Fuel Economy of Automatic vs. Manual Transmission Cars (in 1974!)

Andrew Pratt

Last updated: 2017-09-23

This report is for the final project in the Regression Models course from the John Hopkins University Data Science Specialization on Coursera. I was tasked with using the `mtcars` dataset from the R `datasets` package to address the following question:

> Is an automatic or a manual transmission better for gas mileage?

In addition, I was tasked with quantifying the difference in miles per gallon (mpg) between automatic transmission cars and manual transmission cars.

Start by taking a peek at the data.

```
# Load the data and take a quick peek at it
data(mtcars)
head(mtcars)
```

```
##                    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
```

```
# Take a look at its structure. All of the variable
# are numeric. This will have to change later.
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 ...
```

```
sum(!complete.cases(mtcars)) # Make sure there are no missing values
```

```
## [1] 0
```

The variables in the `mtcars` dataset are as follows:
- `mpg` : miles per gallon
- `cyl` : number of cylinders in the engine
- `disp` : engine displacement volume in cubic inches
- `hp` : horsepower
- `drat` : rear axle ratio
- `wt` : car weight (1000 lbs.)
- `qsec` : quarter mile time
- `vs` : engine type (v = V-engine, s = straight engine)
- `am` : type of transmission (0 = automatic, 1 = manual)
- `gear` : number of forward gears
- `carb` : number of carburetors

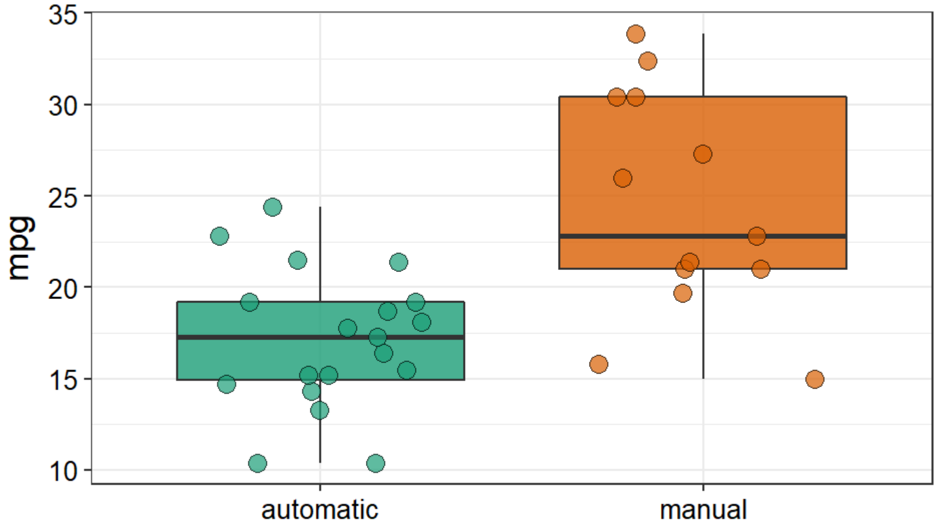
To learn more about the dataset, you can type `?mtcars` in RStudio.

Now that I'm ready to start, load up the libraries I'll be using and change a couple of the variable to factor variable instead of numeric variables.

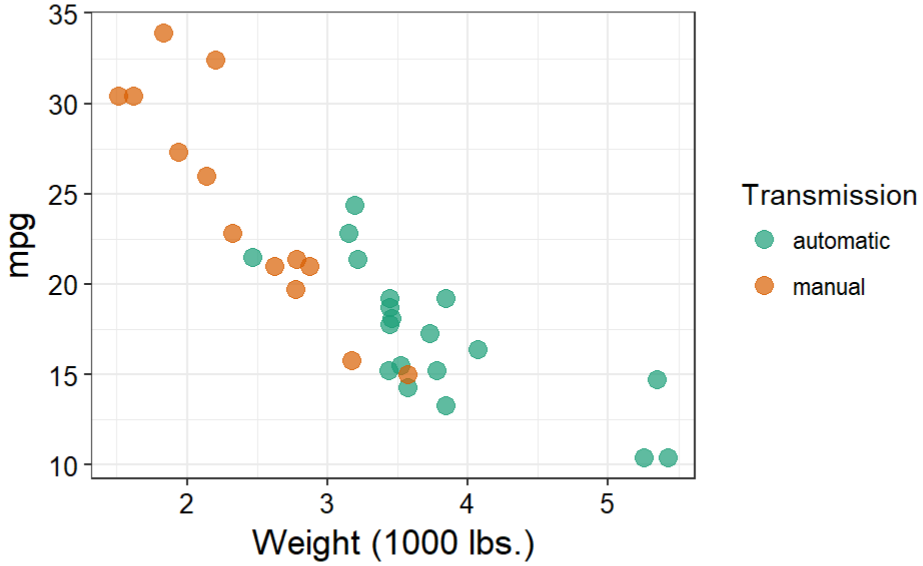
```
# Load up libraries
library(ggplot2)
library(dplyr)
library(tidyr)

# Make transmission (am) and engine type (vs)
# factor variables instead of numeric
mtcars <- mtcars %>% mutate(am = factor(am, levels = c("0", "1"), labels = c("automatic", "manua
l")),
                            vs = factor(vs, c("0", "1"), c("V", "S")))
```

Now onto the analysis. The first thing I want to illustrate about the data is that it probably isn't sufficient to just directly compare the mpg of manual transmission cars to automatic transmission cars. If I did that, it would look something like this:



Just by looking at this, you might think that manual transmission cars definitely get better gas mileage than automatic transmission cars! Let me quickly show you another plot that might make you a little less certain of that:

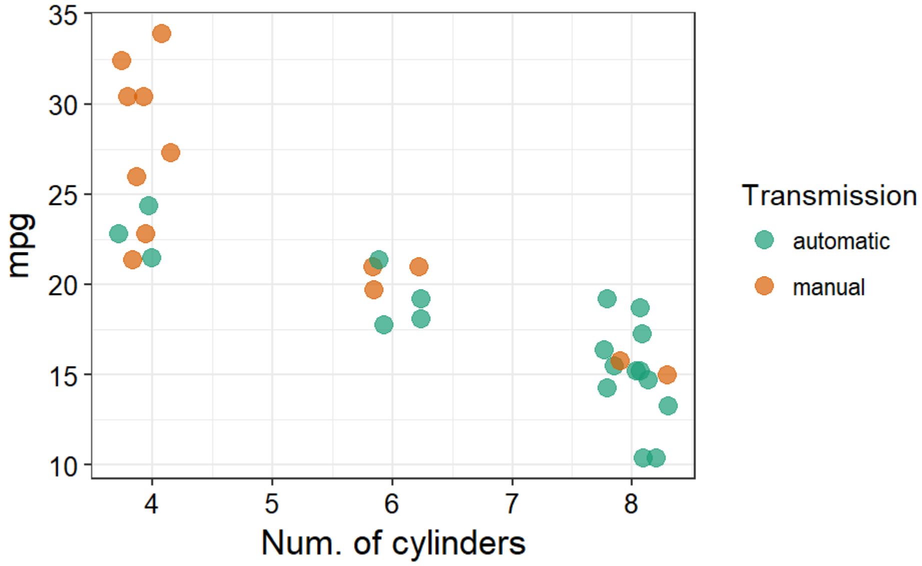


Notice how, in this plot, you see three things:
1. Heavier cars get fewer miles to the gallon.
2. Manual transmission cars are lighter than automatic transmission cars.
3. There’s not a lot of overlap between the two groups, but where there are cases of automatic and manual cars having similar weights, they seem to get similar gas mileage.

This made me immediately wonder if the difference in average gas mileage between manual and automatic cars is mostly due to other variables. So I asked myself, what other variables might possibly affect the fuel efficiency of a car, and do manual and automatic cars tend to show differences in those variables as well? As it turns out, there are a few such variable in the dataset, and I’ll go through a couple of them here before I talk about the regression model I used at the end of the report.

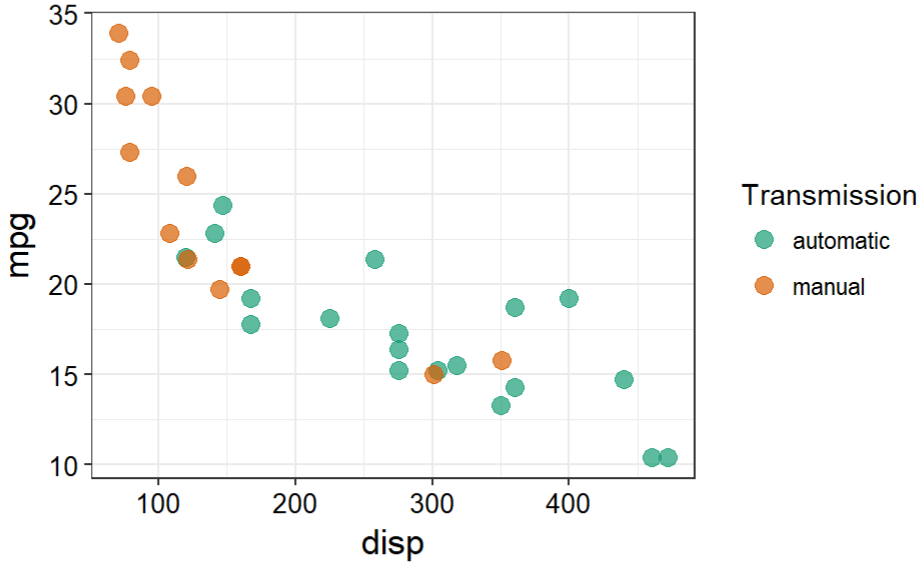
Number of engine cylinders

My guess is that adding more cylinders adds a little bit of weight to the car and therefore decreases the fuel efficiency of the car. The data shows that there is, indeed, a trend of fewer mpg at higher cylinder numbers. Furthermore, most of the cars with four cylinders have a manual transmission and most of the cars with eight cylinders have an automatic transmission. This could definitely affect any conclusion I make about the difference of fuel economy in manual vs. automatic transmission cars!



Engine displacement volume

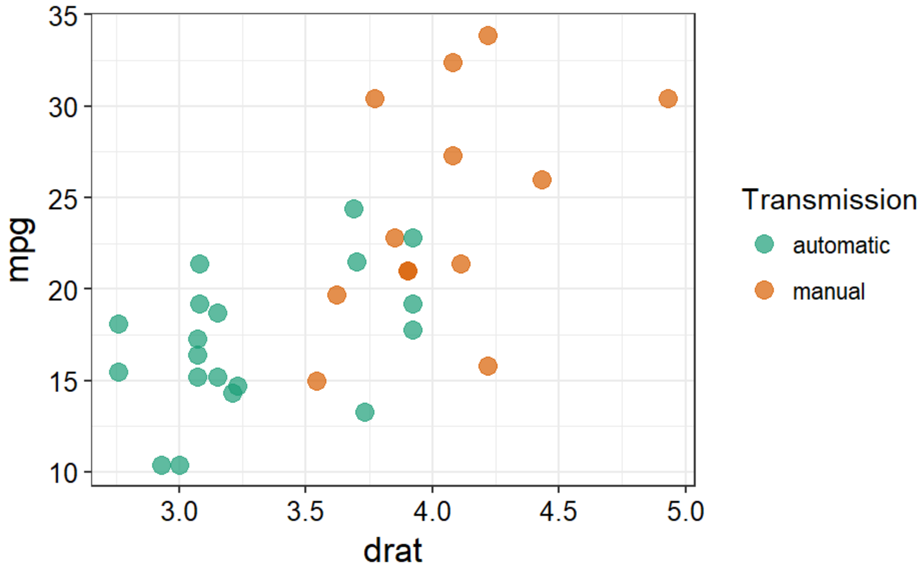
If my memory serves me correctly, cars with bigger, more powerful engines tend to get fewer miles per gallon. Based off of this, my guess is that cars with larger displacement volumes will tend to be less fuel efficient. The data supports my guess. And again, you can see that most of the cars with small engine displacement volumes have manual transmission whereas most of the cars with higher engine displacement volumes have automatic transmissions.



A quick note: It’s also a good idea to think about how some variables might be measuring the same thing, like engine displacement and number of cylinders. If you add more cylinders into an engine, then you can displace more fuel. As a result, it may be that engine displacement is largely dependent on the number of cylinders in the car's engine. This is important when considering which variables to include in a regression model!

Rear axle ratio

I'll show one last interesting example before briefly summarizing a few other variables and moving on to the regression model selection process. I did a little bit of reading and the rear axle ratio is often talked about in the context of pickup trucks. What it means is this: it is the ratio of the number of teeth on the rear axle gear to the number of teeth on the pinion gear of the driveshaft, which is turned by the transmission. When the pinion gear is turned by the transmission, the axle gear is subsequently turned by the pinion gear, causing the rear wheels to turn. So, if a car has a ratio of 4.11, for example, then it would take 4.11 revolutions of the pinion gear to turn the axle gear once. For trucks, the significance is this: the higher this ratio is, the more pulling power the truck has but the lower the top-speed of the car becomes. And the inverse is also true. What I expected to see in the `mtcars` is that cars with higher ratios get fewer miles per gallon because it takes more energy to turn the rear axle gear one revolution. Instead, here's what the data showed:



Interestingly, I saw exactly the opposite of what I expected to see. Why is this? Well, the short answer is I don't know. Everything I read about axle ratios (which wasn't that much, I admit) suggested that I should see the opposite trend. It could be that another variable is closely related to the axle ratio and is confounding the trend. At any rate, however, you can again see that manual transmission cars in the `mtcars` dataset tend to have higher rear axle ratios than automatic transmission cars, and this could potentially affect comparisons between the fuel efficiencies of manual vs. automatic cars.

Here's a brief summary of some of the other variables in the `mtcars` dataset:
1. More horsepower corresponds to worse fuel efficiency
2. More gears mayyyybe corresponds to slightly better fuel effiency
3. More carburetors corresponds to worse fuel efficiency

Now I'll show how I selected a regression model to see how gas mileage compares between the two types of transmission.

Model Selection

What I decided to do was to first select out potentially important variables from the `mtcars` dataset and build a series of regression models with those variables. I started by including all of the following variables: `mpg` (the response variable), `cyl` , `disp` , `hp` , `drat` , `wt` , `am` , `gear` , `carb` (the predictor variables). After fitting a linear model that included all of these predictor variables, I determined the variable which had the highest p-value and removed it from the following regression. The only variable that I did not eliminate from the regression was `am` , the variable which indicates whether the car had automatic or manual transmission. I systematically eliminated one variable in this manner until the only variable left was `am` . I then ran an analysis of variance comparing the smallest model (with only `am` as a predictor) to each of the bigger models and selected the last model where the addition of a new predictor significantly decreased the residual error of the model. As you will see, this model turned out to be the following:

`lm(mpg ~ am + cyl + wt, data = mtcars)`

Here’s the code:

```
# Generate sequential models
model.data <- list()
models <- list()
model.summaries <- list()
model.data[[1]] <- mtcars %>% select(mpg, cyl, disp, hp, drat, wt, am, gear, carb)
models[[1]] <- lm(mpg ~ ., model.data[[1]])
model.summaries[[1]] <- summary(models[[1]])
for(i in 1:(length(model.data[[1]])-2)){
        parms <- coef(model.summaries[[i]])
        amindex <- which(rownames(parms) == "ammanual")
        parms <- parms[-amindex,]
        var <- names(which.max(parms[,4]))
        varindex <- which(names(model.data[[i]]) == var)
        model.data[[i+1]] <- model.data[[i]][,-varindex]
        models[[i+1]] <- lm(mpg ~ ., model.data[[i+1]])
        model.summaries[[i+1]] <- summary(models[[i+1]])
}

# Analysis of variance
anova(models[[8]], models[[7]], models[[6]],
      models[[5]], models[[4]], models[[3]],
      models[[2]], models[[1]])
```

```
## Analysis of Variance Table
##
## Model 1: mpg ~ am
## Model 2: mpg ~ cyl + am
## Model 3: mpg ~ cyl + wt + am
## Model 4: mpg ~ cyl + wt + am + carb
## Model 5: mpg ~ cyl + hp + wt + am + carb
## Model 6: mpg ~ cyl + hp + drat + wt + am + carb
## Model 7: mpg ~ cyl + hp + drat + wt + am + gear + carb
## Model 8: mpg ~ cyl + disp + hp + drat + wt + am + gear + carb
##   Res.Df    RSS Df Sum of Sq       F    Pr(>F)
## 1     30 720.90
## 2     29 271.36  1    449.53 65.0632 3.723e-08 ***
## 3     28 191.05  1     80.32 11.6244  0.002403 **
## 4     27 168.71  1     22.34  3.2336  0.085283 .
## 5     26 162.72  1      5.99  0.8667  0.361529
## 6     25 160.70  1      2.02  0.2925  0.593839
## 7     24 159.61  1      1.09  0.1576  0.695024
## 8     23 158.91  1      0.70  0.1007  0.753822
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
# Pick the final model
final.model <- models[[6]]
summary(final.model)
```

```
##
## Call:
## lm(formula = mpg ~ ., data = model.data[[i + 1]])
##
## 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 ***
## cyl          -1.5102     0.4223  -3.576  0.00129 **
## wt           -3.1251     0.9109  -3.431  0.00189 **
## ammanual      0.1765     1.3045   0.135  0.89334
## ---
## 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
```

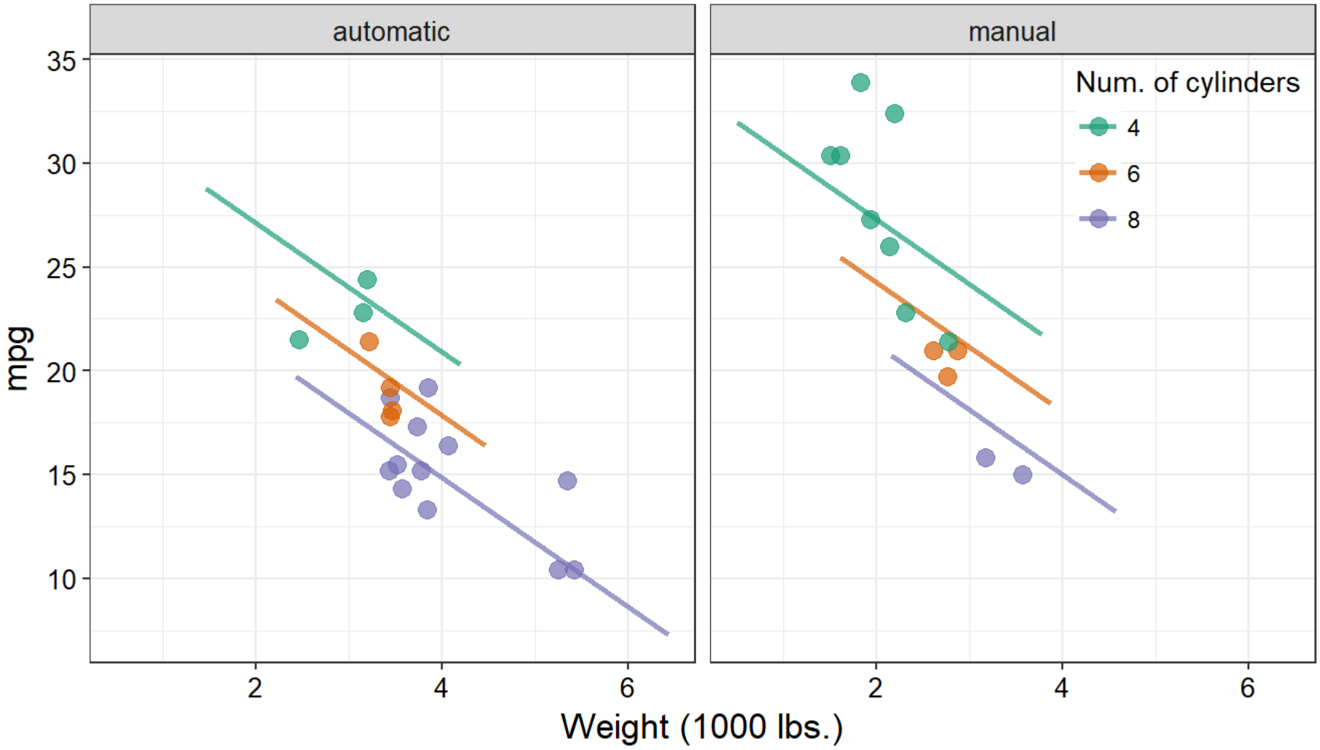
According to this model, when all other factors are held constant, the gas mileage of a car decreases by about 1.5 mpg per additional cylinder you put in its engine and it decreases by about 3.1 mpg for every 1000 pounds of weight added to the car. On the other hand, the difference in gas mileage of manual transmissions and automatic transmission, when all other factors are held constant, is estimated to be somewhere between -2.5 and 2.8.

```
confint(final.model)[4,]
```

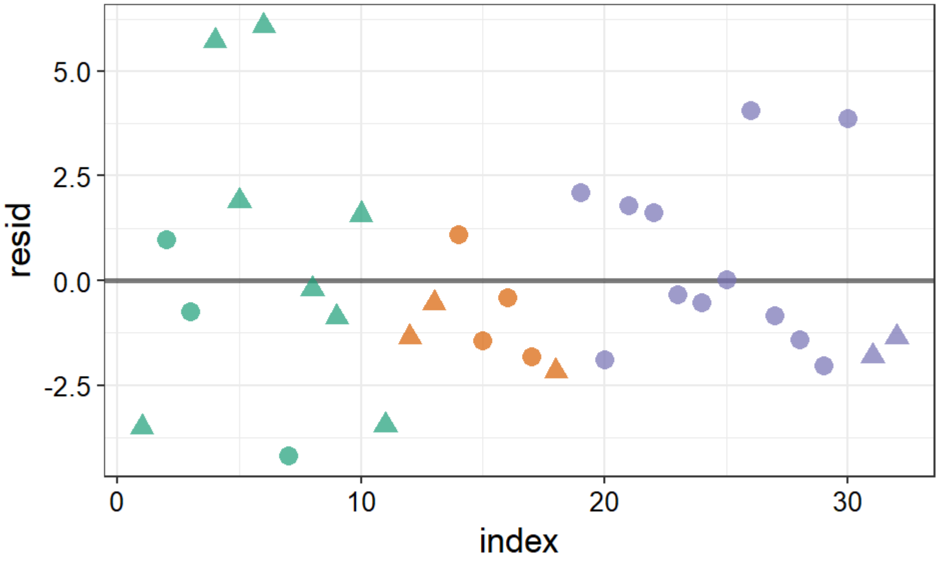
```
##     2.5 %    97.5 %
## -2.495555  2.848541
```

In other words, the model isn't sure if manual transmission cars get worse gas mileage than automatic cars or if they get better gas mileage than automatic cars!

Before I officially accept this model, though, I want to plot it and see how it looks.

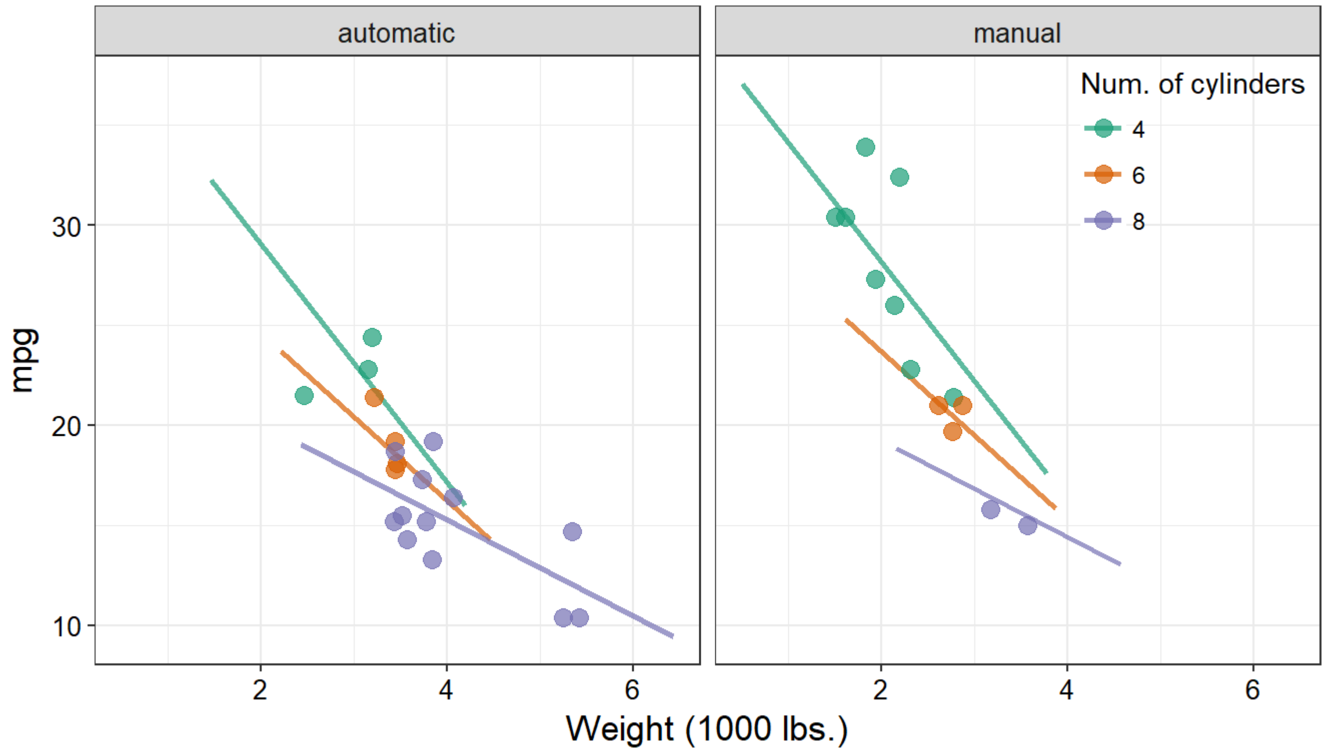


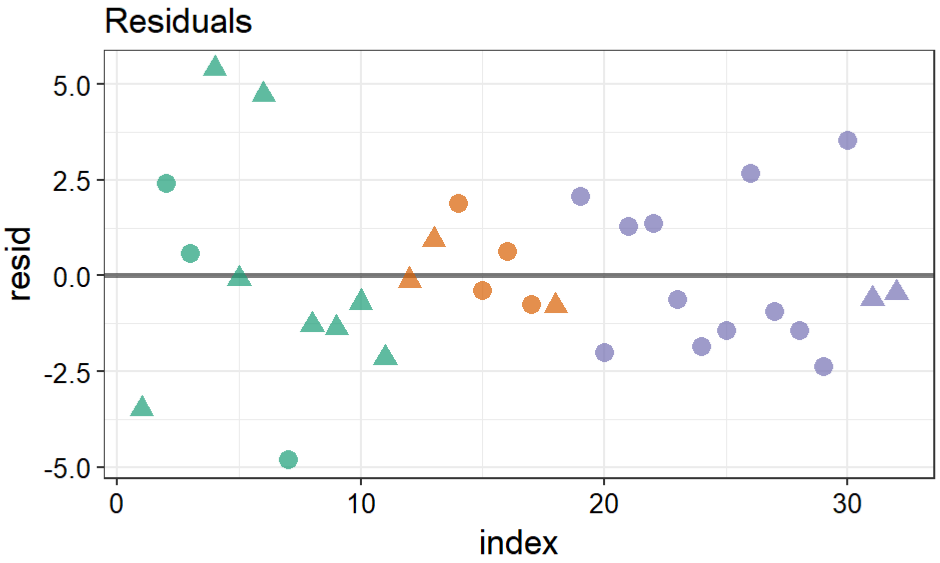
Hmm. Well it doesn't look great. Because I assumed there were no interactions between variables, the slope of the predicted regression lines are all the same and don't look very reasonable. Look at the residuals:



Notice how most of the orange points (meaning six cylinders) are one one side of the line zero? Now look at the following model and its residuals:

```
## 
## Call:
## lm(formula = mpg ~ am + cyl * wt, data = mtcars)
## 
## Residuals:
##     Min      1Q  Median      3Q     Max 
## -4.7887 -1.3812 -0.5268  1.3155  5.3945 
## 
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)    
## (Intercept)  57.0173     7.3507   7.757 2.43e-08 ***
## ammanual     -0.8619     1.2622  -0.683 0.500524    
## cyl          -4.0111     1.0594  -3.786 0.000777 ***
## wt           -9.5003     2.6492  -3.586 0.001308 ** 
## cyl:wt        0.8858     0.3494   2.535 0.017337 *  
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## Residual standard error: 2.391 on 27 degrees of freedom
## Multiple R-squared:  0.863,  Adjusted R-squared:  0.8427 
## F-statistic: 42.51 on 4 and 27 DF,  p-value: 2.815e-11
```





That looks better. Now here’s a summary of this new final model:

```
summary(lm(mpg ~ am + cyl*wt, mtcars))
```

```
## 
## Call:
## lm(formula = mpg ~ am + cyl * wt, data = mtcars)
## 
## Residuals:
##     Min      1Q  Median      3Q     Max 
## -4.7887 -1.3812 -0.5268  1.3155  5.3945 
## 
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)    
## (Intercept)  57.0173     7.3507   7.757 2.43e-08 ***
## ammanual     -0.8619     1.2622  -0.683 0.500524    
## cyl          -4.0111     1.0594  -3.786 0.000777 ***
## wt           -9.5003     2.6492  -3.586 0.001308 ** 
## cyl:wt        0.8858     0.3494   2.535 0.017337 *  
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## Residual standard error: 2.391 on 27 degrees of freedom
## Multiple R-squared:  0.863,  Adjusted R-squared:  0.8427 
## F-statistic: 42.51 on 4 and 27 DF,  p-value: 2.815e-11
```

Again, we see that true effect of the transmission appears to be small, if it even exists at all. The model estimates that, holding all other factors equal, manual transmission cars can at the least get about 3.5 FEWER miles per gallon than automatic transmission cars and at the most get about 1.7 MORE miles per gallon than automatic transmission cars.

```
confint(lm(mpg ~ am + cyl*wt, mtcars))[2,]
```

```
##      2.5 %    97.5 %
## -3.451589  1.727878
```

So there there you have it! I hope my boss at Motor Trends magazine is happy.

Code Appendix

```
# Make some theme elements to make the plots look nice and
# easy to read:

# Make custom palettes just because I feel like seeing different colors.
color.palette <- scale_color_brewer(palette = "Dark2", guide = FALSE)
fill.palette <- scale_fill_brewer(palette = "Dark2", guide = FALSE)

# plot.theme creates a simple theme that's easy to look at
plot.theme <- theme_bw() +
        theme(axis.title = element_text(size = 13),
              axis.text = element_text(size = 10, color = "black"),
              strip.text = element_text(size = 10))

# legend.top.right puts the legend in the top right corner of the plot
legend.top.right <- theme(legend.justification = c(1,1),
                          legend.position = c(1,1),
                          legend.background = element_rect(fill = "transparent"))



# PLOT 1: boxplot of mpg vs. am
ggplot(mtcars, aes(am, mpg)) +
        geom_boxplot(aes(fill = am), alpha = .8) +
        geom_jitter(aes(fill = am), shape = 21, size = 3, height = 0, width = .3, alpha = .7) +
        fill.palette +
        labs(x = "") +
        plot.theme

# PLOT 2: mpg vs. wt
ggplot(mtcars, aes(wt, mpg)) +
        geom_point(aes(color = am), size = 3, alpha = .7) +
        labs(x = "Weight (1000 lbs.)") +
        scale_color_brewer(palette = "Dark2", name = "Transmission") +
        plot.theme

# PLOT 3: mpg vs. cyl
ggplot(mtcars, aes(cyl, mpg)) +
        geom_jitter(aes(color = am), size = 3, alpha = .7, height = 0, width = .3) +
        labs(x = "Num. of cylinders") +
        scale_color_brewer(palette = "Dark2", name = "Transmission") +
        plot.theme

# PLOT 4: mpg vs. disp
ggplot(mtcars, aes(disp, mpg)) +
        geom_point(aes(color = am), size = 3, alpha = .7) +
        scale_color_brewer(palette = "Dark2", name = "Transmission") +
        plot.theme

# PLOT 5: mpg vs. drat
ggplot(mtcars, aes(drat, mpg)) +
        geom_point(aes(color = am), size = 3, alpha = .7) +
        scale_color_brewer(palette = "Dark2", name = "Transmission") +
        plot.theme
```

```r
# PLOT 6: mpg ~ am + cyl + wt
# Create new data containing predicted values from final.model
# to mtcars dataset that extend slightly beyond the range of the
# data so the lines don't look stupid.
pred <- mtcars %>% group_by(am, cyl) %>%
        mutate(lower = range(wt)[1]-1,
               upper = range(wt)[2]+1) %>%
        select(cyl, am, lower, upper) %>%
        gather("bound", "wt", lower:upper) %>%
        select(-bound) %>%
        ungroup
pred$pred <- predict(final.model, newdata = pred)

# Plot mpg vs. weight, colored by number of cylinders
# Lines are predicted values from final.model
ggplot(mtcars, aes(wt, mpg)) +
        geom_line(data = pred, aes(wt, pred, color = factor(cyl)), size = 1, alpha = .7) +
        geom_point(aes(color = factor(cyl)), size = 3, alpha = .7) +
        scale_color_brewer(palette = "Dark2", name = "Num. of cylinders") +
        facet_wrap(~ am) +
        labs(x = "Weight (1000 lbs.)",
             y = "mpg") +
        plot.theme +
        legend.top.right

# PLOT 7: residuals of first model
mtcars$resid <- resid(final.model)
mtcars <- mtcars %>% arrange(cyl) %>% mutate(index = 1:nrow(mtcars))
ggplot(mtcars, aes(index, resid)) +
        geom_hline(yintercept = 0, size = 1, alpha = .5) +
        geom_point(aes(color = factor(cyl), shape = am), size = 3, alpha = .7) +
        scale_shape_discrete(guide = FALSE) +
        color.palette +
        plot.theme

# PLOT 8: mpg ~ am + cyl*wt
# This time I'll fit a model that includes an interaction
# term between number of cylinders and car weight. The
# residuals look much better.
final.model2 <- lm(mpg ~ am + cyl*wt, mtcars)
summary(final.model2)
mtcars2 <- mtcars %>% mutate(resid = resid(final.model2)) %>%
        arrange(cyl) %>%
        mutate(index = 1:nrow(mtcars))

pred2 <- mtcars %>% group_by(am, cyl) %>%
        mutate(lower = range(wt)[1]-1,
               upper = range(wt)[2]+1) %>%
        select(cyl, am, lower, upper) %>%
        gather("bound", "wt", lower:upper) %>%
        select(-bound) %>%
        ungroup
pred2$pred <- predict(final.model2, newdata = pred2)
```

```
ggplot(mtcars2, aes(wt, mpg)) +
        geom_line(data = pred2, aes(wt, pred, color = factor(cyl)), size = 1, alpha = .7) +
        geom_point(aes(color = factor(cyl)), size = 3, alpha = .7) +
        scale_color_brewer(palette = "Dark2", name = "Num. of cylinders") +
        facet_wrap(~ am) +
        labs(x = "Weight (1000 lbs.)",
             y = "mpg") +
        plot.theme +
        legend.top.right

# PLOT 9: residuals from second model
ggplot(mtcars2, aes(index, resid)) +
        geom_hline(yintercept = 0, size = 1, alpha = .5) +
        geom_point(aes(color = factor(cyl), shape = am), size = 3, alpha = .7) +
        labs(title = "Residuals") +
        scale_shape_discrete(guide = FALSE) +
        color.palette +
        plot.theme
```