MRLM_Coursera.R

pcbrom

Wed Oct 26 16:18:17 2016

```
setwd("/home/pcbrom/Dropbox/Trabalho e Estudo/Cursos Livres/Regression Models/CurseProject")

## Instructions

# You work for Motor Trend, a magazine about the automobile industry. Looking at
# a data set of a collection of cars, they are interested in exploring the
# relationship between a set of variables and miles per gallon (MPG) (outcome).
# They are particularly interested in the following two questions:

# "Is an automatic or manual transmission better for MPG"
# "Quantify the MPG difference between automatic and manual transmissions"

# do multiple cores
doMC::registerDoMC(4)

# get data
data(mtcars)

# dictionary
# [, 1]  mpg    Miles/(US) gallon
# [, 2]  cyl    Number of cylinders
# [, 3]  disp   Displacement (cu.in.)
# [, 4]  hp     Gross horsepower
# [, 5]  drat   Rear axle ratio
# [, 6]  wt     Weight (1000 lbs)
# [, 7]  qsec   1/4 mile time
# [, 8]  vs     V/S
# [, 9]  am     Transmission (0 = automatic, 1 = manual)
# [,10]  gear   Number of forward gears
# [,11]  carb   Number of carburetors

# see some lines
head(mtcars)
```

```
##                   mpg cyl disp  hp drat    wt  qsec vs am gear carb
## Mazda RX4         21.0   6  160 110 3.90 2.620 16.46  0  1    4    4
## Mazda RX4 Wag     21.0   6  160 110 3.90 2.875 17.02  0  1    4    4
## Datsun 710        22.8   4  108  93 3.85 2.320 18.61  1  1    4    1
## Hornet 4 Drive    21.4   6  258 110 3.08 3.215 19.44  1  0    3    1
## Hornet Sportabout 18.7   8  360 175 3.15 3.440 17.02  0  0    3    2
## Valiant           18.1   6  225 105 2.76 3.460 20.22  1  0    3    1
```

```
# see structure of data
str(mtcars)
```

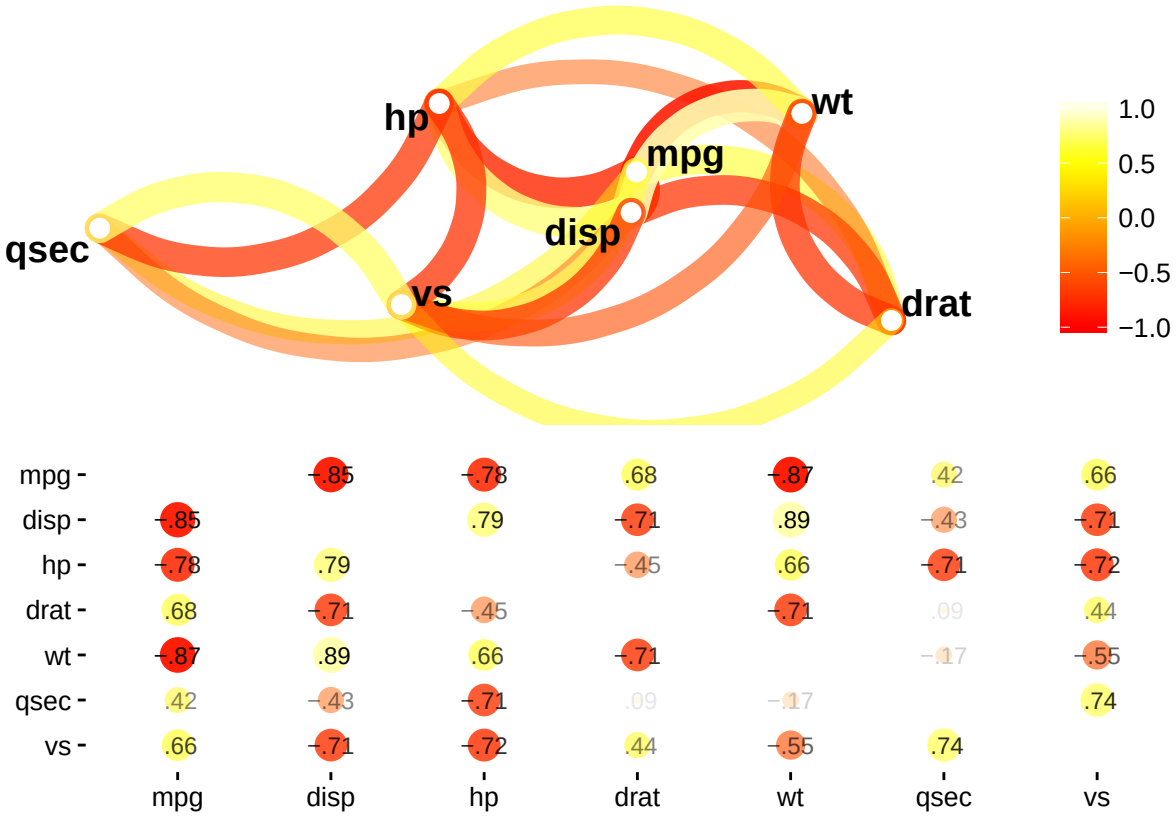
```
## 'data.frame':    32 obs. of  11 variables:
```

```
##  $ mpg : num  21 21 22.8 21.4 18.7 18.1 14.3 24.4 22.8 19.2 ...
##  $ cyl : num  6 6 4 6 8 6 8 4 4 6 ...
##  $ disp: num  160 160 108 258 360 ...
##  $ hp  : num  110 110 93 110 175 105 245 62 95 123 ...
##  $ drat: num  3.9 3.9 3.85 3.08 3.15 2.76 3.21 3.69 3.92 3.92 ...
##  $ wt  : num  2.62 2.88 2.32 3.21 3.44 ...
##  $ qsec: num  16.5 17 18.6 19.4 17 ...
##  $ vs  : num  0 0 1 1 0 1 0 1 1 1 ...
##  $ am  : num  1 1 1 0 0 0 0 0 0 0 ...
##  $ gear: num  4 4 4 3 3 3 3 4 4 4 ...
##  $ carb: num  4 4 1 1 2 1 4 2 2 4 ...
```

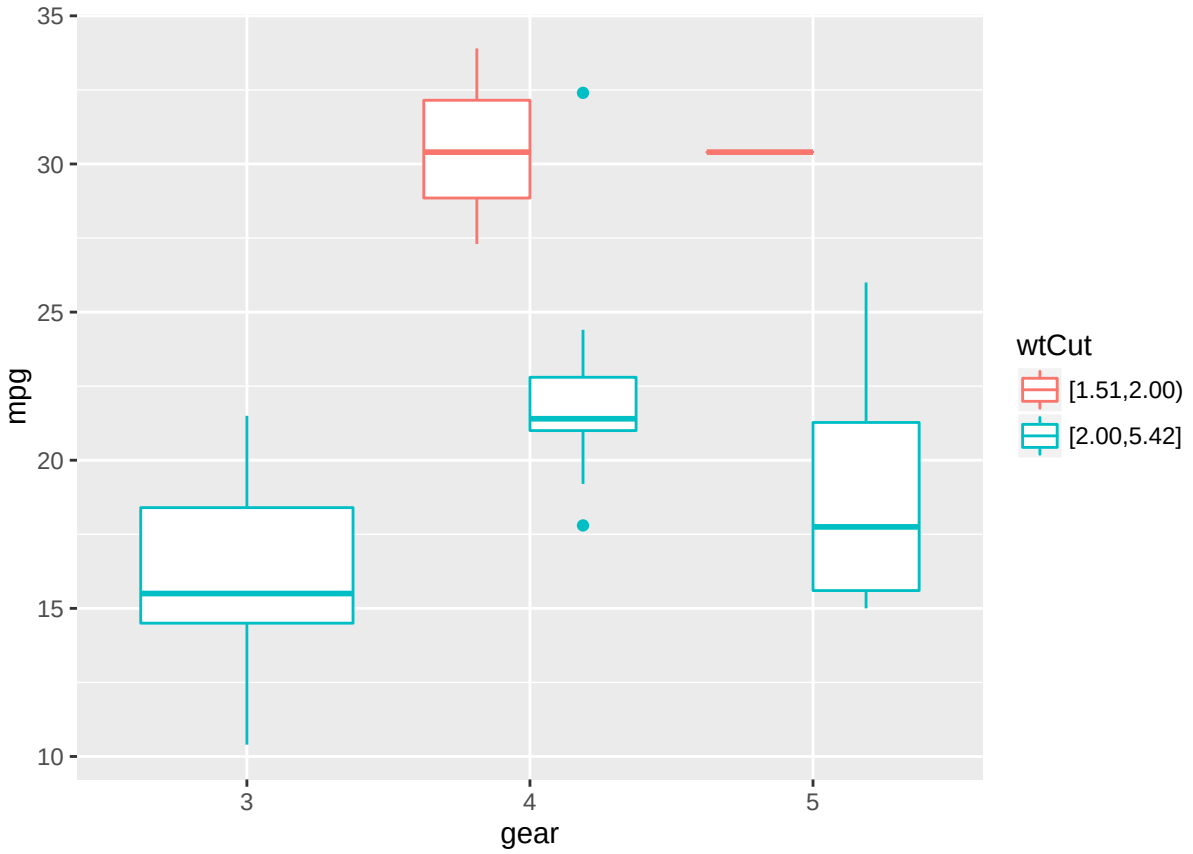
```
# adjust some variables to factor / adjust some factors
mtcars$cyl = as.factor(mtcars$cyl)
mtcars$gear = as.factor(mtcars$gear)
mtcars$carb = as.factor(mtcars$carb)
mtcars$am = as.factor(mtcars$am)
levels(mtcars$am) = c("auto", "manu")

# summary of data
summary(mtcars)
```

```
##       mpg        cyl         disp             hp             drat
##  Min.   :10.40   4:11   Min.   : 71.1   Min.   : 52.0   Min.   :2.760
##  1st Qu.:15.43   6: 7   1st Qu.:120.8   1st Qu.: 96.5   1st Qu.:3.080
##  Median :19.20   8:14   Median :196.3   Median :123.0   Median :3.695
##  Mean   :20.09          Mean   :230.7   Mean   :146.7   Mean   :3.597
##  3rd Qu.:22.80          3rd Qu.:326.0   3rd Qu.:180.0   3rd Qu.:3.920
##  Max.   :33.90          Max.   :472.0   Max.   :335.0   Max.   :4.930
##        wt             qsec             vs            am     gear   carb
##  Min.   :1.513   Min.   :14.50   Min.   :0.0000   auto:19   3:15   1: 7
##  1st Qu.:2.581   1st Qu.:16.89   1st Qu.:0.0000   manu:13   4:12   2:10
##  Median :3.325   Median :17.71   Median :0.0000             5: 5   3: 3
##  Mean   :3.217   Mean   :17.85   Mean   :0.4375                    4:10
##  3rd Qu.:3.610   3rd Qu.:18.90   3rd Qu.:1.0000                    6: 1
##  Max.   :5.424   Max.   :22.90   Max.   :1.0000                    8: 1
```

```
# exploratory data analyses

suppressWarnings(suppressMessages({require(corrr); require(gridExtra)}))
rdf = correlate(mtcars[, -c(2,9:11)])
p1 = rplot(rdf, print_cor = T, legend = F, colours = heat.colors(20))
p2 = network_plot(rdf, legend = T, colours = heat.colors(20))
grid.arrange(p2, p1, ncol = 1, nrow = 2)
```




```
# Here mpg correlated with all numeric variables, shows an explanatory potential
# for consumption, but overall the database has correlation between the
# variables. This can lead us to an inflation undesirable variance. So here we
# have two paths. The first is to evaluate a lean model can u still provide good
# degree of variability explanation and the other is to make an approach using
# the PCA matrix.

suppressWarnings(suppressMessages({require(ggplot2); require(Hmisc)}))
mtcars$wtCut = cut2(mtcars$wt, 2)
qplot(gear, mpg, col = wtCut, data = mtcars, geom = "boxplot")
```




```
# mpg influenced when considering the amount of speed and vehicle weight. It is
# remarkable that better performance meets vehicle 4 gears and weighing less
# than 2000 lbs A pect that calls the attention is the outlier with next
# performance 32 mpg with the vehicle of higher lbs.

mtcars[mtcars$gear == 4 & mtcars$wtCut == "[2.00,5.42]" & mtcars$mpg > 30, ]
```

```
##           mpg cyl disp hp drat  wt  qsec vs   am gear carb       wtCut
## Fiat 128 32.4   4 78.7 66 4.08 2.2 19.47  1 manu    4    1 [2.00,5.42]
```

```
# As well a Fiat 128 weighs 2200 lbs ??? I believe this is an error in the
# database, for this car, according to the manufacturer, weighs about 1600 lbs.
# It's explain the outlier.

mtcars[mtcars$gear == 4 & mtcars$wtCut == "[2.00,5.42]" & mtcars$mpg < 20, ]
```

```
##            mpg cyl  disp  hp drat   wt qsec vs   am gear carb       wtCut
## Merc 280  19.2   6 167.6 123 3.92 3.44 18.3  1 auto    4    4 [2.00,5.42]
## Merc 280C 17.8   6 167.6 123 3.92 3.44 18.9  1 auto    4    4 [2.00,5.42]
```

```
suppressWarnings(suppressMessages({require(jpeg); require(grid)}))
img1 = readJPEG("Fiat_128_Kent_UK2.JPG")
img2 = readJPEG("800px-MercedesBenz_250C_1970.JPG")
g1 = rasterGrob(img1, interpolate = T)
g2 = rasterGrob(img2, interpolate = T)
```

```
p3 = qplot(1:10, 1:10, geom = "blank",
           xlab = "This is a bug database.\nIt is not an outlier.\nImage from Wikipedia.",
           ylab = "", main = "Fiat 128\nWeighs about 1600 lbs") +
    annotation_custom(g1, xmin = -Inf, xmax = Inf, ymin = -Inf, ymax = Inf) +
    theme(axis.line=element_blank(),axis.text.x=element_blank(),
          axis.text.y=element_blank(),axis.ticks=element_blank(),
          #axis.title.x=element_blank(),
          axis.title.y=element_blank(),legend.position="none",
          panel.background=element_blank(),panel.border=element_blank(),
          panel.grid.major=element_blank(),
          panel.grid.minor=element_blank(),plot.background=element_blank())
p4 = qplot(1:10, 1:10, geom = "blank",
           xlab = "This really is an outlier.\nImage from Wikipedia.",
           ylab = "", main = "Mercedes 280C") +
    annotation_custom(g2, xmin = -Inf, xmax = Inf, ymin = -Inf, ymax = Inf) +
    theme(axis.line=element_blank(),axis.text.x=element_blank(),
          axis.text.y=element_blank(),axis.ticks=element_blank(),
          #axis.title.x=element_blank(),
          axis.title.y=element_blank(),legend.position="none",
          panel.background=element_blank(),panel.border=element_blank(),
          panel.grid.major=element_blank(),
          panel.grid.minor=element_blank(),plot.background=element_blank())
grid.arrange(p3, p4, ncol = 2, nrow = 1)
```

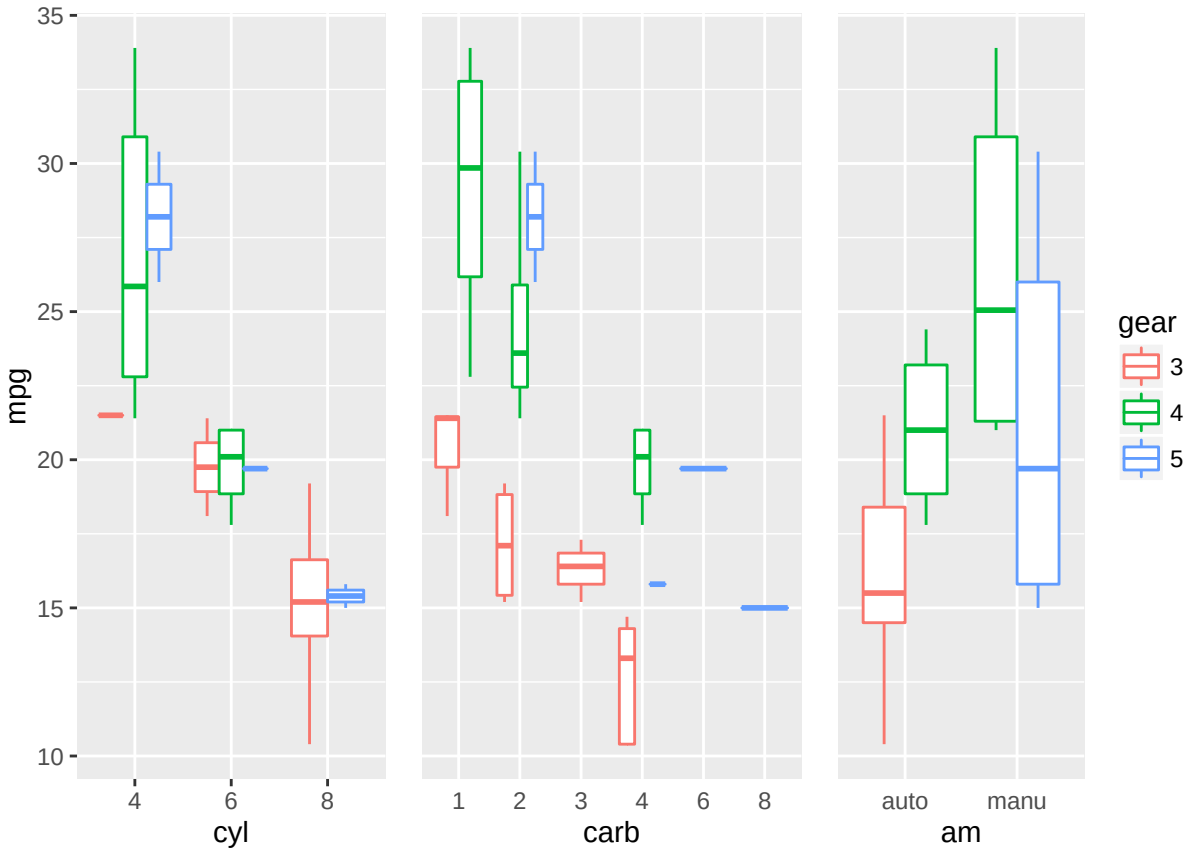
Fiat 128
Weighs about 1600 lbs

Mercedes 280C

This is a bug database.
It is not an outlier.
Image from Wikipedia.

This really is an outlier.
Image from Wikipedia.

```
# It is remarkable as the amount of speed influences the variability of
# consumption to categorize cars cúmero cylinders. 4 cylinder is Earl found the
# best performances and, in general, cars with 5 speed though not necessarily
# have the best consumption shows little variability. So if you buy a car and do
# not know much about technical data give preference for the 5-speed you'll be
# fine. Or rather, prefer lighter cars with 5-speed and 4 cylinders.

p5 = qplot(cyl, mpg, col = gear, data = mtcars, geom = "boxplot") +
    theme(legend.position = "none")
p6 = qplot(carb, mpg, col = gear, data = mtcars, geom = "boxplot")  +
    theme(legend.position = "none", axis.title.y = element_blank(),
          axis.text.y = element_blank(), axis.ticks = element_blank())
p7 = qplot(am, mpg, col = gear, data = mtcars, geom = "boxplot")  +
    theme(axis.title.y = element_blank(), axis.text.y = element_blank(),
          axis.ticks = element_blank())
grid.arrange(p5, p6, p7, ncol = 3, nrow = 1)
```




```
# Adding information to the previous paragraph, if you can choose your car with
# 2 carburetors and manual shift is better.

## strategy for model selection

# Right. So far we have learned that the weight amount of speed and number of
# cylinders become information that really differentiate the car performance. So
# the strategy is basically to use this data to model mpg.
```

```r
db = subset(mtcars, select = -c(wtCut))

fit1 = lm(mpg ~ ., data = db)
summary(fit1)
```

```
##
## Call:
## lm(formula = mpg ~ ., data = db)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -3.5087 -1.3584 -0.0948  0.7745  4.6251
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 23.87913   20.06582   1.190   0.2525
## cyl6        -2.64870    3.04089  -0.871   0.3975
## cyl8        -0.33616    7.15954  -0.047   0.9632
## disp         0.03555    0.03190   1.114   0.2827
## hp          -0.07051    0.03943  -1.788   0.0939 .
## drat         1.18283    2.48348   0.476   0.6407
## wt          -4.52978    2.53875  -1.784   0.0946 .
## qsec         0.36784    0.93540   0.393   0.6997
## vs           1.93085    2.87126   0.672   0.5115
## ammanu       1.21212    3.21355   0.377   0.7113
## gear4        1.11435    3.79952   0.293   0.7733
## gear5        2.52840    3.73636   0.677   0.5089
## carb2       -0.97935    2.31797  -0.423   0.6787
## carb3        2.99964    4.29355   0.699   0.4955
## carb4        1.09142    4.44962   0.245   0.8096
## carb6        4.47757    6.38406   0.701   0.4938
## carb8        7.25041    8.36057   0.867   0.3995
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.833 on 15 degrees of freedom
## Multiple R-squared:  0.8931, Adjusted R-squared:  0.779
## F-statistic:  7.83 on 16 and 15 DF,  p-value: 0.000124
```

```r
fit2 = lm(mpg ~ cyl + hp + wt + gear + carb, data = db)
summary(fit2)
```

```
##
## Call:
## lm(formula = mpg ~ cyl + hp + wt + gear + carb, data = db)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -4.2236 -1.3324 -0.0677  0.8930  4.3673
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
```

```
## (Intercept) 35.37743    4.63859   7.627 2.42e-07 ***
## cyl6        -2.60335    1.91793  -1.357   0.1898
## cyl8         1.03185    3.64041   0.283   0.7797
## hp          -0.05129    0.03255  -1.576   0.1307
## wt          -2.43230    1.00337  -2.424   0.0249 *
## gear4        2.05907    2.32904   0.884   0.3872
## gear5        3.61278    2.83533   1.274   0.2172
## carb2       -1.96198    1.63603  -1.199   0.2445
## carb3       -1.48841    2.43815  -0.610   0.5484
## carb4       -1.27924    2.87577  -0.445   0.6612
## carb6       -0.97365    4.42151  -0.220   0.8279
## carb8        0.84337    6.42230   0.131   0.8968
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.648 on 20 degrees of freedom
## Multiple R-squared:  0.8754, Adjusted R-squared:  0.8069
## F-statistic: 12.78 on 11 and 20 DF,  p-value: 9.236e-07
```

```
fit3 = lm(mpg ~ cyl + wt, data = db)
summary(fit3)
```

```
##
## Call:
## lm(formula = mpg ~ cyl + wt, data = db)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -4.5890 -1.2357 -0.5159  1.3845  5.7915
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  33.9908     1.8878  18.006  < 2e-16 ***
## cyl6         -4.2556     1.3861  -3.070 0.004718 **
## cyl8         -6.0709     1.6523  -3.674 0.000999 ***
## wt           -3.2056     0.7539  -4.252 0.000213 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.557 on 28 degrees of freedom
## Multiple R-squared:  0.8374, Adjusted R-squared:   0.82
## F-statistic: 48.08 on 3 and 28 DF,  p-value: 3.594e-11
```

```
fit4 = lm(mpg ~ cyl * wt, data = db)
summary(fit4)
```

```
##
## Call:
## lm(formula = mpg ~ cyl * wt, data = db)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -4.1513 -1.3798 -0.6389  1.4938  5.2523
```

```
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   39.571      3.194  12.389 2.06e-12 ***
## cyl6         -11.162      9.355  -1.193 0.243584
## cyl8         -15.703      4.839  -3.245 0.003223 **
## wt            -5.647      1.359  -4.154 0.000313 ***
## cyl6:wt        2.867      3.117   0.920 0.366199
## cyl8:wt        3.455      1.627   2.123 0.043440 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.449 on 26 degrees of freedom
## Multiple R-squared:  0.8616, Adjusted R-squared:  0.8349
## F-statistic: 32.36 on 5 and 26 DF,  p-value: 2.258e-10
```

```
anova(fit3, fit4, fit1, fit2)
```

```
## Analysis of Variance Table
##
## Model 1: mpg ~ cyl + wt
## Model 2: mpg ~ cyl * wt
## Model 3: mpg ~ cyl + disp + hp + drat + wt + qsec + vs + am + gear + carb
## Model 4: mpg ~ cyl + hp + wt + gear + carb
##   Res.Df    RSS Df Sum of Sq      F Pr(>F)
## 1     28 183.06
## 2     26 155.89  2    27.170 1.6924 0.2174
## 3     15 120.40 11    35.486 0.4019 0.9337
## 4     20 140.25 -5   -19.853 0.4947 0.7754
```

```
fit5 = lm(mpg ~ cyl + wt - 1, data = db)
summary(fit5)
```

```
##
## Call:
## lm(formula = mpg ~ cyl + wt - 1, data = db)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -4.5890 -1.2357 -0.5159  1.3845  5.7915
##
## Coefficients:
##      Estimate Std. Error t value Pr(>|t|)
## cyl4  33.9908     1.8878  18.006  < 2e-16 ***
## cyl6  29.7352     2.5410  11.702 2.69e-12 ***
## cyl8  27.9199     3.0915   9.031 8.68e-10 ***
## wt    -3.2056     0.7539  -4.252 0.000213 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.557 on 28 degrees of freedom
## Multiple R-squared:  0.987,  Adjusted R-squared:  0.9851
## F-statistic:   530 on 4 and 28 DF,  p-value: < 2.2e-16
```

```
# Considering the models worked, we use the fit3 as a reference to assess
# whether it is worthwhile to add variables to the Anova tool. See the results
# suggest that we should keep only the most simplified model (mpg ~ cyl + wt).
# For this we obtain an Adjusted R² 0.82, ie, there is a good explanation of
# variability of vehicle performance when we use the reference weight and number
# of cylinders. If we consider a new model in which we remove the intercept
# (mpg ~ cyl + wt - 1), we obtain Adjusted R-squared: 0.9851, and we have to
# compare the reference level to see if there was a change in the expected
# value. To fit3 the intercept has the same value as the level of 4-cylinder,
# ie, using a model without the intercept allows the coefficients have a
# marginal interpretation more interpretable.

# Interpretation of coefficients. (fit5)

# A regression model returns on an average equation ous is, for each unit of
# weight in lbs, have a marginal reduction, average mpg in 3.2056 units,
# considering the other variables fixed. When evaluating the cylinders have a
# categorization of performance. Suppose the best performance, 4 cyl. The
# average marginal gain is about 34 mpg when considering fixed wt. If we want
# to compare the gain in mpg performance between the cylinders, we practice
# the following:

cyl4 = coef(fit5)[[1]]
cyl6 = coef(fit5)[[2]]
cyl8 = coef(fit5)[[3]]

cyl4/cyl6
```

```
## [1] 1.143116
```

```
cyl4/cyl8
```

```
## [1] 1.217438
```

```
cyl6/cyl8
```

```
## [1] 1.065017
```

```
# This means that the engine with 4 cylinders has a yield of 14% if compared to
# 6. Similarly have 4 to 8 cyl approximately 22% more yield and 6 to 8 has a
# yield about 6.5%.

# Here we create a model with coefficients evaluated standard deviation. The
# purpose is to study which factor is more important for vehicle performance
# analysis.

fit6 = lm(I((mpg - mean(mpg))/sd(mpg)) ~ cyl + I((wt - mean(wt))/sd(wt)) - 1,
          data = db); summary(fit6)
```

```
##
## Call:
```

```
## lm(formula = I((mpg - mean(mpg))/sd(mpg)) ~ cyl + I((wt - mean(wt))/sd(wt)) -
##     1, data = db)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.76141 -0.20503 -0.08559  0.22971  0.96094
##
## Coefficients:
##                           Estimate Std. Error t value Pr(>|t|)
## cyl4                        0.5951     0.1730   3.440 0.001844 **
## cyl6                       -0.1109     0.1608  -0.690 0.495995
## cyl8                       -0.4121     0.1497  -2.752 0.010267 *
## I((wt - mean(wt))/sd(wt))  -0.5204     0.1224  -4.252 0.000213 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.4242 on 28 degrees of freedom
## Multiple R-squared:  0.8374, Adjusted R-squared:  0.8142
## F-statistic: 36.06 on 4 and 28 DF,  p-value: 1.146e-10
```

```
# The car 4 cylinders has a positive effect suggests that MPG is an improvement
# for this condition. On the other hand, we have wt indicating about the same
# order of magnitude, that when we increase the weight will naturally lose the
# car performance.

fit7 = lm(mpg ~ am - 1, data = db)
summary(fit7)
```

```
##
## Call:
## lm(formula = mpg ~ am - 1, data = db)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -9.3923 -3.0923 -0.2974  3.2439  9.5077
##
## Coefficients:
##        Estimate Std. Error t value Pr(>|t|)
## amauto   17.147      1.125   15.25 1.13e-15 ***
## ammanu   24.392      1.360   17.94  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 4.902 on 30 degrees of freedom
## Multiple R-squared:  0.9487, Adjusted R-squared:  0.9452
## F-statistic: 277.2 on 2 and 30 DF,  p-value: < 2.2e-16
```

```
coef(fit7)[[2]]/coef(fit7)[[1]]
```

```
## [1] 1.42251
```

```r
# Using only the type of transmission reference has generated a new model (fit7)
# to describe mpg. It is easy to see that on average, the marginal mpg for
# manual cars is higher than the automatic. When we compare numerically we have
# a 42% yield of hand vehicles more compared to automatic. Here we emphasize the
# idea that a manual car is more economical than an automatic car.

# So which model should you use?
# R: fit3
summary(fit3)
```

```
##
## Call:
## lm(formula = mpg ~ cyl + wt, data = db)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -4.5890 -1.2357 -0.5159  1.3845  5.7915
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  33.9908     1.8878  18.006  < 2e-16 ***
## cyl6         -4.2556     1.3861  -3.070 0.004718 **
## cyl8         -6.0709     1.6523  -3.674 0.000999 ***
## wt           -3.2056     0.7539  -4.252 0.000213 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.557 on 28 degrees of freedom
## Multiple R-squared:  0.8374, Adjusted R-squared:   0.82
## F-statistic: 48.08 on 3 and 28 DF,  p-value: 3.594e-11
```

```r
# What is the confidence interval for the coefficients?
confint(fit3)
```

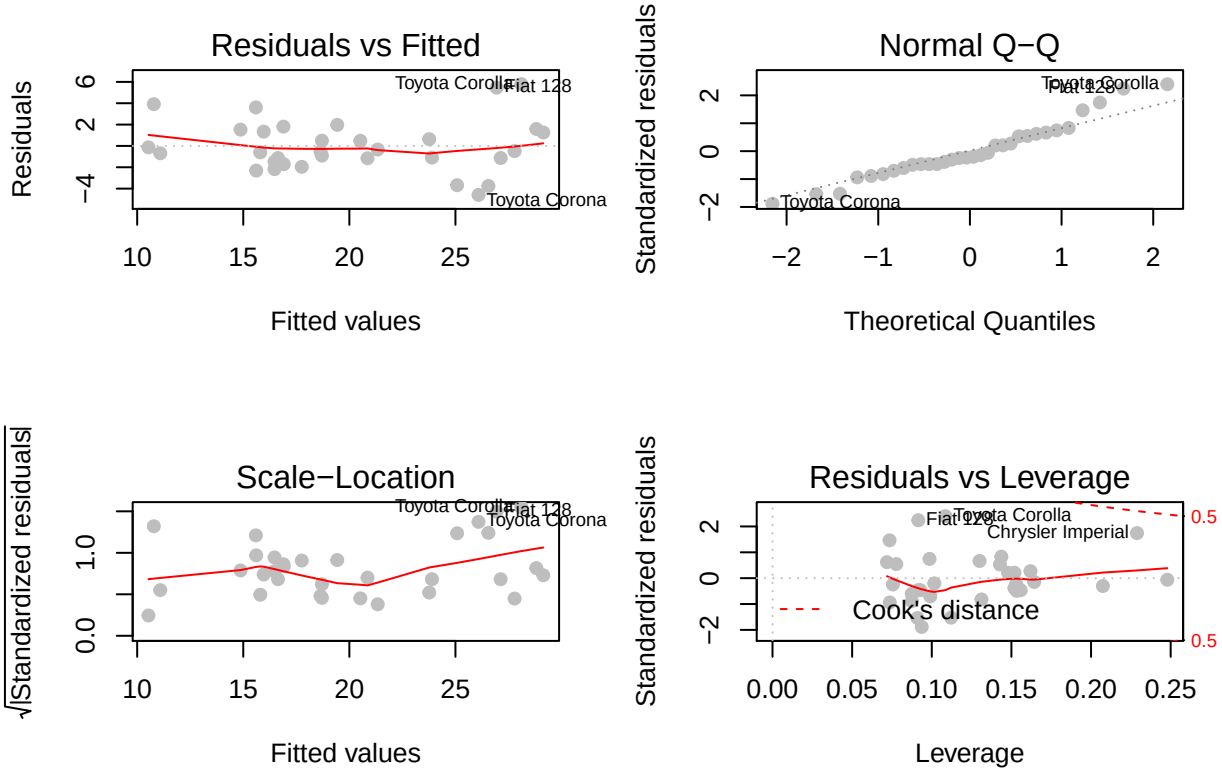
```
##                 2.5 %    97.5 %
## (Intercept) 30.123824 37.857764
## cyl6        -7.094824 -1.416341
## cyl8        -9.455418 -2.686301
## wt          -4.749898 -1.661328
```

```r
# This indicates that given this sample, the performance "mpg" can vary for the
# wt variable -4.75 to -1.66, that is, if we repeat the sampling experiment many
# times, we have the true value of the coefficient (population and unknown) 95%
# often within the indicated range. The same interpretation is for the cylinders.

# The model is good or have some kind of problem?

# The risiduals meet the assumption of normality?
# The test (p-value = 0.259) accept the hypothesis of normality.
shapiro.test(residuals(fit3))
```

```
##
```

```
##  Shapiro-Wilk normality test
##
## data:  residuals(fit3)
## W = 0.95907, p-value = 0.259
```

```
# Residual plot and some diagnostics
par(mfrow = c(2,2)); plot(fit3, pch = 19, col = "grey"); par(mfrow = c(1,1))
```




```
# Evaluating Residuals vs Fitted apparently have possible interference, on
# quadratic dependency problem. Approximately normal distribution is ok.
# Scale-Location is within up to 2 nd, so is reasonably ok. Residuals vs
# Leverage. Here are some points "pulling" / potentially influencing the model,
# but are still within an appropriate range.

# We have to make a homoscedasticity test to draw any conclusion.
suppressWarnings(suppressMessages(require(lmtest)))
bptest(fit3, studentize = F)
```

```
##
##  Breusch-Pagan test
##
## data:  fit3
## BP = 7.3931, df = 3, p-value = 0.06037
```

```
# Here it depends on what the researcher must accept in his working hypothesis.
# To run performance is reasonable mágido number of 5%, so it would be
# inconclusive because 6% is very close to 5%. In this case we must increase the
# sample size to assess the consistency of constant variance test or consider
```

```r
# the possibility of studying vehicle categories separately. Anyway ...
# apparently would not absurd, for this model suggest that the variance of the
# errors suffer some kind of inflation as the Residuals chart vs Fitted does not
# demonstrate any "cone". So we can consider that the constructed model is
# minimally reasonable and is good enough to explain the variability of vehicle
# performance considering only wt and cyl.

## Creating the same model by machine learning

set.seed(123)
suppressWarnings(suppressMessages(require(caret)))
inTrain = createDataPartition(y = db$mpg, p = .7, list = F)
training = db[inTrain, ]; testing = db[-inTrain, ]
modFit = train(mpg ~ cyl + wt, data = training, method = "lm")
modFit$modelType; modFit$metric; modFit$finalModel
```

```
## [1] "Regression"
```

```
## [1] "RMSE"
```

```
##
## Call:
## lm(formula = .outcome ~ ., data = dat)
##
## Coefficients:
## (Intercept)         cyl6         cyl8           wt
##      33.791       -4.638       -6.420       -3.019
```

```r
modFit
```

```
## Linear Regression
##
## 24 samples
##  2 predictors
##
## No pre-processing
## Resampling: Bootstrapped (25 reps)
## Summary of sample sizes: 24, 24, 24, 24, 24, 24, ...
## Resampling results:
##
##   RMSE      Rsquared
##   2.690386  0.8514624
##
## Tuning parameter 'intercept' was held constant at a value of TRUE
##
```

```r
# Predicting new values

pred = predict(modFit, testing)
testing$predRigth = pred == testing$mpg

# Evaluating prediction
```

```r
table("predict values" = round(pred, 1), "original values" = testing$mpg)
```

```
##               original values
## predict values 13.3 15.2 15.5 18.7 21.4 22.8 24.4 32.4
##           15.8    1    0    0    0    0    0    0    0
##           16      0    1    0    0    0    0    0    0
##           16.7    0    0    1    0    0    0    0    0
##           17      0    0    0    1    0    0    0    0
##           24.2    0    0    0    0    0    0    1    0
##           25.4    0    0    0    0    1    0    0    0
##           26.8    0    0    0    0    0    1    0    0
##           27.1    0    0    0    0    0    0    0    1
```

```r
# Okay. That was interesting. Here we evaluate the quality of the model looking
# directly at the matrix of predicted and original values. The model will have a
# good quality when the results are closer to the main diagonal. Note that those
# who are not in the main, are "practically stuck" in it. This confirms that the
# model is well adjusted. In general it is recommended that have more lines of
# information to ensure the robustness of the model and stability of the
# coefficients.

suppressWarnings({
    modFitPCA = train(mpg ~ ., data = training, method = "lm",
                      preProcess = "pca")
})
modFitPCA
```

```
## Linear Regression
##
## 24 samples
## 10 predictors
##
## Pre-processing: principal component signal extraction (16), centered
##  (16), scaled (16)
## Resampling: Bootstrapped (25 reps)
## Summary of sample sizes: 24, 24, 24, 24, 24, 24, ...
## Resampling results:
##
##   RMSE      Rsquared
##   2.882095  0.8126418
##
## Tuning parameter 'intercept' was held constant at a value of TRUE
##
```

```r
# Initially I commented that use PCA might be useful. We will test a model using
# the rotated matrix. The RMSE had no big change and the gain on the adjustment
# was also not the best. In other words the simplest model (fit3) ended up being
# the best and most interpretable among all presented and built by ML (modFit)
# the result will be slightly higher than using classical statistics.


# To conclude we will clearly answer the questions of Motor Trend.
```

```
# "Is an automatic or manual transmission better for MPG"
# R: Manual transmission is better for MPG.

# "Quantify the MPG difference between automatic and manual transmissions"
# R: Auto = 17.147, Manu = 24.392.
#    Manu, Auto ratio = 1.42251, ie, 42% yield over manual for the automatic.
```