Regression Models Coursera Project

Carlos Barco Blanco

June 18, 2017

Executive Summary

The *Motor-Trend* Magazine is interested in publish an investigation about the relationship between miles per gallon (*mpg*) efficiency and a set of variables, one of the most important is transmission type (manual or automatic). #Main Analysis First, we need to load the data & explore the relationship between different transmissions and MPG.

Pairs plotting.

```
library(knitr)
library(rgl) #to produce interactive 3-D plots
knit_hooks$set(webgl = hook_webgl)
```

```
data(mtcars)
library(car)
library(GGally)
library(ggplot2)
pairs.plot<-ggpairs(mtcars) +
            theme(axis.line=element_blank(),
            axis.text=element_blank(),
            axis.ticks=element_blank())
```

Cleaning up the data set and running the first regression.

```
colnames(mtcars)<-c("Miles/Gallon","Number of Cylinders",
                    "Displacement (cu.in.)","Gross Horsepower","Rear Axle Ratio",
                    "Weight (lb/1000)","1/4 Mile Time","Engine Type","Trans Type",
                    "Number of Forward Gears","Number of Carburetors")
corr.matrix<-cor(mtcars)
mtcars$`Trans Type`<-gsub(0,"Automatic",mtcars$`Trans Type`)
mtcars$`Trans Type`<-gsub(1,"Manual",mtcars$`Trans Type`)
mtcars$`Engine Type`<-gsub(0,"V-Engine",mtcars$`Engine Type`)
mtcars$`Engine Type`<-gsub(1,"Straight Engine",mtcars$`Engine Type`)
```

Model 1

- An ordinary least squares (OLS) model.
- Dependent variable: “Miles/Gallon”
- Independent variable: “Trans type” This is done after renaming the columns and binary variables for better readability.

```
fit1 <- lm(`Miles/Gallon`~`Trans Type`,data=mtcars)
fit1.manual.mpg <- round(sum(fit1$coefficients),2)
fit1.manual.mpg
```

```
## [1] 24.39
```

```
fit1.automatic.mpg<-unname(round(fit1$coefficients[1],2))
fit1.automatic.mpg
```

```
## [1] 17.15
```

It appears that on average, a manual transmission will yield **24.39** Miles/Gallon and an automatic transmission will yield **17.15** Miles/Gallon.

```
fit1.qqplot<-ggplot(fit1, aes(sample=fit1$residuals)) +
             stat_qq() +
             geom_abline(intercept=0,slope=5) +
             xlab("Theoretical Quantiles") +
             ylab("Miles/Gallon ~ Trans Type Residuals")
```

The erros appears to be normally distributed (see Simple OLS Residuals), but there are many other covariables affecting *mpg* tham just *TransType*.

Model 2 and probability summaries.

Uses all of the variables to explain `Miles/Gallon`.

```
fit2<-lm(mtcars$`Miles/Gallon`~.,data=mtcars)
summary.fit2<-summary.lm(fit2)
summary.fit2
```

```
##
## Call:
## lm(formula = mtcars$`Miles/Gallon` ~ ., data = mtcars)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -3.4506 -1.6044 -0.1196  1.2193  4.6271
##
## Coefficients:
##                           Estimate Std. Error t value Pr(>|t|)
## (Intercept)               12.62114   19.02842   0.663   0.5144
## `Number of Cylinders`     -0.11144    1.04502  -0.107   0.9161
## `Displacement (cu.in.)`    0.01334    0.01786   0.747   0.4635
## `Gross Horsepower`        -0.02148    0.02177  -0.987   0.3350
## `Rear Axle Ratio`          0.78711    1.63537   0.481   0.6353
## `Weight (lb/1000)`        -3.71530    1.89441  -1.961   0.0633 .
## `1/4 Mile Time`            0.82104    0.73084   1.123   0.2739
## `Engine Type`V-Engine     -0.31776    2.10451  -0.151   0.8814
## `Trans Type`Manual         2.52023    2.05665   1.225   0.2340
## `Number of Forward Gears`  0.65541    1.49326   0.439   0.6652
## `Number of Carburetors`   -0.19942    0.82875  -0.241   0.8122
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
```

```
## Residual standard error: 2.65 on 21 degrees of freedom
## Multiple R-squared:  0.869,  Adjusted R-squared:  0.8066
## F-statistic: 13.93 on 10 and 21 DF,  p-value: 3.793e-07
```

We see that the R-squared value is **0.869**, meaning that our model explains near to **87%** of the variance. The average probability of the hypothesis that the different coefficients have a population mean = 0, equivalent to no effect on `Miles/Gallon` is **0.53** (distribution with a true mean of zero).

```
fit2.p.mean<-round(mean(summary.fit2$coefficients[,4]),2)
fit2.p.mean
```

```
## [1] 0.53
```

The regression coefficient for `Weight (lb/1000)` has an estimated probability of **0.06** (distribution with a true mean of zero).

```
fit2.p.min<-round(min(summary.fit2$coefficients[,4]),2)
fit2.p.min
```

```
## [1] 0.06
```

The population coefficient has an **0.92** as estimated probability of being from a distribution that is centered around zero.

```
fit2.p.max<-round(max(summary.fit2$coefficients[,4]),2)
fit2.p.max
```

```
## [1] 0.92
```

According to the model, here occurs a Variance Inflation Factor, due to the high collinearity of the predictors, so not all the variables are needed to predict *mpg* efficiency, but it must be considered the bias due to exclude regressors correlated wth others.

Model 3

Using a matrix of correlations (see Apendix 3) we can avoid variance inflation due to collinearity.

```
library(reshape2)
melt.corr<-melt(corr.matrix)
corr.plot<-ggplot(data =  melt.corr, aes(x = Var1, y = Var2))+
  geom_tile(aes(fill = value), colour = "white") +
  geom_text(aes(label = sprintf("%1.2f",value)), vjust = 1) +
  scale_fill_gradient(low = "white", high = "steelblue") +
  xlab(NULL) +
  ylab(NULL) +
  theme(axis.text.x = element_text(angle = 90, hjust = 1)) +
  theme(axis.text = element_text(colour = "black"))
```

So, the variables that has to be considered are:`Trans Type`and `Weight (lb/1000)`, the last one with a great linear relationship with `Miles/Gallon` = **-0.87**

```
cor.1<-round(cor(mtcars$`Weight (lb/1000)`,mtcars$`Miles/Gallon`),2)
cor.1
```

```
## [1] -0.87
```

```
library(car)
fit3<-lm(`Miles/Gallon`~`1/4 Mile Time`+`Weight (lb/1000)`
         +`Gross Horsepower`+`Number of Carburetors`
```

```
        +`Weight (lb/1000)`*`Trans Type`, data=mtcars)
fit3.summary<-summary.lm(fit3)
fit3.coefficients<-round(fit3.summary$coefficients,2)
fit3.coefficients
```

```
##                                    Estimate Std. Error t value Pr(>|t|)
## (Intercept)                           10.06       8.70    1.16     0.26
## `1/4 Mile Time`                        1.00       0.40    2.50     0.02
## `Weight (lb/1000)`                    -2.93       0.82   -3.58     0.00
## `Gross Horsepower`                     0.00       0.01   -0.12     0.91
## `Number of Carburetors`                0.06       0.43    0.14     0.89
## `Trans Type`Manual                    14.06       3.92    3.58     0.00
## `Weight (lb/1000)`:`Trans Type`Manual -4.16       1.43   -2.91     0.01
```

The variables to be excluded (according to their p-value) are: `Gross Horsepower` (**0.91**) and `Number of Carburetors` (**0.89**), because are highly correlated with `1/4 Mile Time`

```
fit3.qqplot<-ggplot(fit1, aes(sample=fit3$residuals)) +
             stat_qq() +
             geom_abline(intercept=0,slope=2) +
             xlab("Theoretical Quantiles") +
             ylab("Miles/Gallon~1/4 Mile Time+Weight*Trans Type")
```

Final model

```
round(fit3.summary$coefficients,4)
```

```
##                                    Estimate Std. Error t value Pr(>|t|)
## (Intercept)                         10.0619     8.7032  1.1561   0.2586
## `1/4 Mile Time`                      1.0021     0.4008  2.5000   0.0193
## `Weight (lb/1000)`                  -2.9271     0.8166 -3.5844   0.0014
## `Gross Horsepower`                  -0.0017     0.0142 -0.1183   0.9068
## `Number of Carburetors`              0.0606     0.4272  0.1419   0.8883
## `Trans Type`Manual                  14.0597     3.9219  3.5849   0.0014
## `Weight (lb/1000)`:`Trans Type`Manual -4.1610     1.4297 -2.9103   0.0075
```

```
round
```

```
## function (x, digits = 0)  .Primitive("round")
```

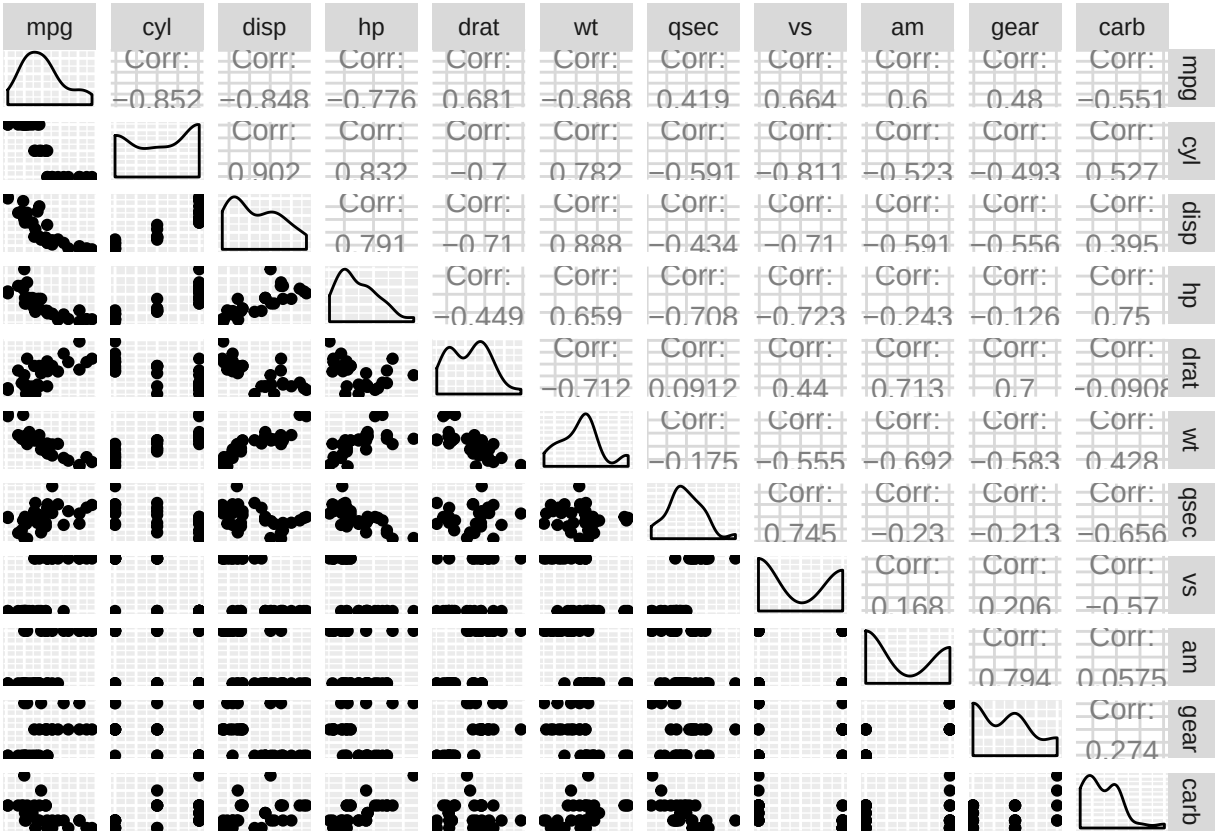
The slope and intercept change in this model depending on the `Trans Type`.

Is an automatic or manual transmission better for MPG?

It depends on the weight of the car. Lighter cars benefit from having manual transmissions while heavier cars don't, holding `1/4 Mile Time` constant. ###Quantify the MPG difference between automatic and manual transmissions. When the transmission is Automatic the intercept is 10.0619, a one unit increase in `Weight (lb/1000)` results in a -2.9271 decrease in `Miles/Gallon`, with `1/4 Mile Time`constant. When it is Manual, the rate of change in `Miles/Gallon` intercept is $10.0619 + 14.0597 = 24.1216$, a one unit increase in the `1/4 Mile Time` results in a 1.0021 increase in `Miles/Gallon` and a one unit increase in `Weight (lb/1000)` results in a fall in the rate of change of $-2.9271 + -4.1610 = -7.0881$ in `Miles/Gallon`, with `1/4 Mile Time` constant.

#Appendix A.1 Pairs Plot

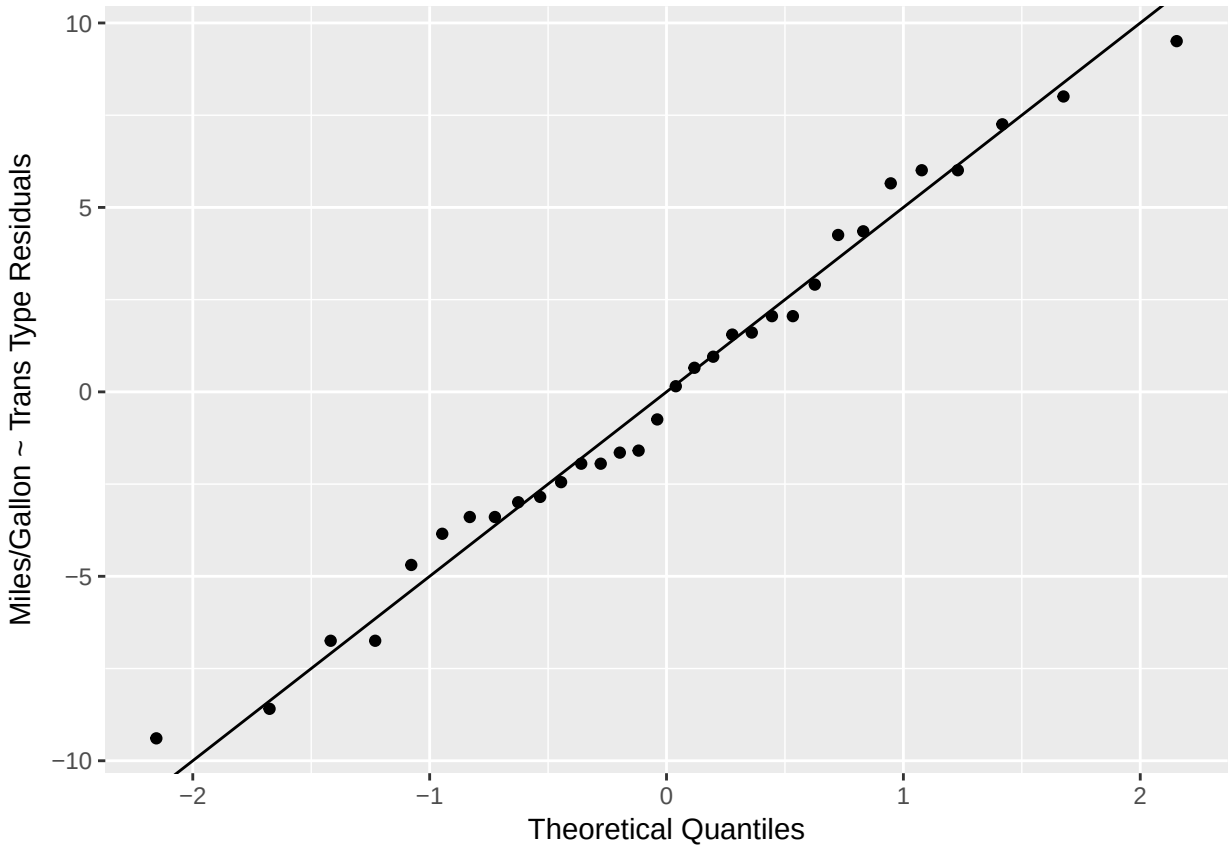
```
pairs.plot
```



The highly variable distributions and small sample sizes may distort the accuracy of any model. Also, since the sample was not selected using a randomized process this means that the sample is biased. This also affects the accuracy of any model.

A.2 Simple OLS Residuals

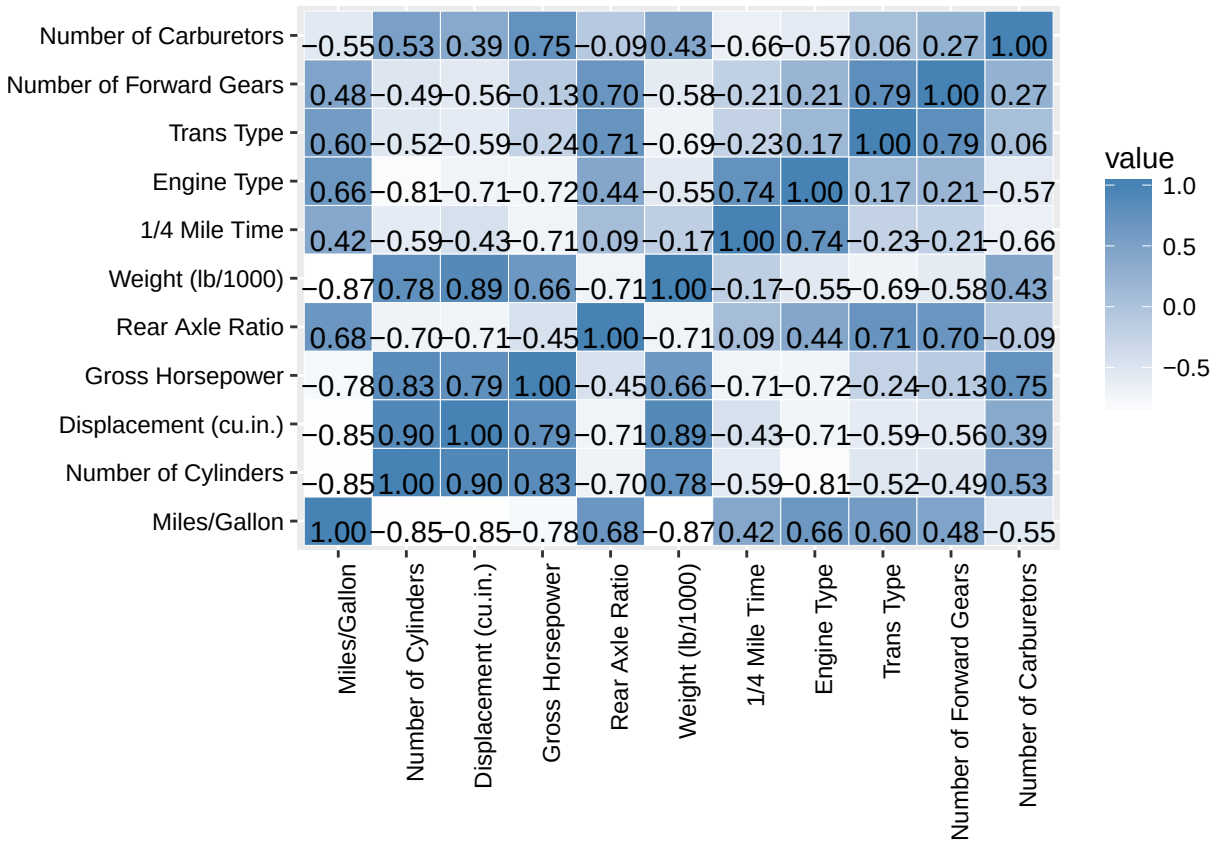
```
fit1.qqplot
```



Miles/Gallon as the dependent and Trans Type as the independent variable. This is done after renaming the columns and binary variables for better readability. It appears that on average, a manual transmission will yield 24.39 Miles/Gallon and an automatic transmission will yield 17.15 Miles/Gallon. There are many other linear variables that affect Miles/Gallon that are offsetting each other in the residuals.

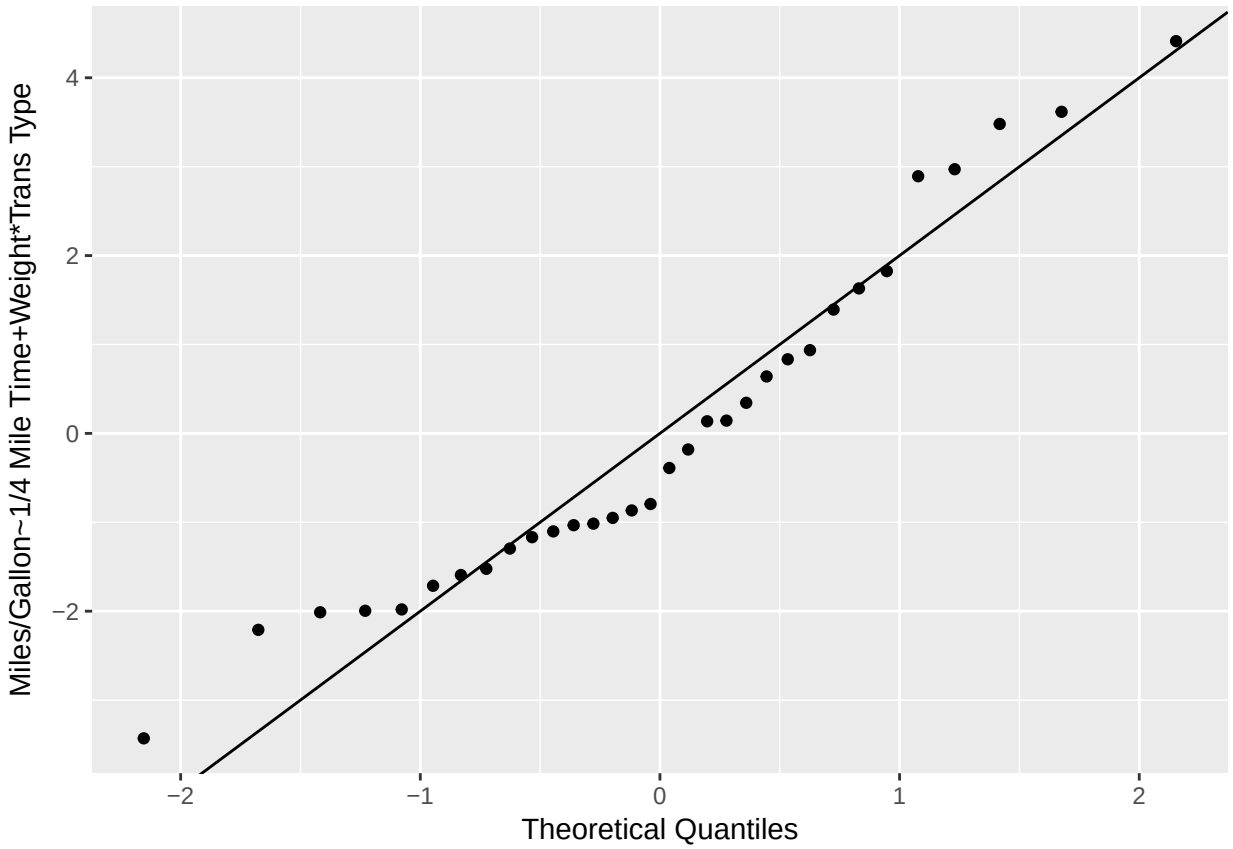
A.3 Full Correlation Matrix

```
corr.plot
```



A.4 Final Model Residuals

```
fit3.qqplot
```



the residual distribution doesn't appear to be normal. The model's intercept is also not very significant with a probability of 0.2586 that it comes from a distribution with a population mean of zero. The intercept's meaning doesn't have much weight in this model because none of the predictors can be zero by construction. Again, the small and highly variable sample can also distort the significance and meaning of the coefficients.

The bias in the sample can also artificially inflate the adjusted coefficient of determination of the multiple linear regression model for the data set = 0.87

```
round(fit3.summary$adj.r.squared,2)
```

```
## [1] 0.87
```

Even after being adjusted for the number of covariate parameters, this seems a bit too high. There is likely a mechanistic and nonlinear relationship between some of the variables.

A.5 Final Model

```
scatter3d(`Miles/Gallon`~`1/4 Mile Time`+`Weight (lb/1000)`*as.factor(`Trans Type`),data=mtcars)
```