Weight Alone is Better Predictor of Car Mileage than Weight and Transmission Type

Phillip Rowe, 10/8/2019

Executive Summary

We compared two linear models to explain the change in miles per gallon of the mtcars data on other variables. In the first model, mpg depends only on the car weight. The second includes the transmission type as a dummy variable, which results in a different intercept and slope from interaction terms. Both models can be used with 95% confidence and pass ANOVA tests for their included regressors. However, because of the lack of datapoints for very light automatic cars and very heavy manual cars, we cannot be sure that the second model does not overfit the data. Intuitively, the second model does appear nonsensical or at least not very useful. In particular, though the second model would predict that manual cars have better mileage when their weight is less than the crossover point of the fitted lines (at slightly above 2,800 lbs.), it would also predict that a manual car of 5000 pounds would have a mileage of less than 1 mpg (which seems mechanically pessimistic/unlikely). Moreover, the second model does not differentiate between most of the datapoints, as only 2 very light manual cars fall outside the prediction intervals of the auto fit, and only 3 very heavy cars with automatic transmission fall outside the prediction intervals of the manual fit. In other words, the model fails to reliably quantify the mileage advantage of a car with a given weight and transmission type. We decided not to try to fit more complex, nonlinear models. **Thus, we conclude that the first, simpler model is adequate (37.3 mpg intercept, slope of -5.34 mpg per 1000 lb increase in weight), and that the transmission type does not have a statistically significant impact on the mileage.**

Exploratory Data Analysis

We calculated the correlations between all the variables and the mileage variable (mpg) and found that weight (wt) has the largest (negative) correlation, of -0.868 (see first code block in Appendix). Other large magnitude correlations such as displacement, number of cylinders, and horsepower are intuitively related to weight (i.e., a larger or more powerful engine would result in greater weight overall), so we decline to examine these variables for the sake of simplicity. In Figure 1 and 2, we can see the scatterplot of mileage vs. the car's weight and transmission type, and the fitted linear model and its prediction intervals.

We note that we did do a boxplot of mileage by transmission type (not shown here due to space constraints), and that a t-test of the mean mileage did reject the null hypothesis that the means were equal. However, such analysis is obviously too simplistic given the other variables.

Models and ANOVA tests

The ANOVA test comparison between mpg_wt and mpg_wt_am2 shows that adding the am and am*wt terms to the linear model are statistically significant. However, the t-values for the intercept and slope coefficient are orders of magnitude more significant for the mpg_wt model than the mpg_wt_am2 model. The mpg intercept for mpg_wt is 37.3 mpg and the slope coefficient is -5.34 mpg per 1000 pound increase in car weight. For the mpg_wt_am2 model, the intercept and slope for automatic cars (am=0) is 31.4 and -3.79 mpg per 1,000 pound increase. For manual cars, the intercept is 46.3 mpg and the slope is -9.1 mpg/1,000 pounds.

Residuals

We plot the residuals of mpg_wt (see Appendix) and calculate the model's dfbetas and hatvalues, finding that the most extreme values are not orders of magnitude larger that the others (e.g., the most negative dfbeta is the Toyota Corolla at -0.636 vs. Chrysler Imperial being the largest at 1.006, while the range of hatvalues is 0.031 to 0.195). In other words, no single weight datapoint changes the individual coefficients much more than the others when excluded from the dataset (dfbetas). No single datapoint has much more leverage than the others (leverage is measured by hatvalues). We also plot the residuals of both models vs. their predictions and see no obvious pattern.

Appendix (Charts and code only)

Exploratory Data Analysis

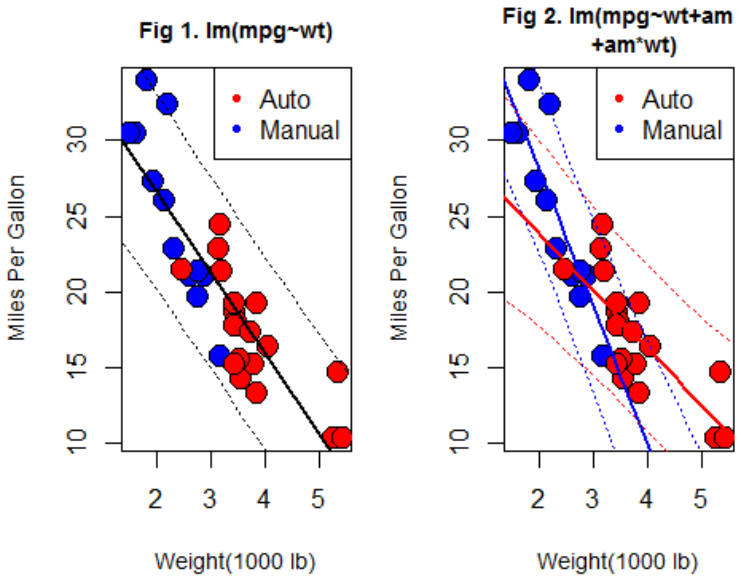
```
sort(round(cor(mtcars)[-1,1],3)) # correlation of mpg with all other vars
```

```
##     wt    cyl   disp     hp   carb   qsec   gear     am     vs   drat
## -0.868 -0.852 -0.848 -0.776 -0.551  0.419  0.480  0.600  0.664  0.681
```

Models

```
par(mfrow=c(1,2),mai = c(1, 0.9, 0.5, .1))

plot(mtcars$wt,mtcars$mpg,main="Fig 1. lm(mpg~wt)",cex.lab=0.9,cex.main=.9,
     xlab="Weight(1000 lb)", ylab="Miles Per Gallon",)
legend('topright', col=c('red','blue'), pch=20,
legend=c('Auto','Manual'),box.lty=1,cex=1)

man<-mtcars[mtcars$am==1,]
auto<-mtcars[mtcars$am==0,]
points(man$wt,man$mpg, pch = 21, col = "black", bg = "blue", cex = 2)
points(auto$wt,auto$mpg, pch = 21, col = "black", bg = "red", cex = 2)

# First, simpler model
mpg_wt<-lm(mpg~wt,mtcars)
abline(mpg_wt, lwd=2)

# Second model including interaction term
mpg_wt_am2<-lm(mpg~wt+am+am*wt,mtcars)

# prediction intervals of first model
xVals<- seq(min(mtcars$wt)-.2, max(mtcars$wt)+.2, by = .2)
newdata<-data.frame(wt=xVals)
p2 <- predict(mpg_wt, newdata=newdata, interval = ("prediction"))
lines(xVals, p2[,2],lty=3); lines(xVals, p2[,3],lty=3)

# plot of second model's prediction intervals
plot(mtcars$wt,mtcars$mpg,main="Fig 2. lm(mpg~wt+am
     +am*wt)", xlab="Weight(1000 lb)", ylab="Miles Per Gallon",cex.lab=0.9,cex.main=.9)
legend('topright', col=c('red','blue'), pch=20,
legend=c('Auto','Manual'),box.lty=1,cex=1)

points(man$wt,man$mpg, pch = 21, col = "black", bg = "blue", cex = 2)
points(auto$wt,auto$mpg, pch = 21, col = "black", bg = "red", cex = 2)

newdata2<-newdata
```

```
newdata2$am<-1 # setting am to 'manual' so we can draw prediction intervals
manpred<-predict(mpg_wt_am2,newdata=newdata2,interval=("prediction"))
newdata2$am<-0  # setting am to 'automatic' so we can draw prediction intervals
autopred<-predict(mpg_wt_am2,newdata=newdata2,interval=("prediction"))
lines(xVals,manpred[,1],lwd=2,col='blue')
lines(xVals,manpred[,2],lty=3,col='blue') ; lines(xVals,manpred[,3],lty=3,col='blue')
lines(xVals,autopred[,1],lwd=2,col='red')
lines(xVals,autopred[,2],lty=3,col='red') ; lines(xVals,autopred[,3],lty=3,col='red')
```



ANOVA Tests and Model Coefficients

```
anova(mpg_wt,mpg_wt_am2)

## Analysis of Variance Table
##
## Model 1: mpg ~ wt
## Model 2: mpg ~ wt + am + am * wt
##   Res.Df    RSS Df Sum of Sq      F   Pr(>F)
## 1     30 278.32
## 2     28 188.01  2    90.314 6.7253 0.004119 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

summary(mpg_wt);

##
## Call:
## lm(formula = mpg ~ wt, data = mtcars)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -4.5432 -2.3647 -0.1252  1.4096  6.8727
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  37.2851     1.8776  19.858  < 2e-16 ***
## wt           -5.3445     0.5591  -9.559 1.29e-10 ***
## ---
```

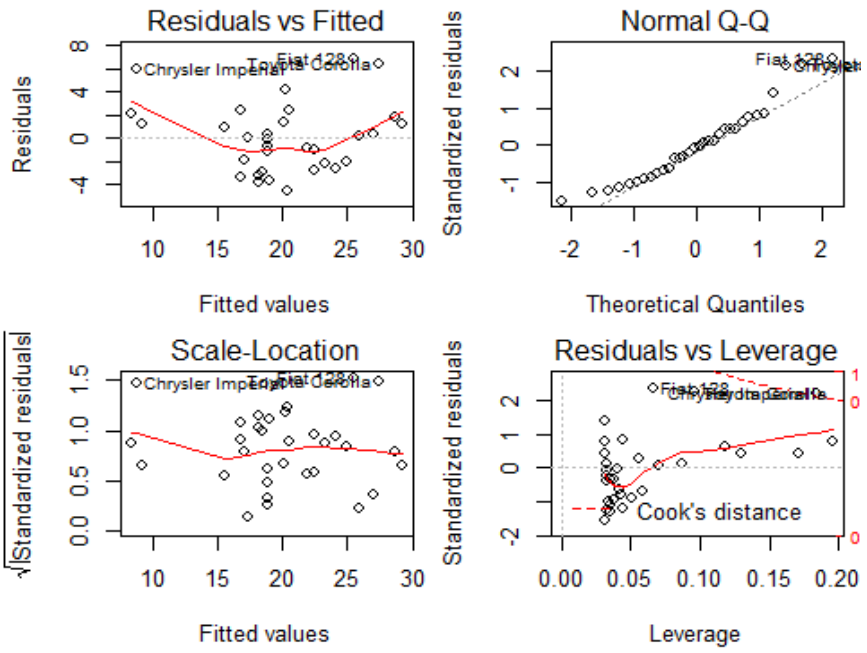
```
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.046 on 30 degrees of freedom
## Multiple R-squared:  0.7528, Adjusted R-squared:  0.7446
## F-statistic: 91.38 on 1 and 30 DF,  p-value: 1.294e-10
```

```
summary(mpg_wt_am2);
```

```
##
## Call:
## lm(formula = mpg ~ wt + am + am * wt, data = mtcars)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -3.6004 -1.5446 -0.5325  0.9012  6.0909
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  31.4161     3.0201  10.402 4.00e-11 ***
## wt           -3.7859     0.7856  -4.819 4.55e-05 ***
## am           14.8784     4.2640   3.489  0.00162 **
## wt:am        -5.2984     1.4447  -3.667  0.00102 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.591 on 28 degrees of freedom
## Multiple R-squared:  0.833,  Adjusted R-squared:  0.8151
## F-statistic: 46.57 on 3 and 28 DF,  p-value: 5.209e-11
```

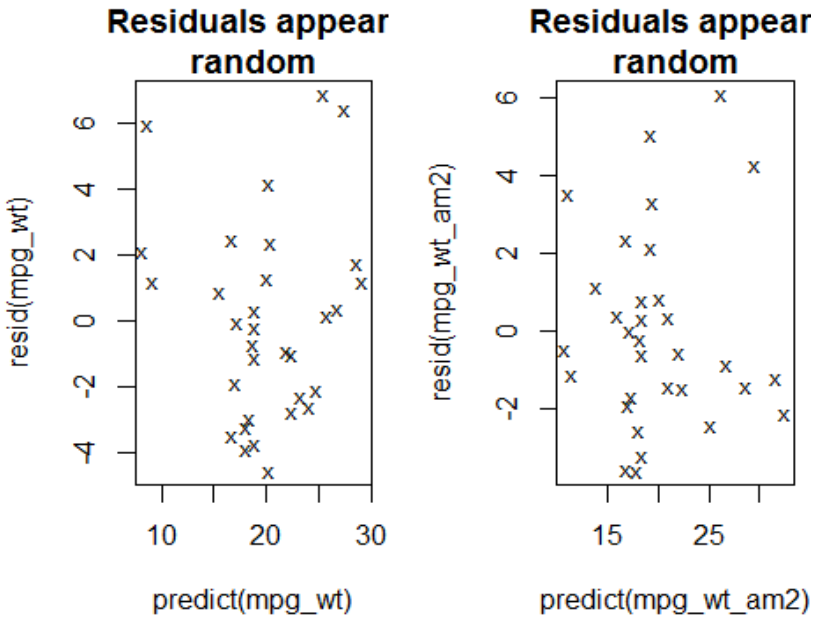
Residuals and Model Fit without Outliers

```
par(mfrow=c(2,2),mai = c(.7, 0.7, 0.3, 0.1))
plot(mpg_wt)
```




```
par(mfrow=c(1,2),mai = c(1,1,.5,.1))
plot(predict(mpg_wt),resid(mpg_wt),pch='x', main='Residuals appear \nrandom',cex=.8)
```

```
plot(predict(mpg_wt_am2),resid(mpg_wt_am2),pch='x',
     main='Residuals appear \nrandom',cex=.8)
```




```
# showing most extreme dfbetas and hatvalues to see which cars
sort(round(dfbetas(mpg_wt)[, 2], 3))[c(1,2,3,31,32)]
```

```
##      Toyota Corolla            Fiat 128         Honda Civic
##              -0.636              -0.490              -0.189
## Lincoln Continental   Chrysler Imperial
##               0.345               1.006
```

```
sort(round(hatvalues(mpg_wt), 3))[c(1,31,32)]
```

```
##      Hornet 4 Drive   Chrysler Imperial Lincoln Continental
##               0.031               0.184               0.195
```