RegressionModelProject

Nima Taqidust

2/23/2020

```
library(dplyr)
```

```
## Warning: package 'dplyr' was built under R version 3.6.2
```

```
library(ggplot2)
```

Summary

First of all we can look at the data that we want to do the analysis for:

```
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 ...
```

```
summary(mtcars$mpg)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   10.40   15.43   19.20   20.09   22.80   33.90
```

Exploratory Analysis

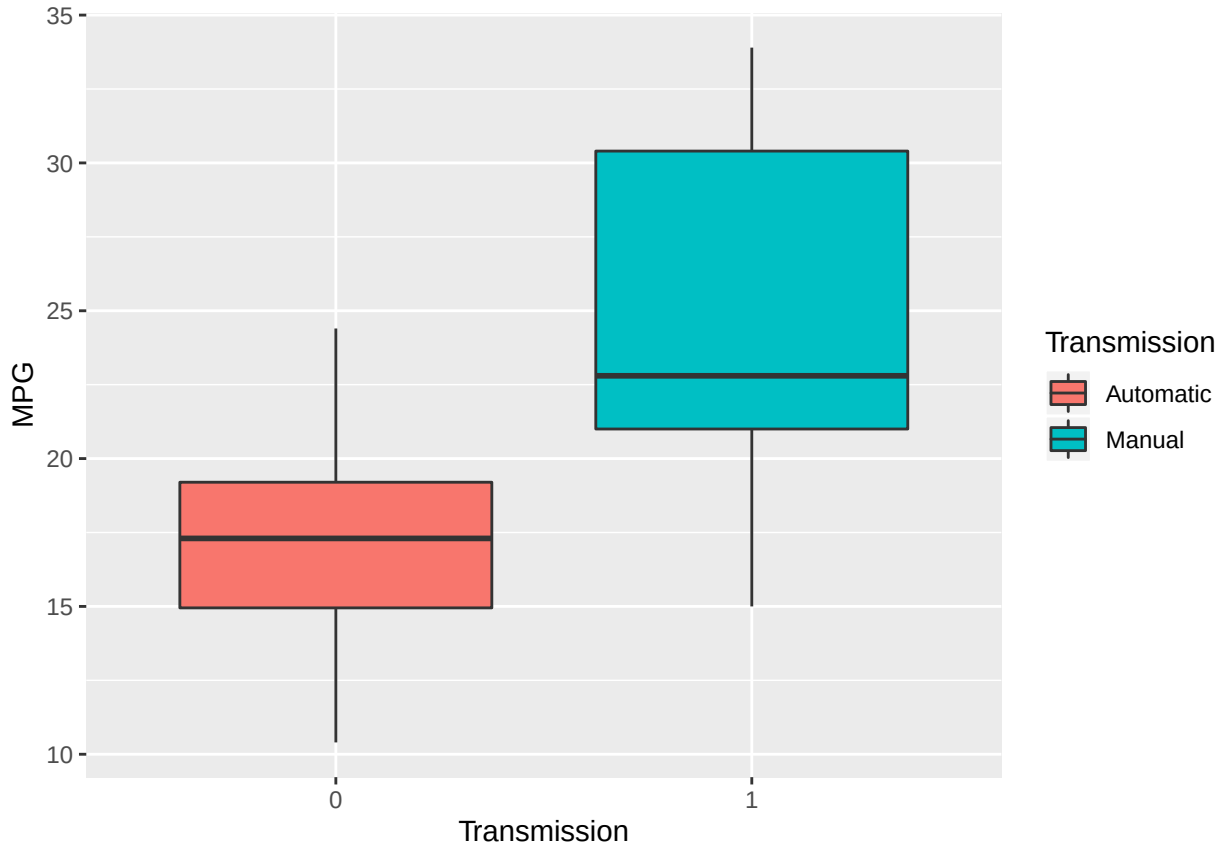
```
group_by(mtcars,am) %>% summarise(mean(mpg),sd(mpg)) %>% as.data.frame()
```

```
##   am mean(mpg)  sd(mpg)
## 1  0  17.14737 3.833966
## 2  1  24.39231 6.166504
```

We can see that there a distance betwwen the mean of two types of the variable “am”

Now we can get help from the boxplot to see the defference between the mgp for automaic and manual transmission:

```
ggplot(mtcars,aes(x=factor(mtcars$am),y=mtcars$mpg,fill=factor(mtcars$am))) +
  geom_boxplot()+
  scale_fill_discrete(name = "Transmission", labels = c("Automatic", "Manual"))+
  xlab("Transmission") + ylab("MPG")
```



We can split the data into two groups to run the T-test: automatic and manual transmission

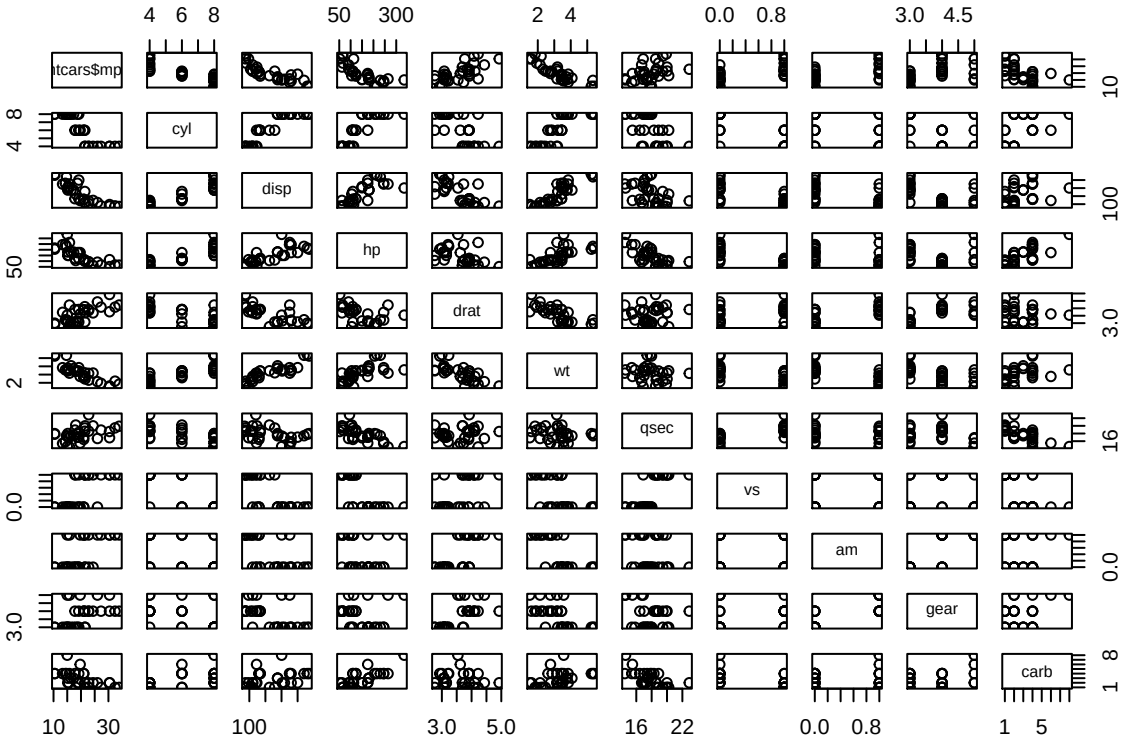
```
mtcarsAutomatic <- mtcars[mtcars$am==0,]

mtcarsManual <- mtcars[mtcars$am==1,]

t.test(mtcarsAutomatic$mpg,mtcarsManual$mpg,paired = FALSE)
```

```
##
##  Welch Two Sample t-test
##
## data:  mtcarsAutomatic$mpg and mtcarsManual$mpg
## 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 of x mean of y
##  17.14737  24.39231
```

As the T-test shows, the mpg variable for automatic and manual transmission are not the same.

Now we can see the relation between all the variables in the dataset:

```
pairs(mtcars$mpg ~ .,data=mtcars)
```



Now we see some strong relations between the variables, for a better understanding we can see the correlation between some variables:

```
correlation <- as.data.frame(cor(mtcars$cyl,mtcars$disp))
correlation <- as.data.frame(rbind(correlation, cor(mtcars$cyl,mtcars$hp)))
correlation <- as.data.frame(rbind(correlation,cor(mtcars$cyl,mtcars$wt)))
correlation <- as.data.frame(rbind(correlation,cor(mtcars$qsec,mtcars$drat)))
names(correlation) <- "Correlation"
correlation
```

```
##   Correlation
## 1  0.90203287
## 2  0.83244745
## 3  0.78249579
## 4  0.09120476
```

The results show a signinficant correlation between these variables so it is better not to use all these variables together.

Regression Models

First of all we can do a linear regression with the independent variable "am" and the dependent variable "mpg":

```
fit1 <- lm(mpg~factor(am),mtcars)
CoefFit1 <- summary(fit1)$coef
summary(fit1)
```

```
##
## Call:
## lm(formula = mpg ~ factor(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 ***
## factor(am)1    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
```

Note that R-squared is 0.3598

From the above we know that the variables "cyl","dis","hp" and "wt" have strong correlations so I added one of them into the model named "cyl":

```
fit2 <- lm(mpg~factor(am)+cyl-1,mtcars)
CoefFit2 <- summary(fit2)$coef
summary(fit2)
```

```
##
## Call:
## lm(formula = mpg ~ factor(am) + cyl - 1, data = mtcars)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -5.6856 -1.7172 -0.2657  1.8838  6.8144
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## factor(am)0  34.5224     2.6032  13.262 7.69e-14 ***
## factor(am)1  37.0895     2.0188  18.372  < 2e-16 ***
## cyl          -2.5010     0.3608  -6.931 1.28e-07 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
```

```
## Residual standard error: 3.059 on 29 degrees of freedom
## Multiple R-squared:  0.9807, Adjusted R-squared:  0.9787
## F-statistic: 490.6 on 3 and 29 DF,  p-value: < 2.2e-16
```

Now the R-squared is 0.9807 which is way better than before. We use tha anova function to see if the new variable have an improvement on the model or not:

```
anova(fit1,fit2)
```

```
## Analysis of Variance Table
##
## Model 1: mpg ~ factor(am)
## Model 2: mpg ~ factor(am) + cyl - 1
##   Res.Df    RSS Df Sum of Sq      F    Pr(>F)
## 1     30 720.90
## 2     29 271.36  1    449.53 48.041 1.285e-07 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

It had a significant effect on the model so we keep the "cyl".

Now we add the other variables step by step to see their impact on the model.

```
fit3 <- lm(mpg~factor(am)+cyl+drat-1,mtcars)
anova(fit1,fit2,fit3)
```

```
## Analysis of Variance Table
##
## Model 1: mpg ~ factor(am)
## Model 2: mpg ~ factor(am) + cyl - 1
## Model 3: mpg ~ factor(am) + cyl + drat - 1
##   Res.Df    RSS Df Sum of Sq       F    Pr(>F)
## 1     30 720.90
## 2     29 271.36  1    449.53 46.4518 2.101e-07 ***
## 3     28 270.97  1      0.39  0.0407    0.8415
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
fit4 <- lm(mpg~factor(am)+cyl+qsec-1,mtcars)
anova(fit1,fit2,fit4)
```

```
## Analysis of Variance Table
##
## Model 1: mpg ~ factor(am)
## Model 2: mpg ~ factor(am) + cyl - 1
## Model 3: mpg ~ factor(am) + cyl + qsec - 1
##   Res.Df    RSS Df Sum of Sq       F    Pr(>F)
## 1     30 720.90
## 2     29 271.36  1    449.53 47.0556 1.873e-07 ***
## 3     28 267.49  1      3.87  0.4052    0.5296
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
fit5 <- lm(mpg~factor(am)+cyl+factor(vs)-1,mtcars)
anova(fit1,fit2,fit5)
```

```
## Analysis of Variance Table
##
## Model 1: mpg ~ factor(am)
## Model 2: mpg ~ factor(am) + cyl - 1
## Model 3: mpg ~ factor(am) + cyl + factor(vs) - 1
##   Res.Df    RSS Df Sum of Sq       F    Pr(>F)
## 1     30 720.90
## 2     29 271.36  1    449.53 46.9619 1.906e-07 ***
## 3     28 268.02  1      3.34  0.3486    0.5596
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
fit6 <- lm(mpg~factor(am)+cyl+factor(gear)-1,mtcars)
anova(fit1,fit2,fit6)
```

```
## Analysis of Variance Table
##
## Model 1: mpg ~ factor(am)
## Model 2: mpg ~ factor(am) + cyl - 1
## Model 3: mpg ~ factor(am) + cyl + factor(gear) - 1
##   Res.Df    RSS Df Sum of Sq      F    Pr(>F)
## 1     30 720.90
## 2     29 271.36  1    449.53 46.291 2.623e-07 ***
## 3     27 262.20  2      9.17  0.472    0.6288
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
fit7 <- lm(mpg~factor(am)+cyl+factor(carb)-1,mtcars)
anova(fit1,fit2,fit7)
```

```
## Analysis of Variance Table
##
## Model 1: mpg ~ factor(am)
## Model 2: mpg ~ factor(am) + cyl - 1
## Model 3: mpg ~ factor(am) + cyl + factor(carb) - 1
##   Res.Df    RSS Df Sum of Sq       F    Pr(>F)
## 1     30 720.90
## 2     29 271.36  1    449.53 54.6240 1.245e-07 ***
## 3     24 197.51  5     73.85  1.7948    0.1521
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

We can see that none of the above variables do not have a significant impact on the model so we ommit them all.

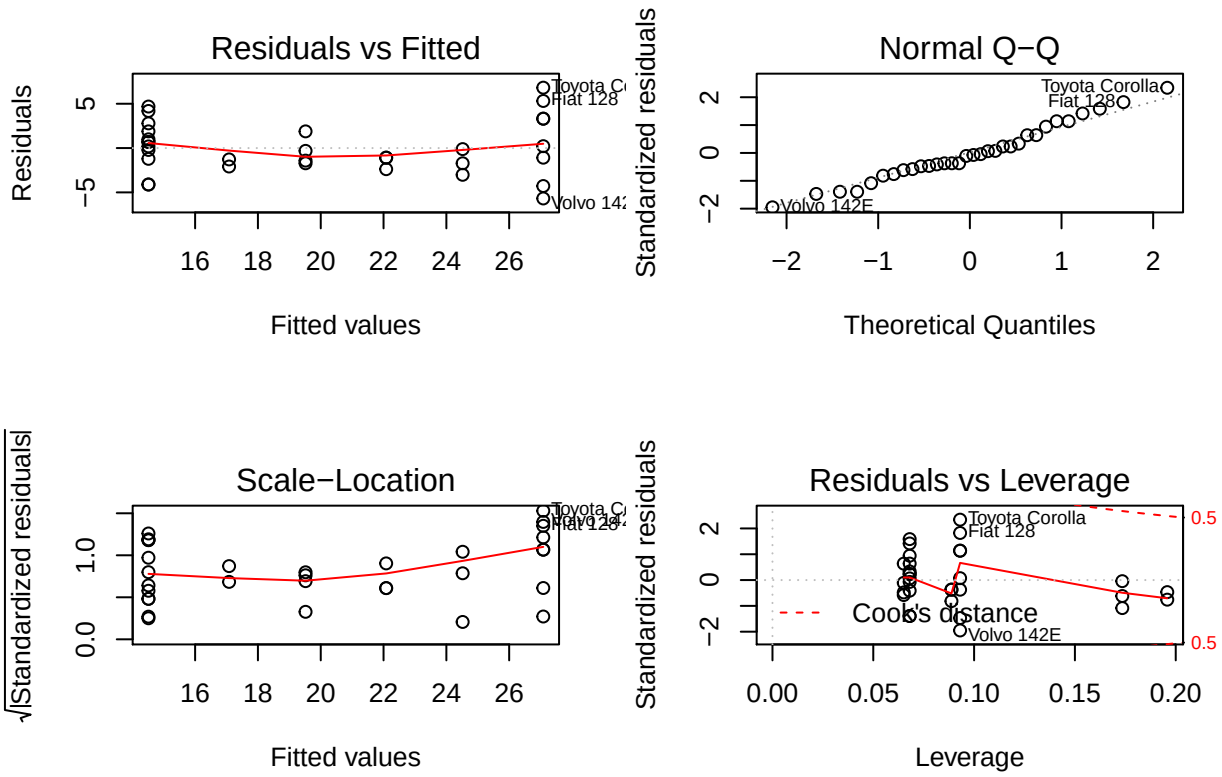
So our model is based on the variables “cyl” and “am”.

Appendix

```
FittedMPGs <- as.integer(predict(fit2))
mtcars <- mutate(mtcars,FittedMPGs)
mtcars
```

```
##     mpg cyl  disp  hp drat    wt  qsec vs am gear carb FittedMPGs
## 1  21.0   6 160.0 110 3.90 2.620 16.46  0  1    4    4         22
## 2  21.0   6 160.0 110 3.90 2.875 17.02  0  1    4    4         22
## 3  22.8   4 108.0  93 3.85 2.320 18.61  1  1    4    1         27
## 4  21.4   6 258.0 110 3.08 3.215 19.44  1  0    3    1         19
## 5  18.7   8 360.0 175 3.15 3.440 17.02  0  0    3    2         14
## 6  18.1   6 225.0 105 2.76 3.460 20.22  1  0    3    1         19
## 7  14.3   8 360.0 245 3.21 3.570 15.84  0  0    3    4         14
## 8  24.4   4 146.7  62 3.69 3.190 20.00  1  0    4    2         24
## 9  22.8   4 140.8  95 3.92 3.150 22.90  1  0    4    2         24
## 10 19.2   6 167.6 123 3.92 3.440 18.30  1  0    4    4         19
## 11 17.8   6 167.6 123 3.92 3.440 18.90  1  0    4    4         19
## 12 16.4   8 275.8 180 3.07 4.070 17.40  0  0    3    3         14
## 13 17.3   8 275.8 180 3.07 3.730 17.60  0  0    3    3         14
## 14 15.2   8 275.8 180 3.07 3.780 18.00  0  0    3    3         14
## 15 10.4   8 472.0 205 2.93 5.250 17.98  0  0    3    4         14
## 16 10.4   8 460.0 215 3.00 5.424 17.82  0  0    3    4         14
## 17 14.7   8 440.0 230 3.23 5.345 17.42  0  0    3    4         14
## 18 32.4   4  78.7  66 4.08 2.200 19.47  1  1    4    1         27
## 19 30.4   4  75.7  52 4.93 1.615 18.52  1  1    4    2         27
## 20 33.9   4  71.1  65 4.22 1.835 19.90  1  1    4    1         27
## 21 21.5   4 120.1  97 3.70 2.465 20.01  1  0    3    1         24
## 22 15.5   8 318.0 150 2.76 3.520 16.87  0  0    3    2         14
## 23 15.2   8 304.0 150 3.15 3.435 17.30  0  0    3    2         14
## 24 13.3   8 350.0 245 3.73 3.840 15.41  0  0    3    4         14
## 25 19.2   8 400.0 175 3.08 3.845 17.05  0  0    3    2         14
## 26 27.3   4  79.0  66 4.08 1.935 18.90  1  1    4    1         27
## 27 26.0   4 120.3  91 4.43 2.140 16.70  0  1    5    2         27
## 28 30.4   4  95.1 113 3.77 1.513 16.90  1  1    5    2         27
## 29 15.8   8 351.0 264 4.22 3.170 14.50  0  1    5    4         17
## 30 19.7   6 145.0 175 3.62 2.770 15.50  0  1    5    6         22
## 31 15.0   8 301.0 335 3.54 3.570 14.60  0  1    5    8         17
## 32 21.4   4 121.0 109 4.11 2.780 18.60  1  1    4    2         27
```

```
  par(mfrow=c(2,2))
  plot(fit2)
```


Residuals vs Fitted
Residuals
Fitted values
Toyota C
Fiat 128
Volvo 142
Normal Q−Q
Standardized residuals
Theoretical Quantiles
Toyota Corolla
Fiat 128
Volvo 142E
Scale−Location
√|Standardized residuals|
Fitted values
Residuals vs Leverage
Standardized residuals
Leverage
Cook's distance
Toyota Corolla
Fiat 128
Volvo 142E
0.5