Regression Models Course

Peer graded project - Motor trend MPG data analysis

Albert Fradera Sola

2020-06-25

Contents

Motor trend MPG data analysis **1**
Overview 1
1: Data load, exploratory analysis and summary 1
2: Data analyis 5
Testing for differences 5
Fitting a regression model 6
3: Conclusions: 14

Motor trend MPG data analysis

Overview

You work for Motor Trend, a magazine about the automobile industry. Looking at a data set of a collection of cars, they are interested in exploring the relationship between a set of variables and miles per gallon (MPG) (outcome). They are particularly interested in the following two questions:

- Is an automatic or manual transmission better for MPG
- Quantify the MPG difference between automatic and manual transmissions

1: Data load, exploratory analysis and summary

We start by loading the data and storing it into a data frame:

```r
data("mtcars") # Load the data
mtcars_df <- as.data.frame(mtcars) # Store it on a data frame
```

First thing we do is to check how our data looks like:

```r
kable(head(mtcars_df, n = 4), caption = "First 4 entries on the data set")
```

Table 1: First 4 entries on the data set

	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

We have 11 variables; but looking at the head, we cannot grasp the full information on the variables. Thus, and looking for more details on the data set, we go to the package documentation to find the following information on the variables:

- 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

Next step is to check wether the variables are numerical or categorical:

```r
pander(as.data.frame(t(apply(mtcars_df,2,class))),
       split.table = 80,
       style = "rmarkdown",
       caption = "Class of the variables")
```

Table 2: Class of the variables (continued below)

mpg	cyl	disp	hp	drat	wt	qsec	vs	am	gear	carb
numeric	numeric	numeric	numeric	numeric	numeric	numeric	numeric	numeric	numeric	numeric

We relabel those variables that are categorical and whose classes are assigned as numerical:

```r
# Change attributes from numeric to factor
mtcars_df$cyl <- factor(mtcars_df$cyl)
mtcars_df$vs <- factor(mtcars_df$vs)
mtcars_df$am <- factor(mtcars_df$am, labels = c("AT","MT")) #Relabel: 0, automatic (AT) and 1, manual (
mtcars_df$gear <- factor(mtcars_df$gear)
mtcars_df$carb <- factor(mtcars_df$carb)
```

Thus we could consider which is the influnce of the the type of transmission (automatic or manual) to the mpg and if other factors may have an influence. The head of the data frame only gives information about the structure of the data frame. We will get more details on how many items of each we have by doing a data summary:

```r
# Summary of our data
pander(summary(mtcars_df),
       split.table = 80,
       style = 'rmarkdown',
       caption = "Data set summary",
       missing = "")
```

Table 4: Data set summary (continued below)

mpg	cyl	disp	hp	drat	wt	qsec	vs	am	gear	carb
Min. :10.40	4:11	Min. : 71.1	Min. : 52.0	Min. :2.760	Min. :1.513	Min. :14.50	0:18	AT:19	3:15	1: 7
1st Qu.:15.43	6: 7	1st Qu.:120.8	1st Qu.: 96.5	1st Qu.:3.080	1st Qu.:2.581	1st Qu.:16.89	1:14	MT:13	4:12	2:10
Median :19.20	8:14	Median :196.3	Median :123.0	Median :3.695	Median :3.325	Median :17.71			5: 5	3: 3
Mean :20.09		Mean :230.7	Mean :146.7	Mean :3.597	Mean :3.217	Mean :17.85				4:10
3rd Qu.:22.80		3rd Qu.:326.0	3rd Qu.:180.0	3rd Qu.:3.920	3rd Qu.:3.610	3rd Qu.:18.90				6: 1
Max. :33.90		Max. :472.0	Max. :335.0	Max. :4.930	Max. :5.424	Max. :22.90				8: 1

For the continous variables we obtain the mean, the median and the quantiles. For the discrete variables we obtain the number of entries. Next step is to see if we have an equal number of entries per each transmission:

```r
kable(table(mtcars_df$am), caption = "Number of observations")
```

Table 6: Number of observations

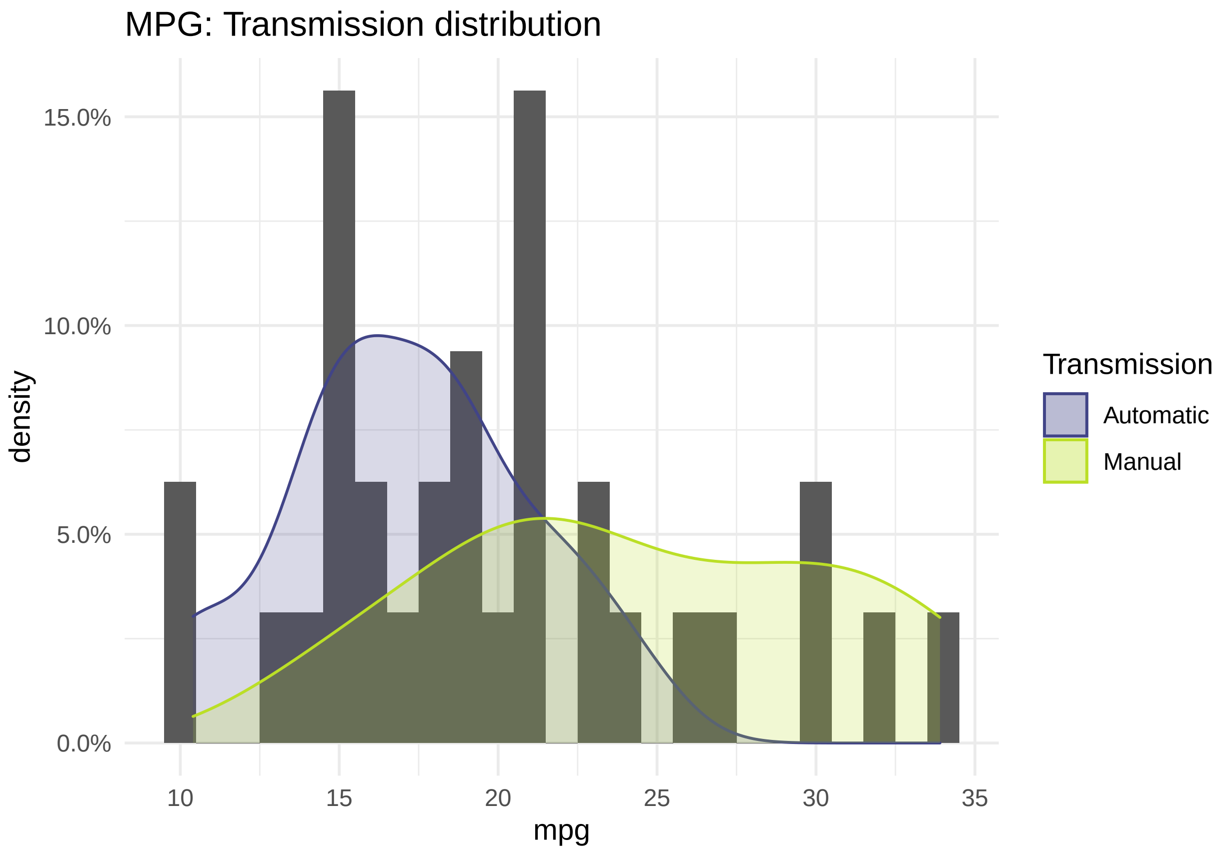
Var1	Freq
AT	19
MT	13

Next would be to look the distribution of our countinous variable and overlay the density of our discrete variables of interest:

```r
colours <- viridis_pal(alpha = 1, begin = 0.2, end = 0.9, direction = 1,option = "D")(2)

densityplot_1 <-  ggplot(data = mtcars_df, mapping = aes(x = mpg))+
                  geom_histogram(binwidth = 1,aes(y = ..density..))+
                  geom_density(data = mtcars_df[mtcars_df$am == "AT", ],
                               aes(color = "Automatic",
                                   fill="Automatic"),
                               alpha=.2)+
                  geom_density(data = mtcars_df[mtcars_df$am == "MT", ],
                               aes(color = "Manual",
                                   fill="Manual"),
                               alpha=.2)+
                  scale_y_continuous(labels = percent_format())+
                  scale_colour_manual("Transmission", values = c("Automatic" = colours[1],
                                                                 "Manual" = colours[2]))+
                  scale_fill_manual("Transmission", values = c("Automatic" = colours[1],
                                                               "Manual" = colours[2]))+
                  ggtitle("MPG: Transmission distribution")+
                   theme_minimal()

print(densityplot_1)
```



Now that we know how our data looks like, we can start exploring its properties. We start by obtaining the mpg mean and the variance for the different transmissions:

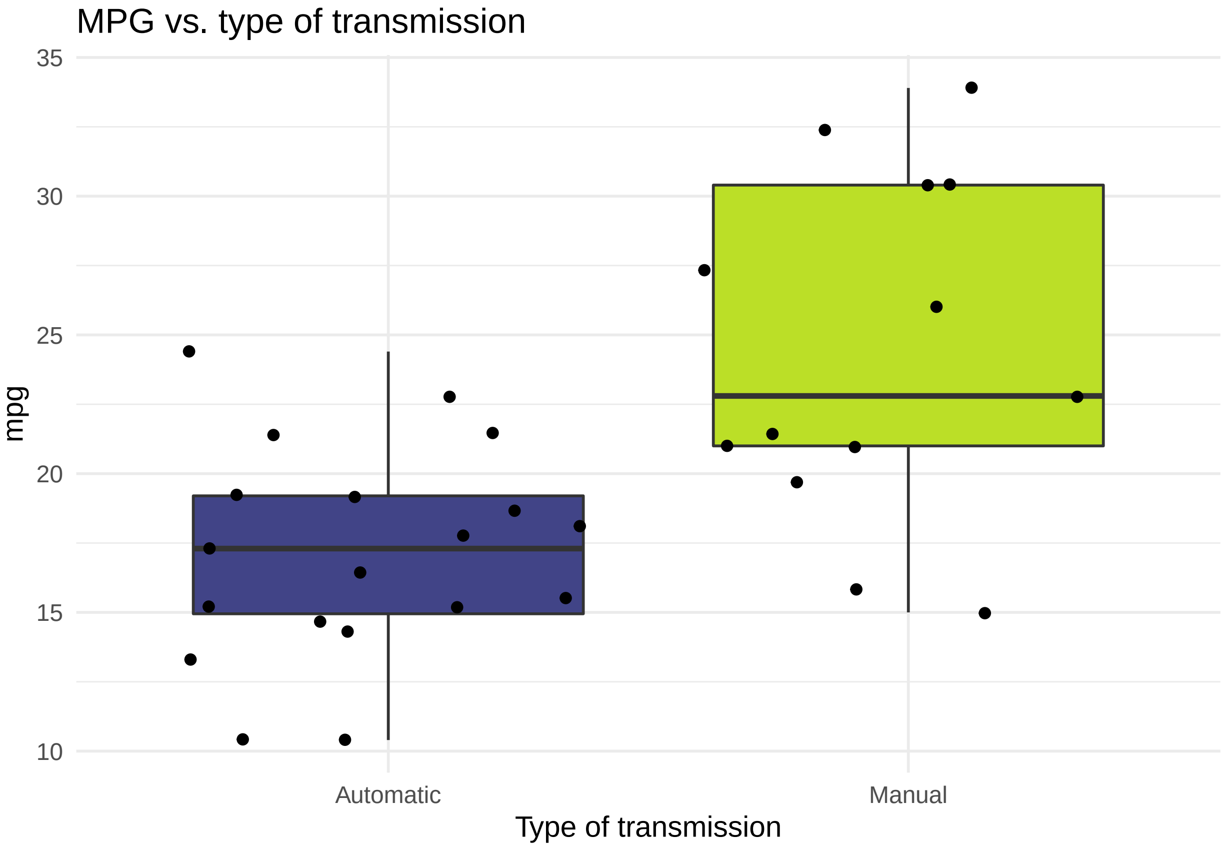
```
# Transmission infered values:
kable(cbind(tapply(mtcars_df$mpg, list(mtcars_df$am), mean),
            tapply(mtcars_df$mpg, list(mtcars_df$am), var)),
      digits = 2,
      caption = "Transmssion estimate values",
      col.names = c("Mean", "Variance"))
```

Table 7: Transmssion estimate values

	Mean	Variance
AT	17.15	14.70
MT	24.39	38.03

A better way to visualize the quantiles and means of our data is via graphical exploration. Thus we draw boxplots of the mpg depending on the transmission:

```
boxplot_1 <-  ggplot(data = mtcars_df, mapping = aes(x = am, y = mpg, fill = am))+
              geom_boxplot()+
              geom_jitter()+
              scale_fill_viridis(discrete = T, option = "D", end = 0.9, begin = 0.2)+
              ggtitle("MPG vs. type of transmission")+
              scale_x_discrete(labels = c('Automatic','Manual'))+
              xlab("Type of transmission")+
              theme_minimal()+
              theme(legend.position = "none")
print(boxplot_1)
```



Overall, we observe that the type of transmission change the shape of the distribution of mpg. This is supported by different mpg means and all the graphical visual exploration. But are those differences signigicant? We will test that on the following block.

2: Data analyis

Testing for differences

During he exploratory analysis we saw that there are differences on the MPG when looking at the inlunece of the transmission. To test wether those differences are significant or not, we are going to perform a t test to accept or reject the following hyphothesis:

$$H_0 : \mu_1 = \mu_2$$

$$H_\alpha : \mu_1 \neq \mu_2$$

We assume that the distribution follows normality and that there is no equality on the variances.

```
ttest <- t.test(mpg ~ am, data = mtcars, paired = FALSE)
print(ttest)
```

```
##
##  Welch Two Sample t-test
##
## data:  mpg by am
## t = -3.7671, df = 18.332, p-value = 0.001374
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -11.280194  -3.209684
## sample estimates:
## mean in group 0 mean in group 1
##        17.14737        24.39231
```

We oserve that 0 is not included in our confidence interaval and that we obatin a p-value lower than our significance threshold of 0.05 (p-value = 0.0014). Thus we **reject the null hyphothesis**. Thus, mpg is defiened by the transmission value. But, is it the only variable explaining MPG?

Fitting a regression model

We know that MPG is influenced by the transmission, but we need to asses wether is the only variable explaining it or if there are more variables which have an influence on MPG. First thing we do, is to model MPG in function of the transmission:

```
single_fit <- lm(mpg ~ am, mtcars_df) # linear model including only transmission
print(summary(single_fit))
```

```
##
## Call:
## lm(formula = mpg ~ am, data = mtcars_df)
##
## 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 ***
## amMT           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
```

Our coefficients show us the estimate for the automatic transmission (intercept = 17.47) and the manual transmission (amMT = 24.39 or 17.147 + 7.245). This estimations match with the infered mean obtained previously on the hypothesis testing. We also observe that the linear model including a single regressor, am, returns significant pvalues for the coefficients but a low R^2 value ($R^2 = 0.36$). An interepretation of this result would be that our regressor, tansmission type, only explains 36% of the MPG differences. Thus, we should explore the influence of the other variables on MPG. We start by showing how they correlate:

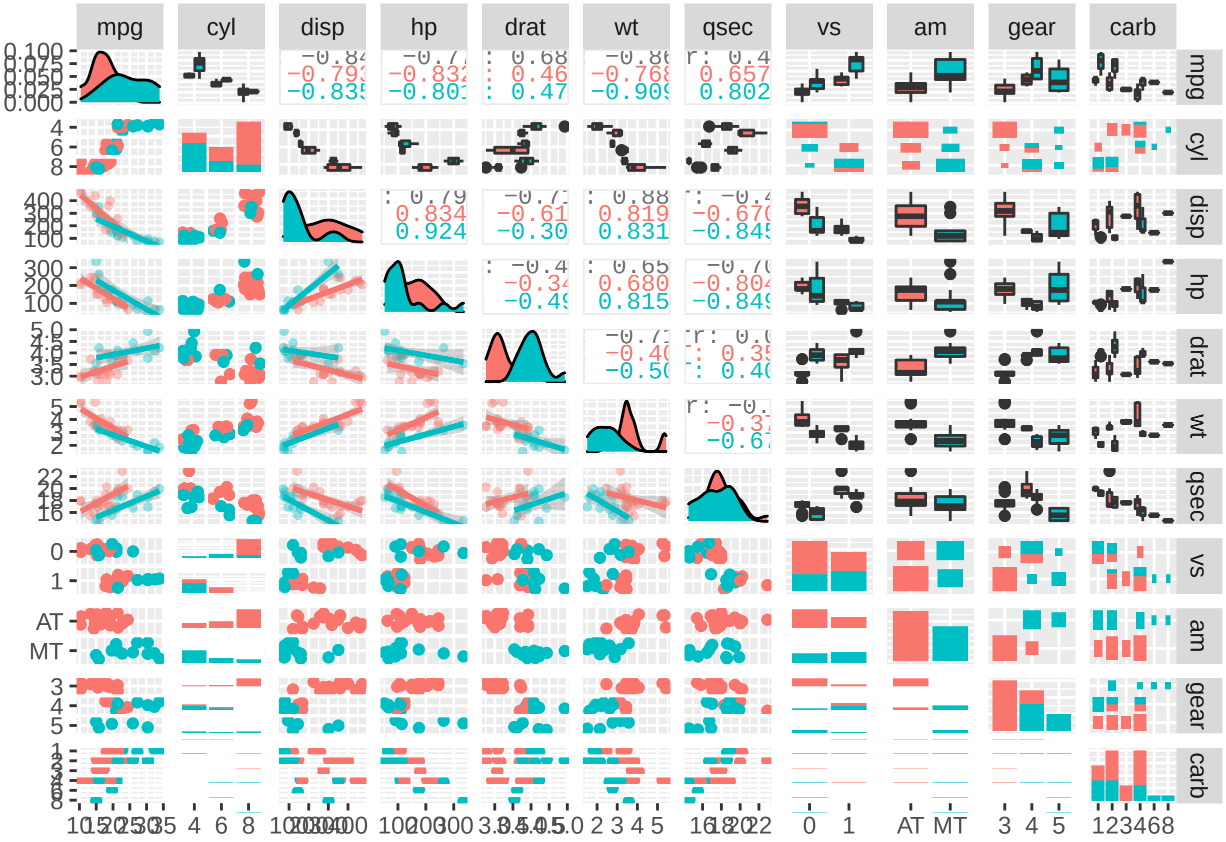
```r
cor <- cor(mtcars$mpg, mtcars) # Compute the correlation between samples
corord <- as.data.frame(t(cor[,order(-abs(cor[1,]))]))
cor_test <- cor
for (i in 1:ncol(cor_test)) {
  cor_test_loop <- cor.test(mtcars$mpg, mtcars[,colnames(cor_test)[i]])
  pval <- cor_test_loop$p.value
  cor_test[i] <- pval
}
table <- rbind(corord,cor_test)
rownames(table) <- c("Coeficient", "P-value")
# Print correlation coeficient in decreasing order with they p-values
pander(table,
       split.table = 80,
       style = 'rmarkdown',
       caption = "Correlation coeficient and its associated p-value",
       missing = "",
       digits = 2)
```

Table 8: Correlation coeficient and its associated p-value (continued below)

	mpg	wt	cyl	disp	hp	drat	vs	am	carb	gear	qsec
Coeficient	1	-0.87	-0.85	-0.85	-0.78	0.68	0.66	0.6	-0.55	0.48	0.42
P-value	0	1.3e-10	6.1e-10	9.4e-10	1.8e-07	1.8e-05	3.4e-05	0.00029	0.0011	0.0054	0.017

All the variables seem to be more or less significantly correlated to MPG. This can also be shown on a pairs plot:

```r
mtcars_df %>%
ggpairs(.,
     mapping = ggplot2::aes(color = am),
     upper = list(continuous = wrap("cor", size = 3), combo = wrap("box_no_facet")),
     lower = list(continuous = wrap("smooth", alpha=0.4, size=1), combo = wrap("dot"))
   )
```



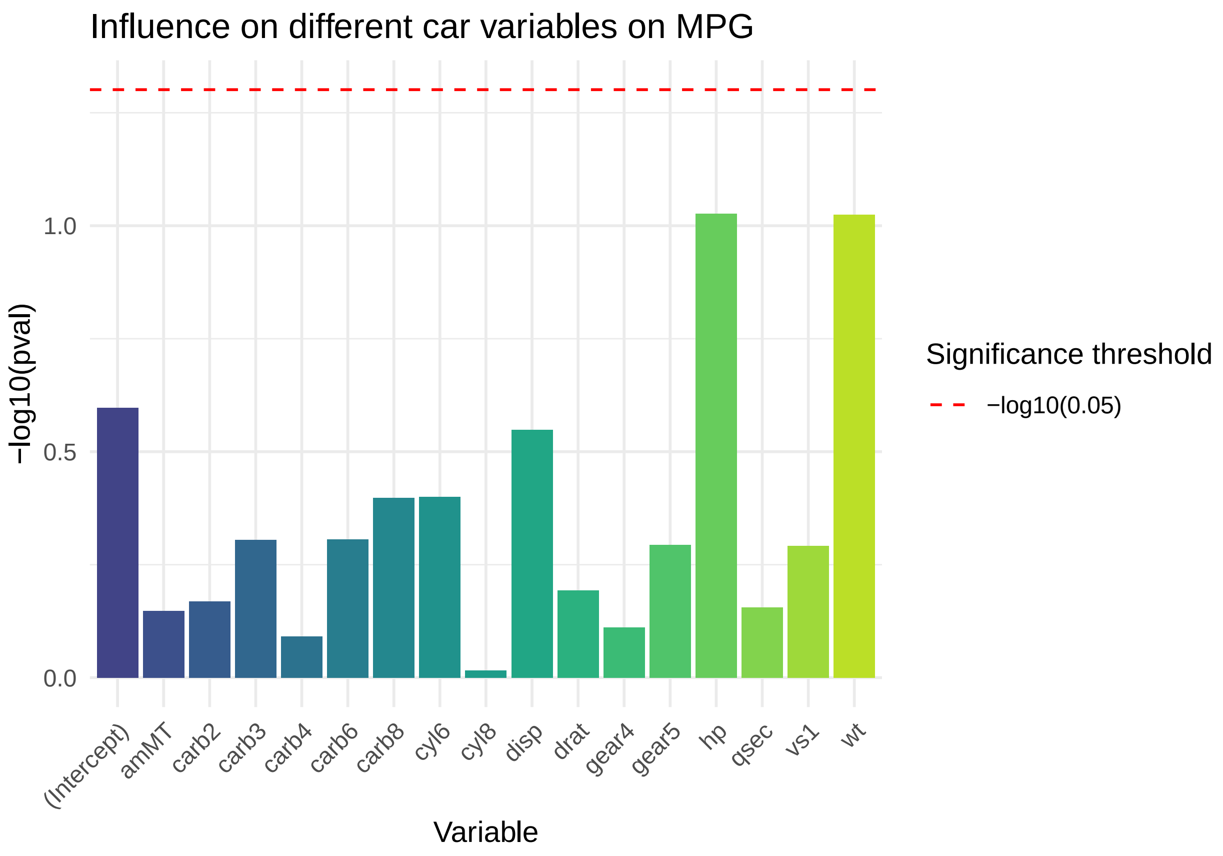
Knowing that, in a higher or lower degree, all variables have an influence on MPG we are going to try to fit a new model including all variables:

```
all_fit <- lm(mpg ~ ., mtcars_df)
print(summary(all_fit))
```

```
##
## Call:
## lm(formula = mpg ~ ., data = mtcars_df)
##
## 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
## amMT         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
```

We can see now that our R^2 is much higher ($\mathrm{R}^2 allfit = 0.89$)) than before ($\mathrm{R}^2 singlefit = 0.36$). But even now we are able to explain more differences on MPG, none of our coefficients are significant, as none is lower than 0.05:

```
df <- as.data.frame((-log10(summary(all_fit)$coefficients[,4])))
colnames(df) <- "pval"
df$Variable <- rownames(df)
barplot <-  ggplot(data = df, aes(x = Variable, y = pval, fill = Variable))+
            geom_bar(stat = "identity", show.legend = F)+
            scale_fill_viridis(discrete = T, option = "D", end = 0.9, begin = 0.2)+
            geom_hline(aes(yintercept = -log10(0.05),
            linetype = "-log10(0.05)"),
            color = "red")+
            scale_linetype_manual(name = "Significance threshold", values = c(2, 2),
            guide = guide_legend(override.aes = list(color = c("red"))))+
            ggtitle("Influence on different car variables on MPG")+
            ylab("-log10(pval)")+
            theme_minimal()+
            theme(axis.text.x = element_text(angle = 45, hjust = 1))
print(barplot)
```



We can observe that none of the variables reach our significance threshold. Since regressors can correlate among them and not only to MPG, we might be overfitting the model and not predicting the MPG differences in a correct way. To check which regressors are helpful in our model, we should check all the model combinations with the different regressors and choose which fits better. We explore three different criteria to choose which model to keep:

- Akaike information criterion (AIC), as implemented in step function.
- Mallows's Cp, as implemented in regsubsets function. The lower is this value, the better a model fits. Similiar to AIC
- Bayesian information criterion (BIC), as implemented in regsubsets function. The lower is this value, the better a model fits. More restricitve (higher penalty for parameter) than AIC.

Thus we proceed to compute the values for the different criteria so we can make a good model decision:

```
# Compute AIC for all models
step_fit <- step(all_fit,direction="both",trace=F)
print(summary(step_fit))
```

```
##
## Call:
## lm(formula = mpg ~ cyl + hp + wt + am, data = mtcars_df)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -3.9387 -1.2560 -0.4013  1.1253  5.0513
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 33.70832    2.60489  12.940 7.73e-13 ***
## cyl6        -3.03134    1.40728  -2.154  0.04068 *
## cyl8        -2.16368    2.28425  -0.947  0.35225
## hp          -0.03211    0.01369  -2.345  0.02693 *
## wt          -2.49683    0.88559  -2.819  0.00908 **
## amMT         1.80921    1.39630   1.296  0.20646
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.41 on 26 degrees of freedom
## Multiple R-squared:  0.8659, Adjusted R-squared:  0.8401
## F-statistic: 33.57 on 5 and 26 DF,  p-value: 1.506e-10
```

```r
# Compute Cp and BIC for all models
subset_fit <- regsubsets(mpg ~ ., mtcars_df, nvmax = 25)
subset_fit_summary <- summary(subset_fit)
adjr2 <- which.max(subset_fit_summary$adjr2) # Model with higher Rsquared
cp <- which.min(subset_fit_summary$cp) # Model with lower Cp
bic <- which.min(subset_fit_summary$bic) # Model with lower BIC
best_set <- subset_fit_summary$outmat[c(adjr2,cp,bic),] # Filter summary following our criteria
rownames(best_set) <- c(paste0("Rsquared model (", adjr2,")"),
                        paste0("Cp Model (", cp,")"),
                        paste0("BIC model (", bic,")"))
pander(best_set,
       split.table = 80,
       style = 'rmarkdown',
       caption = "Selected models with Rsquared, Cp and BIC",
       missing = "",
       digits = 2)
```

Table 10: Selected models with Rsquared, Cp and BIC (continued below)

	cyl6	cyl8	disp	hp	drat	wt	qsec
Rsquared model (5)	*			*		*	
Cp Model (3)						*	*
BIC model (3)						*	*

Table 11: Table continues below

	vs1	amMT	gear4	gear5	carb2	carb3
Rsquared model (5)	*	*				
Cp Model (3)		*				
BIC model (3)		*				

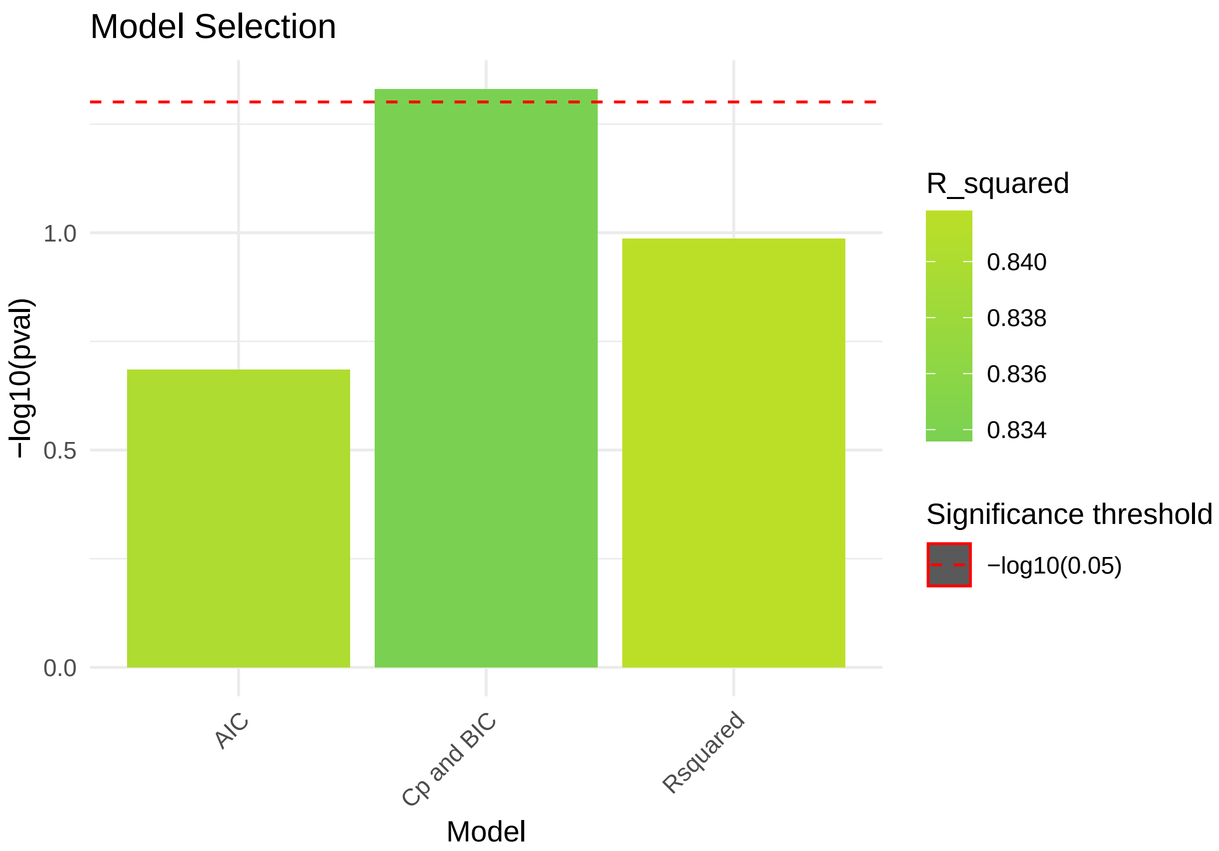
	carb4	carb6	carb8
Rsquared model (5)			
Cp Model (3)			
BIC model (3)			

We can observe that, while Cp and BIC return the same model, AIC gives us a different model. Besides we also selected the one that gave us the best R^2. Thus we ended up with the following models:

- AIC: mpg ~ cyl + hp + wt + am
- R^2: mpg ~ am + cyl + hp + wt + vs
- Cp and BIC: mpg ~ am + wt + qsec

Our next step will be to explore the different models and keep the one with a better fit. Thus we fit the trhee models and select the R^2 value and the p-value for our regressor of interest (am) and keep the model that gives us a combined better performance:

```r
# Models
model_AIC <- lm(mpg ~ cyl + hp + wt + am, mtcars_df)
model_R2 <- lm(mpg ~ am + cyl + hp + wt + vs, mtcars_df)
model_CB <- lm(mpg ~ am + wt + qsec, mtcars_df)
# Keep Rsquared
R_squared <- round(c(summary(model_AIC)$adj.r.squared,
                         summary(model_R2)$adj.r.squared,
                         summary(model_CB)$adj.r.squared),4)
# Keep p-values
amMT_Pvalues <- round(c(-log10(summary(model_AIC)$coefficients["amMT",4]),
                        -log10(summary(model_R2)$coefficients["amMT",4]),
                        -log10(summary(model_CB)$coefficients["amMT",4])),4)
# Build data frame
Model <- c("AIC", "Rsquared", "Cp and BIC")
df <- data.frame(Model, R_squared, amMT_Pvalues)
barplot <-  ggplot(data = df, aes(x = Model, y = amMT_Pvalues, fill = R_squared))+
            geom_bar(stat = "identity", show.legend = T)+
            scale_fill_viridis(discrete = F, option = "D", end = 0.9, begin = 0.8)+
            geom_hline(aes(yintercept = -log10(0.05),
            linetype = "-log10(0.05)"),
            color = "red")+
            scale_linetype_manual(name = "Significance threshold", values = c(2, 2),
            guide = guide_legend(override.aes = list(color = c("red"))))+
            ggtitle("Model Selection")+
            ylab("-log10(pval)")+
            theme_minimal()+
            theme(axis.text.x = element_text(angle = 45, hjust = 1))
print(barplot)
```

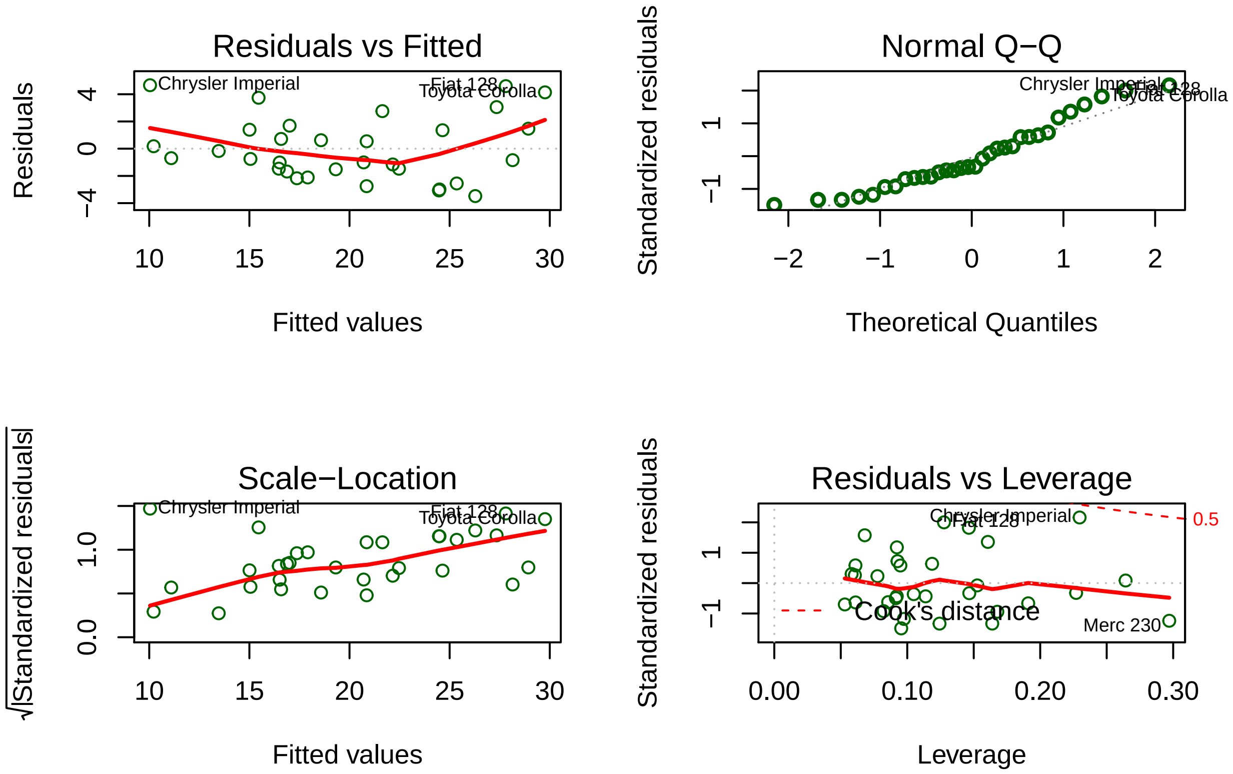


While all the models show a similiar amount of MPG variation explanation (very close R^2 values) there is only one model that accounts for transmission as signigicant. Since this is our regressor of interest, we choose it as our final model. Thus our final model results from both Cp and BIC criteria and includes the following regressors:

- wt: Weight (1000 lbs)
- qsec: 1/4 mile time
- am: Transmission (0 = automatic, 1 = manual)

Our last step is to check the diagnostic plots for our selected model:

```r
par(mfrow = c(2,2))
plot(model_CB, col = "darkgreen", lwd = 2)
```



The diagnosis plots show us the following:

- Residuals vs Fitted: They do not follow any particular pattern, looks like randomly distributted which is a good sign.
- Normal Q-Q: The points only deviate slightly from the diagonal, indicating that data follows normality
- Scale-location: The residues spread slightly wider on the upward slope
- Residuals vs leverage: No point appears to have leverage nor point has to be considered an outlayer

3: Conclusions:

Before taking our final conclusions, we look at the summary of our selected model:

```
print(summary(model_CB))
```

```
##
## Call:
## lm(formula = mpg ~ am + wt + qsec, data = mtcars_df)
##
## 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
## amMT          2.9358     1.4109   2.081 0.046716 *
## wt           -3.9165     0.7112  -5.507 6.95e-06 ***
## qsec          1.2259     0.2887   4.247 0.000216 ***
## ---
## 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
```

If you want to optimize the MPG on the road the best way to go is to select **a manual transmission**. It is confirmed to have a higher MPG by t-test with a significance value lower than 0.05 (p-value = 0.0014)). Our best regression model predicts an increase of 2.935 MPG when using manual transmission with a significance value lower than 0.05 (p-value = 0.0467) and a high R^2 ($R^2 = 0.8336$). The model also shows that **transmission** is not the only variable that explains MPG, as both **weight** and **qsec** are also included in our model.

We have to take these results carefully, as the scale-location diagnosis plot show that we might be missing something in our model (tends to spread slightly wider on the upward slope), which might be related to low number of observations. Thus we cannot guarantee that with an increased number of observations this model stands true.