Regression-model-for-automobile-industry

luis

11/7/2020

Description

This project consists on a regression model for analysis of a data collection of automobile industry. The main objectivo is to explore the relationship between a set of variables and miles per gallon (MPG). Particularly in the answer of the following questions:

1. Is an automatic or manual transmission better for MPG
2. Quantify the MPG difference between automatic and manual transmissions

The data contains a set of 32 observations on 11 different variables - mpg: Miles/(US) gallon - cyl: Number of cylinders - disp: Displacement (cu.in.) - hp: Gross horsepower - drat: Rear axle ratio - wt: Weight (1000 lbs) - qsec: 1/4 mile time - vs: Engine (0 = V-shaped, 1 = straight) - am: Transmission (0 = automatic, 1 = manual) - gear: Number of forward gears - carb: Number of carburetors

Data and libraries loading

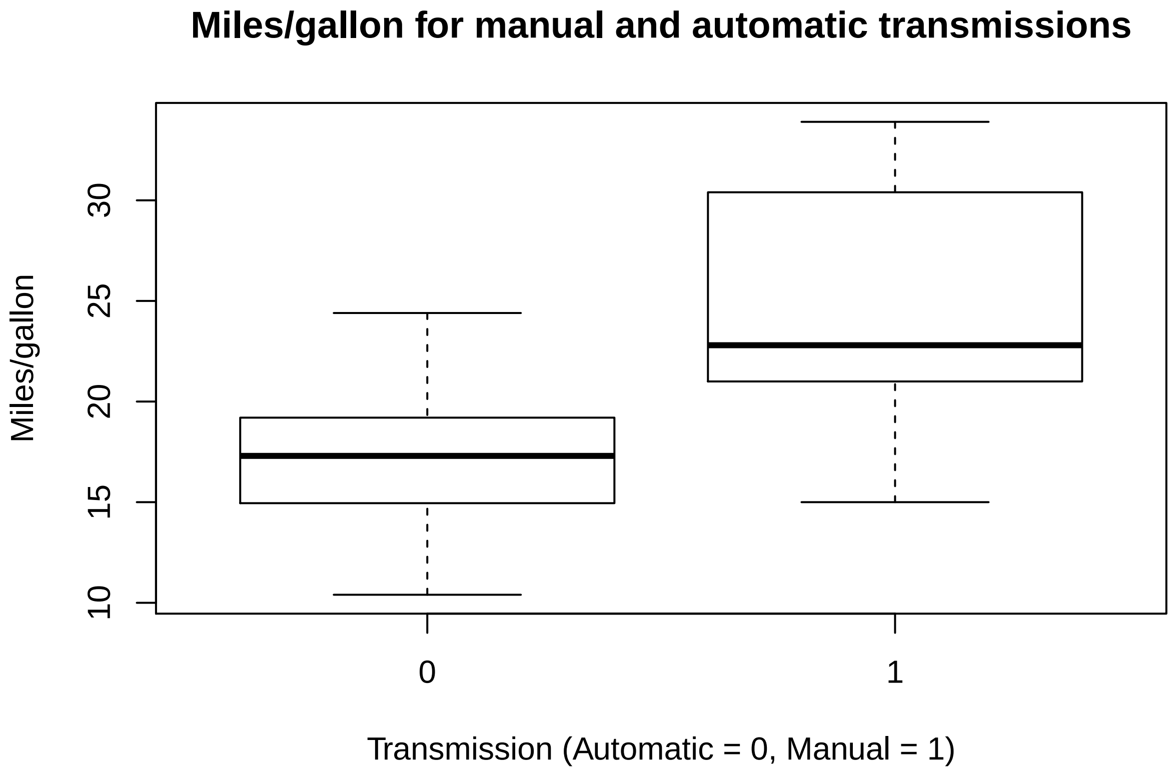
The libraries and set of data are loaded

```
library(ggplot2)
library(dplyr)
data("mtcars")
```

Exploratory Data Analyses

A first aproximation is made by the boxplot that differentiates the automatic and manual transmissions.

```
boxplot(mpg ~ am, data = mtcars, xlab = "Transmission (Automatic = 0, Manual = 1)", ylab = "Miles/gallon
```



The boxplot shows the manual transmission has in general greater values (mean = 24.39) for Miles/Gallon (mpg) than automatic transmissions (mean = 17.14). Nevertheless, the range of values is bigger for the manual transmission.

Model Selection

For the model selection, a first guess is made by fitting all the variables in the dataframe in order to look the diagnostics and decide whicho ones are needed.

```
allData <- mtcars %>% mutate(cyl = as.factor(cyl), vs = as.factor(vs), am = as.factor(am), gear = as.fa
fitAllData <- lm(mpg ~ ., data = allData)
summary(fitAllData)
```

```
##
## Call:
## lm(formula = mpg ~ ., data = allData)
##
## 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
## vs1          1.93085    2.87126   0.672   0.5115
## am1          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
```

The model shows a residual standard error of 2.833 on 15 degrees of freedom. The R-squared value 0.8931 indicates 89.3% of the data is explained by this model. Nevertheless, none of the variables show significant values less than 5%.

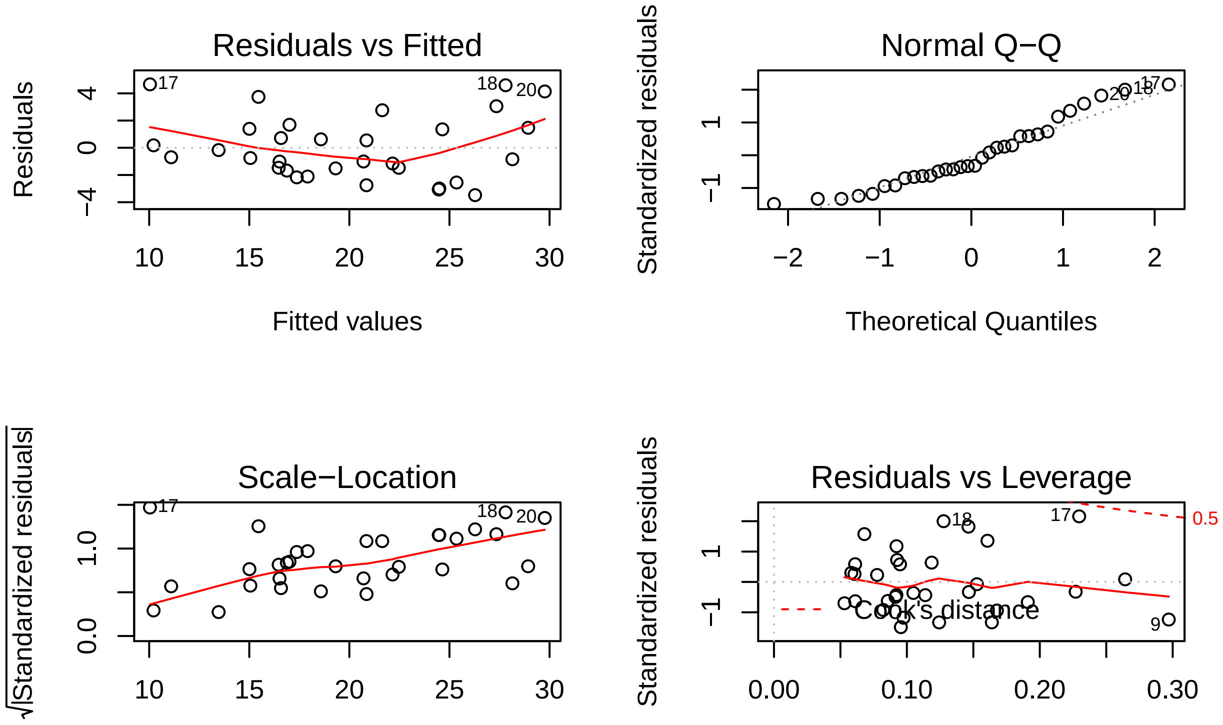
The followed procedure was to find the most insignificant variable in data set and remove it. The line “which.max(summary(*model*)$coef[, 4])” can find the least significant variable. The process was repeated until all the variables in the model resulted significant.

```r
dataBetter <- allData %>% select(-cyl); fitBetter <- lm(mpg ~ ., data = allData)
dataBetter <- dataBetter %>% select(-carb); fitBetter <- lm(mpg ~ ., data = dataBetter)
dataBetter <- dataBetter %>% select(-gear); fitBetter <- lm(mpg ~ ., data = dataBetter)
dataBetter <- dataBetter %>% select(-vs); fitBetter <- lm(mpg ~ ., data = dataBetter)
dataBetter <- dataBetter %>% select(-drat); fitBetter <- lm(mpg ~ ., data = dataBetter)
dataBetter <- dataBetter %>% select(-disp); fitBetter <- lm(mpg ~ ., data = dataBetter)
dataBetter <- dataBetter %>% select(-hp); fitBetter <- lm(mpg ~ ., data = dataBetter)
summary(fitBetter)
```

```
##
## Call:
## lm(formula = mpg ~ ., data = dataBetter)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -3.4811 -1.5555 -0.7257  1.4110  4.6610
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   9.6178     6.9596   1.382 0.177915
## wt           -3.9165     0.7112  -5.507 6.95e-06 ***
## qsec          1.2259     0.2887   4.247 0.000216 ***
## am1           2.9358     1.4109   2.081 0.046716 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.459 on 28 degrees of freedom
## Multiple R-squared:  0.8497, Adjusted R-squared:  0.8336
## F-statistic: 52.75 on 3 and 28 DF,  p-value: 1.21e-11
```

After the process is concluded, the final model shows a residual standard error of 2.459 on 28 degrees of freedom. Even though we removed 8 variables of the data out of 11 initial variables, the R-squared value is 0.8497 so the 84.97% of the data variance is still explained with all the variables show significance values less than 5%. The p-value indicates that we fail to reject the null hypothesis.

```
par(mfrow = c(2, 2))
plot(fitBetter)
```



In the QQ plot we can see there is no special pattern for the residuals, also the standarized and theoretical residuals have a good correlation.

Conclusions

This model states that if given the other vairables constant (qsec: 1/4 mile time, wt: Weight(1000 lbs)), the Miles/gallon (mpg) for the transmissions is: - Automatic: 9.6178 + 0*(2.9358) = 9.6178 Miles/Gallon - Manual: 9.6178 + 1*(2.9358) = 12.5536 Miles/Gallon Concluding, the manual transmission is better for Miles/Gallon, with a 2.9358 Miles/Gallon difference with a constant weight and 1/4 mile time.