Regression Models Course Project

Andrew E. Davidson

May 23, 2015

I ran into problems running Knit PDF in R Studio

Unfortunatly my report is not formated as nicely as I might like. Knit did not work on my Mac. As a work around I set the Knit format to html and used my browser to convert it into PDF. This makes the report take up much more space

Explore the mtcars data set

Probably the most useful thing to do is read the data set description. Using ? mtcars we see the data set contains 11 variables, including such as engine size, number of gears, … We'll need to adjust our models to account for these variable.

set up factors

Using str it turns out cyl, gear, am, and carb are numeric, not factors. We'll need to convert them

```
str(mtcars)
```

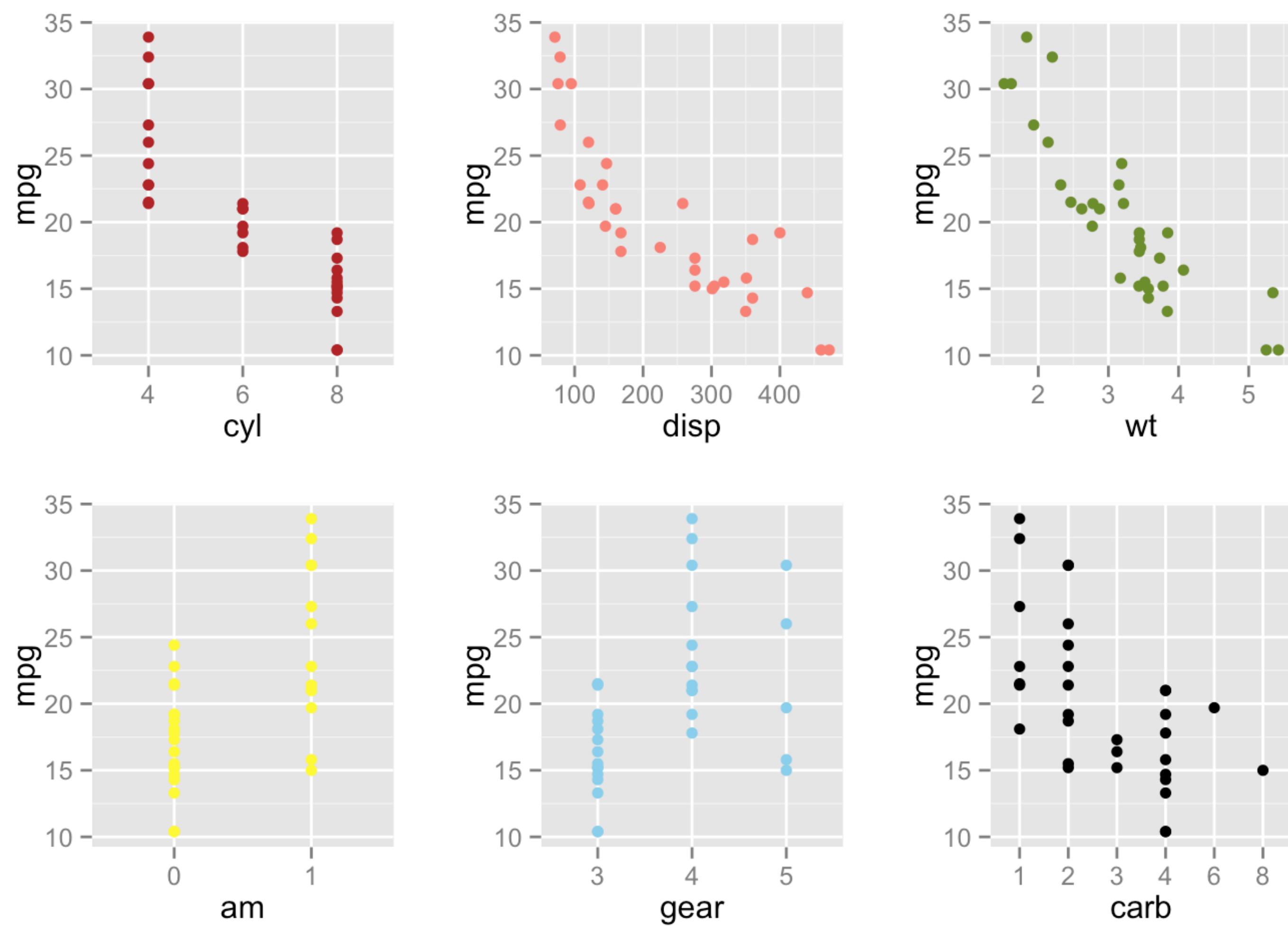
```
## 'data.frame':    32 obs. of  11 variables:
##  $ mpg : num  21 21 22.8 21.4 18.7 18.1 14.3 24.4 22.8 19.2 ...
##  $ cyl : num  6 6 4 6 8 6 8 4 4 6 ...
##  $ disp: num  160 160 108 258 360 ...
##  $ hp  : num  110 110 93 110 175 105 245 62 95 123 ...
##  $ drat: num  3.9 3.9 3.85 3.08 3.15 2.76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

```
mtcars$am <- factor(mtcars$am)
mtcars$cyl <- factor(mtcars$cyl)
mtcars$gear <- factor(mtcars$gear)
mtcars$carb <- factor(mtcars$carb)
```

Lets take a look at some of the raw sample data

```
##                      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
```

Using scatter plots we can identify variables that might be good predictors of miles per gallon. We are looking for variables that seem to have strong linear tendency. For example displayment. We are also looking for variables that might be confounding. For example we would expect the more gears a car has the better the miles per gallon would be. Just looking st the scatter plot, It looks like cars with 4 gears might do better than cars with either 3 or 5 gears suggesting that something else is going on. The confounding effect may have to do with how the test was run. Did they take the car out on track and run at a high speed for long time? If so we would expect cars with 5 gears to consistently outperform the others holding things like weight, engine size, … constant



Is an automatic or manual transmission better for MPG?

To solve this problem, we need to create a linear model that predicts MPG. We can not simply using the factor variable am, (automatic or manual transmision) as apredictor. We'll need to adjust for other factors such as weight, number of gears,engine size, … To keep things simple we will assume there are no interaction between variables. That is to say our model does not have predictors variable of the form wt * disp.

First we need to decide what variables to use in our model. If we create a linear model using all the variable and look at the col Pr(>|t|) we do not find a beta that appears to be signifigant. The best predictor seems to be vehicle weight. the Pr() value is still lower than we would normally like. The Estimate for wt is -4.5, which means for every 1000 lbs increase in vehicle weight we would expect MPG to decrese by 4.5. (Assuming we did not change any of the other variables like engine size, transmission, …)

```
summary(lm(mpg ~ ., data=mtcars))$coefficients
```

```
##             Estimate Std. Error  t value Pr(>|t|)
## (Intercept) 23.87913   20.06582  1.19004  0.25253
## cyl6        -2.64870    3.04089 -0.87103  0.39747
## cyl8        -0.33616    7.15954 -0.04695  0.96317
## disp         0.03555    0.03190  1.11433  0.28267
## hp          -0.07051    0.03943 -1.78835  0.09393
## drat         1.18283    2.48348  0.47628  0.64074
## wt          -4.52978    2.53875 -1.78426  0.09462
## qsec         0.36784    0.93540  0.39325  0.69967
## vs           1.93085    2.87126  0.67248  0.51151
## am1          1.21212    3.21355  0.37719  0.71132
## gear4        1.11435    3.79952  0.29329  0.77332
## gear5        2.52840    3.73636  0.67670  0.50890
## carb2       -0.97935    2.31797 -0.42250  0.67865
## carb3        2.99964    4.29355  0.69864  0.49547
## carb4        1.09142    4.44962  0.24528  0.80956
## carb6        4.47757    6.38406  0.70137  0.49381
## carb8        7.25041    8.36057  0.86722  0.39948
```

Whats interesting is if we look at the unadjust model. That is to say the model with a single predictor variable of weight, the is t value is signifigant! The size of the effect is also much stronger.

```
summary(lm(mpg ~ wt, data=mtcars))$coefficients
```

```
##             Estimate Std. Error t value  Pr(>|t|)
## (Intercept)   37.285     1.8776  19.858 8.242e-19
## wt            -5.344     0.5591  -9.559 1.294e-10
```

The next step to try a couple of different models. We will use the "nested model search approach to evaluate different models". I did not consider 'vs' becuase I did not know what 'V/S' ment. I also did not consider qsec.

```
f1 <- lm(mpg ~ am, data=mtcars)
f2 <- update(f1, mpg ~ am + cyl, data=mtcars)
f3 <- update(f2, mpg ~ am + cyl + disp, data=mtcars)
f4 <- update(f3, mpg ~ am + cyl + disp + hp, data=mtcars)
f5 <- update(f4, mpg ~ am + cyl + disp + hp + drat, data=mtcars)
f6 <- update(f5, mpg ~ am + cyl + disp + hp + drat + wt, data=mtcars)
f7 <- update(f6, mpg ~ am + cyl + disp + hp + drat + wt + gear, data=mtcars)
f8 <- update(f7, mpg ~ am + cyl + disp + hp + drat + wt + gear + carb, data=mtcar
s)

anova(f1, f2, f3, f4, f5, f6, f7, f8)
```

```
## Analysis of Variance Table
##
## Model 1: mpg ~ am
## Model 2: mpg ~ am + cyl
## Model 3: mpg ~ am + cyl + disp
## Model 4: mpg ~ am + cyl + disp + hp
## Model 5: mpg ~ am + cyl + disp + hp + drat
## Model 6: mpg ~ am + cyl + disp + hp + drat + wt
## Model 7: mpg ~ am + cyl + disp + hp + drat + wt + gear
## Model 8: mpg ~ am + cyl + disp + hp + drat + wt + gear + carb
##   Res.Df RSS Df Sum of Sq     F  Pr(>F)
## 1     30 721
## 2     28 264  2       456 30.84 2.2e-06 ***
## 3     27 230  1        34  4.60   0.047 *
## 4     26 183  1        47  6.41   0.022 *
## 5     25 182  1         1  0.09   0.769
## 6     24 150  1        32  4.36   0.052 .
## 7     22 148  2         2  0.11   0.892
## 8     17 126  5        23  0.61   0.693
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Based on the Pr(>F) values in the anova results the variables we should use as predictors are am, cyl, disp, hp, and wt. The Pr(>F) value for wt was actually 0.052. My intutition tells me weight is an important variable. One thing to keep in mind is the unit for wt is 1000 LBS. If weight was represented in actual pounds, it would definietly be signifigant

Becuase am is the factor variable with the lowest level of the coveraients in our model, R will automatically select it as the reference variable. That is to say the expected value in mpg for cars with automatic transmissions

```
model <- lm(mpg ~ am + cyl + disp + hp + wt, data=mtcars)
summary(model)$coefficients
```

```
##              Estimate Std. Error t value  Pr(>|t|)
## (Intercept) 33.864276    2.69542 12.5637 2.668e-12
## am1          1.806099    1.42108  1.2709 2.155e-01
## cyl6        -3.136067    1.46909 -2.1347 4.277e-02
## cyl8        -2.717781    2.89815 -0.9378 3.573e-01
## disp         0.004088    0.01277  0.3202 7.515e-01
## hp          -0.032480    0.01398 -2.3228 2.862e-02
## wt          -2.738695    1.17598 -2.3289 2.825e-02
```

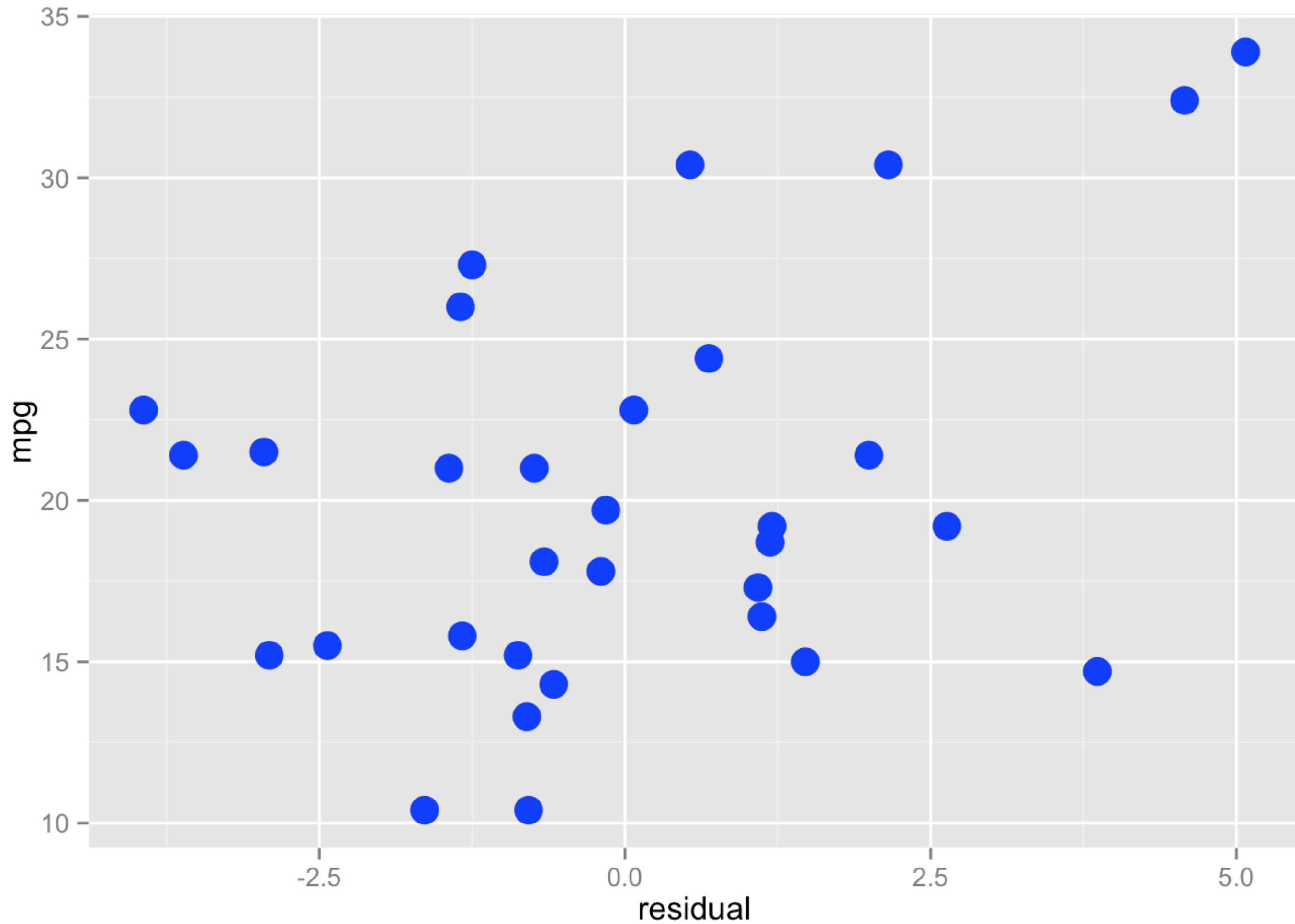
Conclusion

The expect MPG for cars with automatic transmission is 33.9. Having a manual transmision increases the MPG by 1.8 MPG.

Graph of Residuals

If our model is good, we would expect our residual point graph to appear random. That is to say we should not find any sort of pattern. (Our model looks good)

```
r <- resid(model)
df <- data.frame(mtcars$mpg, r)
colnames(df) <- c("mpg", "residual")
g <- ggplot(data=df, aes(y=mpg, x=residual))
g <- g + geom_point(size=5, colour="blue")
g
```


35
30
25
20
15
10
mpg
-2.5
0.0
2.5
5.0
residual