Influence of car transmission type on mpg

Ricardo Leon

4/26/2020

Summary

Ecology and efficiency are important aspects in this times when talking about car manufacturing. With Motor Trend available data, we will measure the influence of car transmission types on mpg. After the analysis, it is concluded that for cars with weight greater than 3400lb automatic transmission is the best option for mpg, otherwise (less than 3400lb) manual transmission is the right choice.

Setup for analysis

As the first step, dplyr and ggplot2 libraries and mtcars dataset will be loaded.

```r
library(dplyr)
library(ggplot2)
data(mtcars)
```

Exploratory Analysis

Checking the data, it can be noticed that cyl, vs, am, gear and carb are in fact factor variables (see Appendix A). So lets transform that to corresponding type.

```r
tidy.data <- mtcars %>% mutate(cyl = as.factor(cyl),
                               vs = as.factor(vs),
                               am = as.factor(am),
                               gear = as.factor(gear),
                               carb = as.factor(carb))
```

The variable we are interested in is am (tramission type). In Appendix B, it is shown a plot with the mpg values for each transmission type.

Model Selection

The first model will fit mpg ~ am. Consecutively, it will add one predictor to check if the model can fit the data with the use of the anova function (numeric results shown in Appendix C).

```r
fit1 <- lm(mpg ~ am, data = tidy.data)
fit2 <- update(fit1, mpg ~ am + wt)
fit3 <- update(fit1, mpg ~ am + wt + qsec)
fit4 <- update(fit1, mpg ~ am + wt + qsec + cyl)
anova(fit1, fit2, fit3, fit4)
```

We have that wt and qsec worth to be included in the model considering a significance value of alpha = 0.05. As am is a factor value, we can test if including mixed predictors add value (am * wt & am * qsec).

```
fit4 <- update(fit1, mpg ~ am + wt + qsec + am * wt)
fit5 <- update(fit1, mpg ~ am + wt + qsec + am * wt + am * qsec)
anova(fit3, fit4, fit5)
```

am * wt worths to be included. looking at the p values for each coefficient, each of them is significant expect for the intercept (Appendix C). Removing the independent term, we get for am value 0 a coefficient to make easier the interpretation.

```
fit <- lm(mpg ~ am + wt + qsec + am * wt - 1, data = tidy.data)
modelCoef <- round(summary(fit)$coef, 3)
modelCoef
```

```
##         Estimate Std. Error t value Pr(>|t|)
## am0        9.723      5.899   1.648    0.111
## am1       23.802      6.077   3.917    0.001
## wt        -2.937      0.666  -4.409    0.000
## qsec       1.017      0.252   4.035    0.000
## am1:wt    -4.141      1.197  -3.460    0.002
```

Residual analysis

Checking the "Residual vs Fitted" and "Residuals vs Leverage" plots (Appendix D) obtained from the model, no patterns can be detected that contradict the model.

Inferences from model

From model coefficients, we have that automatic and manual cars contribute with 9.723 and 23.802 mpg respectively. So manual cars coefficient have more impact by itself for mpg. However we can see a greater negative impact in weight contribution to mpg when car is manual.

```
wt.interval.auto <- modelCoef[3,1] + c(-1,1) * qt(.975, df = fit$df) * modelCoef[3,2]

wt.manual.coef <- modelCoef[3,1] + modelCoef[5,1]
wt.manual.std <- abs(modelCoef[3,2]) + abs(modelCoef[5,2])
wt.interval.manual <- wt.manual.coef + c(-1,1) * qt(.975, df = fit$df) * wt.manual.std
```

With 95% confidence, we have a decrement for mpg each 1000lb, as follows:

For automatic transmission: (-4.3035191, -1.5704809).

For manual transmission: (-10.9005603, -3.2554397).

Manual cars adds 14.079 mpg over automatic ones but this gain is cancelled for cars over 3400 lbs.

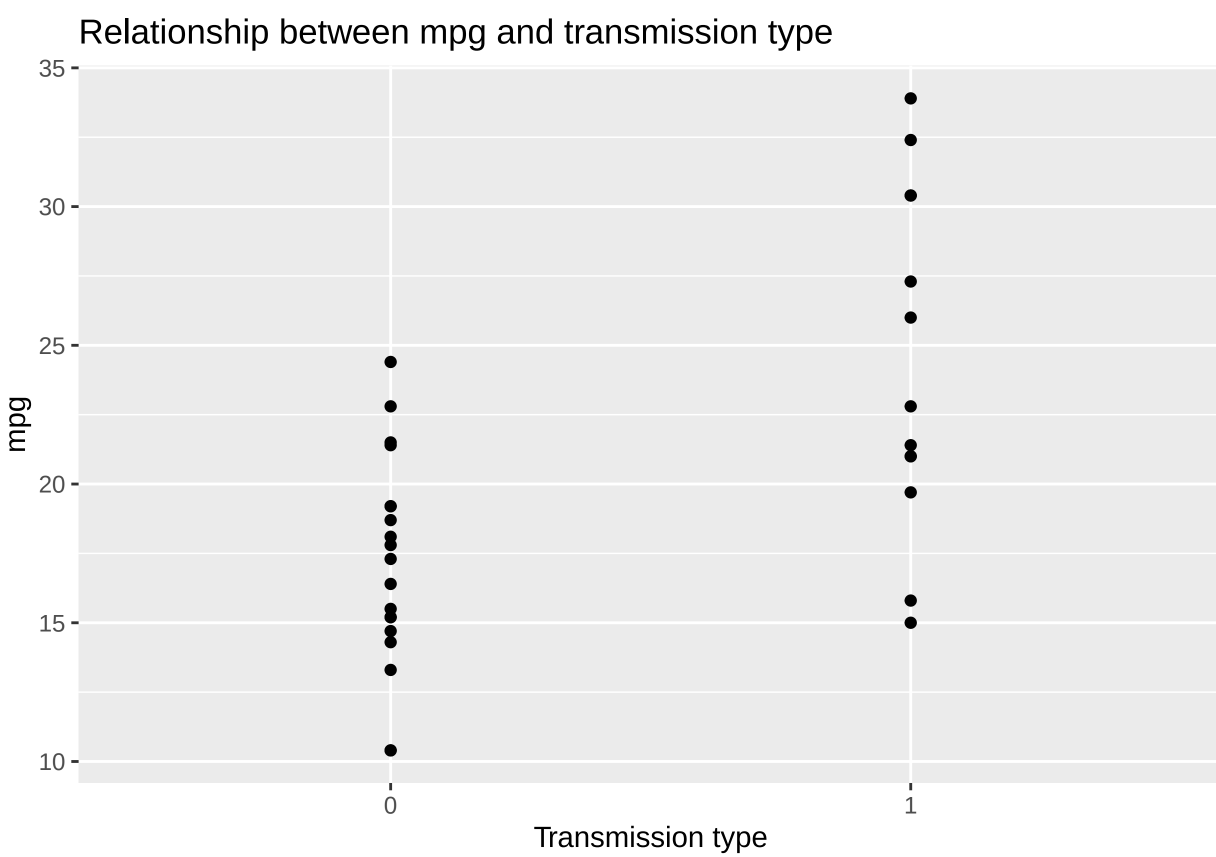
Appendix A. Data structure.

```
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 ...
```

Appendix B. Relationship between mpg and transmission type

```
ggplot(tidy.data, aes(y = mpg, x = am)) +
  geom_point() +
  labs(title="Relationship between mpg and transmission type",
       x ="Transmission type", y = "mpg")
```



Appendix C. Results of model comparison.

If we add any predictor to fit3, anova return a F score greater than our significant value 0.05.

```r
fit1 <- lm(mpg ~ am, data = tidy.data)
fit2 <- update(fit1, mpg ~ am + wt)
fit3 <- update(fit1, mpg ~ am + wt + qsec)
fit4 <- update(fit1, mpg ~ am + wt + qsec + cyl)
anova(fit1, fit2, fit3, fit4)
```

```
## Analysis of Variance Table
##
## Model 1: mpg ~ am
## Model 2: mpg ~ am + wt
## Model 3: mpg ~ am + wt + qsec
## Model 4: mpg ~ am + wt + qsec + cyl
##   Res.Df    RSS Df Sum of Sq       F    Pr(>F)
## 1     30 720.90
## 2     29 278.32  1    442.58 72.1784 5.625e-09 ***
## 3     28 169.29  1    109.03 17.7820  0.000265 ***
## 4     26 159.42  2      9.86  0.8041  0.458289
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
fit4 <- update(fit1, mpg ~ am + wt + qsec + am * wt)
fit5 <- update(fit1, mpg ~ am + wt + qsec + am * wt + am *qsec)
anova(fit3, fit4, fit5)
```

```
## Analysis of Variance Table
##
## Model 1: mpg ~ am + wt + qsec
## Model 2: mpg ~ am + wt + qsec + am:wt
## Model 3: mpg ~ am + wt + qsec + am:wt + am:qsec
##   Res.Df    RSS Df Sum of Sq       F   Pr(>F)
## 1     28 169.29
## 2     27 117.28  1    52.010 11.6099 0.002145 **
## 3     26 116.47  1     0.802  0.1791 0.675630
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
summary(fit4)$coef
```

```
##              Estimate Std. Error   t value     Pr(>|t|)
## (Intercept)  9.723053  5.8990407  1.648243 0.1108925394
## am1         14.079428  3.4352512  4.098515 0.0003408693
## wt          -2.936531  0.6660253 -4.409038 0.0001488947
## qsec         1.016974  0.2520152  4.035366 0.0004030165
## am1:wt      -4.141376  1.1968119 -3.460340 0.0018085763
```

Appendix D. Residual plots.

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