Regression Models Course Project

Angela Frolov

March 8, 2017

In this project we explore a data set, describing a collection of cars, called mtcars. We are interested in exploring the relationship between a set of variables, such as number of cyllinders, horse power, weight etc and their effect on miles per gallon (MPG) outcome. In particular we are looking for the answers to the following two questions:

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

Executive Summary

Based on our analysis it looks like that there is not much difference in MPG between cars with automatic and manual transmission. Adjusted for number of cylinders and weight the manual transmission adds just 0.15 mpg.

I. We start with exploratory analysis

```
# include libraries
library(ggplot2)

library(corrgram)

# check data set
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 ...
```

```
# check average mpg for manual and automatic cars
mpg_m <- mean(mtcars[mtcars$am==1, "mpg"])
mpg_a <- mean(mtcars[mtcars$am==0, "mpg"])
```

We have a data set consisting of 32 observations with 11 numerical variables. The calculation of averaged mpg shows that cars with manual transmission have 24.3923077 mpg vs. 17.1473684 mpg for automatic. This can be visualy verified by Figure1, see an Appendix.

II. Regression models.

Let us fit simple lenear model with MPG as outcome and transmission as a regressor.

```
fit <- lm(mpg~am, mtcars)
summary(fit)$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
```

```
rsquared <- summary(fit)$r.squared
```

This model supports our previous findings regarding automatic vs.manual MPG, but R-squared value 0.3597989 shows that the model is very underfitted and can not be used as good predictor of MPG in general.

To do more modeling, let us compute correlation matrix to see which variables have a greatest impact on MPG, see Figure 2 in the Appendix

We can see that wt, cyl, disp, hp and drat are the top parameters. We will try a few combinations of them. Since we are looking to the effect of am variable we will include it as well.

Let us build some models.

```
wt <- lm(mpg~wt+as.factor(am), mtcars)
wtcyl <- lm(mpg~wt+as.factor(cyl)+as.factor(am), mtcars)
wtdisp <- lm(mpg~wt+disp+as.factor(am), mtcars)
wthp <- lm(mpg~wt+hp+as.factor(am), mtcars)
wtcylhp <- lm(mpg~wt+as.factor(cyl)+hp+as.factor(am), mtcars)
wtdisphp <- lm(mpg~wt+disp+hp+as.factor(am), mtcars)
wtcyldisp <- lm(mpg~wt+as.factor(cyl)+disp+as.factor(am), mtcars)
wtcyldisphp <- lm(mpg~wt+as.factor(cyl)+hp+as.factor(am), mtcars)
```

The anova analysis (see Figure 3) shows, that models 4, 5, 6 and 8 are pretty good contenders. Let us check R-squared values for all of them. For Model4 it is 0.8658799, for Model5 it is 0.7810427, for Model6 it is 0.8375127 and for Model8 it is 0.7528348. As we can see Models 4 and 6 accounts for the most variation in MPG. Before making our final desision let us check models residuals, see Figures 3 and 4 in Appendix.

Looking at the plots we can observe that for Model4, there is some pattern in both left residual plots (they go down and then up), so there is some dependency among the variables. For Model 5 there are no patterns, and all the plots looks fine. Therefore, we can use Model5 as our final model.

```
summary(wtcyl)$coef
```

```
##                   Estimate Std. Error    t value     Pr(>|t|)
## (Intercept)     33.7535920  2.8134831 11.9970836 2.495549e-12
## wt              -3.1495978  0.9080495 -3.4685309 1.770987e-03
## as.factor(cyl)6 -4.2573185  1.4112394 -3.0167231 5.514697e-03
## as.factor(cyl)8 -6.0791189  1.6837131 -3.6105432 1.227964e-03
## as.factor(am)1   0.1501031  1.3002231  0.1154441 9.089474e-01
```

The coefficient for transmission is very small and insignificant. Manual transmission adds just 0.15 mpg.

Appendix

Figure 1.

```
boxplot(mpg~factor(am), data=mtcars, notch=TRUE,
  col=(c("gold","darkgreen")),
  main="MPG in manual vs. automatic", xlab="Transmission Type")

## Warning in bxp(structure(list(stats = structure(c(10.4, 14.95, 17.3,
## 19.2, : some notches went outside hinges ('box'): maybe set notch=FALSE
```

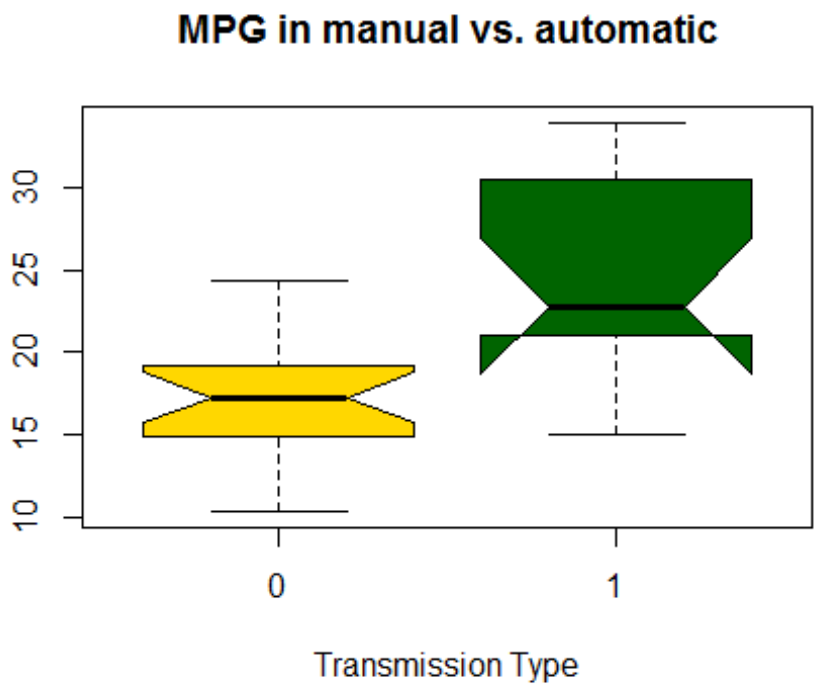

Figure 2.

```
corrgram(mtcars, order=NULL, lower.panel=panel.pie,
  upper.panel=NULL, text.panel=panel.txt,
  main="Car Mileage Data")
```

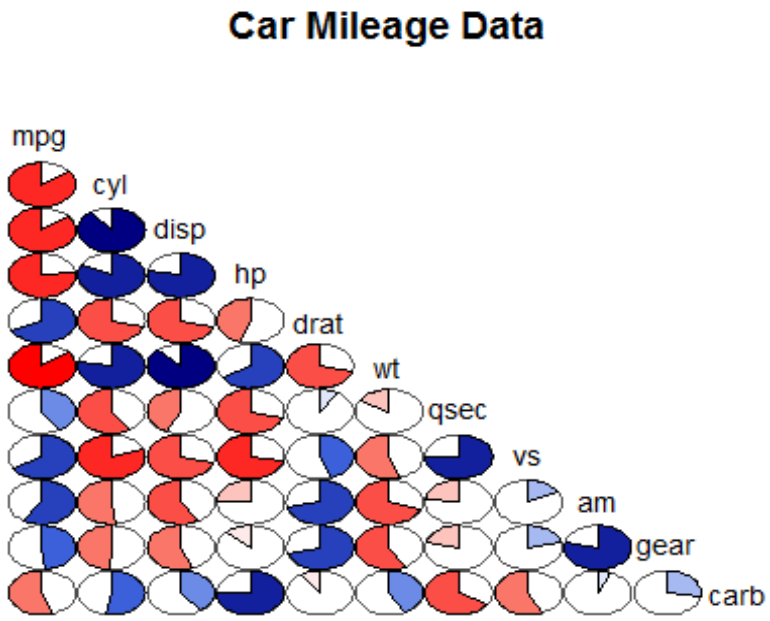

Figure 3

```
anova(wtcyldisphp, wtcyldisp, wtdisphp, wtcylhp, wtdisp, wtcyl, wthp, wt)
```

```
## Analysis of Variance Table
##
## Model 1: mpg ~ wt + as.factor(cyl) + hp + as.factor(am)
## Model 2: mpg ~ wt + as.factor(cyl) + disp + as.factor(am)
## Model 3: mpg ~ wt + disp + hp + as.factor(am)
## Model 4: mpg ~ wt + as.factor(cyl) + hp + as.factor(am)
## Model 5: mpg ~ wt + disp + as.factor(am)
## Model 6: mpg ~ wt + as.factor(cyl) + as.factor(am)
## Model 7: mpg ~ wt + hp + as.factor(am)
## Model 8: mpg ~ wt + as.factor(am)
##   Res.Df    RSS Df Sum of Sq       F    Pr(>F)
## 1     26 151.03
## 2     26 182.87  0   -31.844
## 3     27 179.91 -1     2.962
## 4     26 151.03  1    28.882  4.9722 0.0346066 *
## 5     28 246.56 -2   -95.531  8.2231 0.0017090 **
## 6     27 182.97  1    63.588 10.9471 0.0027486 **
## 7     28 180.29 -1     2.677
## 8     29 278.32 -1   -98.029 16.8762 0.0003525 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Figure 4 (Model4)

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

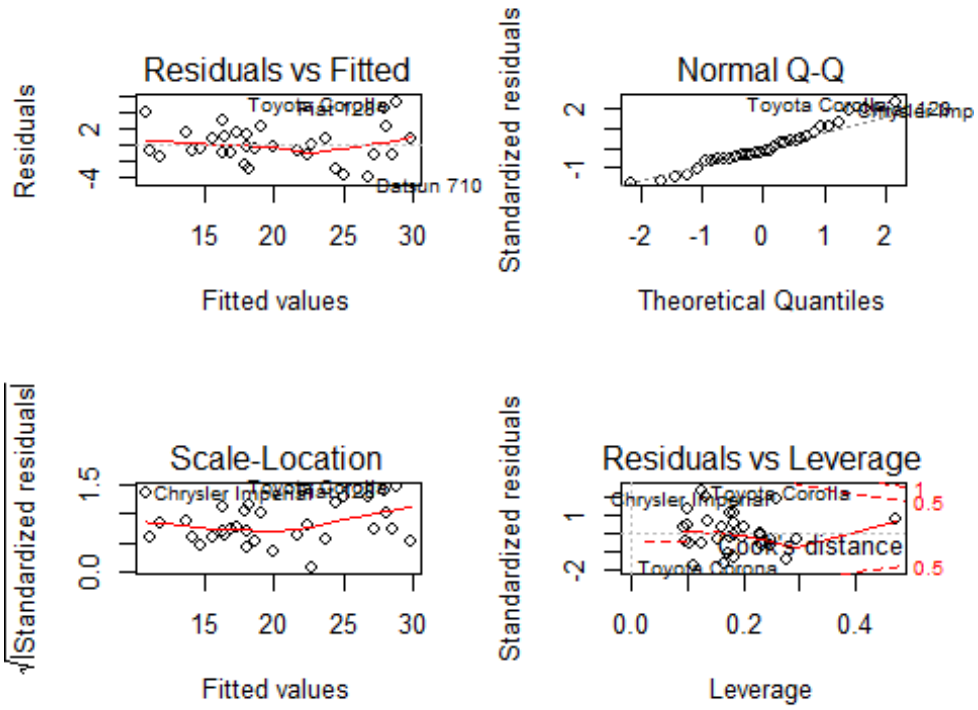

Figure 5(Model6)

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