STATS 101A Section 1B Kaggle Projectt

Karina Santoso, UID- 805291900

3/22/2021

```r
library(ggplot2)
library(GGally)
```

```
## Registered S3 method overwritten by 'GGally':
##   method from
##   +.gg   ggplot2
```

```r
library(corrplot)
```

```
## corrplot 0.84 loaded
```

```r
library(car)
```

```
## Loading required package: carData
```

```r
library(leaps)
cars <- read.csv("carsTrain.csv")
```

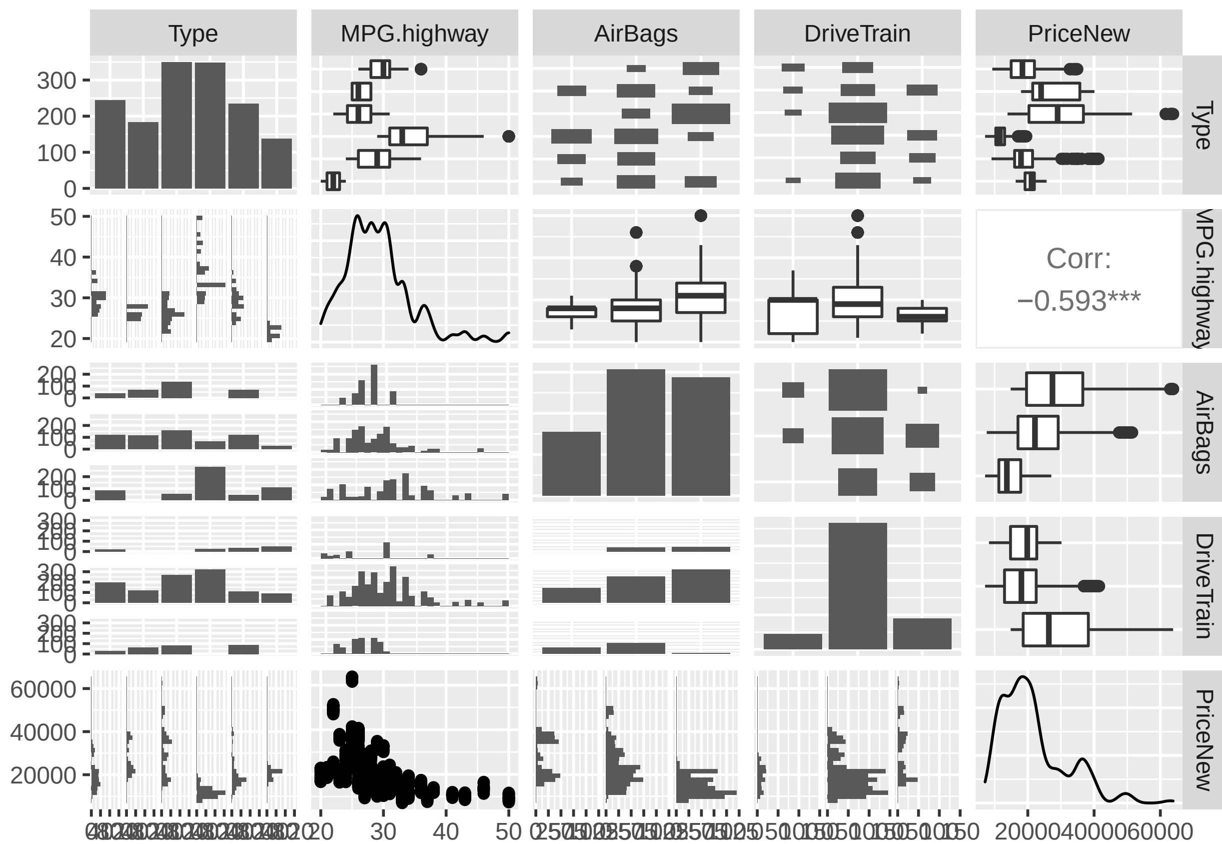
```r
head(cars)
```

```
##   Ob Manufacturer       Model    Type MPG.highway            AirBags DriveTrain
## 1  1   Volkswagen     Corrado  Sporty          25               None      Front
## 2  2        Buick     Riviera Midsize          27        Driver only      Front
## 3  3     Infiniti         Q45 Midsize          22        Driver only       Rear
## 4  4        Mazda         626 Compact          34        Driver only      Front
## 5  5    Chevrolet    Corvette  Sporty          25        Driver only       Rear
## 6  6      Lincoln Continental Midsize          26 Driver & Passenger      Front
##   Cylinders EngineSize Horsepower  RPM Rev.per.mile Man.trans.avail
## 1         6        2.8        178 5800         2385             Yes
## 2         6        3.8        170 4800         1690              No
## 3         8        4.5        278 6000         1955              No
## 4         4        2.5        164 5600         2505             Yes
## 5         8        5.7        300 5000         1450             Yes
## 6         6        3.8        160 4400         1835              No
##   Fuel.tank.capacity Passengers Length Wheelbase Width Turn.circle
## 1               18.5          4    159        97    66          36
## 2               18.8          5    198       108    73          41
## 3               22.5          5    200       113    72          42
## 4               15.5          5    184       103    69          40
## 5               20.0          2    179        96    74          43
## 6               18.4          6    205       109    73          42
##   Rear.seat.room Luggage.room Weight  Origin               Make PriceNew
## 1           26.0           15   2810 non-USA  Volkswagen Corrado 24377.37
## 2           26.5           14   3495     USA      Buick Riviera 28625.33
## 3           29.0           15   4000 non-USA       Infiniti Q45 50390.28
```

```
## 4           29.5           14    2970 non-USA           Mazda 626 18868.78
## 5           24.5           15    3380     USA  Chevrolet Corvette 38989.67
## 6           30.0           19    3695     USA Lincoln Continental 36427.55
```

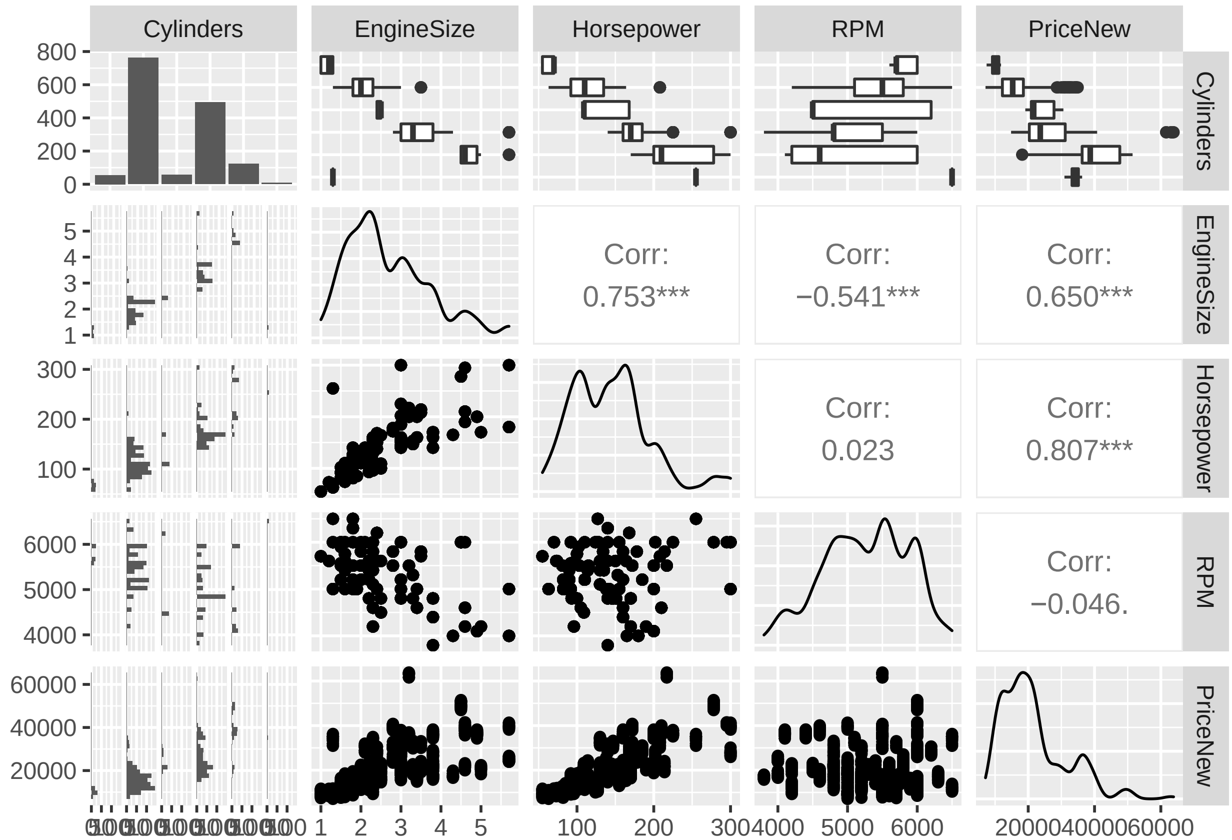
```r
ggpairs(cars, columns = c(4, 5, 6, 7, 25))
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



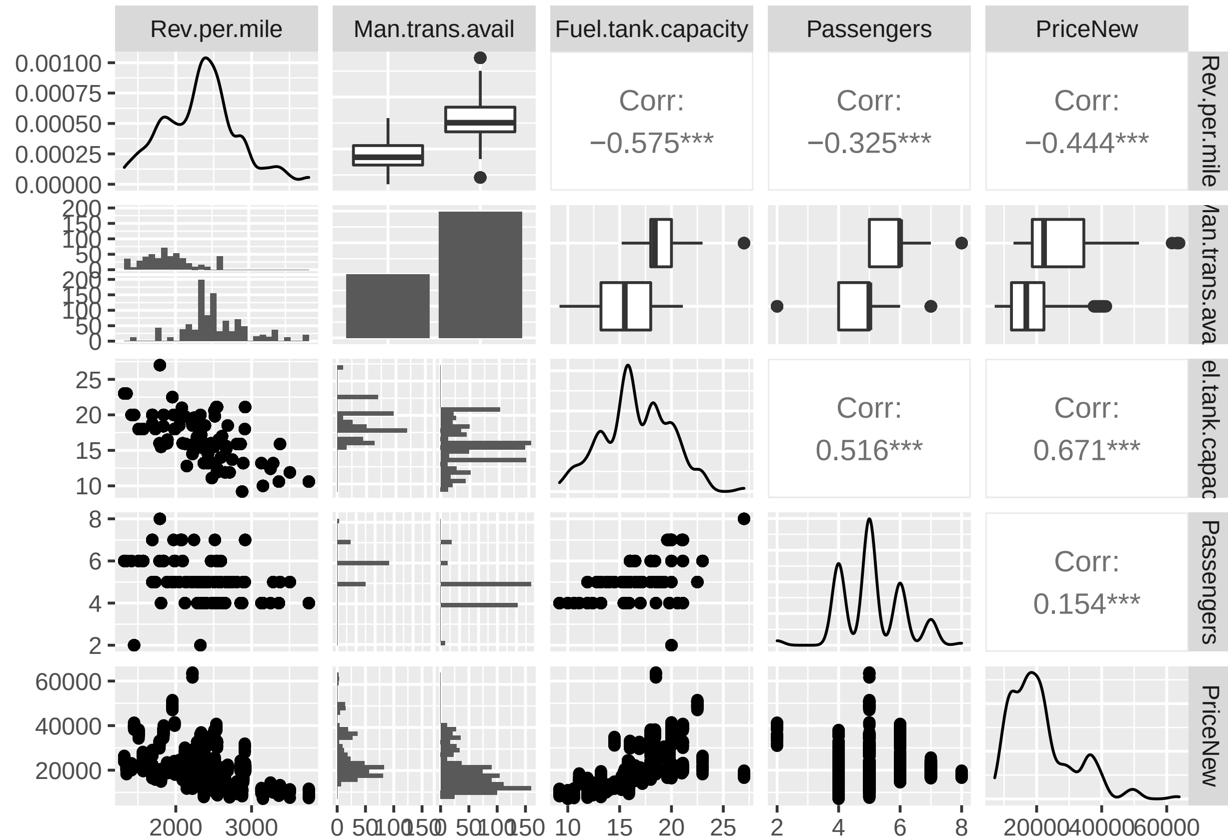

```r
ggpairs(cars, columns = c(8, 9, 10, 11, 25))
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```

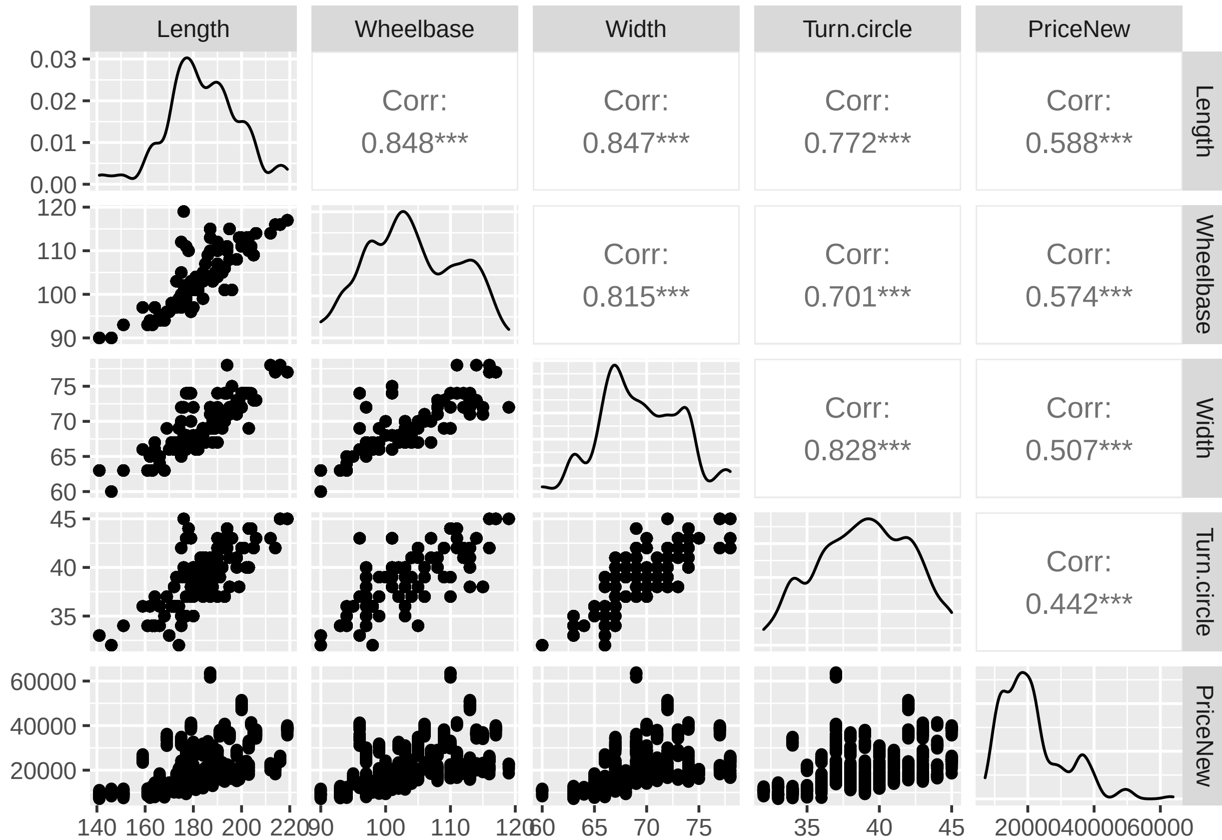



```r
ggpairs(cars, columns = c(12, 13, 14, 15, 25))
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```

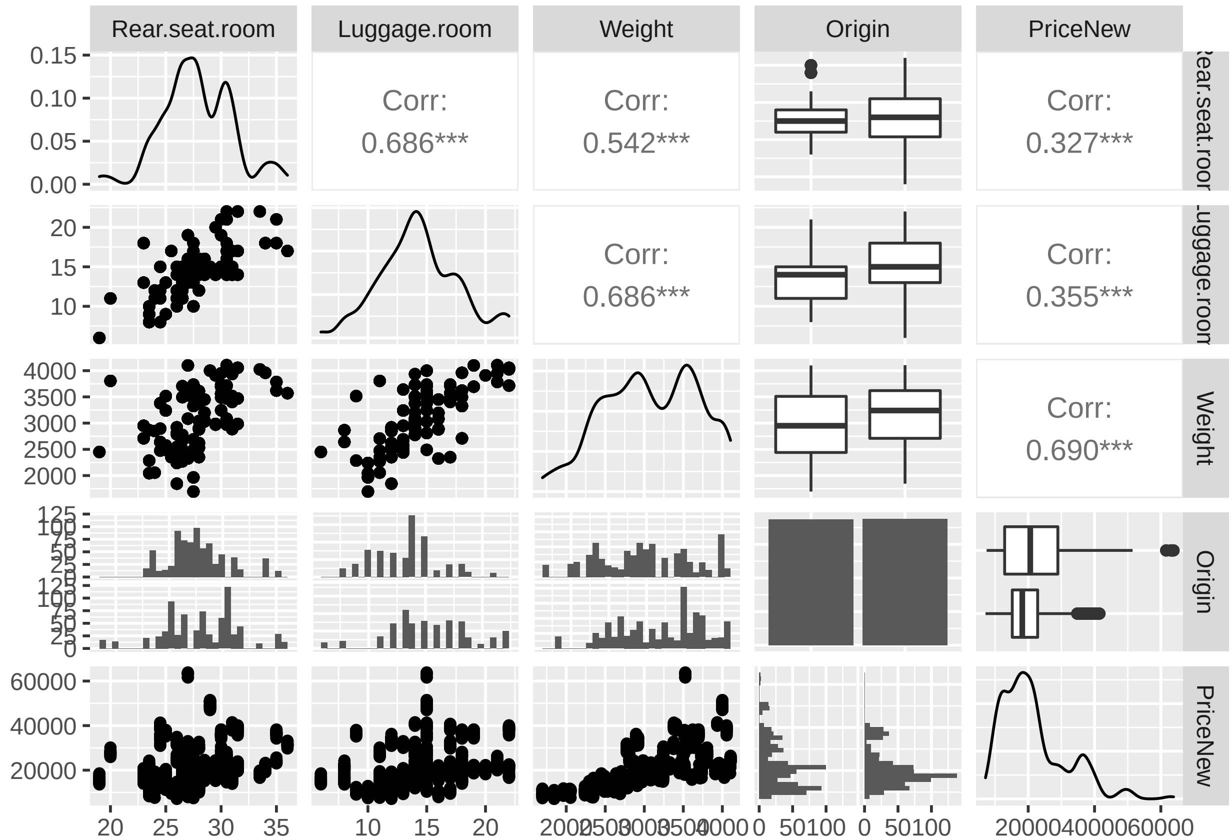



```r
ggpairs(cars, columns = c(16, 17, 18, 19, 25))
```




```r
ggpairs(cars, columns = c(20, 21, 22, 23, 25))
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```




```
cars$Cylinders2 <- ifelse(cars$Cylinders == 8 | cars$Cylinders == "rotary", 1, 2)
table(factor(cars$Cylinders2))
```

```
##
##    1    2
##  132 1368
```

```
cars$Engine3 <- ifelse(cars$EngineSize == 1.5, 1, ifelse(cars$EngineSize == 2.1, 2,
    ifelse(cars$EngineSize == 2.8, 3, ifelse(cars$EngineSize == 3.2, 4, ifelse(cars$EngineSize == 4.9, 5
    ifelse(cars$EngineSize == 5, 6, 7))))))
cars$Man <- ifelse(cars$Manufacturer == "Mercedes-Benz", 1, ifelse(cars$Manufacturer == "Mercury",2,
    ifelse(cars$Manufacturer == "Lexus", 3, ifelse(cars$Model == "Crown_Victoria", 4,
    ifelse(cars$Model == "Imperial", 5, ifelse(cars$Model == "Continental", 6,
    ifelse(cars$Manufacturer == "Audi", 7, ifelse(cars$Manufacturer == "Volvo", 8, 9))))))))
```

```
library(caTools)
set.seed(123456)
Cars.split = sample.split(as.numeric(rownames(cars)), SplitRatio= 0.7)
train.Cars= subset(cars, Cars.split==TRUE)
test.Cars= subset(cars, Cars.split==FALSE)



m0 <- lm((PriceNew)^(1/3)~Type+Weight+factor(Engine3):Horsepower+factor(Man)+ factor(Cylinders2), data =

summary(m0)
```

```
##
## Call:
## lm(formula = (PriceNew)^(1/3) ~ Type + Weight + factor(Engine3):Horsepower +
##     factor(Man) + factor(Cylinders2), data = train.Cars)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -2.9058 -0.6519  0.0275  0.6352  2.7448
##
## Coefficients:
##                              Estimate Std. Error t value Pr(>|t|)
## (Intercept)                 2.747e+01  5.404e-01  50.826  < 2e-16 ***
## TypeLarge                   2.001e-01  1.740e-01   1.150 0.250282
## TypeMidsize                 1.068e+00  1.278e-01   8.361  < 2e-16 ***
## TypeSmall                  -6.992e-01  1.375e-01  -5.087 4.33e-07 ***
## TypeSporty                  4.771e-01  1.280e-01   3.727 0.000204 ***
## TypeVan                    -6.295e-01  2.063e-01  -3.052 0.002334 **
## Weight                      2.346e-03  1.776e-04  13.205  < 2e-16 ***
## factor(Man)2               -7.601e+00  3.557e-01 -21.368  < 2e-16 ***
## factor(Man)3               -3.456e+00  3.574e-01  -9.672  < 2e-16 ***
## factor(Man)4               -1.205e+01  5.751e-01 -20.953  < 2e-16 ***
## factor(Man)5               -2.473e+00  4.720e-01  -5.239 1.96e-07 ***
## factor(Man)6               -2.390e+00  3.534e-01  -6.765 2.24e-11 ***
## factor(Man)7               -3.875e+00  5.370e-01  -7.217 1.03e-12 ***
## factor(Man)8               -3.632e+00  3.565e-01 -10.189  < 2e-16 ***
## factor(Man)9               -6.559e+00  2.652e-01 -24.731  < 2e-16 ***
## factor(Cylinders2)2        -4.382e+00  1.932e-01 -22.679  < 2e-16 ***
## factor(Engine3)1:Horsepower  1.505e-02  2.252e-03   6.680 3.90e-11 ***
## factor(Engine3)2:Horsepower  5.761e-02  3.013e-03  19.120  < 2e-16 ***
## factor(Engine3)3:Horsepower  3.021e-02  2.734e-03  11.051  < 2e-16 ***
## factor(Engine3)4:Horsepower  3.468e-02  1.748e-03  19.840  < 2e-16 ***
## factor(Engine3)5:Horsepower  1.774e-02  2.313e-03   7.672 3.94e-14 ***
## factor(Engine3)6:Horsepower -1.690e-02  3.119e-03  -5.420 7.41e-08 ***
## factor(Engine3)7:Horsepower  1.655e-02  1.502e-03  11.013  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.053 on 1027 degrees of freedom
## Multiple R-squared:  0.9305, Adjusted R-squared:  0.929
## F-statistic: 624.6 on 22 and 1027 DF,  p-value: < 2.2e-16
```

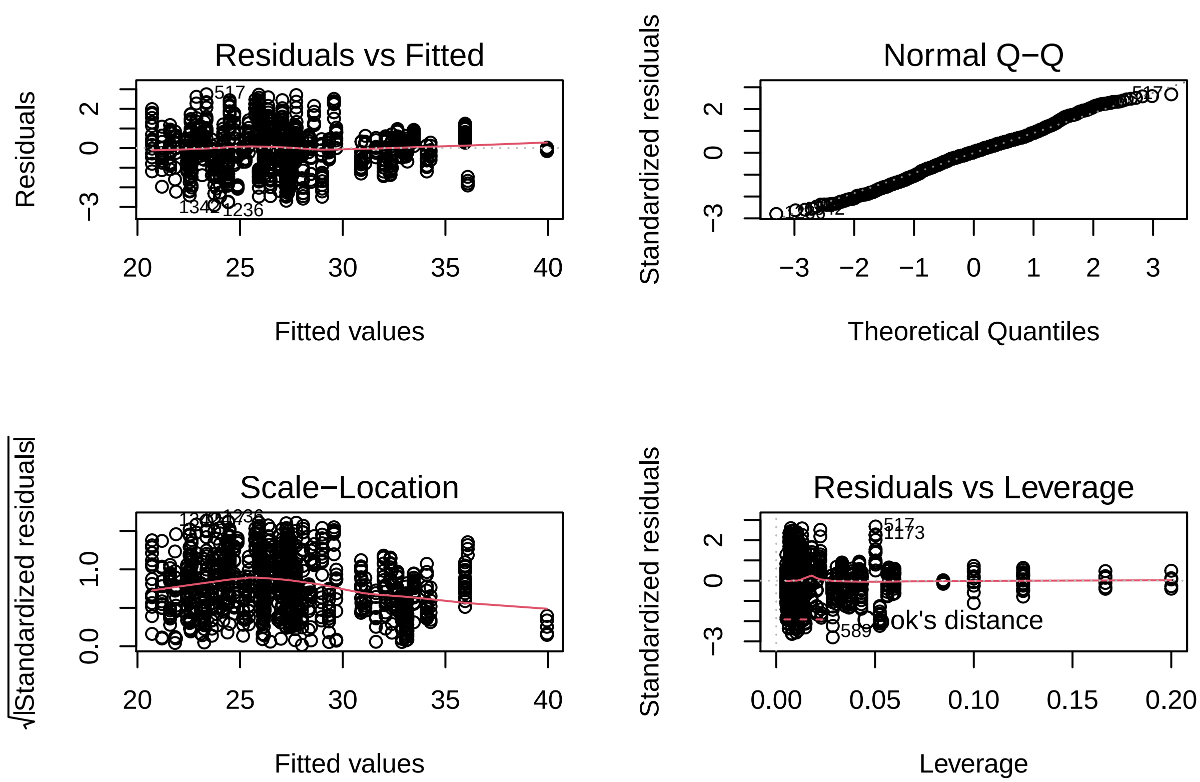
```
anova(m0)
```

```
## Analysis of Variance Table
##
## Response: (PriceNew)^(1/3)
##                              Df Sum Sq Mean Sq F value    Pr(>F)
## Type                          5 8716.5 1743.30 1572.27 < 2.2e-16 ***
## Weight                        1 2662.6 2662.60 2401.37 < 2.2e-16 ***
## factor(Man)                   8 1593.0  199.13  179.59 < 2.2e-16 ***
## factor(Cylinders2)            1 1061.4 1061.36  957.23 < 2.2e-16 ***
## factor(Engine3):Horsepower    7 1202.3  171.76  154.90 < 2.2e-16 ***
## Residuals                  1027 1138.7    1.11
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
vif(m0)
```

```
##                                 GVIF Df GVIF^(1/(2*Df))
## Type                        17.67408  5        1.332704
## Weight                      10.74694  1        3.278252
## factor(Man)                 14.73999  8        1.183126
## factor(Cylinders2)           2.74317  1        1.656252
## factor(Engine3):Horsepower 72.66560  7        1.358164
```

```r
par(mfrow=c(2,2))
plot(m0)
```




```r
AIC(m0)
```

```
## [1] 3112.94
```

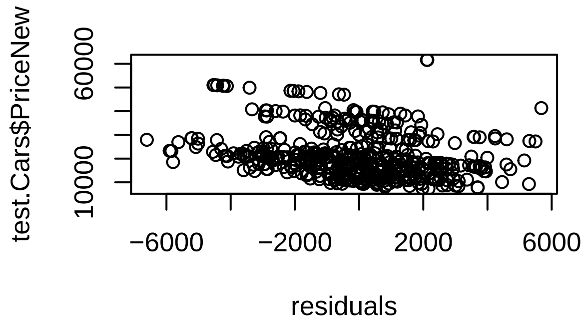
```r
extractAIC(m0,k=log(length(cars$Ob)))
```

```
## [1]  23.0000 253.3727
```

```r
predictions <- predict(m0, newdata = test.Cars)
predictions <- predictions^3

residuals <- (predictions - test.Cars$PriceNew)
plot(residuals, test.Cars$PriceNew)

bigerror <- which(abs(residuals) > 6000)
```



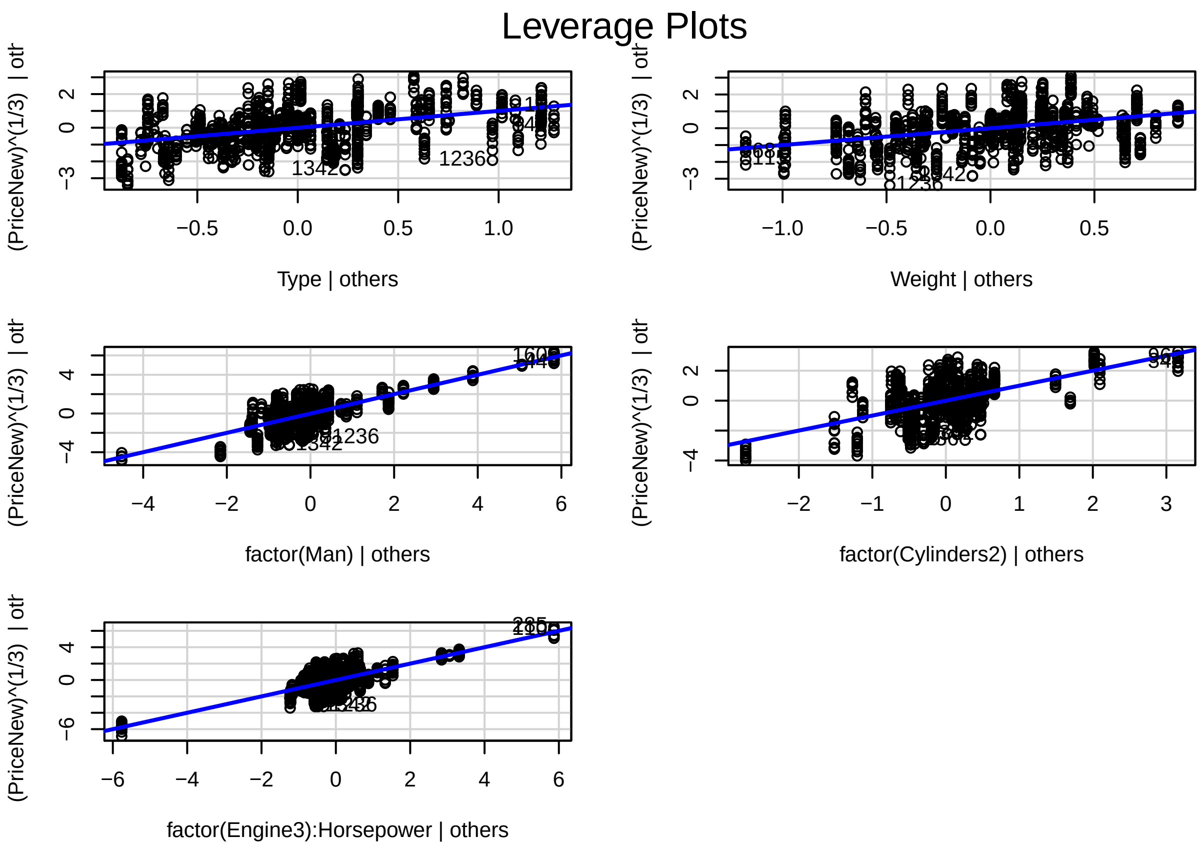

```
test.Cars[bigerror,]
```

```
##      Ob Manufacturer      Model  Type MPG.highway            AirBags DriveTrain
## 653 653      Pontiac Bonneville Large          28 Driver & Passenger      Front
##     Cylinders EngineSize Horsepower  RPM Rev.per.mile Man.trans.avail
## 653         6        3.8        170 4800         1565              No
##     Fuel.tank.capacity Passengers Length Wheelbase Width Turn.circle
## 653                 18          6    177       111    74          43
##     Rear.seat.room Luggage.room Weight Origin               Make PriceNew
## 653           30.5           18   3495    USA Pontiac Bonneville 27948.05
##     Cylinders2 Engine3 Man
## 653          2       7   9
```

```
predictions[bigerror]
```

```
##      653
## 21336.87
```

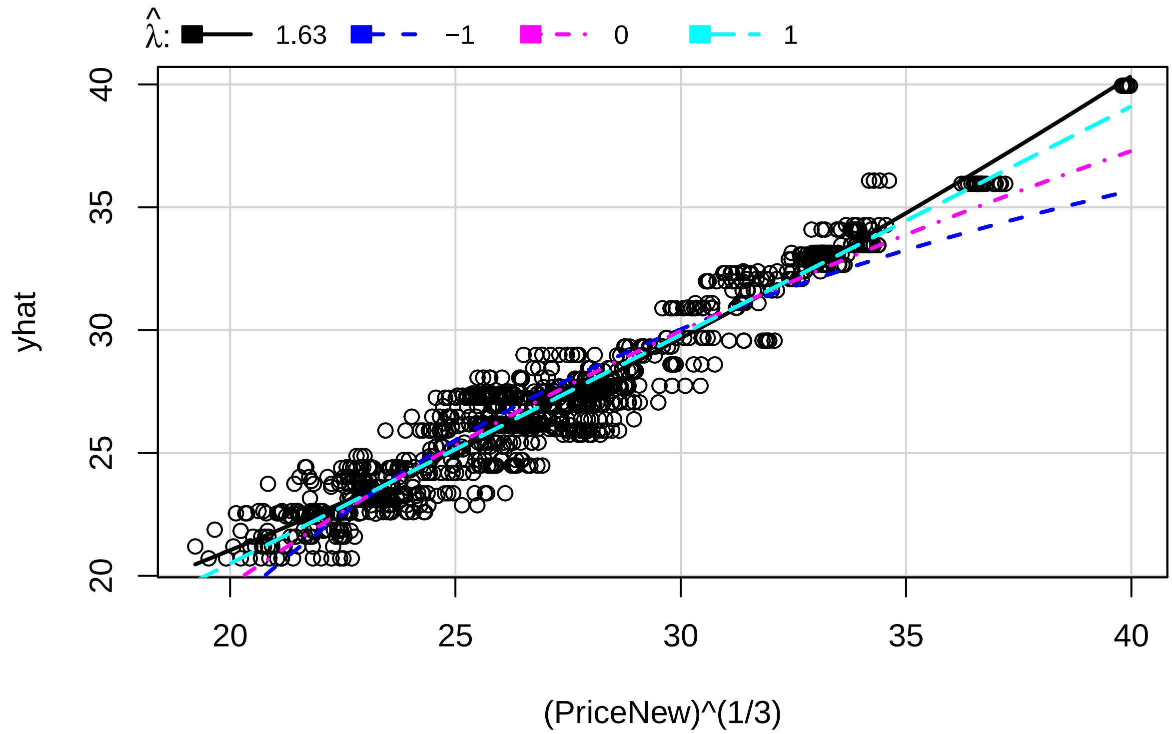
```
leveragePlots(m0)
```




```r
#mmps(m0)
powerTransform(cbind(cars$Horsepower, cars$Fuel.tank.capacity, cars$Width, cars$Weight))
```

```
## Estimated transformation parameters
##          Y1          Y2          Y3          Y4
## -0.40545433  0.08131719 -2.86441413  0.17626690
```

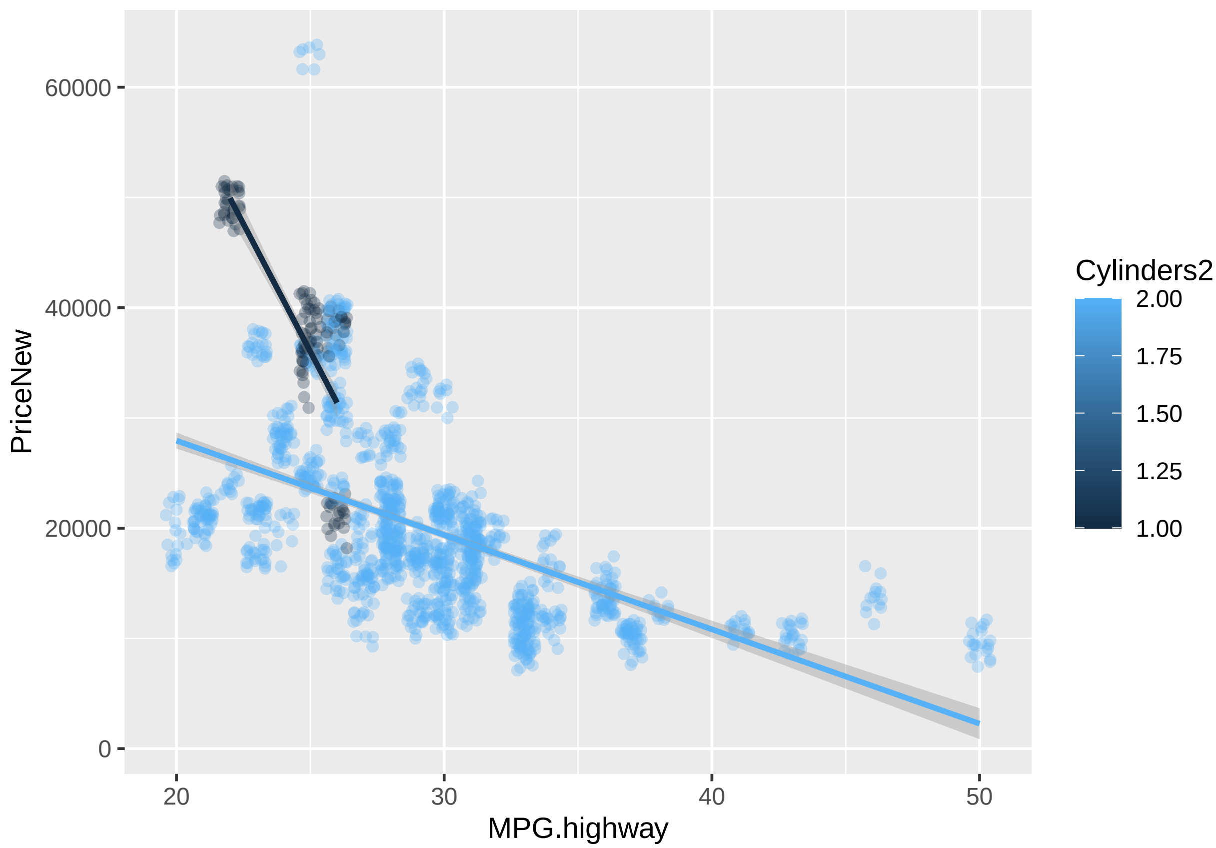
```r
inverseResponsePlot(m0)
```




```
##      lambda      RSS
## 1  1.630472 1017.351
## 2 -1.000000 1724.314
## 3  0.000000 1295.873
## 4  1.000000 1059.529
```

```
ggplot(cars, aes(x=MPG.highway, y=PriceNew, group=Cylinders2, color=Cylinders2))+geom_point(alpha = 0.3
```

```
## `geom_smooth()` using formula 'y ~ x'
```




```r
test <- read.csv("carsTestNoY.csv")
test$Engine <- ifelse(test$EngineSize < 2.8 | test$EngineSize == 3 | test$EngineSize == 3.5, 1,
                      ifelse(test$EngineSize == 3.2 | test$EngineSize == 4.5, 2, test$EngineSize))
test$Cylinders2 <- ifelse(test$Cylinders == 8 | test$Cylinders == "rotary", 1, 2)

test$Man <- ifelse(test$Manufacturer == "Mercedes-Benz", 1, ifelse(test$Manufacturer == "Mercury",2, if
test$Engine3 <- ifelse(test$EngineSize == 1.5, 1, ifelse(test$EngineSize == 2.1, 2, ifelse(test$EngineS
predictions <- predict(m0, newdata = test)
predictions <- predictions^3
submission <- data.frame(Ob = 1:500, PriceNew = predictions)
write.csv(submission, file = "~/Desktop/predictions.csv", row.names = F)
```