Multiple Regression using mtcars dataset

Venkatesh Vedam

Summary

Multiple regression is an extension of linear regression. While there are several more powerful algorithms than multi regression available to the data scientist, this is often useful during exploratory analysis and also for establishing a baseline for the model accuracy. As compared to other advanced algorithms, linear/multiple regression have the advantage of being simple to explain.

In this exercise, we explore multiple regression using the mtcars dataset as we answer two questions:

1) Is an automatic or manual transmission better for MPG ?
2) How different (quantitatively) is the MPG difference between automatic and manual transmissions?

We start off with a layman approach of taking a simple mean of mpg by transmission type and found that manual transmission is better. We follow this up with a two sample, one tailed t-test which confirms that the difference between the two means is indeed statistically signficant, thus conclusively answering the first question above in favor of manual transmission.

Next we turn our attention to model building for answering the second question above. Our base model involves of modeling mpg with only am as the predictor, which explains 34% of the variation in the response variable. Subsequently we use the bestglm package to figure out the best subset of variables for the model, which turns out to be wt,qsec and am. Essentially we pick a combination of categorical and continuous variables as predictors.

The final model thus created could explain 83% of the variation in mpg. As expected wt (weight) is negatively correlated with mpg whereas qsec and am (manual option) yield positive coefficients.

Approach

Step 1

As the first step, we load the required libraries and examine the dataset

```
library(ggplot2)
library(dplyr)
library(GGally)
library(bestglm)
```

```
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 ...
```

The R help menu gives addnl description about each column:

[, 1] mpg Miles/(US) gallon [, 2] cyl Number of cylinders [, 3] disp Displacement (cu.in.) [, 4] hp Gross horsepower [, 5] drat Rear axle ratio [, 6] wt Weight (1000 lbs) [, 7] qsec 1/4 mile time [, 8] vs V/S [, 9] am Transmission (0 = automatic, 1 = manual) [,10] gear Number of forward gears [,11] carb Number of carburetors

We note that the categorical columns are also stored as numbers. It's a good idea to convert these to factors before moving forward:

```
cols <- c("cyl","vs", "am", "gear", "carb")
mtcars[cols] <- lapply(mtcars[cols], factor)
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 : Factor w/ 3 levels "4","6","8": 2 2 1 2 3 2 3 1 1 2 ...
##  $ 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  : Factor w/ 2 levels "0","1": 1 1 2 2 1 2 1 2 2 2 ...
##  $ am  : Factor w/ 2 levels "0","1": 2 2 2 1 1 1 1 1 1 1 ...
##  $ gear: Factor w/ 3 levels "3","4","5": 2 2 2 1 1 1 1 2 2 2 ...
##  $ carb: Factor w/ 6 levels "1","2","3","4",..: 4 4 1 1 2 1 4 2 2 4 ...
```

Whever we start working on a new dataset, it helps to examine the relationship between the target (mpg) and the different response variables (remaining columns in the mpg dataset). In addition, we also look at the relationship between differnt pairs of response variables.While pairs in base R works here, we will use ggpairs since it's more informative. A truncated form of the output(for better clarity)is captured in Figure 1 below

Step 2

At this point, let us examine the question as above - “Is an automatic or manual transmission better for MPG ?”

Before getting into model building, we would want to examine if there is any difference between the mean of the MPG for automatic transmission and manual transmission. Hence we will proceed with calculating the means afer grouping the records by the am column.

```
ams <- group_by(mtcars, am)
as.data.frame(summarize(ams, mpgmean = mean(mpg)))

##   am  mpgmean
## 1  0 17.14737
## 2  1 24.39231
```

The means are also depicted graphically in Figure 2. We see that the mean mpg for manual transmission is about 42% higher as compared to automatic transmission.

However, the mere difference in the means is not sufficient to conclusively answer the first question on whether manual or automatic transmission is better. This is where we will turn to hypothesis testing.Since the sample size here is small (32), we will turn to t-test, specifically the one-tailed t-test since the question here boils down to: Is mpg for manual transmission > mpg for automatic transmission and is the difference statistically significant? So we will use a two sample, one-tailed t-test here to check if there is a significant difference between the group means. First, we subset mpg by am and create two different vectors for manual and automatic

```
data.auto = subset(mtcars, am == "0")[,c(1)]
data.manual = subset(mtcars, am == "1")[,c(1)]
```

We then invoke the t.test function. The parameters we choose are important: => var.equal is set to FALSE, implying unequal variances in the two vectors. In this case R uses Welch t.test which is supposed to be better. => alternative = “greater”" imples a one-tailed test, i.e. the first group to the t.test function (manaul in this case) being greater than the second group (auto).

```
t.test(data.manual,data.auto,var.equal=FALSE, alternative = "greater")

##
##  Welch Two Sample t-test
##
## data:  data.manual and data.auto
## t = 3.7671, df = 18.332, p-value = 0.0006868
## alternative hypothesis: true difference in means is greater than 0
## 95 percent confidence interval:
##  3.913256      Inf
## sample estimates:
## mean of x mean of y
##  24.39231  17.14737
```

As we can see from the results, the difference in the two means is significant at p = 0.05 This means we can reject the null hypothesis which says that there is no significant difference in the mpg means for manual and auto, thus establishing that manual transmission indeed gives a better mpg.

Step 3

Having established that manual transmission is better for mpg, let us tackle the second question:

"Quantify the MPG difference between automatic and manual transmissions"

This is where we will use multiple linear regression to build different models and choose the best one. Our first thought would be to build a model using only the predictor variable in question i.e. am

```
model.mtcars1 <- lm(mpg ~ am, data=mtcars)
summary(model.mtcars1)

##
## 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 ***
## am1            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
```

Model interpretation - -The intercept represents the baseline (automatic transmission). Essentially it signifies that for automatic transmission, the mean of the mpg will be about 17.14. -Similary am1 represents manual transmission relative to the baseline. Hence we can expect a mean mpg of 17.14 + 7.24 = 24.38 for manual transmission. -The adjusted R-squared is about 0.34 which means the model is able to explain about 34% of the variance in the response variable.

The 95% confidence interval for Beta1(manual transmission) is the estimate +/- 2 standard errors

```
co = coef(summary(model.mtcars1))
ce1=co[2,1] + 2*(co[2,2])
```

```
ce2=co[2,1] - 2*(co[2,2])
ce1

## [1] 10.77378

ce2

## [1] 3.716096
```

As we can see, the 95% confidence limits are [3.7, 10.8] for Beta1

Step 4

Having built the elementary model with only am as the response variable, let us build a complete model now including the other relevant response variables.

We start by building a model including all response variables available and see if we can better the 34% R-squared which we got by including only the am variable.

```
model.mtcars2 <- lm(mpg ~ ., data=mtcars)
summary(model.mtcars2)

##
## Call:
## lm(formula = mpg ~ ., data = mtcars)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -3.5087 -1.3584 -0.0948  0.7745  4.6251
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 23.87913   20.06582   1.190   0.2525
## cyl6        -2.64870    3.04089  -0.871   0.3975
## cyl8        -0.33616    7.15954  -0.047   0.9632
## disp         0.03555    0.03190   1.114   0.2827
## hp          -0.07051    0.03943  -1.788   0.0939 .
## drat         1.18283    2.48348   0.476   0.6407
## wt          -4.52978    2.53875  -1.784   0.0946 .
## qsec         0.36784    0.93540   0.393   0.6997
## vs1          1.93085    2.87126   0.672   0.5115
## am1          1.21212    3.21355   0.377   0.7113
## gear4        1.11435    3.79952   0.293   0.7733
## gear5        2.52840    3.73636   0.677   0.5089
## carb2       -0.97935    2.31797  -0.423   0.6787
## carb3        2.99964    4.29355   0.699   0.4955
## carb4        1.09142    4.44962   0.245   0.8096
## carb6        4.47757    6.38406   0.701   0.4938
## carb8        7.25041    8.36057   0.867   0.3995
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
##
## Residual standard error: 2.833 on 15 degrees of freedom
## Multiple R-squared:  0.8931, Adjusted R-squared:  0.779
## F-statistic:  7.83 on 16 and 15 DF,  p-value: 0.000124
```

As is evident from the output, the adj R-squared has improved to 0.8165 and the model p-value is significant. However,none of the independent predictors are signficant. Even the wt variable which is otherwise higly correlated with mpg is not signficant with a p-value of 0.0525. So no way that this can be the final model.

To improve the model by removing non-informative variables, we have a few options (and these are not the only ones):

1) Do a backward elimination from the all predictor model as above. This involves removing the least signficant predictor in the current model (gear in the one above), rebuilding the model and repeating the process till the remaining predictors are signficant. While this is a straightforward process, when the original data has a high no of features. Also, we need to be mindful of the interaction between the predictors so dropping a predictor based on low signficance in one iteration may not always be right.
2) We can examine the correlation between the response and the different predictor variables and pick the highest correlated ones. From these we can clean up predictors that are correlated between each other, retaining only one. While this approach will give a fairly good model quickly, it's still not a foolproof solution since a relatively lower correlation for a predictor variable (wrt response variable) may not be reason enough for the predictor to become insignificant in the final model
3) After trying out the above two approaches and not being satisfied with the results, I finally settled on using the bestglm package to pick out the best subset of predictors as below

```
#The response variable needs to be called 'y' for using in bestglm
mtcars.for.bestglm <- within(mtcars, {
  y    <- mpg         # mpg into y
  mpg  <- NULL        # Delete mpg
})

res.bestglm <-
  bestglm(Xy = mtcars.for.bestglm,
          #family =  , # not reqd for muliple regression
          IC = "AIC",                 # Information criteria
          method = "exhaustive")

## Morgan-Tatar search since factors present with more than 2 levels.

res.bestglm$BestModels

##     cyl  disp    hp  drat   wt  qsec    vs    am  gear  carb Criterion
## 1 FALSE FALSE FALSE FALSE TRUE  TRUE FALSE  TRUE FALSE FALSE  59.30730
## 2 FALSE FALSE  TRUE FALSE TRUE  TRUE FALSE  TRUE FALSE FALSE  59.51530
## 3  TRUE FALSE  TRUE FALSE TRUE FALSE FALSE  TRUE FALSE FALSE  59.65483
```

```
## 4  TRUE FALSE  TRUE FALSE TRUE FALSE FALSE FALSE FALSE FALSE  59.65716
## 5  TRUE FALSE  TRUE FALSE TRUE FALSE  TRUE  TRUE FALSE FALSE  60.05921
```

So we will pick wt, qsec and am as our final predictors as suggested by bestglm. Please note in an actual project, the choice of the final predictors will also be driven by the needs of the business, not necessarily by what the program suggests.

```
model.mtcars3 <- lm(mpg ~ wt+qsec+am, data=mtcars)
summary(model.mtcars3)
```

```
##
## 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 ***
## am1           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
```

We see that the adj r-squared has improved to 0.8336 and the 3 predictors are significant at p = 0.05

Let us look at the coefficients closely and additionally calculate the 95% confidence limit range like earlier, i.e. the Estimate +/- 2*Std.Error

```
coef.mtcars <- as.data.frame(coef(summary(model.mtcars3)))
coef.mtcars$confintfrom <- coef.mtcars$Estimate - (2*coef.mtcars$`Std. Error`
)
coef.mtcars$confintto <- coef.mtcars$Estimate + (2*coef.mtcars$`Std. Error`)
coef.mtcars[,c(1,2,4,5,6)]
```

```
##              Estimate Std. Error     Pr(>|t|) confintfrom confintto
## (Intercept)  9.617781  6.9595930 1.779152e-01  -4.3014055 23.536966
## wt          -3.916504  0.7112016 6.952711e-06  -5.3389070 -2.494100
## qsec         1.225886  0.2886696 2.161737e-04   0.6485469  1.803225
## am1          2.935837  1.4109045 4.671551e-02   0.1140282  5.757646
```

Model interpretation

-wt is the most significant predictor, though negative. 1 unit (1000 lbs) increase in the weight of the vehicle (keeping other variables constant) reduces the mpg by about 3.92 miles/gallon on an average. We expect the actual reduction to range between 5.33 miles/gallon and 2.49 miles/gallon (for every 1000 lbs increase in weight) 95% of the time, as defined by the confidence interval.

-qsec comes next and is positively correlated with mpg. qsec here is the time in seconds it takes for the car to go from a stop to complete a quarter mile. So, as per our model, a one second increaes in qsec (keeping other variables constant) increases the mpg by 1.22 miles/gallon on an average. We expect the actual increase in mpg to range between 0.65 miles/gallon and 1.8 miles/gallon 95% of the time, as defined by the confidence interval.

-am is the least signficant of the three. Being a categorical variable, instead of a unit change, the interpretion is based on the choice of manual mode when compared to automatic mode, other variables remaining constant. So the manual mode of the transmission (compared to auto mode) increases the mpg by 2.93 miles/gallon on an average. We expect the actual increase to range from 0.11 miles/gallon to 5.75 miles/gallon 95% of the time based on the confidence interval.

-interface here cannot be interpreted easily. In some cases the interface represents the value of the dependent variable when all predictors are zero. However, in this case wt/qsec cannot be zero for any car. Also, we have a categorical variable (am) in the fray. If there was only this one categorical variable, the interface would have represented the baseline class (automatic transmission). However, with continuous as well as categorical predictors in the model, interface is an undecipherable mix and best not interpreted further.

Another interesting thing to do would be using anova to compare the initial model (with only am as predictor) with this final model to see if the increase in r-squared is statistically signficant or not.

```
anova(model.mtcars1, model.mtcars3)
```

```
## 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
```

It's evident that by adding another two predictors (wt & qsec) to the final model, the increase in r-squared is statistically signficant.

Figures

Fig 1 - Pair wise scatter plot of the columns in the dataset

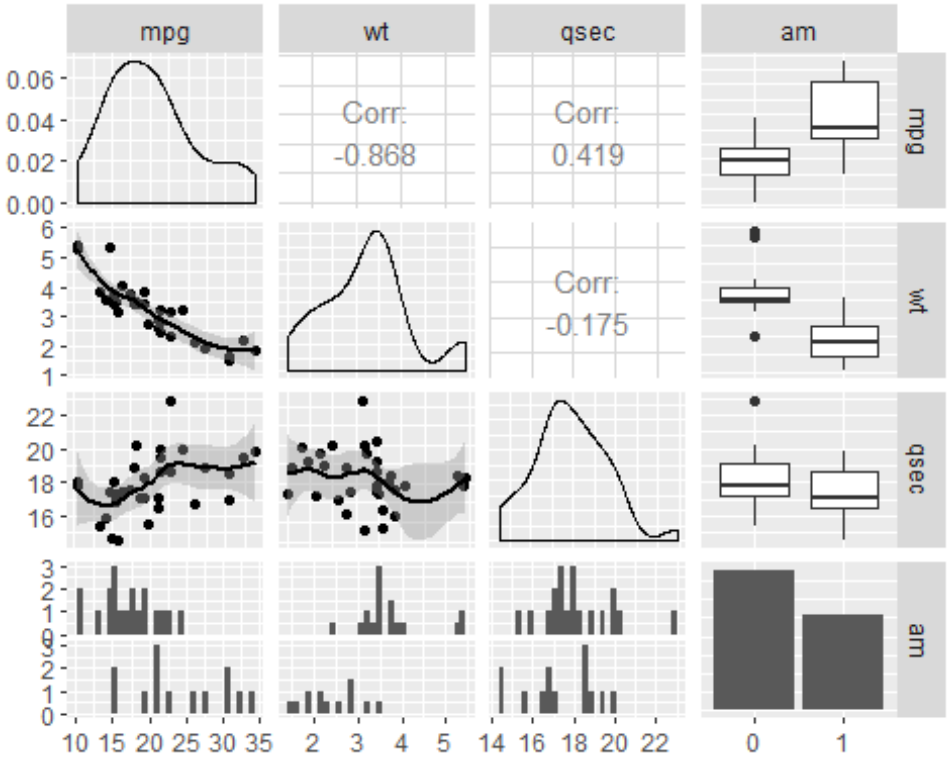


Fig 2 - Scatter plot of mpg vs am

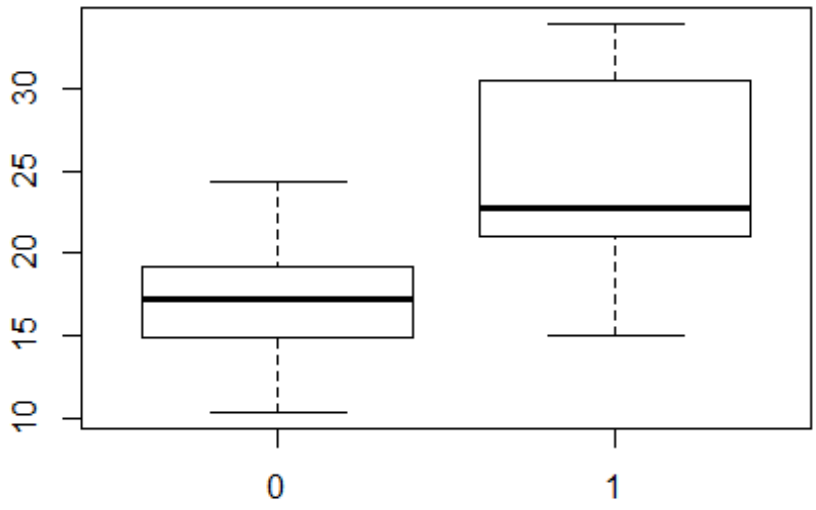
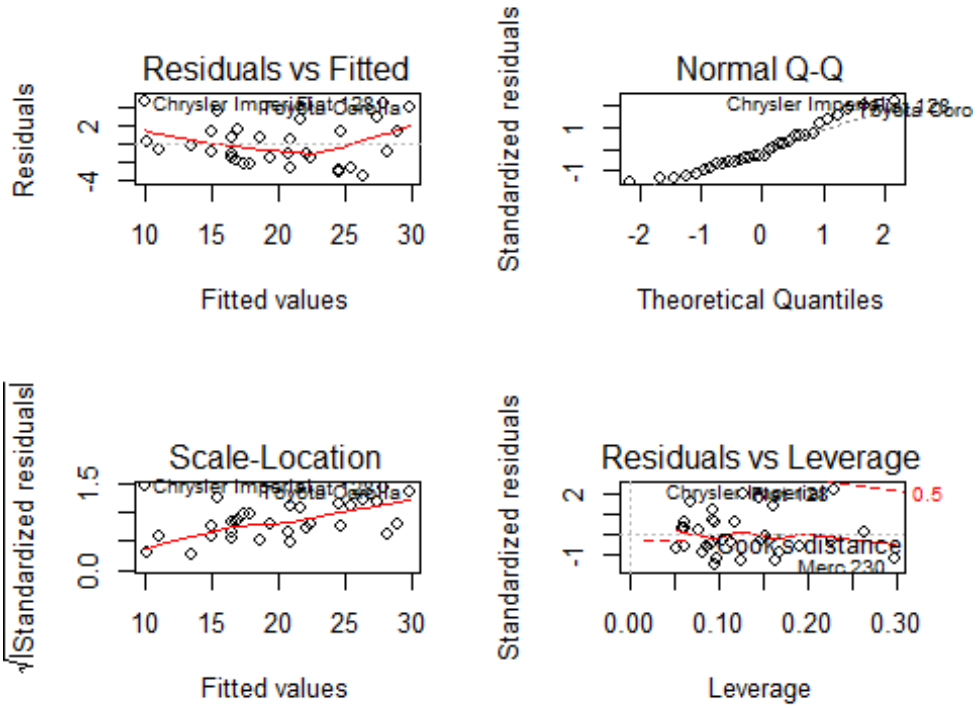


Fig 3 - Diagnostic Plots



First Plot - Residuals vs fitted values - for checking linearity/homoscedasticity. Most of the points here are within [-2,2] so linearity is confirmed. Crysler Imperial, Fiat128 and Totoyota Corolla are called out as outliers though. Also, there is no pattern to the residuals which confirms homoscedasticity (constant variance)

Second Plot - QQ Plot for normality assumption - since observations lie along the 45 degree line, normality holds here

Third Plot - Scale location - for checking homoscedasticity. We don't see a strong pattern here though there is a hint of linearity.

Fourth Plot - Cook's distance to measure the influence of each observation on the regression coefficients. We do see a few observations outside [-1,1] – Chrysler Imperial, Fiat128 and Merc230 are called out here. Ideally these would require further investigation.