8.

(a) We perform a simple linear regression with *mpg* as the response and *horsepower* as a predictor. Here the results :

```
library(MASS)
library(ISLR)
library(car)
lm.fit = lm(mpg~horsepower, data=Auto)
summary(lm.fit)
##
## Call:
## lm(formula = mpg ~ horsepower, data = Auto)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -13.5710  -3.2592  -0.3435   2.7630  16.9240
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 39.935861   0.717499   55.66   <2e-16 ***
## horsepower  -0.157845   0.006446  -24.49   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 4.906 on 390 degrees of freedom
## Multiple R-squared:  0.6059,Adjusted R-squared:  0.6049
## F-statistic: 599.7 on 1 and 390 DF,  p-value: < 2.2e-16
```

We can observe that there is a relationship between the predictor and the response

i. We can observe, that the *p-value* is very low so we can believe that there is a relationship between the predictor and the response.

ii. The RSE equal to 4.906 on 390 degrees of freedom and $R^2 = 60.59\%$ that shows the relationship is a priori strong.

iii. The slope of this simple linear regression is positive thus the relationship between the predictor and the response is positive.

iv.
```
attach(Auto)
#To get confidence interval:
predict(lm.fit, data.frame(horsepower=98), interval="confidence")
##        fit      lwr      upr
## 1 24.46708 23.97308 24.96108
#To get prediction interval:
predict(lm.fit, data.frame(horsepower=98), interval="prediction")
##        fit     lwr      upr
## 1 24.46708 14.8094 34.12476
```

So the associated 95% confidence and predictive intervals are respectively equal to : $[23.97, 24.96]$ and $[14.81, 34.12]$

(b) We will plot the response and the predictor :

```
#To plot
plot(horsepower, mpg, col='red', pch='+')
abline(lm.fit, lwd=3, col='green')
```

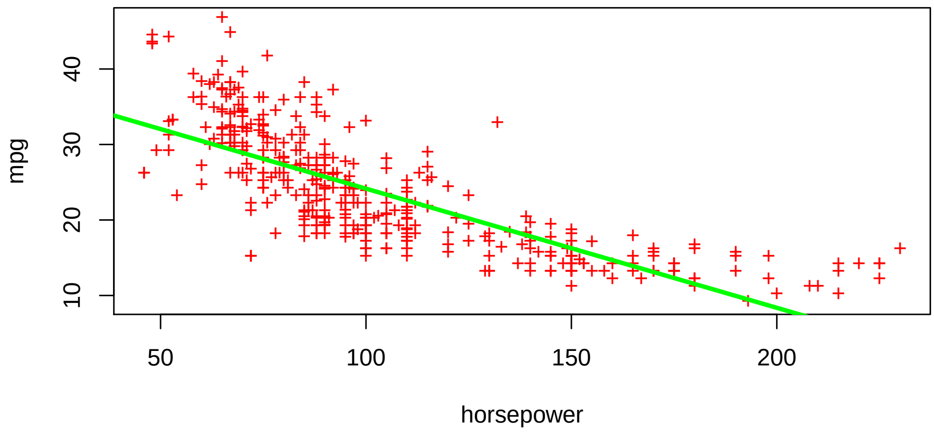

FIGURE 1 – The response and the predictor.

(c) As we are in simple regression settings to check if there is a *Non-linerity of the Data* for this we plot simply residual errors vs predictor :

```
plot(horsepower, residuals(lm.fit), type='o', col='blue')
```

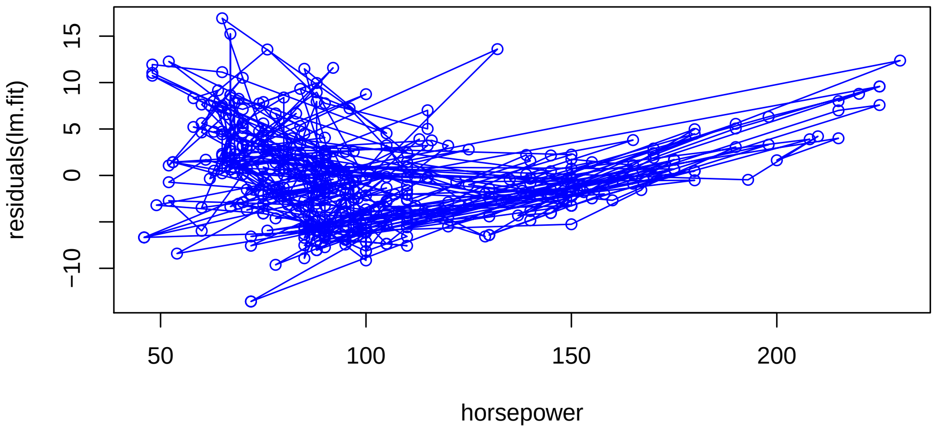

FIGURE 2 – Residuals errors vs predictor.

After this we check if it exists correlation of errors terms, ploting Residuals vs observation number :

```
plot(residuals(lm.fit), type='o', col='red')
title(xlab='Observation number')
```

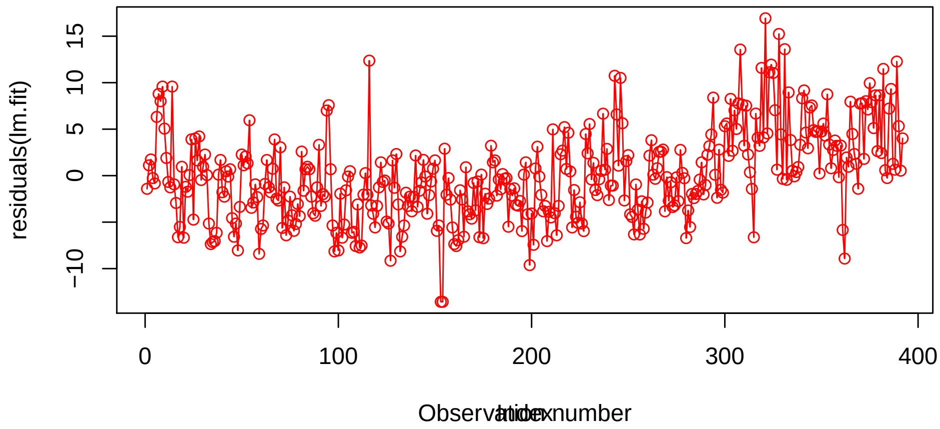

Figure 3 – Residuals vs observation number.

We do not observe pattern in those plots.
The residual errors stay confined so there is no *Non-constant variance of error terms* issues. We do not see neither *Outliers* or *High leverage points*. Finaly as a simple regression there is no matter of *Colinearity*

9.

(a) Here a scatterplot matrix including all variables :

```
pairs(Auto)
```

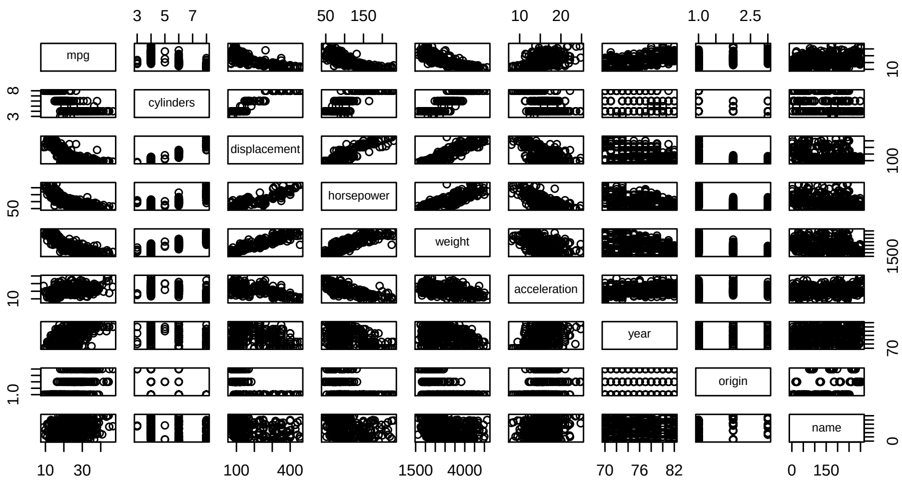

FIGURE 4 – Scatterplot matrix.

(b) For the matrix of correlation :

```
Table = data.frame(Auto)
cor(Table[, -9])
##                     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
## origin        0.5652088 -0.5689316   -0.6145351 -0.4551715 -0.5850054
##              acceleration       year     origin
## mpg             0.4233285  0.5805410  0.5652088
## cylinders      -0.5046834 -0.3456474 -0.5689316
## displacement   -0.5438005 -0.3698552 -0.6145351
## horsepower     -0.6891955 -0.4163615 -0.4551715
## weight         -0.4168392 -0.3091199 -0.5850054
## acceleration    1.0000000  0.2903161  0.2127458
## year            0.2903161  1.0000000  0.1815277
## origin          0.2127458  0.1815277  1.0000000
```

(c) We use multiple linear regression with mpg and the predictors

```r
lm.fit = lm(mpg~.-name, data=Auto)
summary(lm.fit)
##
## Call:
## lm(formula = mpg ~ . - name, 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
```

i. The *F-statistic* is a means to compute hypothesis test to know if there is or not a relationship between the response and the predictors. When $F - statistic$ takes a value close to 1 then there is not relationship, but here *F-statistic* equals to 252.4 on the 7 predictors that are the most significant.

ii. Regarding the *p-value* associated with *F-statistic* the most significant predictors are : weight, year, origin and displacement.

iii. The year coefficient means that in 10 years the distance traveled growth of 75 miles per gallon.

(d) Recall that main problems that we can encountered are : *Non-linerity of the Data, Correlation of error terms, Non-constant variance of error terms, Outliers, High-leverage points, and Colinearity.*

Non-linerity It suffices to plot residual errors vs the predicted response

```r
y_predict = predict(lm.fit)
plot(y_predict, residuals(lm.fit), type='o', col='blue')
```

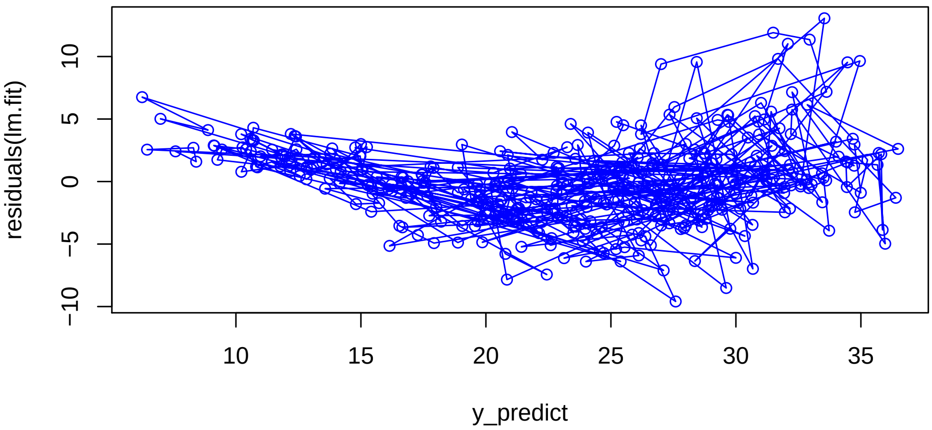

FIGURE 5 – Residuals vs the predicted response.

We do not identify any pattern in the Residuals vs the predicted response plot

Colinearity of error terms We plot residuals vesus observation :

```
plot(residuals(lm.fit), type='o', col='red')
```

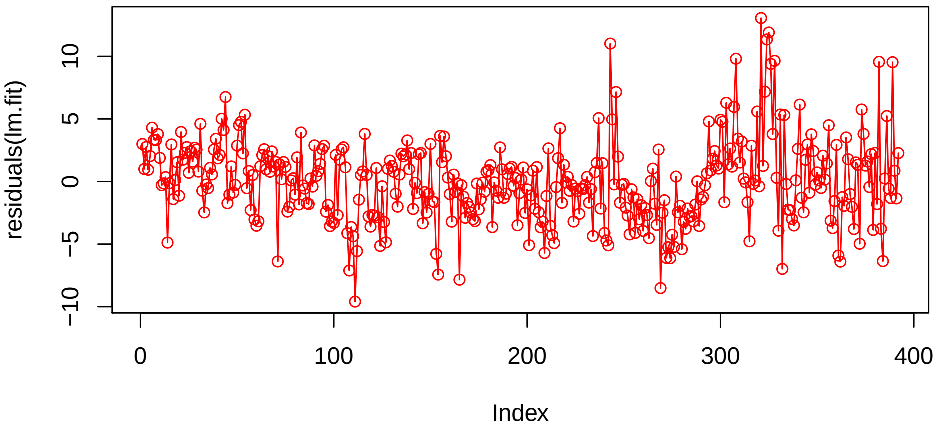

Figure 6 – Residuals vs observation number.

This time we observe that residual values increase with observation, we suspect a correlation in the error terms.

Non-constant variance of error terms We can observe that the residual values tend to stay confined between 5 and -5.

Non-linerity

Non-linerity

Non-linerity

(e)

(f)