Regression Models Course Project

William Matthews

March 10, 2018

Executive Summary

We will be reviewing the mtcars dataset and exploring the relationship between its set of variables and miles per gallon (MPG) as the outcome. Of particular interest are the following questions:

1. Is an automatic or manual transmission better for mpg?
2. Can I quantify the mpg difference between automatic and manual transmission?

Conclusions:

Based on our analysis cars with manual transmission get higher mpg than those with automatic transmission. The difference is about 7.2 mpg. However, transmission type only accounts for about 36% of the variability in mpg. As a result we looked at the other variables in the mtcars dataset to see what affect they had on mpg.

The other variables from mtcars that had the highest affect on mpg were hp, cyl, and wt.

When adjusting for hp, cyl, & wt manual transmission increased mpg by about 1.8 over automatic transmission. Mpg decreased by about 2.5 for every 1000 lb. increase in wt (adjusted for hp, cyl, & am). When cyl (the number of cylinders) increases from 4 to 6 to 8, mpg will decrease by 3 and 2.2 respectively (adjusted for hp, wt, & am).

```r
# Load mtcars dataset
data(mtcars)
```

Exploratory Data Analysis

```r
# Use dim() to obtain the dimensions of the data frame
dim(mtcars)
```

```
## [1] 32 11
```

```r
# Use head() to obtain the first n observations of the dataset.
head(mtcars, 3)
```

```
##                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
```

```r
# Use the str() function to return the structure of the mtcars dataset.
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 ...
```

```
# Display the correlation between mpg and the other variables in mtcars
cor(mtcars)[1,]
```

```
##        mpg        cyl       disp         hp       drat         wt
##  1.0000000 -0.8521620 -0.8475514 -0.7761684  0.6811719 -0.8676594
##       qsec         vs         am       gear       carb
##  0.4186840  0.6640389  0.5998324  0.4802848 -0.5509251
```

The variables with the highest correlation to mpg are cyl, disp, hp, & wt

```
# Recode selected numeric variables as factors
mtcars$cyl <- factor(mtcars$cyl)
mtcars$am <- factor(mtcars$am, labels = c("Automatic", "Manual")) # 0 = automatic, 1 = manual
mtcars$gear <- factor(mtcars$gear)
mtcars$carb <- factor(mtcars$carb)
```

Regression Analysis

```
# Aggregate the mpg by transmission type (auto, manual)
aggregate(mpg~am, data = mtcars, mean)
```

```
##          am      mpg
## 1 Automatic 17.14737
## 2    Manual 24.39231
```

The data indicates that on average mpg is higher with manual transmissions.

Statistical Inference

```
# Quantify the difference in mpg for the am variable with a t-test.
t.test(mpg ~ am, data = mtcars)
```

```
##
##  Welch Two Sample t-test
##
## data:  mpg by am
## t = -3.7671, df = 18.332, p-value = 0.001374
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -11.280194  -3.209684
## sample estimates:
## mean in group Automatic    mean in group Manual
##                17.14737                24.39231
```

The p-value = 0.001374, which is less than .05, indicating that this is a significant difference and thus reject the null hypothesis that automatic and manual transmissions have the same effect on mpg.

Model selection

Linear models

```r
# Use lm function to fit a linear model with mpg as the outcome and am as the predictor.
fit <- lm(mpg~am, data = mtcars)

# View summary of lm fit.
summary(fit)
```

```
##
## 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
```

The lm summary shows that mpg increases by 7.245 for manual transmission. The R-squared: 0.3598 indicates that this model explains only 34% of the variance of mpg.

Multivariate linear model

```r
# Use the lm function to fit the linear model with mpg as the outcome and the other variables as
# predictors.
fit1 <- lm(mpg ~ ., data = mtcars)
```

Use the step fuction to select a formula-based model by AIC of the variables that have the highest correlation to mpg, ie the "best" model. mv_fit <- step(fit1, direction = "both")

The model with the lowest AIC and thus fit is mpg ~ cyl + hp + wt + am

```r
# View summary of mv_fit model
summary(mv_fit)
```

```
##
## Call:
## lm(formula = mpg ~ cyl + hp + wt + am, data = mtcars)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -3.9387 -1.2560 -0.4013  1.1253  5.0513
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 33.70832    2.60489  12.940 7.73e-13 ***
```

```
## cyl6        -3.03134    1.40728  -2.154  0.04068 *
## cyl8        -2.16368    2.28425  -0.947  0.35225
## hp          -0.03211    0.01369  -2.345  0.02693 *
## wt          -2.49683    0.88559  -2.819  0.00908 **
## amManual     1.80921    1.39630   1.296  0.20646
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.41 on 26 degrees of freedom
## Multiple R-squared:  0.8659, Adjusted R-squared:  0.8401
## F-statistic: 33.57 on 5 and 26 DF,  p-value: 1.506e-10
```

When accounting for the other variables (cyl, hp, wt) manual transmission increases mpg by 1.8. The R^2 value indicates tht 86.59% of the variance is explained by the model.

```r
# Use anova function to compare the base model against the best model.
anova(fit, mv_fit)
```

```
## Analysis of Variance Table
##
## Model 1: mpg ~ am
## Model 2: mpg ~ cyl + hp + wt + am
##   Res.Df    RSS Df Sum of Sq      F    Pr(>F)
## 1     30 720.90
## 2     26 151.03  4    569.87 24.527 1.688e-08 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
# Test the confidence of the model.
confint(mv_fit)
```

```
##                   2.5 %       97.5 %
## (Intercept) 28.35390366 39.062744138
## cyl6        -5.92405718 -0.138631806
## cyl8        -6.85902199  2.531671342
## hp          -0.06025492 -0.003963941
## wt          -4.31718120 -0.676477640
## amManual    -1.06093363  4.679356394
```

We can say with 95% confidence that the variables correlations are within the ranges listed.

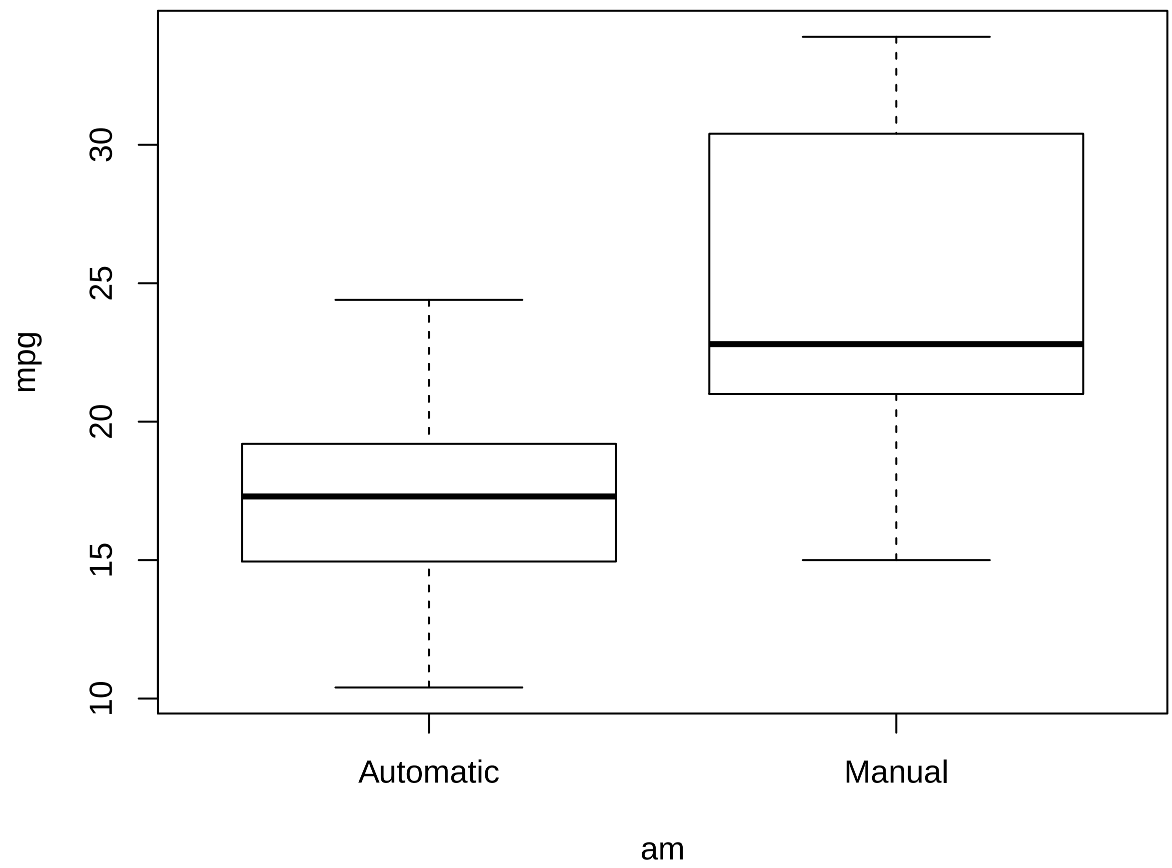
Residual

The Residuals vs Fitted plot (Appendix - Plot 3) shows that the points are randomly distributed indicating independence. The Normal Q-Q plot shows that the distribution is generally normal because the points mostly fall on the normal line. The Scale-Location plot shows the points scattered in a constant pattern indicating a constance variance condition. The Residuals vs Leverage plot shows some outliers

Appendix

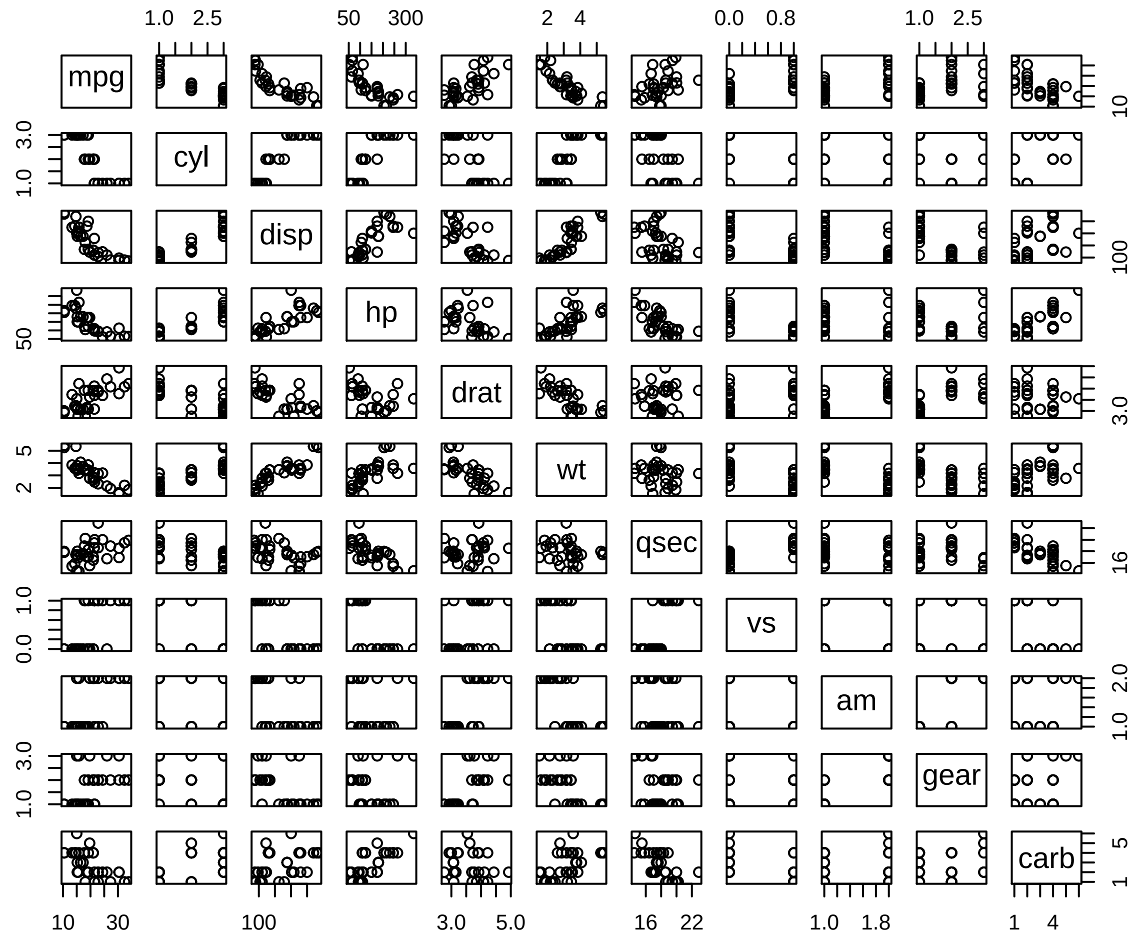
Plot 1: Plot mpg by transmission type with mpg on the y axis and transmission type on the x axis.

```
plot(mpg~am, data = mtcars)
```



Plot 2: Use the pairs function to plot a matrix of the relationship betwen mpg and the other variables.

```r
pairs(mpg~., data = mtcars)
```



Plot 3: View the residual plots for multivariate regression model and compute regression diagnostic of model to uncover outliers.

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