Linear Regression Assignment

Carlos Sanchez

Executive summary

`mtcars` dataset was extracted from the 1974 Motor Trend US magazine, and comprises fuel consumption and 10 aspects of automobile design and performance for 32 automobiles (1973–74 models).

The different variables the dataset has are:

- **mpg**: Miles/(US) gallon
- **cyl**: Number of cylinders
- **disp**: Displacement (cu.in.)
- **hp**: Gross horsepower
- **drat**: Rear axle ratio
- **wt**: Weight (1000 lbs)
- **qsec**: 1/4 mile time
- **vs**: Engine (0 = V-shaped, 1 = straight)
- **am**: Transmission (0 = automatic, 1 = manual)
- **gear**: Number of forward gears
- **carb**: Number of carburetors

If we analyze the head of the dataset and its dimensions, we see that `mtcars` has 32 rows and 11 variables.

```
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
```

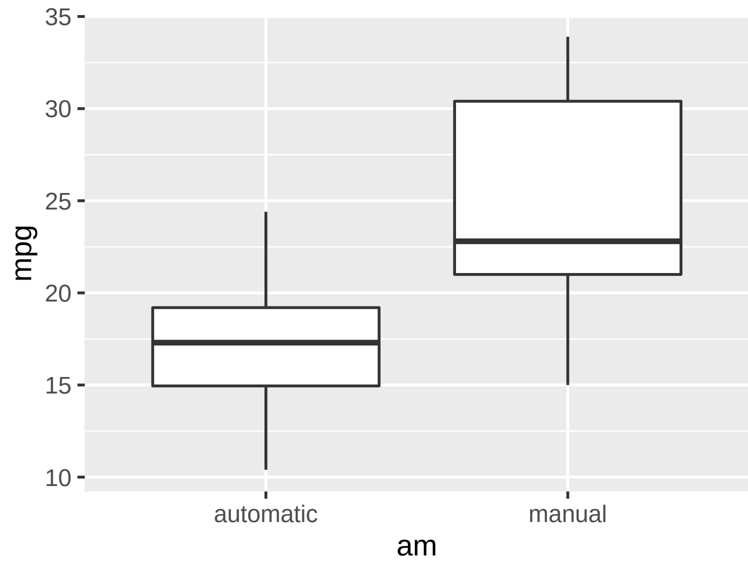
```
dim(mtcars)
```

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

Exploratory analysis

On a first step, we will compare the means of each group, Manual and Automatic transmissions, both graphically and numerically. One previous step to be done is to renamevariable `am` and make it a factor.

```
mtcars$am <- factor(mtcars$am, labels = c("automatic", "manual"))
ggplot(mtcars, aes(x=am, y=mpg))+
  geom_boxplot()
```




```
mtcars %>% group_by(am) %>% summarize(mean=mean(mpg))
```

```
## # A tibble: 2 x 2
##   am         mean
## * <fct>     <dbl>
## 1 automatic  17.1
## 2 manual     24.4
```

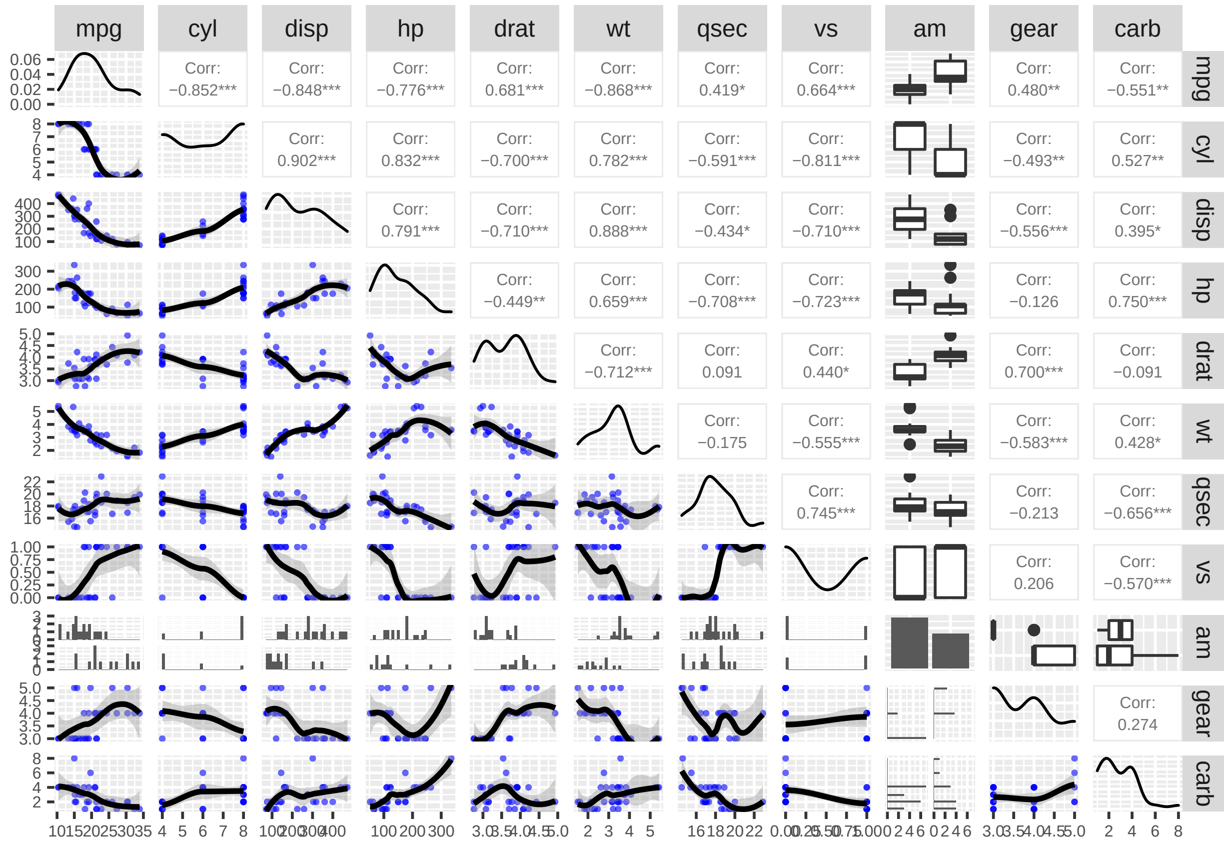
Model of mpg ~ am

```
fit <- lm(mpg ~ am, data = mtcars)
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
```

If we check the values of R-Squared for this model, we see that it's 0.36, explaining too little variance. If we plot the correlation between mpg with the rest of the variables, we can see that some of them have a high correlation which suggest that the final model should include more predictors, not only `am`.

```
ggpairs(mtcars,
        lower = list(continuous = wrap("smooth", size = 0.5, method = "loess", alpha = 0.6, color = "bl
        upper = list(continuous = wrap("cor", size = 2))) +
  theme(
  axis.text = element_text(size = 6),
  axis.title = element_text(size = 6))
```



Model mpg ~ all (include the maximum number of predictors necessary)

In order to select the minimal number of predictors that our model will use without compromising the result, we will use the stepwise regression method. For doing that, we will use the function `stepAIC` from the `MASS` package with the parameter **both** that will perform backward and forward stepwise mode.

```
final.lm <- lm(mpg ~., data = mtcars)
step <- stepAIC(final.lm, direction = "both", trace = FALSE)
step
```

```
##
## Call:
## lm(formula = mpg ~ wt + qsec + am, data = mtcars)
##
## Coefficients:
## (Intercept)           wt         qsec     ammanual
##       9.618       -3.917        1.226        2.936
```

As a result of the stepwise regression, we obtain that the best model for predicting MPG consumption includes **Weight (wt)**, **Acceleration (qsec)** and **Transmission type (am)**.

```
summary(step)
```

```
##
## Call:
## lm(formula = mpg ~ wt + qsec + am, data = mtcars)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -3.4811 -1.5555 -0.7257  1.4110  4.6610
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   9.6178     6.9596   1.382 0.177915
## wt           -3.9165     0.7112  -5.507 6.95e-06 ***
## qsec          1.2259     0.2887   4.247 0.000216 ***
## ammanual      2.9358     1.4109   2.081 0.046716 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.459 on 28 degrees of freedom
## Multiple R-squared:  0.8497, Adjusted R-squared:  0.8336
## F-statistic: 52.75 on 3 and 28 DF,  p-value: 1.21e-11
```

As we can observe, all 3 variables are significant since p-value are below 0.05, and the value of R-squared is `round(summary(step)$r.squared, 3)`, much more higher than the previous moldel.

If now we compare the final model with 3 predictors with the model only including the `am` predictor:

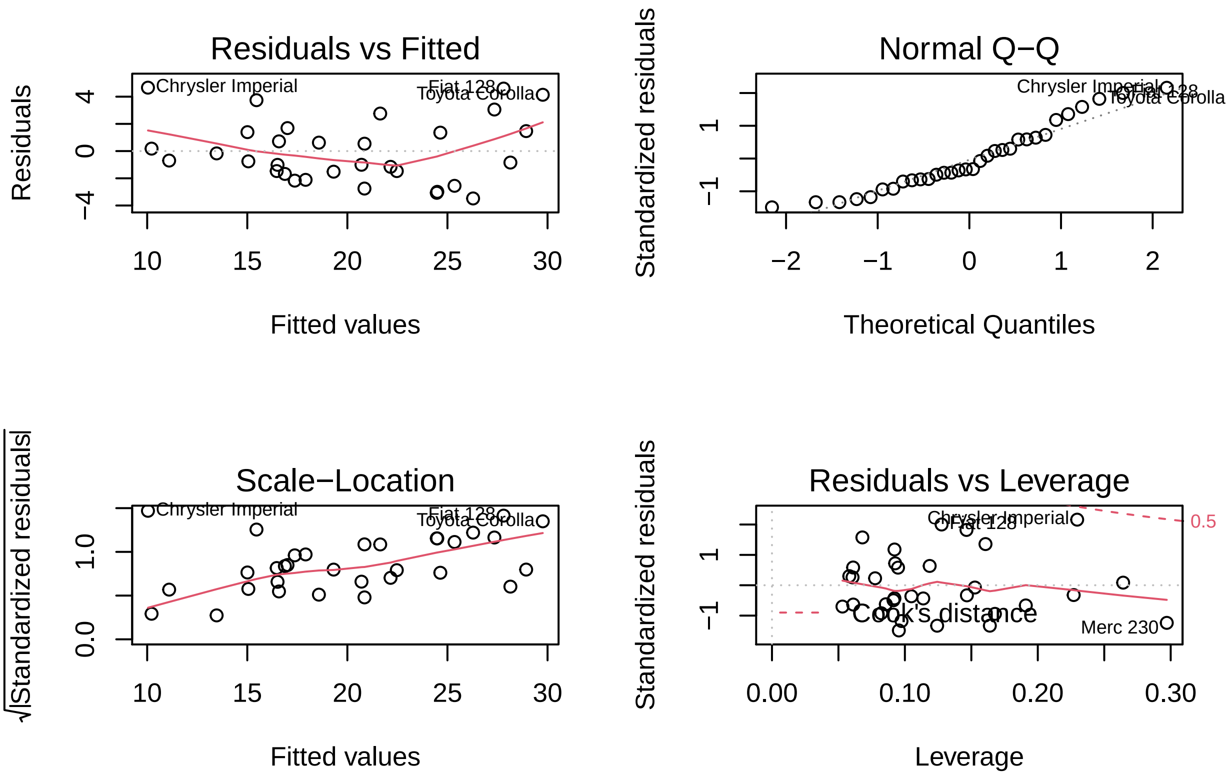
```
anova(fit, step)
```

```
## Analysis of Variance Table
##
## Model 1: mpg ~ am
## Model 2: mpg ~ wt + qsec + am
##   Res.Df    RSS Df Sum of Sq      F   Pr(>F)
## 1     30 720.90
## 2     28 169.29  2    551.61 45.618 1.55e-09 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

We obtain a p-value for the **anova** near to zero, so it fails to accept the null hypothesis of equal means, indicating that the new added predictors really affect the result in **MPG**.

If we run some residuals plots at the final model:

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



Conclusion

When only consider `am` as a predictor, we obtain that manual es 7.25 MPG better on fuel consumption and if we consider `qsec`, `wt` (best model) this value drops to 2.94 again for manual transmission.

We can observe that for each mille per gallon (MPG) on an automatic transmission, Manual has **2.9358372 MPG**.