Implications of manual and automatic transmission in the performance in miles/gallon units

Juan Sebastián Beleño Díaz

25 de marzo de 2016

Executive summary

This document presents the results of a study over the performance(in miles/gallon units) of cars deppending if a car has manual or automatic transmission, the results were that a car with manual transmission has a worst performance than a car that has automatic transmission, but in this text you also will find some interesting analysis that were performed to get more information in the data.

Exploratory analysis

In this part of the inform will be exposed some steps that were followed to obtain more information about the dataset that was used in this project, the information shown in this part works as input to develop models that explain the impact of the type of transmission(`am`) in the the performance measured in miles per gallon(‘mpg), the code used to find some valuable information was the following:

```
# Shows information about what mean variables in the dataset
?mtcars

# This give us an idea in what's the content of the datase
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
```

```
# This shows us the data type of each variable and its range of values
str(mtcars)
```

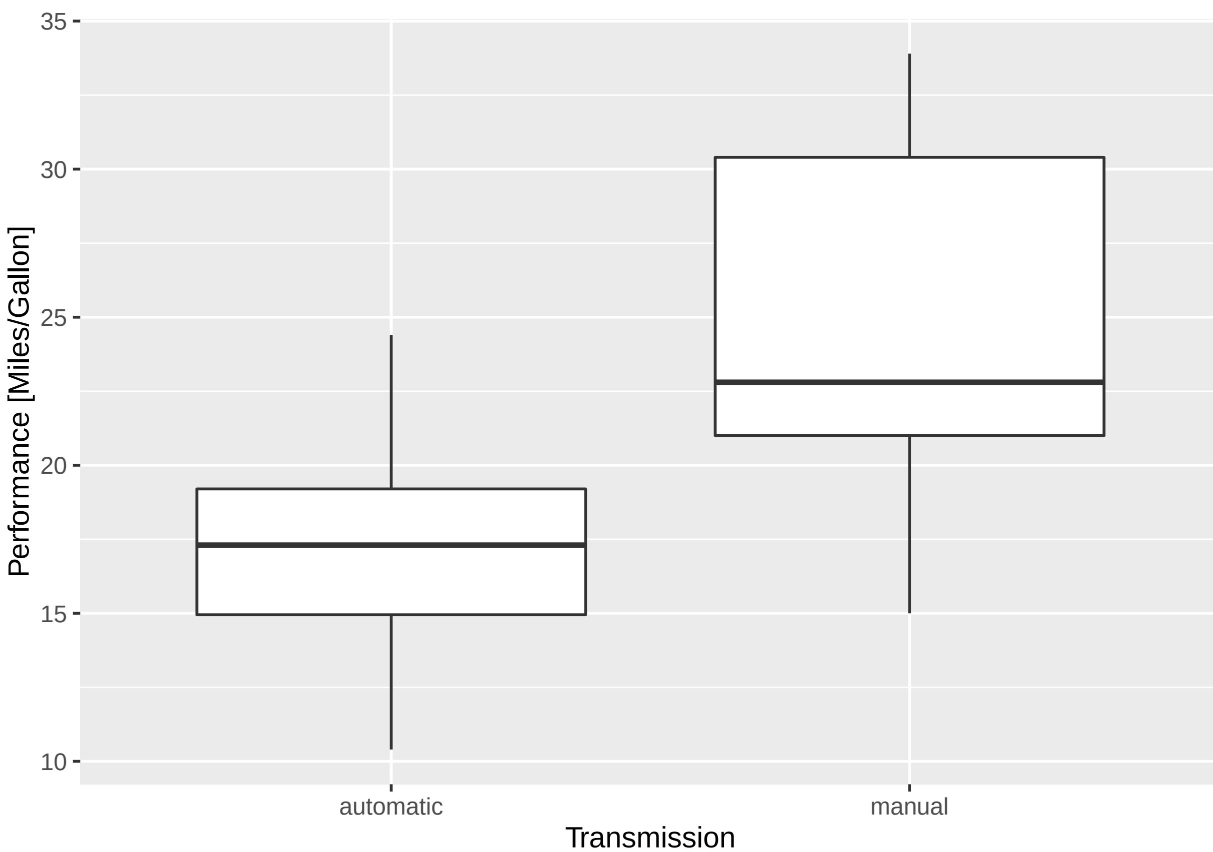
```
## 'data.frame':    32 obs. of  11 variables:
##  $ mpg : num  21 21 22.8 21.4 18.7 18.1 14.3 24.4 22.8 19.2 ...
##  $ cyl : num  6 6 4 6 8 6 8 4 4 6 ...
##  $ disp: num  160 160 108 258 360 ...
##  $ hp  : num  110 110 93 110 175 105 245 62 95 123 ...
##  $ drat: num  3.9 3.9 3.85 3.08 3.15 2.76 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 ...
```

With the information obtained in the chunk of code shown above shows us that the variable called `am` should be converted into `factor` type, although exist a set of other variables that could be converted in `factor` type as `cyl`, `vs`, `gear`, `carb`, it was taken the decision of let those variables a `numeric` due to the behaivor of these variables could also be modeled by a `numeric` type. the conversion of types is done by the following code:

```
# Load dplyr library
require(dplyr)

# Convert the factor data from numeric values in automatic/manual transmission values
mtcars <- mutate(mtcars, am = factor(am, labels = c("automatic", "manual")))
```

In the appendix A is shown a correlation graphic that shows all the correlations among variables in the dataset, but the `factor` type shows a boxplot instead of a points and linear graphic that could be seen in more detail using the following code:

```
# Load ggplot2 library
library(ggplot2)

# BoxPlot
g <- ggplot(mtcars, aes(x=am, y=mpg))
g <- g + geom_boxplot()
g <- g + xlab("Transmission") + ylab("Performance [Miles/Gallon]")
g
```



Thanks to the graphic above we can say that manual transmission has a worst impact in performance [Miles/Gallon] than automatic transmission.

Regression models

A first aproximation to the regression model, it's used a simple model with just a variable as predictor `am` and the outcome will be `mpg`, this is shown below this text:

```r
# simple model
modelSimple <- lm(mpg ~ am, data = mtcars)

# T-test: This data is inside summary(modelSimple)
# t.test( mpg ~ am, data = mtcars)

# get some information about this model
summary(modelSimple)
```

```
##
## Call:
## lm(formula = mpg ~ am, data = mtcars)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -9.3923 -3.0923 -0.2974  3.2439  9.5077
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   17.147      1.125  15.247 1.13e-15 ***
## ammanual       7.245      1.764   4.106 0.000285 ***
## ---
## 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:  0.3598, Adjusted R-squared:  0.3385
## F-statistic: 16.86 on 1 and 30 DF,  p-value: 0.000285
```

We can see that the intercept is the average value for mpg when transmission is automatic the mean value is equal to `17.147` and the `ammanual` variable represents the data when the transmission when is manual, but is necessary to add the intercept to get the right value that is `24.392`, this means that manual transmission has worst performance in [Miles/Gallon] than automatic transmission as we can see our results are p-value significant because `p-value < 0.05`, but watching at `Adjusted R-squared` we know that our model just explain a 33.85% of the variance in the outcome, for this reason is better to increase the vairables in the predictor to increase `Adjusted R-squared` variable.

To do this job it was necessary to calculate models including all variables to see the behaivor as shown below:

```r
modelMultiple <- lm(mpg ~ ., data = mtcars)

summary(modelMultiple)
```

To create a model with multiple variables we use an aproximation by fixing the `am` variable and adding iterativelly variables with low p-values in the global model, until reach a model where the p-values are significant, this model is shown below:

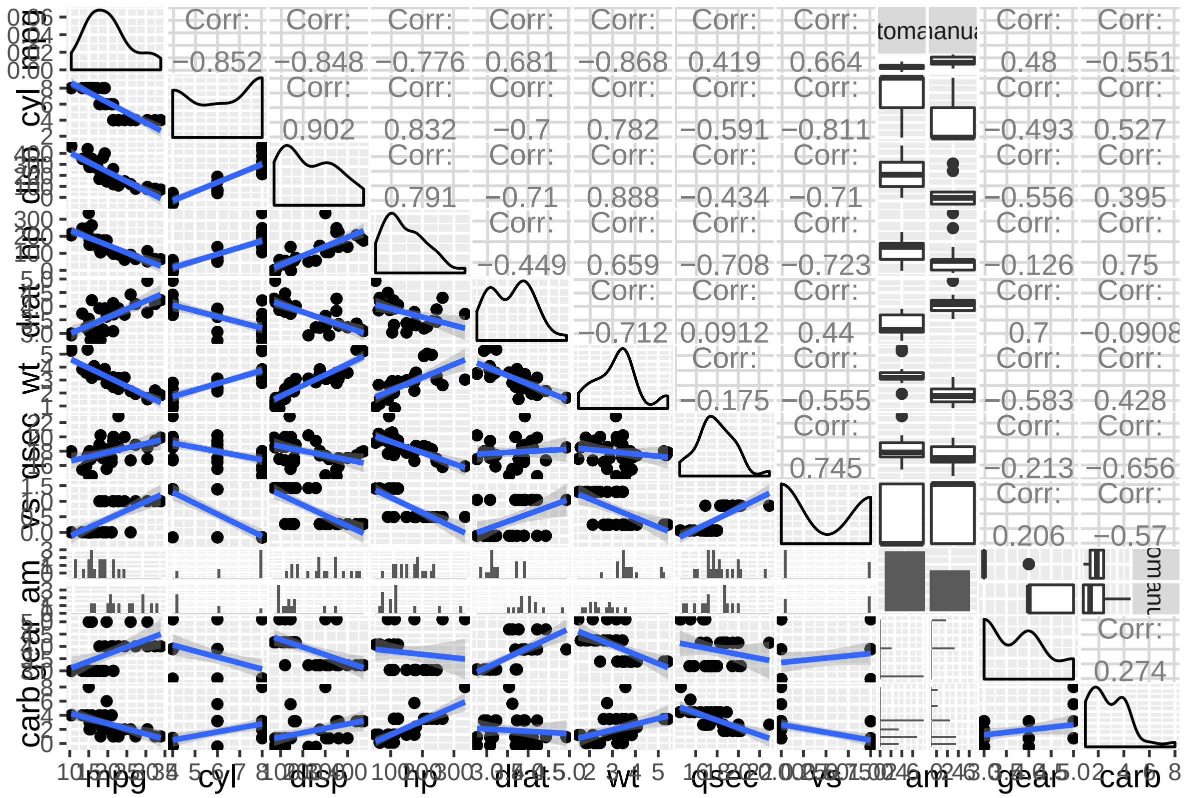
```r
modelAdjusted <- lm(mpg ~ am + wt + qsec, data = mtcars)

summary(modelAdjusted)
```

```
##
## Call:
## lm(formula = mpg ~ am + wt + qsec, data = mtcars)
##
## 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
## ammanual      2.9358     1.4109   2.081 0.046716 *
## wt           -3.9165     0.7112  -5.507 6.95e-06 ***
## qsec          1.2259     0.2887   4.247 0.000216 ***
## ---
## 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
```

As we can see in the adjusted model, the variables are significant according with p-values and the Adjusted R-squared shows us that our model explain the 83,36% of variance in the outcome. In the Appendix B is hsown the results of residuals in the adjusted model.

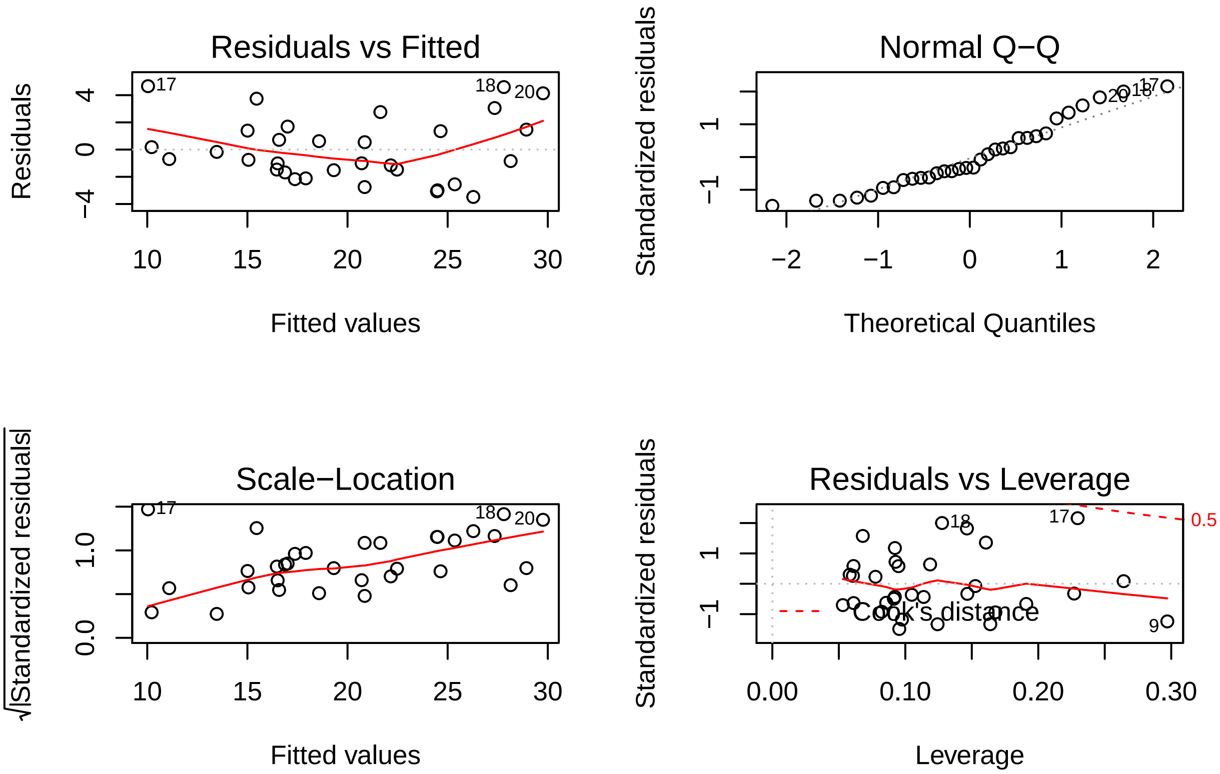
Appendix A

```r
# Load GGally library
require(GGally)

# Function to return points and geom_smooth
# allow for the method to be changed
# This was necessary due to some problems in the GGally
# implementation used in the Regression Models videos
#
# source: http://stackoverflow.com/a/35088740
my_fn <- function(data, mapping, method="loess", ...){
    p <- ggplot(data = data, mapping = mapping) +
        geom_point() +
        geom_smooth(method=method, ...)
    p
}

# Quick overview of data correlations
g <- ggpairs(mtcars, lower = list(continuous = wrap(my_fn, method="lm")))
g
```



Appendix B

```r
par(mfrow=c(2, 2))
plot(modelAdjusted)
```



In the first graphic seems to exist heteroskadicity, the second show us the normality of errors and seems to be OK, but exist a little sinusoidal pattern that will be interesting analyze later, the third graphic shows the than the first but with standarized residual values and finally the fourth graphic shows that exist some points like 9 and 17 that is necessary remove because have a different behaivor.