Regression Models Course Project

Luciano Lattes

August, 2015

Summary

Given a dataset with a collection of cars, we are interested in exploring the relationship between a set of variables and the **miles per US gallon**. In particular, we would like to focus in the effect of the type of transmissions and the mentioned outcome and answer the following questions.

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

To be able to answer those questions we will explore the data and fit linear models to evaluate to what extent we can explain the variance of `mpg` in terms of `manual` or `automatic` transmission.

Main report

Effect of transmission in the miles per US gallon

As a foundation for this analysis, let's start by fitting a linear model using `mpg` as outcome and `am` as predictor. This could be a very basic approach to achieve the goal of explaining the relationship between `mtcars` variables and miles per US gallon (outcome) but since the main interest is centered in answering the questions in the summary, we will use this linear model as the starting point.

```r
# Load the data and fit the linear model
data(mtcars)
fit1 <- lm(mpg ~ am, data = mtcars)
summary(fit1)$coef
```

```
##              Estimate Std. Error   t value     Pr(>|t|)
## (Intercept) 17.147368   1.124603 15.247492 1.133983e-15
## am           7.244939   1.764422  4.106127 2.850207e-04
```

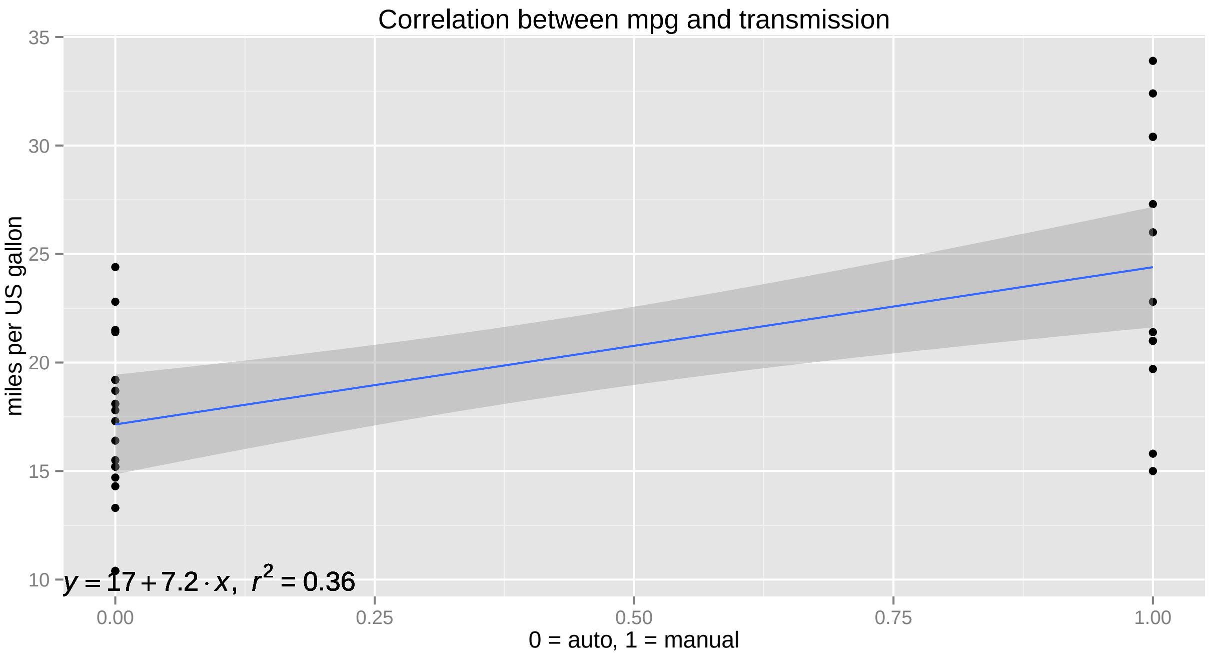
Let's also summarise the data using `dplyr` to obtain the mean of mpg for each group of automobiles: automatic and manual.

```r
# Group data by transmission type and get the mean of each group
summarise(group_by(mtcars, am), mn = mean(mpg))
```

```
## Source: local data frame [2 x 2]
##
##   am       mn
## 1  0 17.14737
## 2  1 24.39231
```

The mean of the groups is useful to interpret the coefficients of the linear model. It's easy to figure out that the sum of the estimates (intercept and `am`) equals the mean of the manual group of automobiles.

To illustrate the model and coefficients shown above, let's plot the data using `ggplot2`. The figure shows all the data points in the dataset, the `lm` smooth and, at the bottom-left corner, the equation of the smooth and the R-squared value for the fitted model.



The first thing to notice is that the correlation between these two variables, according to **Hair et al. (2011) & Hair et al. (2013)** rule of thumb for R-squared, is **low to moderate**. The plot also suggests that manual transmission (`[, 9] am Transmission (0 = automatic, 1 = manual)`) has a positive outcome on Miles/(US) gallon. According to the coefficients of the fitted model, a switch from automatic to manual (or from 0 to 1) results in an increase of the `mpg` of about `7.2mpg`.

However, it is indeed more interesting to obtain a 95% confidence interval for the intercept and the slope of the smooth to draw some more conclussions.

```r
coefSummary <- summary(fit1)$coefficients
# Intercept confidence interval
interval <- (coefSummary[1, 1] + c(-1, 1) * qt(.975, df = fit1$df) * coefSummary[1, 2])
interval
```

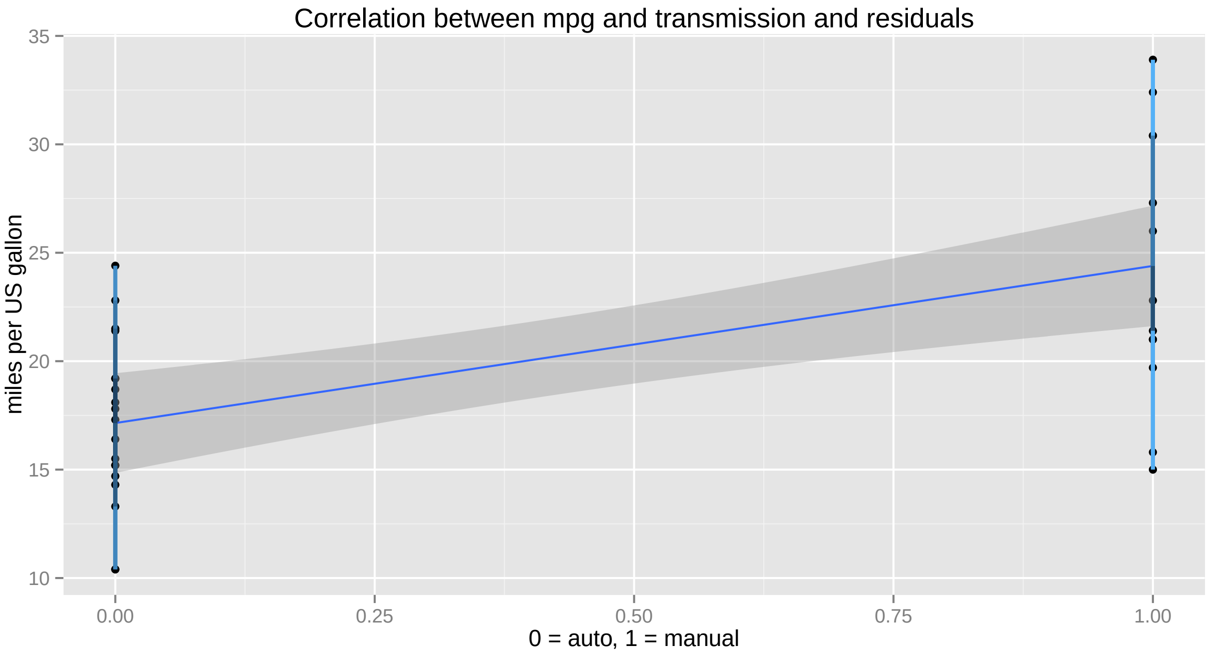
```
## [1] 14.85062 19.44411
```

```r
# Slope confidence interval
interval <- (coefSummary[2, 1] + c(-1, 1) * qt(.975, df = fit1$df) * coefSummary[2, 2])
interval
```

```
## [1]  3.64151 10.84837
```

With 95% confidence, we can estimate that a `1` increase (switching from automatic to manual) in `am` results in a 3.64 to 10.85 increase in `Miles/(US) gallon`. It's easy to observe that the interval is somewhat wide, what suggests what we already mentioned about the strength of the correlation between the two variables.

Finally, it is also interesting to take a look at the residuals. Due to the binarity of the am variable it will be kind of difficult to see the results clearly as they will overlap, but we can get the idea that most of the residuals (light-blue represents higher residuals, black represents lower) are too high, showing the poor correlation between these covariates.



Find a better model

As we could not explain all the variance of mpg by just including am in the model, let's find a better model adding more significant variables.

We'll use the ANOVA function to compare different options (adding one variable of interest at a time).

To start with this approach it is useful to look at the **Figure 1** of the appendix. It describes, in a single plot, the correlation between each pair of variables in the mtcars dataset which is very helpful when we have to choose which variables we should start with.

For example, the number of cylinders (cyl) and the displacement in cu.in. (disp) have a very high positive correlation. According to the course lectures, **including any new variables increases (actual, not estimated) standard errors of other regressors. So we don't want to idly throw variables into the model**. We can verify that a model that includes cyl should not include disp too as it would cause variance inflation:

```
fitAll <- lm(cyl ~ ., data = mtcars)
# Obtain the variance inflation factors of the model << cyl ~ . >>
factors <- vif(fitAll)
factors
```

```
##       mpg      disp        hp      drat        wt      qsec        vs
##  7.630375 20.619661  9.973784  3.159952 17.713860  7.436403  4.463041
##        am      gear      carb
##  4.679750  4.749432  7.523478
```

The value 20.6196611 for `disp` is the increase in the variance for the regressor compared to the ideal setting where it is orthogonal to the other regressors, which is orders of magnitude higher than most of the other factors, that's why we said the model should not include `disp` if it already includes `cyl`.

Following that principle let's choose some variables for our model trying to avoid **variance inflation**.

```r
# Starting model
fit1 <- lm(mpg ~ am, data = mtcars)
# Add 1 variable (cyl)
fit2 <- lm(mpg ~ am + cyl, data = mtcars)
# Add 1 variable (wt)
fit3 <- lm(mpg ~ am + cyl + wt, data = mtcars)
# Add 1 variable (hp)
fit4 <- lm(mpg ~ am + cyl + wt + hp, data = mtcars)
# Run ANOVA with the 4 models to compare
anova(fit1, fit2, fit3, fit4)
```

```
## Analysis of Variance Table
##
## Model 1: mpg ~ am
## Model 2: mpg ~ am + cyl
## Model 3: mpg ~ am + cyl + wt
## Model 4: mpg ~ am + cyl + wt + hp
##   Res.Df    RSS Df Sum of Sq       F    Pr(>F)
## 1     30 720.90
## 2     29 271.36  1    449.53 71.3976 4.619e-09 ***
## 3     28 191.05  1     80.32 12.7561  0.001358 **
## 4     27 170.00  1     21.05  3.3432  0.078553 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

According to the results above, adding `hp` is not significant (`p-value` too high, low `RSS` variation compared to `fit3`), so we can say that `fit3` is the best of the selected 4 models to explain variation in `mpg`.

```
##
## Call:
## lm(formula = mpg ~ am + cyl + wt, data = mtcars)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -4.1735 -1.5340 -0.5386  1.5864  6.0812
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  39.4179     2.6415  14.923 7.42e-15 ***
## am            0.1765     1.3045   0.135  0.89334
## cyl          -1.5102     0.4223  -3.576  0.00129 **
## wt           -3.1251     0.9109  -3.431  0.00189 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.612 on 28 degrees of freedom
## Multiple R-squared:  0.8303, Adjusted R-squared:  0.8122
## F-statistic: 45.68 on 3 and 28 DF,  p-value: 6.51e-11
```

Compared to our starting point (`fit1`), all coefficients look much "better" in this new model. The adjusted R-squared varied from `0.3385` to `0.8122`, the p-value from `0.000285` to `6.51e-11` and the residual standard error from `4.902 on 30 degrees of freedom` to `2.612 on 28 degrees of freedom`.

Appendix

Figure 1. Pairs of variables and their linear correlation

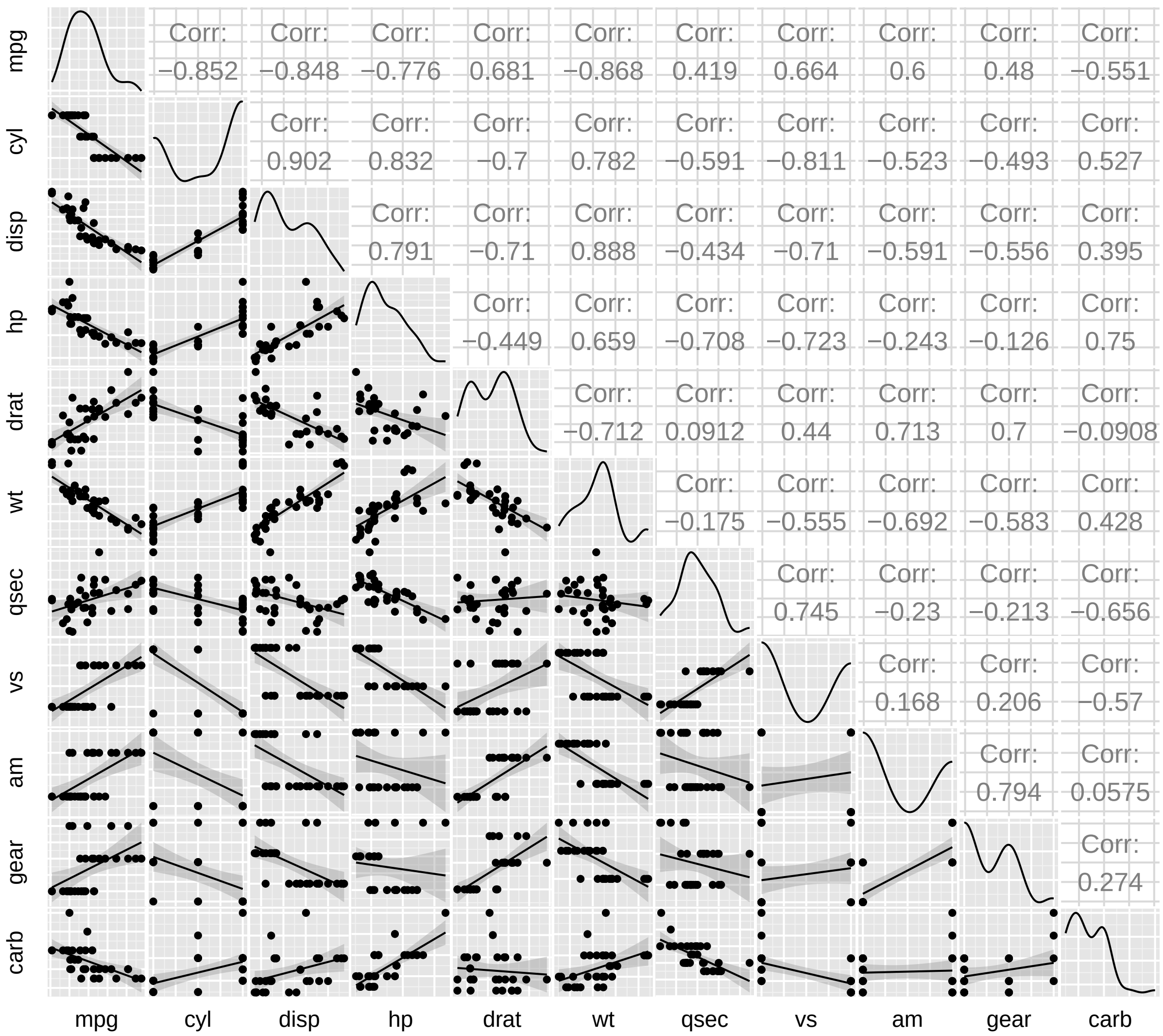
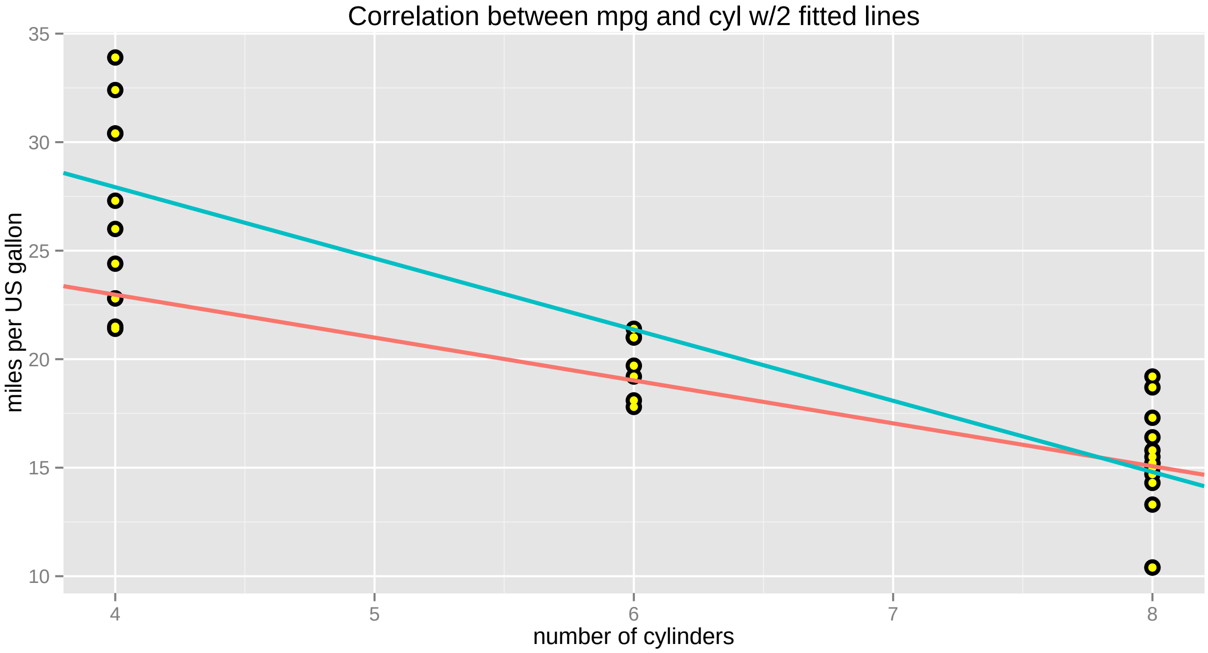


Figure 2. Alternative to fit2 model, with one regression line per transmission type

```
# Alternative 'fit2' model.
fit2Alt <- lm(mpg ~ cyl * factor(am), data = mtcars)
fit2Alt
```

```
##
## Call:
## lm(formula = mpg ~ cyl * factor(am), data = mtcars)
##
## Coefficients:
##     (Intercept)              cyl      factor(am)1  cyl:factor(am)1
##          30.874           -1.976           10.175           -1.305
```



Note: The `red` line is for automatic cars and the `blue-ish` for manual.