displacement -0.02519    0.00857  -2.940 0.003483 **
## horsepower   -0.09029    0.01583  -5.705 2.31e-08 ***
## acceleration -0.41056    0.11597  -3.540 0.000449 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 4.458 on 387 degrees of freedom
## Multiple R-squared:  0.6771, Adjusted R-squared:  0.6738
## F-statistic: 202.9 on 4 and 387 DF,  p-value: < 2.2e-16
##
##
## $model4
##
## Call:
## lm(formula = mpg ~ cylinders + displacement + horsepower + acceleration +
##     weight, data = Auto)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -11.5816  -2.8618  -0.3404   2.2438  16.3416
##
## Coefficients:
##                Estimate Std. Error t value Pr(>|t|)
## (Intercept)   4.626e+01  2.669e+00  17.331   <2e-16 ***
## cylinders    -3.979e-01  4.105e-01  -0.969   0.3330
## displacement -8.313e-05  9.072e-03  -0.009   0.9927
## horsepower   -4.526e-02  1.666e-02  -2.716   0.0069 **
## acceleration -2.910e-02  1.258e-01  -0.231   0.8171
## weight       -5.187e-03  8.167e-04  -6.351    6e-10 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 4.247 on 386 degrees of freedom
## Multiple R-squared:  0.7077, Adjusted R-squared:  0.7039
## F-statistic: 186.9 on 5 and 386 DF,  p-value: < 2.2e-16
##
##
## $model5
##
## Call:
## lm(formula = mpg ~ cylinders + weight + displacement + year,
##     data = Auto)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -9.0169 -2.2958 -0.0967  2.0400 14.4239
##
## Coefficients:
##                Estimate Std. Error t value Pr(>|t|)
```

```
## (Intercept)  -1.369e+01  4.079e+00  -3.357 0.000868 ***
## cylinders    -3.217e-01  3.299e-01  -0.975 0.330182
## weight       -6.612e-03  5.735e-04 -11.531  < 2e-16 ***
## displacement  4.888e-03  6.695e-03   0.730 0.465727
## year          7.586e-01  5.101e-02  14.872  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.432 on 387 degrees of freedom
## Multiple R-squared:  0.8087, Adjusted R-squared:  0.8067
## F-statistic: 408.9 on 4 and 387 DF,  p-value: < 2.2e-16
##
##
## $model6
##
## Call:
## lm(formula = mpg ~ cylinders + displacement + horsepower + acceleration +
##     weight + year, data = Auto)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -8.6927 -2.3864 -0.0801  2.0291 14.3607
##
## Coefficients:
##                Estimate Std. Error t value Pr(>|t|)
## (Intercept)  -1.454e+01  4.764e+00  -3.051  0.00244 **
## cylinders    -3.299e-01  3.321e-01  -0.993  0.32122
## displacement  7.678e-03  7.358e-03   1.044  0.29733
## horsepower   -3.914e-04  1.384e-02  -0.028  0.97745
## acceleration  8.527e-02  1.020e-01   0.836  0.40383
## weight       -6.795e-03  6.700e-04 -10.141  < 2e-16 ***
## year          7.534e-01  5.262e-02  14.318  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.435 on 385 degrees of freedom
## Multiple R-squared:  0.8093, Adjusted R-squared:  0.8063
## F-statistic: 272.2 on 6 and 385 DF,  p-value: < 2.2e-16
```

```
# Choose a model
lapply(models, AIC)
```

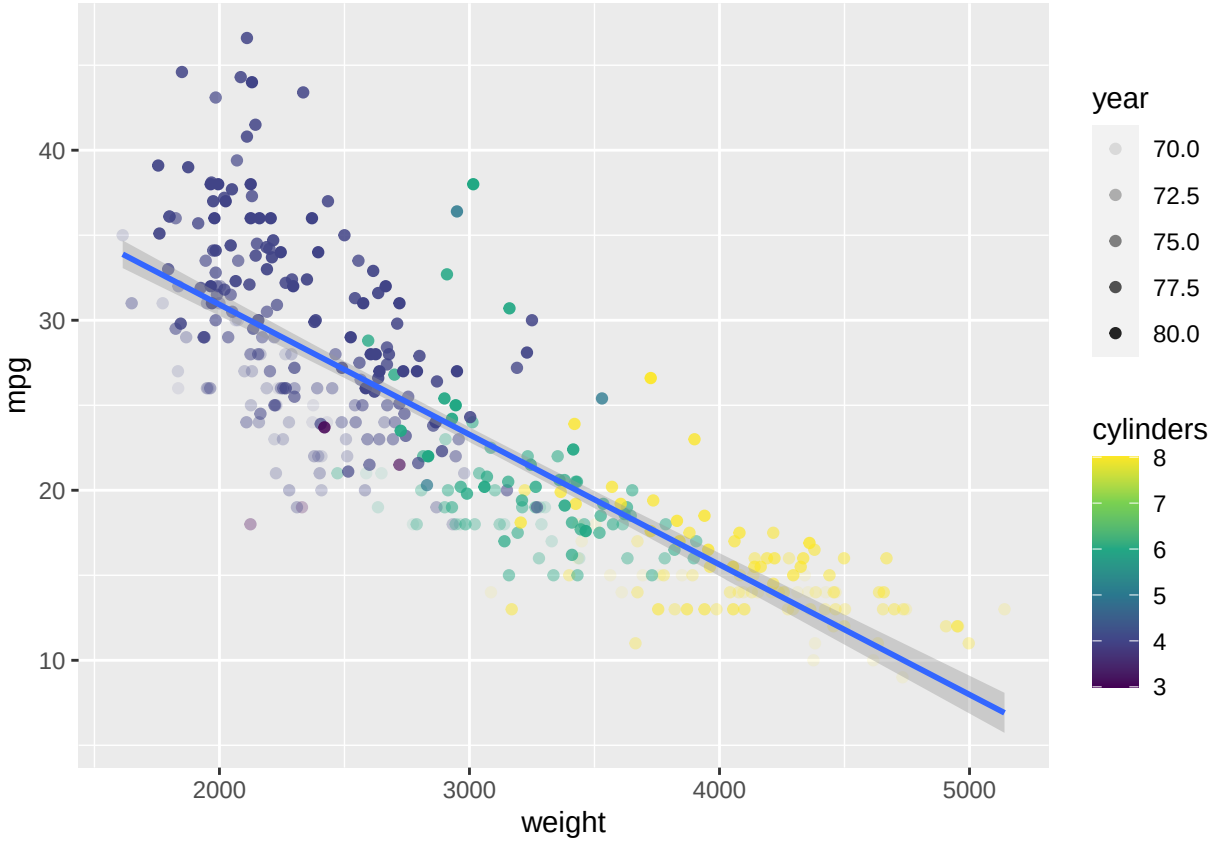
```
## $model1
## [1] 2319.131
##
## $model2
## [1] 2301.702
##
## $model3
## [1] 2291.209
##
## $model4
## [1] 2254.243
##
## $model5
```

```
## [1] 2086.147
##
## $model6
## [1] 2088.91
```

NOTE: Considering the AIC, I will choose model 5 to develop exercises 3 and 4

3. Use the `ggplot()` package to plot your findings.

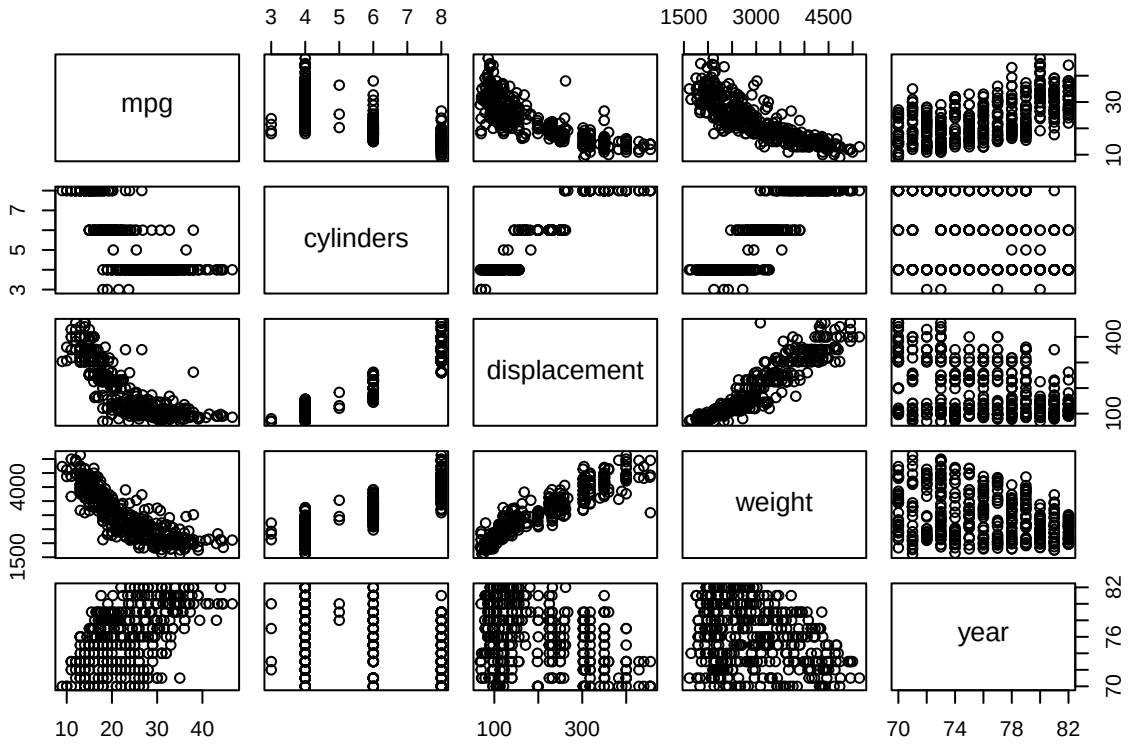
```
# Plot model5: mpg ~ cylinders +  displacement + weight + year
ggplot(Auto, aes(x = weight, y = mpg)) +
  geom_point(aes(col = cylinders, alpha = year)) +
  geom_smooth(method = lm, formula = y ~ x)+
  scale_color_viridis_c()
```



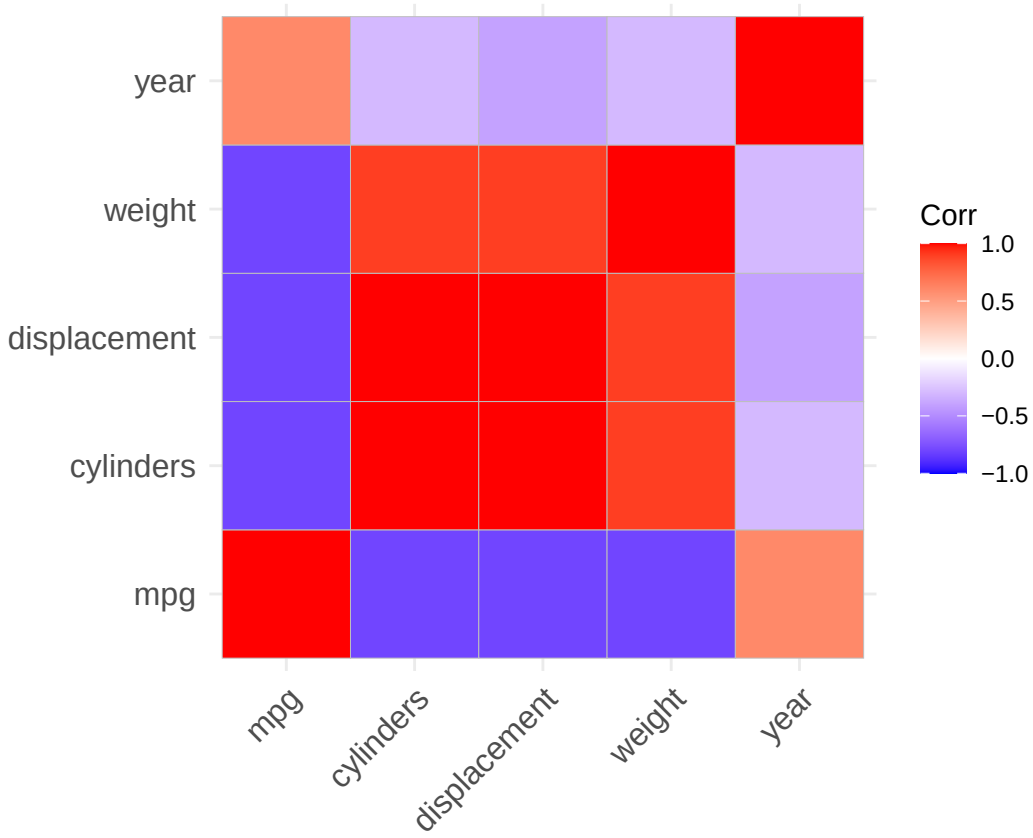
4. Based on the correlation matrix and the scatterplots, try transformation of the predictors (e.g, X^2 or $\sqrt{X}$). Comment on your findings.

```
# filter varibales from model5
var_model5 <- Auto[,c(1,2,3,5,7)]


# scatterplot for variables in model5
pairs(var_model5 )
```




```
# Correlation matrix
cor_model5 <- round(cor(var_model5 ), 1)
ggcorrplot(cor_model5)
```



```r
# Considering different transformation for the variable weight
summary(lm(data = Auto, mpg ~ weight))
```

```
##
## Call:
## lm(formula = mpg ~ weight, data = Auto)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -11.9736  -2.7556  -0.3358   2.1379  16.5194
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 46.216524   0.798673   57.87   <2e-16 ***
## weight      -0.007647   0.000258  -29.64   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 4.333 on 390 degrees of freedom
## Multiple R-squared:  0.6926, Adjusted R-squared:  0.6918
## F-statistic: 878.8 on 1 and 390 DF,  p-value: < 2.2e-16
```

```r
summary(lm(data = Auto, mpg ~ weight^2))
```

```
##
## Call:
## lm(formula = mpg ~ weight^2, data = Auto)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -11.9736  -2.7556  -0.3358   2.1379  16.5194
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 46.216524   0.798673   57.87   <2e-16 ***
## weight      -0.007647   0.000258  -29.64   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 4.333 on 390 degrees of freedom
## Multiple R-squared:  0.6926, Adjusted R-squared:  0.6918
## F-statistic: 878.8 on 1 and 390 DF,  p-value: < 2.2e-16
```

```r
summary(lm(data = Auto, mpg ~ log(weight)))
```

```
##
## Call:
## lm(formula = mpg ~ log(weight), data = Auto)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -12.4315  -2.6752  -0.2888   1.9429  16.0136
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
```

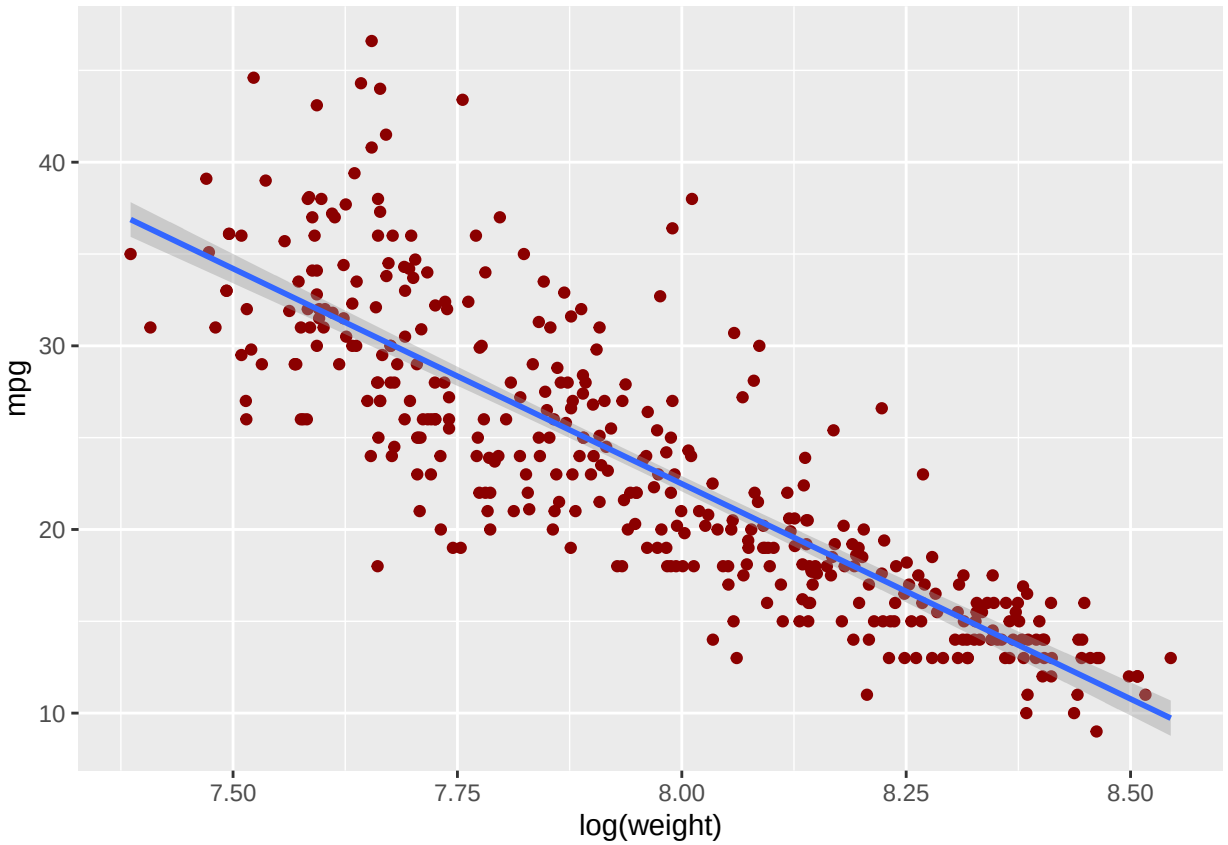
```
## (Intercept) 209.9433     6.0002   34.99   <2e-16 ***
## log(weight) -23.4317     0.7534  -31.10   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 4.189 on 390 degrees of freedom
## Multiple R-squared:  0.7127, Adjusted R-squared:  0.7119
## F-statistic: 967.3 on 1 and 390 DF,  p-value: < 2.2e-16
```

```
summary(lm(data = Auto, mpg ~ sqrt(weight)))
```

```
##
## Call:
## lm(formula = mpg ~ sqrt(weight), data = Auto)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -12.2402  -2.9005  -0.3708   2.0791  16.2296
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  69.67218    1.52649   45.64   <2e-16 ***
## sqrt(weight) -0.85560    0.02797  -30.59   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 4.239 on 390 degrees of freedom
## Multiple R-squared:  0.7058, Adjusted R-squared:  0.705
## F-statistic: 935.4 on 1 and 390 DF,  p-value: < 2.2e-16
```

```
# Plot regression after transformation
ggplot(Auto, aes(y = mpg)) +
  geom_point(aes(x = log(weight)), color = 'darkred')+
  geom_smooth(aes(x = log(weight)), method = 'lm')
```

```
## `geom_smooth()` using formula = 'y ~ x'
```




```
# Considering different transformation for the variable displacement
summary(lm(data = Auto, mpg ~ displacement))
```

```
##
## Call:
## lm(formula = mpg ~ displacement, data = Auto)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -12.9170  -3.0243  -0.5021   2.3512  18.6128
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  35.12064    0.49443   71.03   <2e-16 ***
## displacement -0.06005    0.00224  -26.81   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 4.635 on 390 degrees of freedom
## Multiple R-squared:  0.6482, Adjusted R-squared:  0.6473
## F-statistic: 718.7 on 1 and 390 DF,  p-value: < 2.2e-16
```

```
summary(lm(data = Auto, mpg ~ displacement^2))
```

```
##
## Call:
## lm(formula = mpg ~ displacement^2, data = Auto)
##
```

```
## Residuals:
##      Min       1Q   Median       3Q      Max
## -12.9170  -3.0243  -0.5021   2.3512  18.6128
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  35.12064    0.49443   71.03   <2e-16 ***
## displacement -0.06005    0.00224  -26.81   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 4.635 on 390 degrees of freedom
## Multiple R-squared:  0.6482, Adjusted R-squared:  0.6473
## F-statistic: 718.7 on 1 and 390 DF,  p-value: < 2.2e-16
```

```
summary(lm(data = Auto, mpg ~ log(displacement)))
```

```
##
## Call:
## lm(formula = mpg ~ log(displacement), data = Auto)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -16.1204  -2.5843  -0.4217   2.1979  19.9005
##
## Coefficients:
##                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)        85.6906     2.1422   40.00   <2e-16 ***
## log(displacement) -12.1385     0.4155  -29.21   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 4.377 on 390 degrees of freedom
## Multiple R-squared:  0.6863, Adjusted R-squared:  0.6855
## F-statistic: 853.4 on 1 and 390 DF,  p-value: < 2.2e-16
```

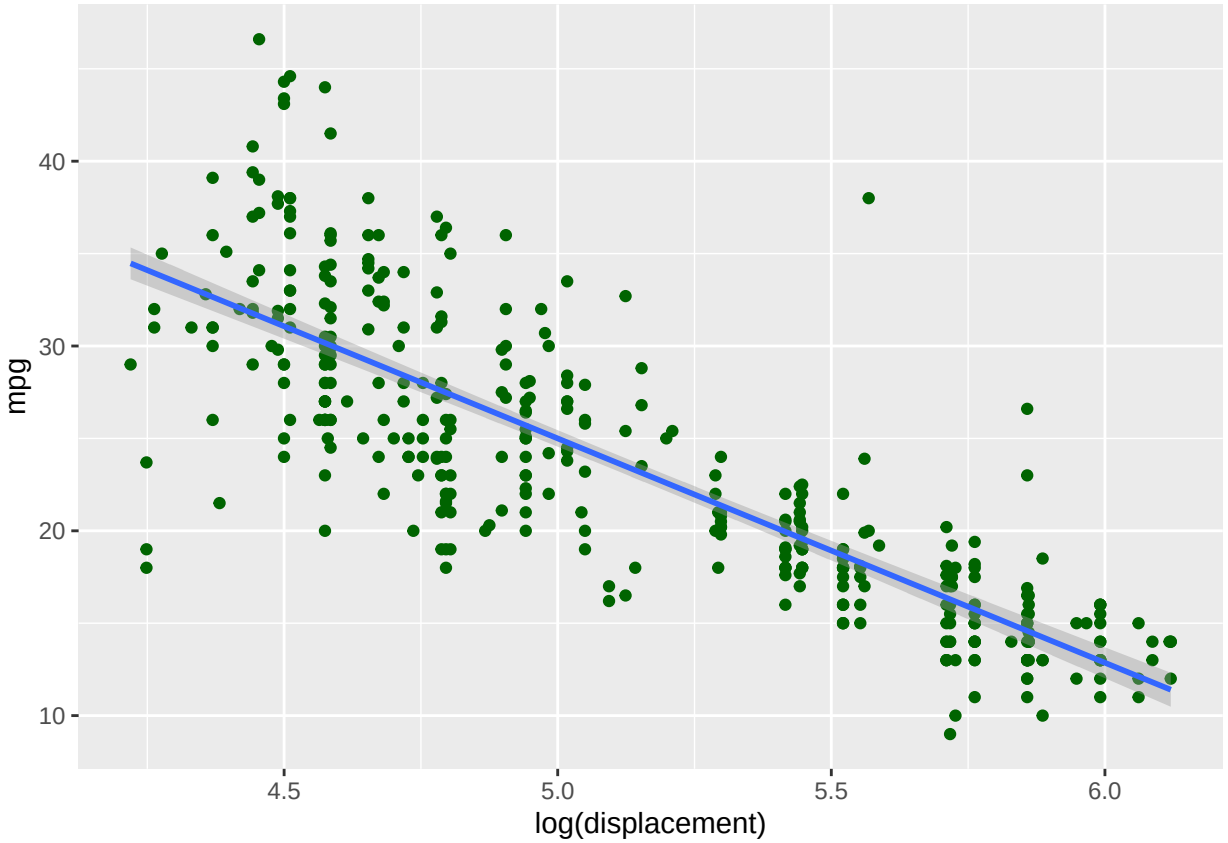
```
summary(lm(data = Auto, mpg ~ sqrt(displacement)))
```

```
##
## Call:
## lm(formula = mpg ~ sqrt(displacement), data = Auto)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -14.4034  -2.7367  -0.4956   2.3207  19.3499
##
## Coefficients:
##                    Estimate Std. Error t value Pr(>|t|)
## (Intercept)        47.11839    0.86246   54.63   <2e-16 ***
## sqrt(displacement) -1.75878    0.06186  -28.43   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 4.458 on 390 degrees of freedom
## Multiple R-squared:  0.6746, Adjusted R-squared:  0.6738
```

```
## F-statistic: 808.5 on 1 and 390 DF,  p-value: < 2.2e-16
```

```
# Plot regression after transformation
ggplot(Auto, aes(y = mpg)) +
geom_point(aes(x = log(displacement)), color = 'darkgreen')+
  geom_smooth(aes(x = log(displacement)), method = 'lm')
```

```
## `geom_smooth()` using formula = 'y ~ x'
```




```
# Fit model without transformation and with transformation in the variables
model5_original <- lm(data = Auto,formula = mpg ~ cylinders  + weight +  displacement + year)
summary(model5_original)
```

```
##
## Call:
## lm(formula = mpg ~ cylinders + weight + displacement + year,
##     data = Auto)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -9.0169 -2.2958 -0.0967  2.0400 14.4239
##
## Coefficients:
##                Estimate Std. Error t value Pr(>|t|)
## (Intercept)  -1.369e+01  4.079e+00  -3.357 0.000868 ***
## cylinders    -3.217e-01  3.299e-01  -0.975 0.330182
## weight       -6.612e-03  5.735e-04 -11.531  < 2e-16 ***
## displacement  4.888e-03  6.695e-03   0.730 0.465727
```

```
## year          7.586e-01  5.101e-02  14.872  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.432 on 387 degrees of freedom
## Multiple R-squared:  0.8087, Adjusted R-squared:  0.8067
## F-statistic: 408.9 on 4 and 387 DF,  p-value: < 2.2e-16
```

```
model5_tranf <- lm(data = Auto,formula = mpg ~ cylinders  + log(weight) +  log(displacement) + year)
summary(model5_tranf)
```

```
##
## Call:
## lm(formula = mpg ~ cylinders + log(weight) + log(displacement) +
##     year, data = Auto)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -9.861 -1.919 -0.019  1.641 13.286
##
## Coefficients:
##                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)       119.37349   10.07726  11.846   <2e-16 ***
## cylinders           0.54452    0.28106   1.937   0.0534 .
## log(weight)       -17.81797    1.69993 -10.482   <2e-16 ***
## log(displacement)  -3.14263    1.25598  -2.502   0.0128 *
## year                0.77684    0.04604  16.872   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.139 on 387 degrees of freedom
## Multiple R-squared:  0.8399, Adjusted R-squared:  0.8383
## F-statistic: 507.7 on 4 and 387 DF,  p-value: < 2.2e-16
```

```
#Compare the how they fit depending on transformation
AIC(model5_original, model5_tranf)
```

```
##                 df      AIC
## model5_original  6 2086.147
## model5_tranf     6 2016.185
```

Note: In this last exercise, we can appreciate how transformation could improve the model fit. In this case, the better transformation was log(), in both variables. It increased the R-square and affected the AIC.