Multivariate Relationship of Transmission to Miles Per Gallons

Executive Summary

After an initial exploratory data analysis, a model selection procedure and some diagnostics, it can be speculated from looking at the boxplot(frame) and the grouped mpg data (grouped by am), that a manual transmission is better for the miles per gallon usage. To quantify this advantage of using manual transmission, a confidence interval of 95% returns that cars with a manual transmission lie within 12.059 and 36.725 miles per gallon, whereas cars with automatic transmission can reach values between 9.479 and 24.815 miles per gallon (e.g. Cadillac Fleetwood and Lincoln Continental both have an mpg of 10.4). This is, however, only a linear approach and thus omits the multivariate dimension of the dataset. I saw, however, no other way to quantify the difference between manual and automatic transmission as my model is wrong. In a correct model the actual change could have been quantified with something similar to the following command: "sumCoef[1,1] + c(-1, 1) * qt(.975, df = fit$df) * sumCoef[1, 2]".

Exploratory Data Analysis

In order to get an overview of the dataset mtcars it is highly recommended to read the help page (?mtcars) first. Here the different variables are explained. It is to be pointed out that some variables are discrete (such as: mpg, qsec, etc) and others are categorical (such as: am, gear, cyl). Different considerations might apply to these types of data during the following analyses. A boxplot comparing the values for mpg with automatic transmission ("mpg_am0") to the once with manual transmission ("mpg_am1") gives an initial non-multivariate hint on whether the transmission influences the miles per gallons used or not. In the appendix the boxplot "# Boxplot mpg ~ am" shows that manual transmission tends to be better, already. This is, however, not considering the multivariate dataset and thus has to be varified in the coming discussion.

Residual Plot and Diagnostics

An initial residual plot of the model shows that there are factors involved or variables missing or something is funny with our model (the red line is not a straight line, see Appendix "# Initial Residual Plot"). Looking at the different diagnostic tools, the hatvalues give a measure of leverage (i.e. potential for influencing the regression) and the dfbetas give a change in individual coefficients when the ith point is deleted in fitting the model. The whole dataset was tested on these three diagnostic tools by finding the mean of the diagnostic outcome and marking all car names which lie outside a 95% interval around this mean diagnostic outcome value (see Appendix: "## Diagnostics"). As the "Merc 230" and the "Ford Pantera L" were significantly outside the boundaries within the dfbetas and the hatvalues both cars were deleted from the dataset. And a new residual plot was created with the adjusted dataset (mtcars2), which slightly adjusted the red line of the residual plot (see Appendix: "# Second Residual Plot").

Interpreting the Coefficients

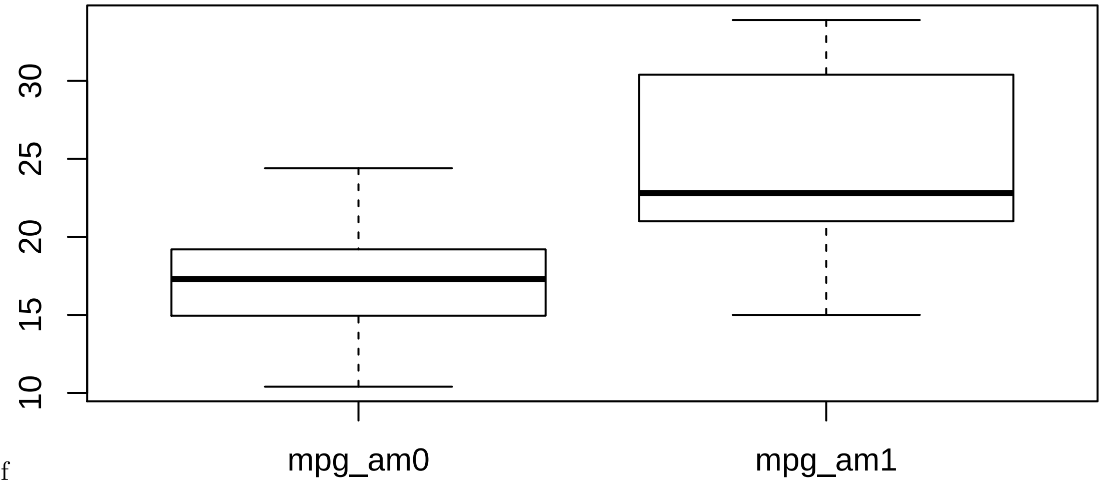
If we look at the adjasent coefficients of the first model (fit) and the second model (fit2, with the cars removed) we can see in the appendix, that the coefficients change drastically and the transmission (am) turns from being second best in terms of probability values to being third last (“# Comparison of Coefficients for Model 1 (fit) and Model 2 (fit2)”). Although in the first dataset no regression coefficient is significant (i.e. no Pr(>|t|) is lower than 0.05) I rather tend to use the initial dataset as there is no test-design reason to remove the cars from the dataset. This is only justified mathematically in the previous paragraph. Also, adjusting the dataset and looking at the significant values of the coefficients for the second fit it becomes clear, that removing the cars from the dataset results in changing only two values to be significant. The model as a whole is still not feasible as there are too many coefficients of the variables not significant.

Conclusion

Thus I have to conclude, that my model has some major unadjusted issues. After doing some transformation (such as log, sqrt etc.) on the data I try to interpred the dataset as it is. Thus I can only quantify the change on a linear, non-multivariate basis as shown in the boxplot. (For the following see Appendix: “# Quantifying the Change”) The 95% confidence interval lies within 12.059 and 36.725 miles per gallon for the manual transission, whereas cars with automatic transmission can reach values between 9.479 and 24.815 miles per gallon (e.g. Cadillac Fleetwood and Lincoln Continental both have an mpg of 10.4). This is, however, only a linear approach and thus omits the multivariate dimension of the dataset. I saw, however, no other way to quantify the difference between manual and automatic transmission as my model is wrong. In a correct model the actual change could have been quantified with something similar to the following command: “sumCoef[1,1] + c(-1, 1) * qt(.975, df = fit$df) * sumCoef[1, 2]”

Appendix

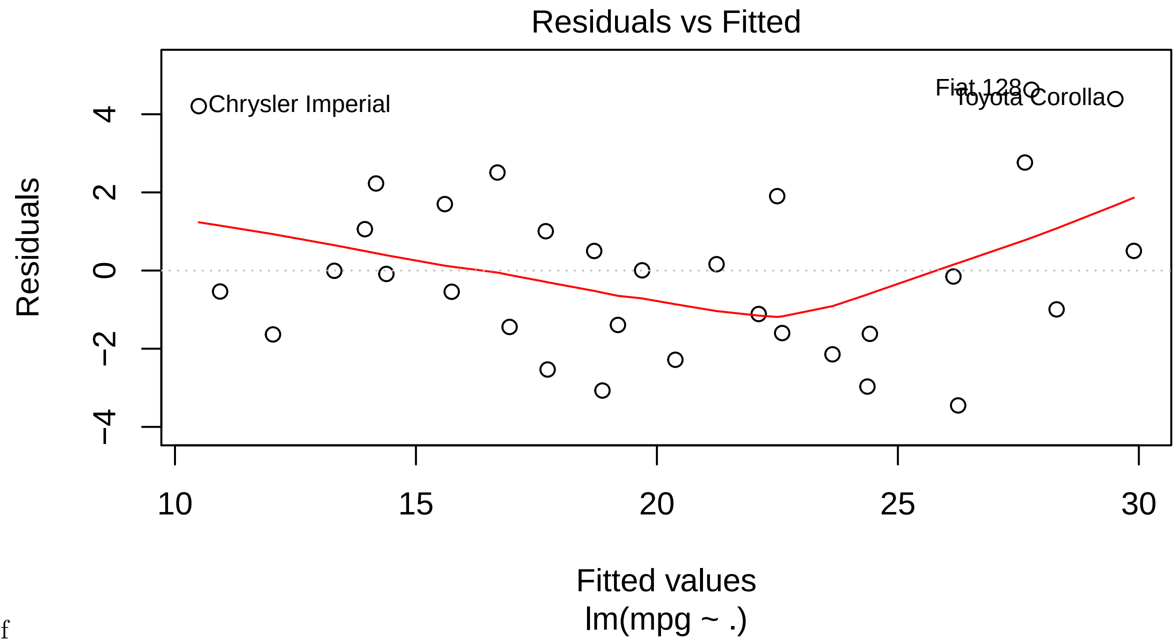
```
# Boxplot mpg ~ am
data <- data.frame(mpg=1:nrow(mtcars), am=1:nrow(mtcars))
data[,1] <- cbind(mtcars$mpg); data[,2] <- cbind(mtcars$am)
split <- split(data, data$am, drop=F)
mpg_am1 <- split(data, data$am, drop=F)[[2]][1]; mpg_am0 <- split(data, data$am, drop=F)[[1]][1]
frame <- data.frame(x=mpg_am0, y=c(mpg_am1, rep(NA,6), recursive =T)); names(frame) <- c("mpg_am0", "mpg
boxplot(frame)
```



here1.pdf

```
# The boxplot shows that manual transmission (mpg_am1) tends to achieve higher mpg values (i.e. is bett


# Initial Residual Plot
fit <- lm(mpg ~ ., data = mtcars) # using all variables (mpg ~ .)
plot(fit,1) # residual plot
```



here2.pdf

```r
cbind(sort(summary(fit)$coef[,4]))
```

```
##                [,1]
## wt          0.06325
## am          0.23399
## qsec        0.27394
## hp          0.33496
## disp        0.46349
## (Intercept) 0.51812
## drat        0.63528
## gear        0.66521
## carb        0.81218
## vs          0.88142
## cyl         0.91609
```

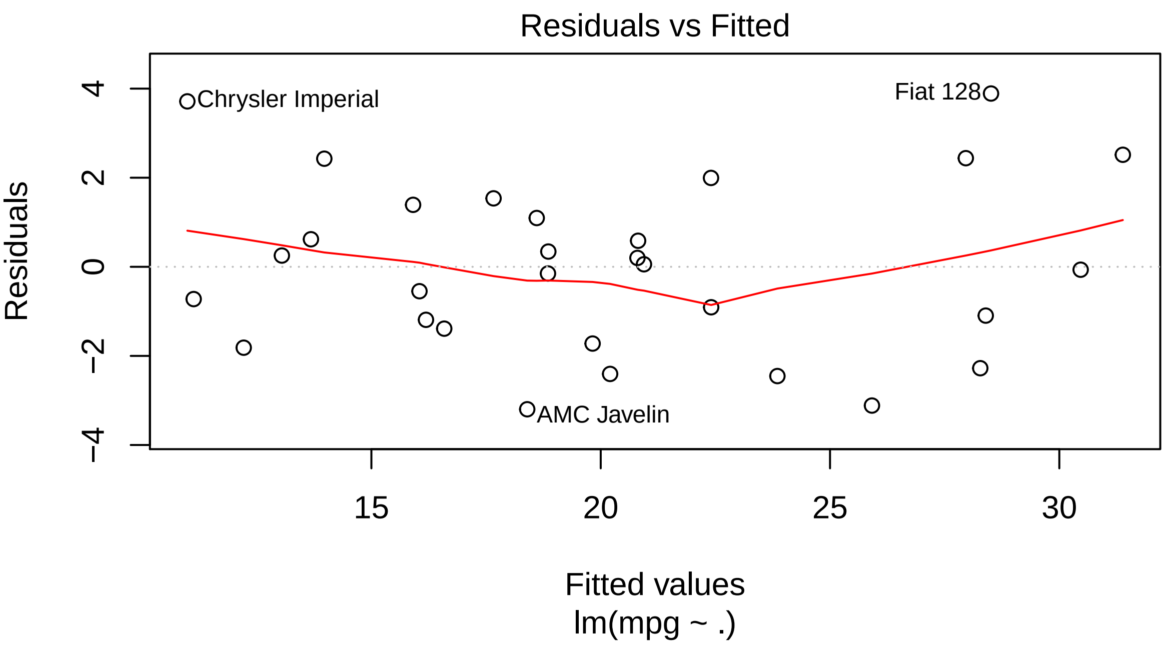
```r
## Diagnostics
# hatvalue
hats <- cbind(round(hatvalues(fit),3))
boundaries <- mean(hats) + c(-1,1) * 2 * sd(hats) # creating 95% confidence interval in which all value
hatvalues <- hats > boundaries[2] # showing where 95% confidence interval was surpassed
head(hatvalues)
```

```
##                    [,1]
## Mazda RX4         FALSE
## Mazda RX4 Wag     FALSE
## Datsun 710        FALSE
## Hornet 4 Drive    FALSE
## Hornet Sportabout FALSE
## Valiant           FALSE
```

```r
# dfbetas
dfs <- round(dfbetas(fit),3) # creating dfbetas
dfs <- dfs[,-1] # deleting intercept
boundaries <- cbind(apply(dfs, 2, mean)) + c(-1,1) * 2 * cbind(apply(dfs, 2, sd)) # creating 95% confid
dfbetas <- dfs > boundaries[2] # showing where 95% confidence interval was surpassed
head(dfbetas[,-3]) # -3 to take out hp as there was only FALSE in the dataset and to reduce the length
```

```
##                     cyl  disp  drat    wt  qsec    vs    am  gear  carb
## Mazda RX4         FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
## Mazda RX4 Wag     FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
## Datsun 710        FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE  TRUE
## Hornet 4 Drive    FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
## Hornet Sportabout FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
## Valiant           FALSE FALSE  TRUE FALSE FALSE FALSE FALSE FALSE FALSE
```

```r
# Second Residual Plot
mtcars2 <- mtcars[-29,]
mtcars2 <- mtcars2[-9,]
fit2 <- lm(mpg ~ ., data = mtcars2) # using all variables (mpg ~ .)
plot(fit2,1) # residual plot
```



here3.pdf

```
# Comparison of Coefficients for Model 1 (fit) and Model 2 (fit2)
data <- data.frame(names_fit=1:11, Significance_fit=1:11, names_fit2=1:11, Significance_fit2=1:11)
data[,1] <- row.names(cbind(sort(summary(fit)$coef[,4])))
data[,2] <- cbind(sort(summary(fit)$coef[,4]))
data[,3] <- row.names(cbind(sort(summary(fit2)$coef[,4])))
data[,4] <- cbind(sort(summary(fit2)$coef[,4]))
data
```

```
##      names_fit Significance_fit  names_fit2 Significance_fit2
## 1           wt          0.06325          wt           0.03123
## 2           am          0.23399        gear           0.04187
## 3         qsec          0.27394        qsec           0.05233
## 4           hp          0.33496        drat           0.07557
## 5         disp          0.46349 (Intercept)           0.15113
## 6  (Intercept)          0.51812        carb           0.20672
## 7         drat          0.63528        disp           0.26865
## 8         gear          0.66521         cyl           0.28941
## 9         carb          0.81218          am           0.36695
## 10          vs          0.88142          vs           0.70121
## 11         cyl          0.91609          hp           0.91413
```

```
# Quantifying the Change
mean(frame[,1]) + c(-1, 1) * 2 * sd(frame[,1]) # creating 95% confidence interval in which all values l
```

```
## [1]  9.479 24.815
```

```
mean(frame[,2], na.rm = T) + c(-1, 1) * 2 * sd(frame[,2], na.rm = T) # creating 95% confidence interval
```

```
## [1] 12.06 36.73
```