stat 350 final project

GROUP 8: Xuefei Li, Kunpeng Wang, Wenzhao Wang, Mengqi Xie

2020/11/24

```
library(readr)
library(MASS)
library(stringr)
library(car)
```

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

```
library(StepReg)
library(ggplot2)
library(performance)
library(Metrics)
```

```
##
## Attaching package: 'Metrics'
```

```
## The following objects are masked from 'package:performance':
##
##     mse, rmse
```

Data Cleaning

```
# Read in the original data
data3 <- read_csv("Car details v3.csv")
```

```
## Parsed with column specification:
## cols(
##   name = col_character(),
##   year = col_double(),
##   selling_price = col_double(),
##   km_driven = col_double(),
##   fuel = col_character(),
##   seller_type = col_character(),
##   transmission = col_character(),
##   owner = col_character(),
##   mileage = col_character(),
##   engine = col_character(),
##   max_power = col_character(),
##   torque = col_character(),
##   seats = col_double()
## )

dim(data3) # 8128 13

## [1] 8128   13

# Only keep observations with complete information
Car_details_v3 <- data3[complete.cases(data3), ]
names(Car_details_v3) # "name" "year" "selling_price" "km_driven"
"fuel" "seller_type" "transmission"

##  [1] "name"          "year"          "selling_price" "km_driven"
##  [5] "fuel"          "seller_type"   "transmission"  "owner"
##  [9] "mileage"       "engine"        "max_power"     "torque"
## [13] "seats"

                      # "owner" "mileage" "engine" "max_power"
"torque" "seats"      13 predictors
dim(Car_details_v3) # 7906 13

## [1] 7906   13

# Print the original data
head(Car_details_v3)

## # A tibble: 6 x 13
##   name   year selling_price km_driven fuel  seller_type
transmission owner
##   <chr> <dbl>         <dbl>     <dbl> <chr> <chr>       <chr>
<chr>
## 1 Maru…  2014        450000    145500 Dies… Individual  Manual
Firs…
## 2 Skod…  2014        370000    120000 Dies… Individual  Manual
Seco…
## 3 Hond…  2006        158000    140000 Petr… Individual  Manual
Thir…
## 4 Hyun…  2010        225000    127000 Dies… Individual  Manual
Firs…
## 5 Maru…  2007        130000    120000 Petr… Individual  Manual
Firs…
## 6 Hyun…  2017        440000     45000 Petr… Individual  Manual
```

```
Firs…
## # … with 5 more variables: mileage <chr>, engine <chr>, max_power
<chr>,
## #   torque <chr>, seats <dbl>
```

```r
# Deal with qualitative variables: fuel, seller type, transmission,
and owner
Car_details_v3$fuel = as.factor(Car_details_v3$fuel)
Car_details_v3$seller_type = as.factor(Car_details_v3$seller_type)
Car_details_v3$transmission = as.factor(Car_details_v3$transmission)
Car_details_v3$owner = as.factor(Car_details_v3$owner)

# Split columns to be numerical part and unit part
Years = 2020 - Car_details_v3$year
Name = str_split_fixed(Car_details_v3$name, " ", 2)
Mileage = str_split_fixed(Car_details_v3$mileage, " ", 2)
Engine = str_split_fixed(Car_details_v3$engine, " ", 2)
Max_power = str_split_fixed(Car_details_v3$max_power, " ", 2)

# Strip off the unit part and keep the plain numerical part
sub_1 = cbind(Name, Years, Mileage, Engine, Max_power)
sub_2 = sub_1[,-c(2,5,7,9)]
car1 <- cbind(sub_2, Car_details_v3)

# Rename four columns and omit five duplicated columns to form "car"
colnames(car1)[which(names(car1) == "V1")] <- "Manufacturer"
colnames(car1)[which(names(car1) == "V3")] <- "Mileage"
colnames(car1)[which(names(car1) == "V4")] <- "Engine"
colnames(car1)[which(names(car1) == "V5")] <- "Max_power"
car <- subset(car1, select = -c(name, year, mileage, engine,
max_power))

# Find unique car manufacturers and categorize them into 5 categories
according to countries
unique(car$Manufacturer)
```

```
##  [1] Maruti        Skoda         Honda         Hyundai       Toyota
##  [6] Ford          Renault       Mahindra      Tata
Chevrolet
## [11] Datsun        Jeep          Mercedes-Benz Mitsubishi    Audi
## [16] Volkswagen    BMW           Nissan        Lexus         Jaguar
## [21] Land          MG            Volvo         Daewoo        Kia
## [26] Fiat          Force         Ambassador    Ashok         Isuzu
## [31] Opel
## 31 Levels: Ambassador Ashok Audi BMW Chevrolet Daewoo Datsun
Fiat ... Volvo
```

```
car$Manufacturer = as.character(car$Manufacturer)
car$Manufacturer[car$Manufacturer %in%

c("Maruti","Honda","Toyota","Mitsubishi","Nissan","Lexus","Isuzu")] <-
"Japan"
car$Manufacturer[car$Manufacturer %in%
                 c("Skoda","Mercedes-Benz","Audi","Volkswagen","BMW")]
<- "Germany"
car$Manufacturer[car$Manufacturer %in%
                 c("Renault", "Land", "MG", "Volvo", "Fiat",
"Opel","Jaguar")] <- "other Europe"
car$Manufacturer[car$Manufacturer %in%
                 c( "Hyudai", "Mahindra", "Tata", "Datsun", "Daewoo",
"Kia", "Force", "Ashok","Hyundai")] <- "other Asia"
car$Manufacturer[car$Manufacturer %in%
c("Ambassador","Ford","Chevrolet","Jeep")] <- "US"

# Change type character to be type double
car$Manufacturer = as.factor(car$Manufacturer)
car$Years = as.double(car$Years)
car$Mileage = as.double(car$Mileage)
car$Engine = as.double(car$Engine)
car$Max_power = as.double(car$Max_power)

# Print the revised data:
# Double type: years, mileage, engine, max power, selling price, km
driven, seats
# Factor type: manufacturer, fuel, seller type, transmission, owner
# Character type: torque (will not be analyzed)
head(car)
```

```
##   Manufacturer Years Mileage Engine Max_power selling_price
km_driven   fuel
## 1        Japan    24     324     14       243        450000
145500 Diesel
```

```
## 2      Germany    24     274     37        14        370000
120000 Diesel
## 3        Japan     7     174     36       252        158000
140000 Petrol
## 4   other Asia     3     316     25       296        225000
127000 Diesel
## 5        Japan     6     132     15       287        130000
120000 Petrol
## 6   other Asia    21     237     11       262        440000
45000 Petrol
##   seller_type transmission        owner                   torque
seats
## 1  Individual       Manual  First Owner           190Nm@ 2000rpm
5
## 2  Individual       Manual Second Owner      250Nm@ 1500-2500rpm
5
## 3  Individual       Manual  Third Owner    12.7@ 2,700(kgm@ rpm)
5
## 4  Individual       Manual  First Owner 22.4 kgm at 1750-2750rpm
5
## 5  Individual       Manual  First Owner    11.5@ 4,500(kgm@ rpm)
5
## 6  Individual       Manual  First Owner        113.75nm@ 4000rpm
5
```

Data Description

```
summary(car$Manufacturer)

##       Germany        Japan   other Asia other Europe           US
##           501         3419         2916          417          653
```

```
summary(car$fuel)

##    CNG Diesel    LPG Petrol
##     52   4299     35   3520
```

```
summary(car$seller_type)

##            Dealer       Individual Trustmark Dealer
##              1107             6563              236
```

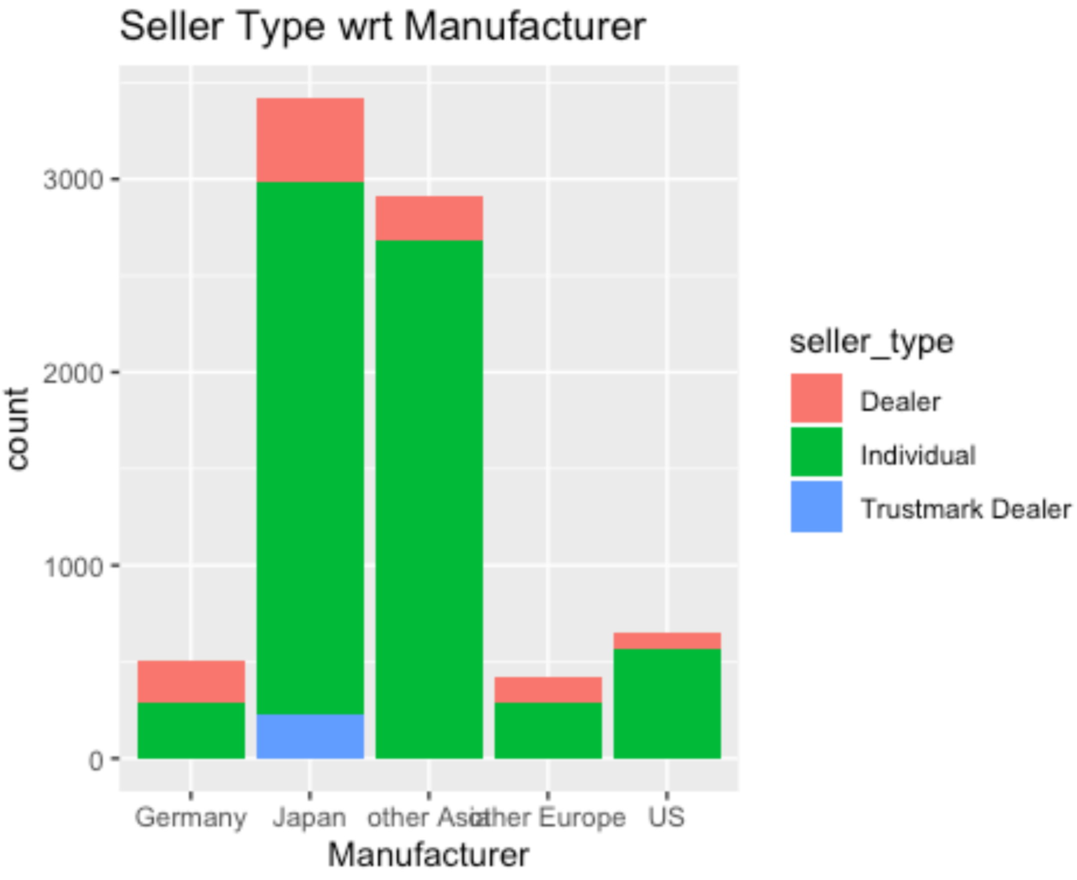
```
summary(car$owner)
```

```
##          First Owner Fourth & Above Owner         Second Owner
##                 5215                  160                 2016
##       Test Drive Car          Third Owner
##                    5                  510
```

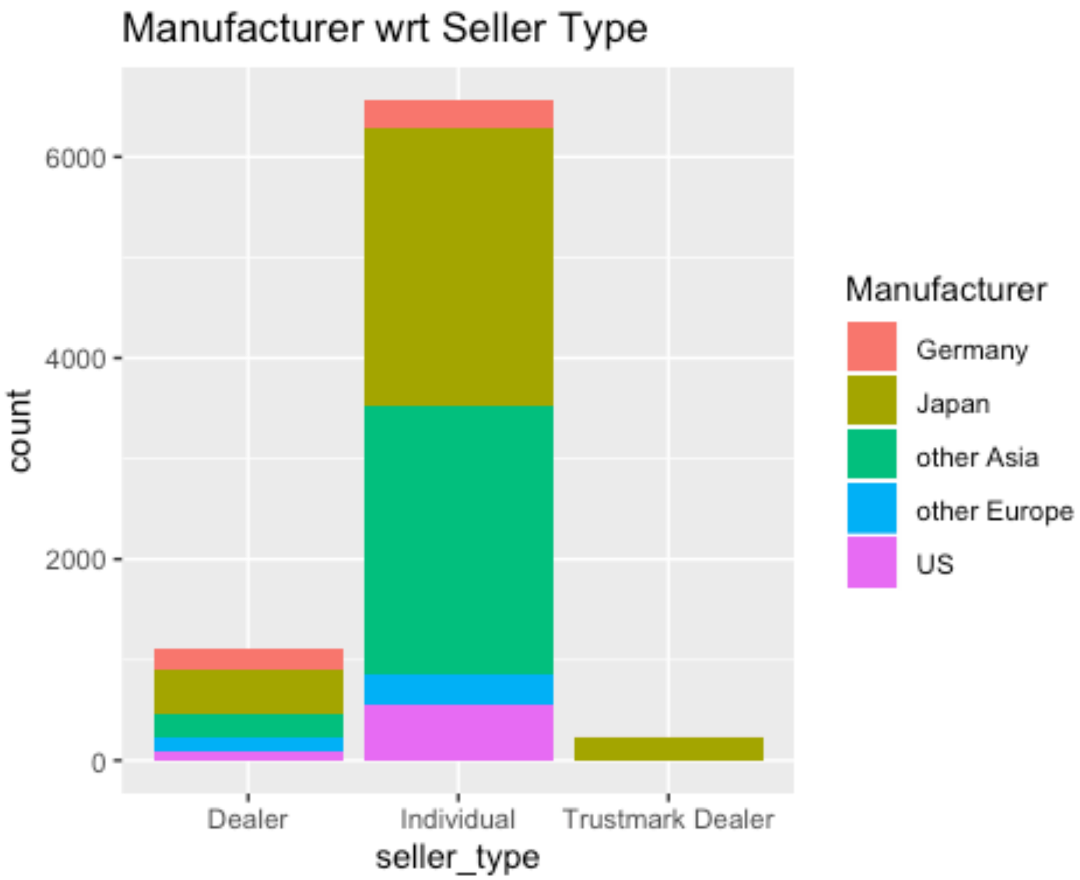
```r
summary(car$transmission)
```

```
## Automatic    Manual
##      1041      6865
```

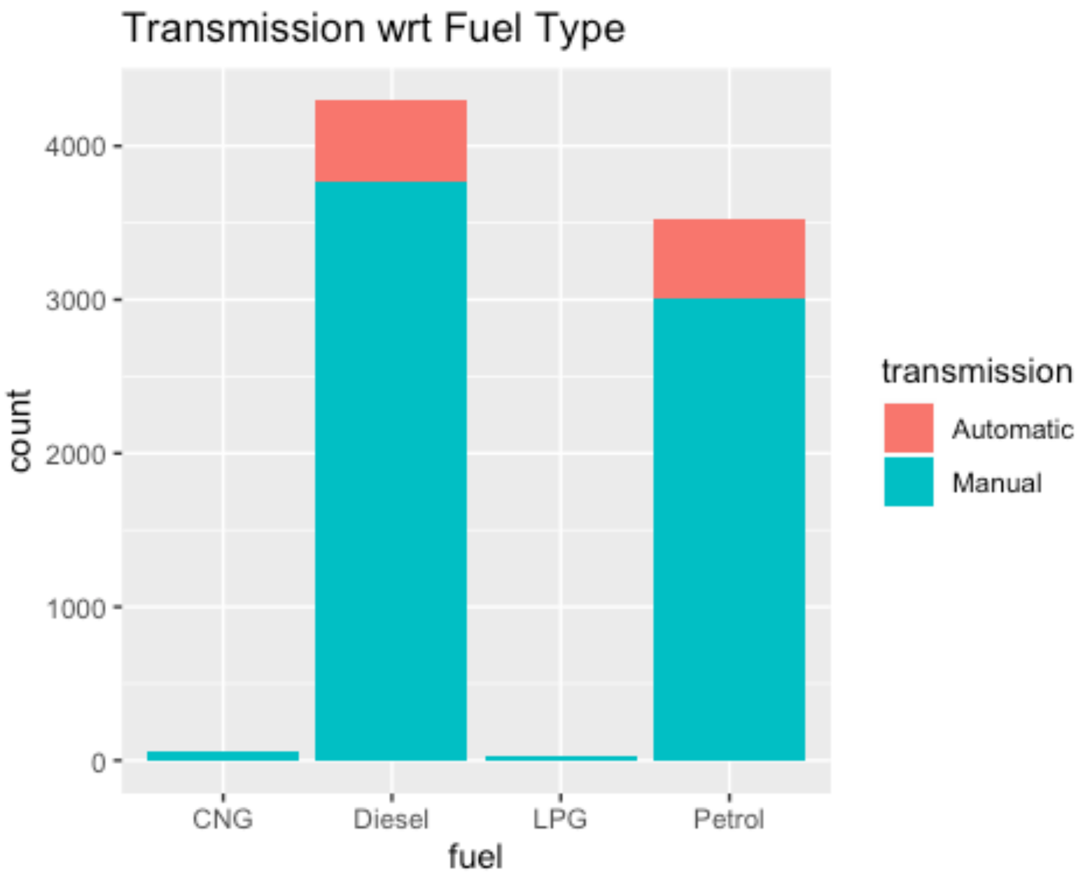
```r
ggplot(car) +
  geom_bar(mapping = aes(x = Manufacturer, fill = seller_type)) +
  ggtitle("Seller Type wrt Manufacturer")
```



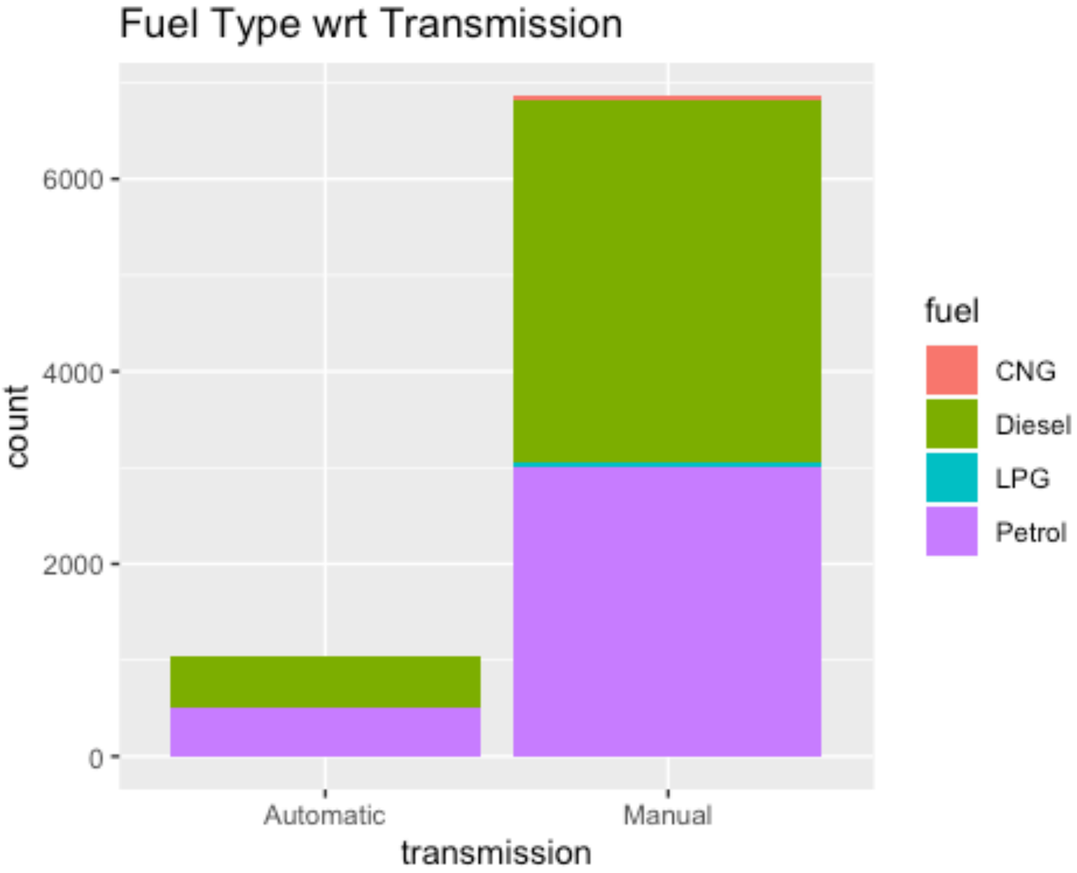
```r
ggplot(car) +
  geom_bar(mapping = aes(x = seller_type, fill = Manufacturer)) +
  ggtitle("Manufacturer wrt Seller Type")
```



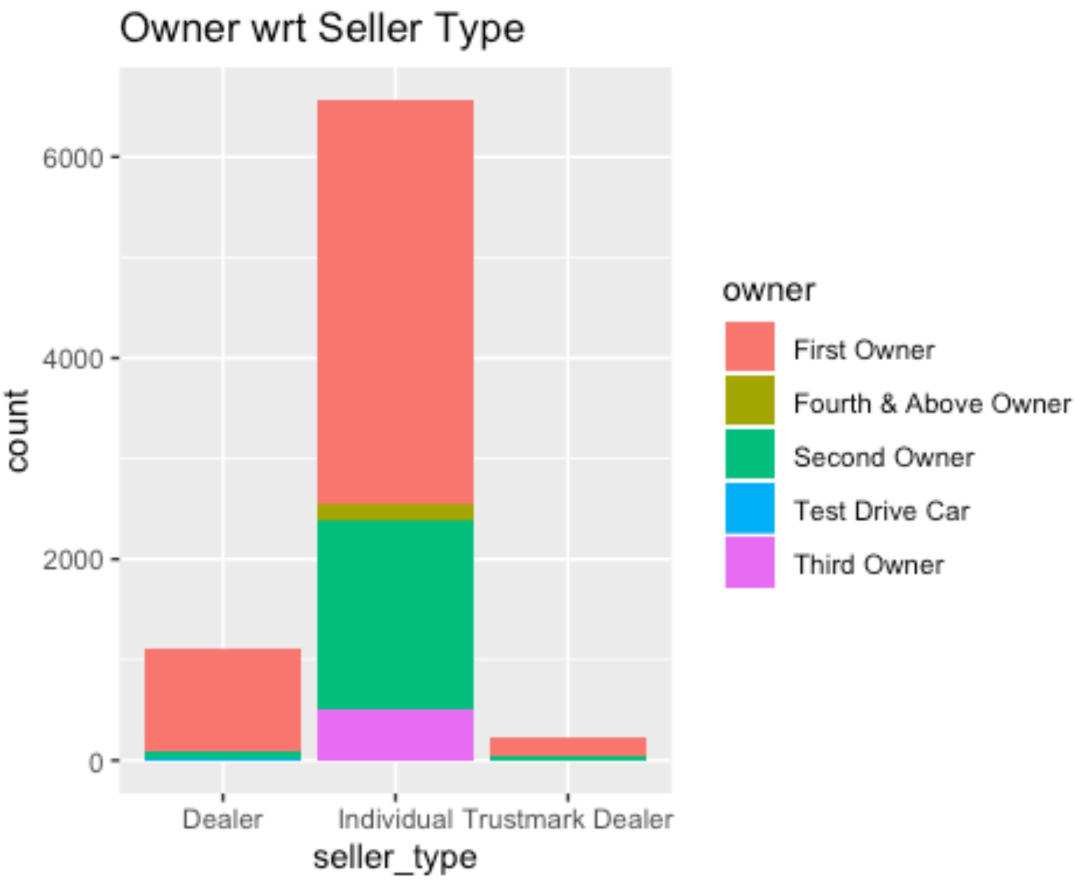
```
ggplot(car) +
  geom_bar(mapping = aes(x = fuel, fill = transmission)) +
  ggtitle("Transmission wrt Fuel Type")
```



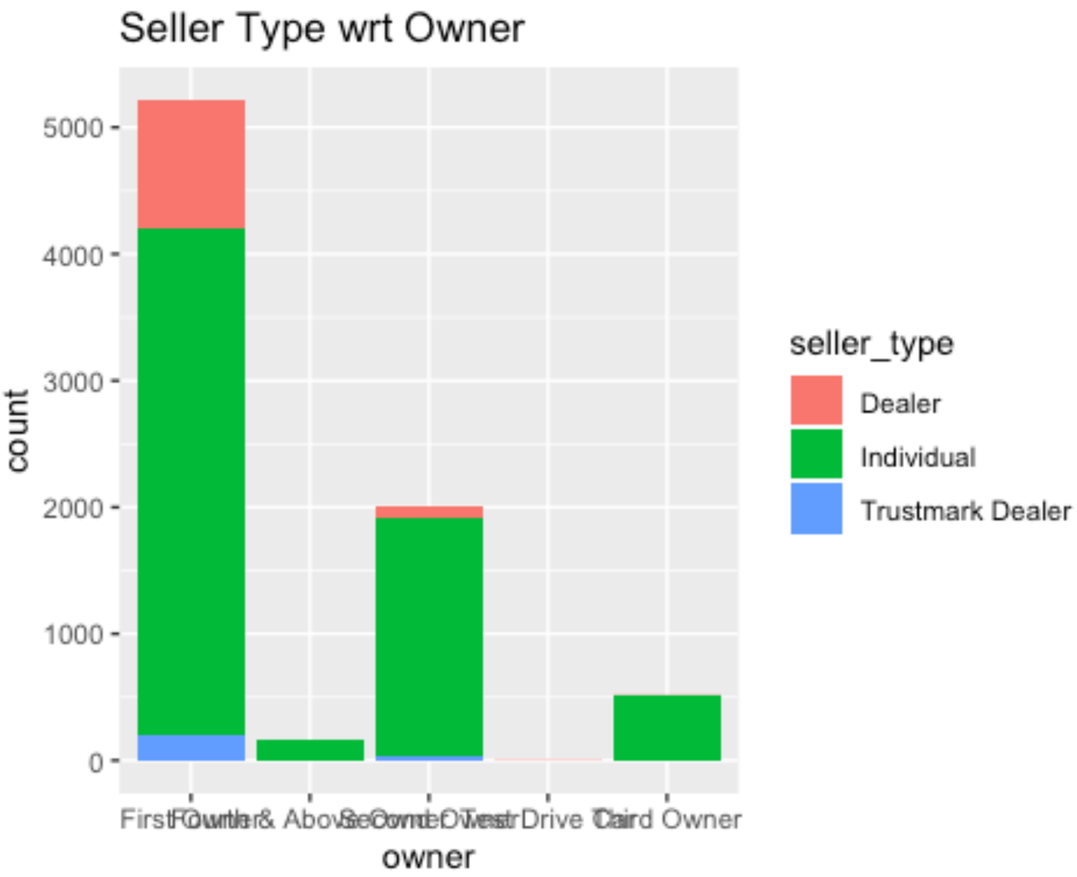
```
ggplot(car) +
  geom_bar(mapping = aes(x = transmission, fill = fuel)) +
  ggtitle("Fuel Type wrt Transmission")
```



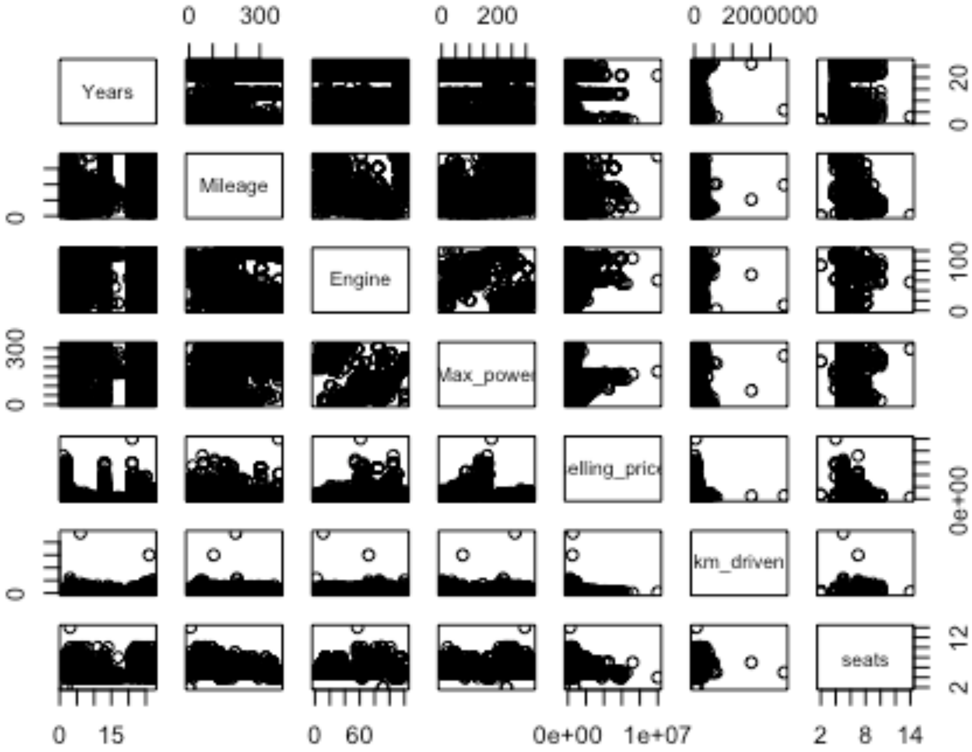
```r
ggplot(car) +
  geom_bar(mapping = aes(x = seller_type, fill = owner)) +
  ggtitle("Owner wrt Seller Type")
```



```r
ggplot(car) +
  geom_bar(mapping = aes(x = owner, fill = seller_type)) +
  ggtitle("Seller Type wrt Owner")
```



```r
# Test colinearity relationship between numerical variables
car2 <- car[, -c(1,8,9,10,11,12)]
pairs(car2)
```



Vriable Selection

```
# Full model (omit torque)
car3 <- subset(car, select = -c(torque))
full.model = lm(selling_price ~., data = car3)
summary(full.model)

##
## Call:
## lm(formula = selling_price ~ ., data = car3)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2640504  -252556   -14037   199528  8535497
##
```

```
## Coefficients:
##                               Estimate Std. Error t value Pr(>|t|)
## (Intercept)                  2.549e+06  9.961e+04  25.589  < 2e-16
***
## ManufacturerJapan           -4.054e+05  3.079e+04 -13.165  < 2e-16
***
## Manufacturerother Asia      -4.947e+05  3.080e+04 -16.061  < 2e-16
***
## Manufacturerother Europe    -1.407e+05  3.812e+04  -3.689 0.000226
***
## ManufacturerUS              -4.940e+05  3.623e+04 -13.634  < 2e-16
***
## Years                       -1.056e+04  7.732e+02 -13.662  < 2e-16
***
## Mileage                     -6.896e+02  9.532e+01  -7.234 5.11e-13
***
## Engine                       1.309e+03  1.851e+02   7.070 1.69e-12
***
## Max_power                   -1.253e+02  9.003e+01  -1.391 0.164124
## km_driven                   -1.974e+00  1.276e-01 -15.478  < 2e-16
***
## fuelDiesel                   3.224e+05  7.955e+04   4.053 5.11e-05
***
## fuelLPG                     -8.414e+04  1.237e+05  -0.680 0.496332
## fuelPetrol                  -1.157e+05  7.934e+04  -1.458 0.144822
## seller_typeIndividual       -3.517e+05  2.032e+04 -17.306  < 2e-16
***
## seller_typeTrustmark Dealer -3.673e+05  4.175e+04  -8.797  < 2e-16
***
## transmissionManual          -9.406e+05  2.268e+04 -41.467  < 2e-16
***
## ownerFourth & Above Owner   -3.485e+05  4.619e+04  -7.544 5.05e-14
***
## ownerSecond Owner           -2.010e+05  1.573e+04 -12.779  < 2e-16
***
## ownerTest Drive Car          2.244e+06  2.533e+05   8.856  < 2e-16
***
## ownerThird Owner            -2.723e+05  2.730e+04  -9.977  < 2e-16
***
## seats                        2.097e+03  9.215e+03   0.228 0.820019
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 563300 on 7885 degrees of freedom
## Multiple R-squared:  0.5219, Adjusted R-squared:  0.5207
## F-statistic: 430.3 on 20 and 7885 DF,  p-value: < 2.2e-16
```

```
full.model_1= lm(selling_price ~ 1 , data = car3)

# Backward Elimination, Forward Selection, Stepwise Regression
step(full.model, direction = "backward")
```

```
## Start:  AIC=209396
## selling_price ~ Manufacturer + Years + Mileage + Engine + Max_power
+
##     km_driven + fuel + seller_type + transmission + owner + seats
##
##                Df  Sum of Sq        RSS    AIC
## - seats         1 1.6425e+10 2.5018e+15 209394
## - Max_power     1 6.1431e+11 2.5024e+15 209396
## <none>                       2.5017e+15 209396
## - Engine        1 1.5857e+13 2.5176e+15 209444
## - Mileage       1 1.6605e+13 2.5184e+15 209446
## - Years         1 5.9222e+13 2.5610e+15 209579
## - km_driven     1 7.6008e+13 2.5778e+15 209631
## - seller_type   2 9.7326e+13 2.5991e+15 209694
## - owner         4 1.0225e+14 2.6040e+15 209705
## - Manufacturer  4 1.0498e+14 2.6067e+15 209713
## - fuel          3 2.2371e+14 2.7255e+15 210067
## - transmission  1 5.4557e+14 3.0473e+15 210954
##
## Step:  AIC=209394
## selling_price ~ Manufacturer + Years + Mileage + Engine + Max_power
+
##     km_driven + fuel + seller_type + transmission + owner
##
##                Df  Sum of Sq        RSS    AIC
## - Max_power     1 6.0954e+11 2.5024e+15 209394
## <none>                       2.5018e+15 209394
## - Engine        1 1.7059e+13 2.5188e+15 209446
## - Mileage       1 2.3201e+13 2.5250e+15 209465
## - Years         1 5.9360e+13 2.5611e+15 209577
## - km_driven     1 7.6014e+13 2.5778e+15 209629
## - seller_type   2 9.7430e+13 2.5992e+15 209692
## - owner         4 1.0436e+14 2.6061e+15 209709
## - Manufacturer  4 1.0987e+14 2.6116e+15 209726
## - fuel          3 2.8876e+14 2.7905e+15 210252
## - transmission  1 5.4907e+14 3.0508e+15 210961
## 
## Step:  AIC=209393.9
## selling_price ~ Manufacturer + Years + Mileage + Engine + km_driven 
+
##     fuel + seller_type + transmission + owner
## 
##                Df  Sum of Sq        RSS    AIC
## <none>                       2.5024e+15 209394
## - Engine        1 2.1812e+13 2.5242e+15 209461
## - Mileage       1 3.2163e+13 2.5345e+15 209493
## - Years         1 5.9039e+13 2.5614e+15 209576
## - km_driven     1 7.6157e+13 2.5785e+15 209629
## - seller_type   2 9.7785e+13 2.6002e+15 209693
## - owner         4 1.0426e+14 2.6066e+15 209709
## - Manufacturer  4 1.1333e+14 2.6157e+15 209736
## - fuel          3 3.0907e+14 2.8114e+15 210309
## - transmission  1 5.5649e+14 3.0589e+15 210980

## 
## Call:
## lm(formula = selling_price ~ Manufacturer + Years + Mileage + 
##     Engine + km_driven + fuel + seller_type + transmission + 
##     owner, data = car3)
## 
## Coefficients:
##                 (Intercept)             ManufacturerJapan  
##                   2.552e+06                    -4.093e+05  
##      Manufacturerother Asia     Manufacturerother Europe  
##                  -4.972e+05                    -1.434e+05  
##              ManufacturerUS                         Years  
##                  -5.023e+05                    -1.052e+04  
##                     Mileage                        Engine  
##                  -7.485e+02                     1.403e+03  
##                   km_driven                    fuelDiesel  
##                   -1.976e+00                    3.227e+05
##                      fuelLPG                   fuelPetrol
##                   -9.040e+04                   -1.215e+05
##        seller_typeIndividual  seller_typeTrustmark Dealer
##                   -3.521e+05                   -3.677e+05
##           transmissionManual    ownerFourth & Above Owner
##                   -9.429e+05                   -3.489e+05
##            ownerSecond Owner          ownerTest Drive Car
##                   -2.010e+05                    2.249e+06
##             ownerThird Owner
##                   -2.729e+05
```

```
step(full.model_1, direction = "forward",scop = formula(full.model))
```

```
## Start:  AIC=215189.7
## selling_price ~ 1
## 
##                 Df  Sum of Sq        RSS    AIC
## + transmission  1 1.8231e+15 3.4094e+15 211805
## + Manufacturer  4 9.5117e+14 4.2813e+15 213612
## + seller_type   2 8.6447e+14 4.3680e+15 213766
## + owner         4 3.9016e+14 4.8423e+15 214585
## + km_driven     1 2.5824e+14 4.9742e+15 214792
## + Max_power     1 2.5343e+14 4.9790e+15 214799
## + fuel          3 2.2309e+14 5.0094e+15 214851
## + Years         1 1.3072e+14 5.1017e+15 214992
## + Mileage       1 8.2942e+13 5.1495e+15 215065
## + Engine        1 3.6691e+13 5.1958e+15 215136
## + seats         1 9.0623e+12 5.2234e+15 215178
## <none>                       5.2325e+15 215190
## 
## Step:  AIC=211805.2
## selling_price ~ transmission
## 
##                 Df  Sum of Sq        RSS    AIC
## + fuel          3 2.5350e+14 3.1559e+15 211200
## + seller_type   2 2.4763e+14 3.1617e+15 211213
## + Manufacturer  4 2.3164e+14 3.1777e+15 211257
## + owner         4 1.6479e+14 3.2446e+15 211422
## + km_driven     1 5.8308e+13 3.3511e+15 211671
## + Max_power     1 5.2296e+13 3.3571e+15 211685
## + seats         1 3.7597e+13 3.3718e+15 211720
## + Years         1 2.0971e+13 3.3884e+15 211758
## + Engine        1 9.5830e+12 3.3998e+15 211785
## + Mileage       1 4.6088e+12 3.4048e+15 211796
## <none>                       3.4094e+15 211805
##
## Step:  AIC=211200.3
## selling_price ~ transmission + fuel
##
##                Df  Sum of Sq        RSS    AIC
## + seller_type   2 2.0757e+14 2.9483e+15 210666
## + owner         4 1.7965e+14 2.9762e+15 210745
## + Manufacturer  4 1.7400e+14 2.9819e+15 210760
## + km_driven     1 1.5475e+14 3.0011e+15 210805
## + Years         1 5.9803e+13 3.0961e+15 211051
## + Engine        1 2.7316e+13 3.1286e+15 211134
## + Max_power     1 2.1771e+13 3.1341e+15 211148
## + Mileage       1 7.0591e+12 3.1488e+15 211185
## <none>                       3.1559e+15 211200
## + seats         1 2.8669e+11 3.1556e+15 211202
##
## Step:  AIC=210666.5
## selling_price ~ transmission + fuel + seller_type
##
##                Df  Sum of Sq        RSS    AIC
## + Manufacturer  4 1.2868e+14 2.8196e+15 210322
## + owner         4 1.2174e+14 2.8266e+15 210341
## + km_driven     1 1.0904e+14 2.8393e+15 210371
## + Years         1 6.2317e+13 2.8860e+15 210500
## + Engine        1 3.3098e+13 2.9152e+15 210579
## + Max_power     1 2.3228e+13 2.9251e+15 210606
## + Mileage       1 1.2336e+13 2.9360e+15 210635
## + seats         1 3.2166e+12 2.9451e+15 210660
## <none>                       2.9483e+15 210666
##
## Step:  AIC=210321.6
## selling_price ~ transmission + fuel + seller_type + Manufacturer
##
##             Df  Sum of Sq        RSS    AIC
## + owner      4 1.1980e+14 2.6998e+15 209986
## + km_driven  1 1.0082e+14 2.7188e+15 210036
## + Years      1 6.6394e+13 2.7532e+15 210135
## + Engine     1 2.0797e+13 2.7988e+15 210265
## + seats      1 1.4827e+13 2.8048e+15 210282
## + Mileage    1 1.4114e+13 2.8055e+15 210284
## + Max_power  1 1.1464e+13 2.8082e+15 210291
## <none>                    2.8196e+15 210322
##
## Step:  AIC=209986.4
## selling_price ~ transmission + fuel + seller_type + Manufacturer +
##     owner
##
##             Df  Sum of Sq        RSS    AIC
## + Years      1 7.3975e+13 2.6259e+15 209769
## + km_driven  1 6.1555e+13 2.6383e+15 209806
## + Mileage    1 3.4694e+13 2.6651e+15 209886
## + Engine     1 2.7843e+13 2.6720e+15 209906
## + Max_power  1 2.0177e+13 2.6796e+15 209929
## + seats      1 1.4975e+13 2.6849e+15 209944
## <none>                    2.6998e+15 209986
##
## Step:  AIC=209768.7
## selling_price ~ transmission + fuel + seller_type + Manufacturer +
##     owner + Years
##
##             Df  Sum of Sq        RSS    AIC
## + km_driven  1 5.4184e+13 2.5717e+15 209606
## + Engine     1 3.1608e+13 2.5942e+15 209675
## + Mileage    1 2.6596e+13 2.5993e+15 209690
## + Max_power  1 2.0389e+13 2.6055e+15 209709
## + seats      1 1.5952e+13 2.6099e+15 209723
## <none>                    2.6259e+15 209769
##
## Step:  AIC=209605.9
## selling_price ~ transmission + fuel + seller_type + Manufacturer +
##     owner + Years + km_driven
##
##             Df  Sum of Sq        RSS    AIC
## + Mileage    1 4.7480e+13 2.5242e+15 209461
## + Engine     1 3.7129e+13 2.5345e+15 209493
## + Max_power  1 2.7609e+13 2.5441e+15 209523
## + seats      1 2.4514e+13 2.5472e+15 209532
## <none>                    2.5717e+15 209606
##
## Step:  AIC=209460.5
## selling_price ~ transmission + fuel + seller_type + Manufacturer +
##     owner + Years + km_driven + Mileage
##
##              Df  Sum of Sq        RSS    AIC
## + Engine     1 2.1812e+13 2.5024e+15 209394
## + Max_power  1 5.3620e+12 2.5188e+15 209446
## + seats      1 1.5318e+12 2.5227e+15 209458
## <none>                    2.5242e+15 209461
##
## Step:  AIC=209393.9
## selling_price ~ transmission + fuel + seller_type + Manufacturer +
##     owner + Years + km_driven + Mileage + Engine
##
##              Df  Sum of Sq        RSS    AIC
## <none>                    2.5024e+15 209394
## + Max_power  1 6.0954e+11 2.5018e+15 209394
## + seats      1 1.1659e+10 2.5024e+15 209396

##
## Call:
## lm(formula = selling_price ~ transmission + fuel + seller_type +
##     Manufacturer + owner + Years + km_driven + Mileage + Engine,
##     data = car3)
##
## Coefficients:
##                 (Intercept)           transmissionManual
##                   2.552e+06                   -9.429e+05
##                  fuelDiesel                      fuelLPG
##                   3.227e+05                   -9.040e+04
##                  fuelPetrol        seller_typeIndividual
##                  -1.215e+05                   -3.521e+05
## seller_typeTrustmark Dealer            ManufacturerJapan
##                  -3.677e+05                   -4.093e+05
##      Manufacturerother Asia     Manufacturerother Europe
##                  -4.972e+05                   -1.434e+05
##              ManufacturerUS    ownerFourth & Above Owner
##                  -5.023e+05                   -3.489e+05
##           ownerSecond Owner          ownerTest Drive Car
##                  -2.010e+05                    2.249e+06
##            ownerThird Owner                        Years
##                  -2.729e+05                   -1.052e+04
##                   km_driven                      Mileage
##                  -1.976e+00                   -7.485e+02
##                      Engine
##                   1.403e+03
```

```r
step(full.model, direction = "both")
```

```
## Start:  AIC=209396
## selling_price ~ Manufacturer + Years + Mileage + Engine + Max_power 
+
##     km_driven + fuel + seller_type + transmission + owner + seats
## 
##                Df  Sum of Sq        RSS    AIC
## - seats         1 1.6425e+10 2.5018e+15 209394
## - Max_power     1 6.1431e+11 2.5024e+15 209396
## <none>                       2.5017e+15 209396
## - Engine        1 1.5857e+13 2.5176e+15 209444
## - Mileage       1 1.6605e+13 2.5184e+15 209446
## - Years         1 5.9222e+13 2.5610e+15 209579
## - km_driven     1 7.6008e+13 2.5778e+15 209631
## - seller_type   2 9.7326e+13 2.5991e+15 209694
## - owner         4 1.0225e+14 2.6040e+15 209705
## - Manufacturer  4 1.0498e+14 2.6067e+15 209713
## - fuel          3 2.2371e+14 2.7255e+15 210067
## - transmission  1 5.4557e+14 3.0473e+15 210954
## 
## Step:  AIC=209394
## selling_price ~ Manufacturer + Years + Mileage + Engine + Max_power 
+
##     km_driven + fuel + seller_type + transmission + owner
## 
##                Df  Sum of Sq        RSS    AIC
## - Max_power     1 6.0954e+11 2.5024e+15 209394
## <none>                       2.5018e+15 209394
## + seats         1 1.6425e+10 2.5017e+15 209396
## - Engine        1 1.7059e+13 2.5188e+15 209446
## - Mileage       1 2.3201e+13 2.5250e+15 209465
## - Years         1 5.9360e+13 2.5611e+15 209577
## - km_driven     1 7.6014e+13 2.5778e+15 209629
## - seller_type   2 9.7430e+13 2.5992e+15 209692
## - owner         4 1.0436e+14 2.6061e+15 209709
## - Manufacturer  4 1.0987e+14 2.6116e+15 209726
## - fuel          3 2.8876e+14 2.7905e+15 210252
## - transmission  1 5.4907e+14 3.0508e+15 210961
##
## Step:  AIC=209393.9
## selling_price ~ Manufacturer + Years + Mileage + Engine + km_driven
+
##     fuel + seller_type + transmission + owner
##
##                Df  Sum of Sq        RSS    AIC
## <none>                       2.5024e+15 209394
## + Max_power     1 6.0954e+11 2.5018e+15 209394
## + seats         1 1.1659e+10 2.5024e+15 209396
## - Engine        1 2.1812e+13 2.5242e+15 209461
## - Mileage       1 3.2163e+13 2.5345e+15 209493
## - Years         1 5.9039e+13 2.5614e+15 209576
## - km_driven     1 7.6157e+13 2.5785e+15 209629
## - seller_type   2 9.7785e+13 2.6002e+15 209693
## - owner         4 1.0426e+14 2.6066e+15 209709
## - Manufacturer  4 1.1333e+14 2.6157e+15 209736
## - fuel          3 3.0907e+14 2.8114e+15 210309
## - transmission  1 5.5649e+14 3.0589e+15 210980

##
## Call:
## lm(formula = selling_price ~ Manufacturer + Years + Mileage +
##     Engine + km_driven + fuel + seller_type + transmission +
##     owner, data = car3)
##
## Coefficients:
##                (Intercept)            ManufacturerJapan
##                  2.552e+06                   -4.093e+05
##      Manufacturerother Asia     Manufacturerother Europe
##                 -4.972e+05                   -1.434e+05
##              ManufacturerUS                        Years
##                  -5.023e+05                   -1.052e+04
##                     Mileage                       Engine
##                  -7.485e+02                    1.403e+03
##                   km_driven                   fuelDiesel
##                  -1.976e+00                    3.227e+05
##                     fuelLPG                   fuelPetrol
##                  -9.040e+04                   -1.215e+05
##       seller_typeIndividual  seller_typeTrustmark Dealer
##                  -3.521e+05                   -3.677e+05
##          transmissionManual    ownerFourth & Above Owner
##                  -9.429e+05                   -3.489e+05
##           ownerSecond Owner          ownerTest Drive Car
##                  -2.010e+05                    2.249e+06
##            ownerThird Owner
##                  -2.729e+05
```

```
# Decide to omit two least important variables: seats and max power.
car4 <- subset(car3, select = -c(seats, Max_power))
head(car4)
```

```
##   Manufacturer Years Mileage Engine selling_price km_driven   fuel
seller_type
## 1        Japan    24     324     14        450000    145500 Diesel
Individual
## 2      Germany    24     274     37        370000    120000 Diesel
Individual
## 3        Japan     7     174     36        158000    140000 Petrol
Individual
## 4   other Asia     3     316     25        225000    127000 Diesel
Individual
## 5        Japan     6     132     15        130000    120000 Petrol
Individual
## 6   other Asia    21     237     11        440000     45000 Petrol
Individual
##   transmission        owner
## 1       Manual  First Owner
## 2       Manual Second Owner
## 3       Manual  Third Owner
## 4       Manual  First Owner
## 5        Manual  First Owner
## 6        Manual  First Owner
```

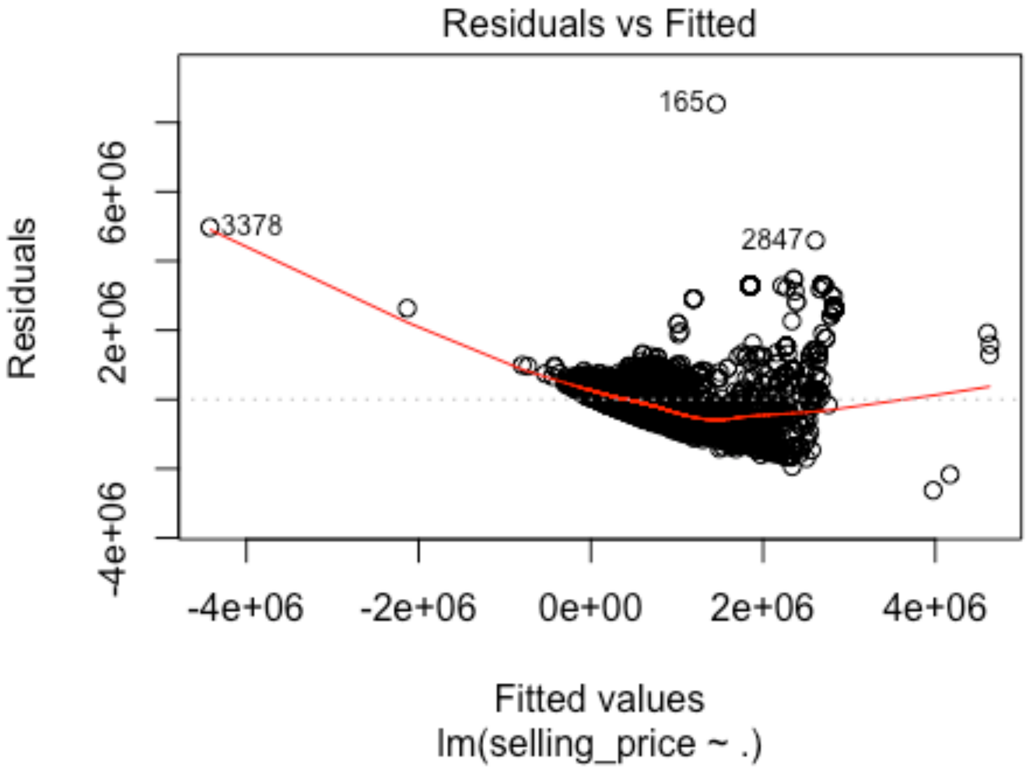
Data Transformation

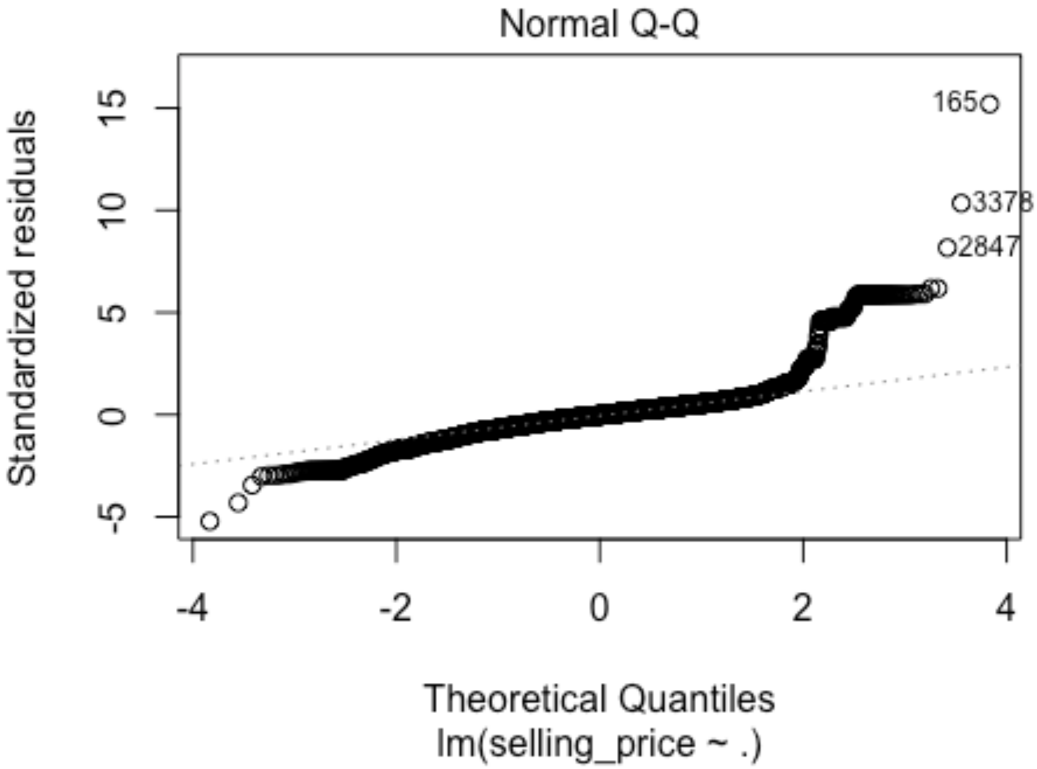
```
lm1 <- lm(selling_price ~ ., data = car4)
summary(lm1)
```

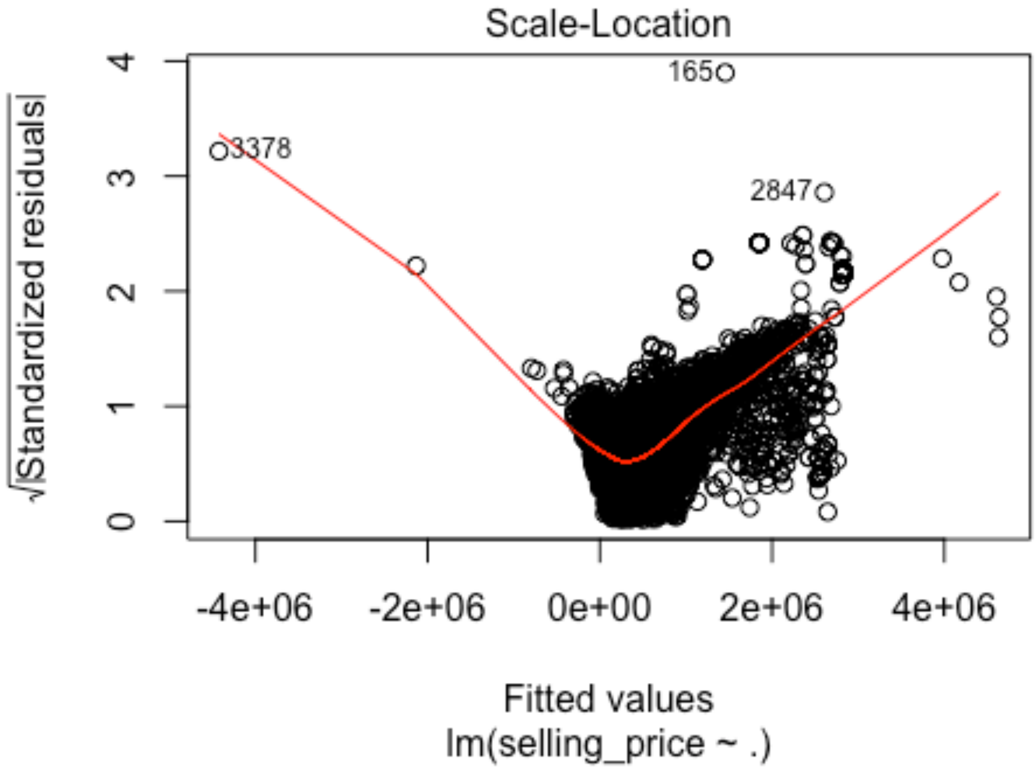
```
## 
## Call:
## lm(formula = selling_price ~ ., data = car4)
## 
## Residuals:
##      Min       1Q   Median       3Q      Max 
## -2625040  -251989   -13873   198894  8543525 
## 
## Coefficients:
##                                Estimate Std. Error t value Pr(>|t|)
## (Intercept)                   2.552e+06  8.759e+04  29.134  < 2e-16
***
## ManufacturerJapan            -4.093e+05  2.966e+04 -13.801  < 2e-16
***
## Manufacturerother Asia       -4.972e+05  2.980e+04 -16.682  < 2e-16
***
## Manufacturerother Europe     -1.434e+05  3.793e+04  -3.781 0.000157
***
## ManufacturerUS               -5.023e+05  3.535e+04 -14.209  < 2e-16
***
## Years                        -1.052e+04  7.710e+02 -13.641  < 2e-16
***
## Mileage                      -7.485e+02  7.434e+01 -10.068  < 2e-16
***
## Engine                        1.403e+03  1.692e+02   8.291  < 2e-16
***
## km_driven                    -1.976e+00  1.276e-01 -15.493  < 2e-16
***
## fuelDiesel                    3.227e+05  7.937e+04   4.066 4.83e-05
***
## fuelLPG                      -9.040e+04  1.235e+05  -0.732 0.464375
## fuelPetrol                   -1.215e+05  7.921e+04  -1.534 0.125035
## seller_typeIndividual        -3.521e+05  2.029e+04 -17.347  < 2e-16
```

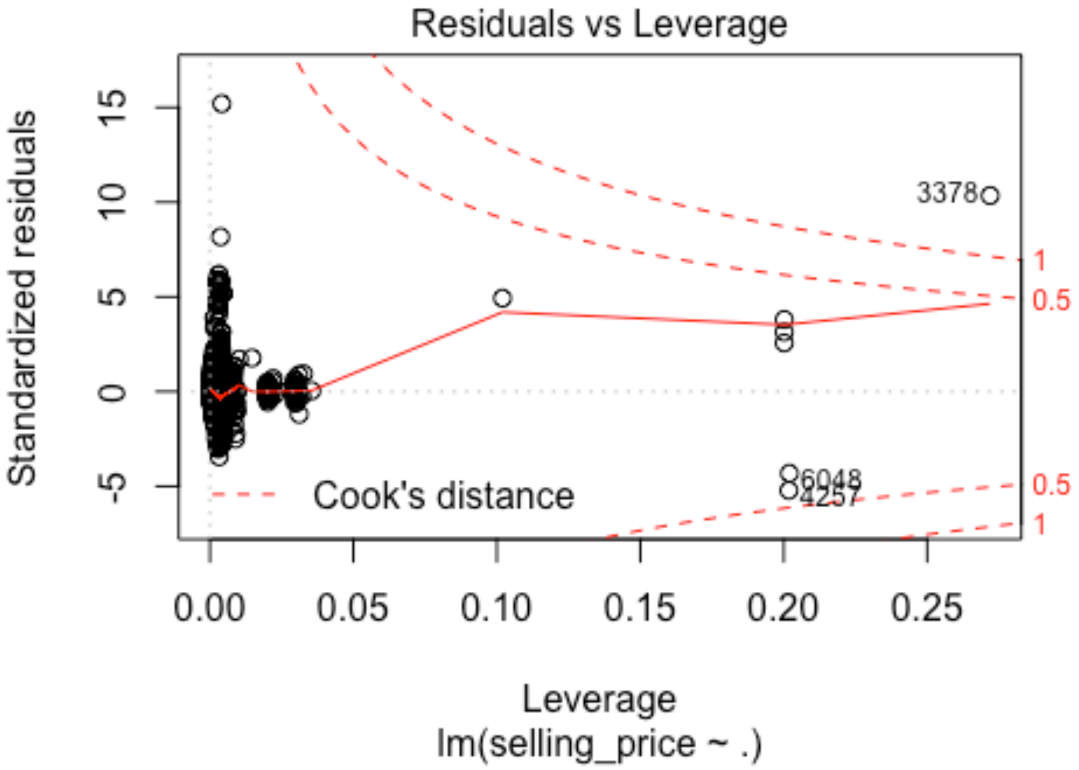
```
***
## seller_typeTrustmark Dealer -3.677e+05  4.174e+04  -8.808  < 2e-16
***
## transmissionManual          -9.429e+05  2.251e+04 -41.880  < 2e-16
***
## ownerFourth & Above Owner   -3.489e+05  4.609e+04  -7.570 4.14e-14
***
## ownerSecond Owner           -2.010e+05  1.562e+04 -12.869  < 2e-16
***
## ownerTest Drive Car          2.249e+06  2.533e+05   8.877  < 2e-16
***
## ownerThird Owner            -2.729e+05  2.717e+04 -10.046  < 2e-16
***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 563300 on 7887 degrees of freedom
## Multiple R-squared:  0.5218, Adjusted R-squared:  0.5207
## F-statistic:   478 on 18 and 7887 DF,  p-value: < 2.2e-16
```

```r
plot(lm1)
```


Residuals vs Fitted
Residuals
-4e+06
2e+06
6e+06
-4e+06
-2e+06
0e+00
2e+06
4e+06
165
3378
2847
Fitted values
lm(selling_price ~ .)


Normal Q-Q
Standardized residuals
-5
0
5
10
15
-4
-2
0
2
4
Theoretical Quantiles
lm(selling_price ~ .)
165
3378
2847

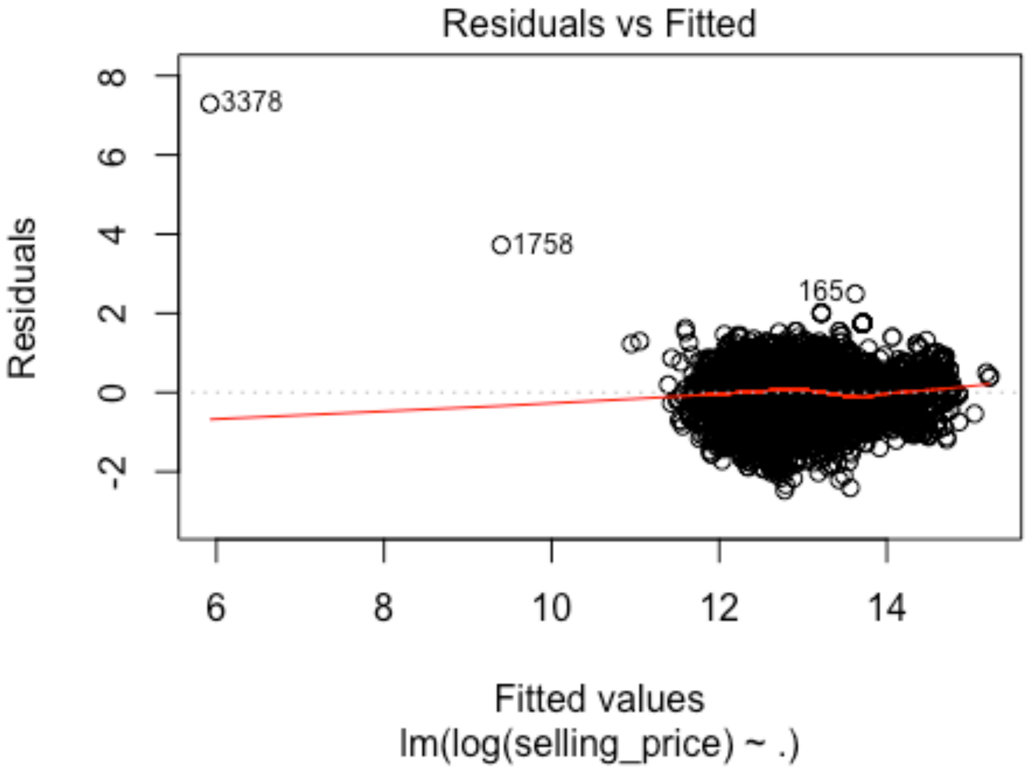

Scale-Location
√|Standardized residuals|
0
1
2
3
4
165
3378
2847
-4e+06
-2e+06
0e+00
2e+06
4e+06
Fitted values
lm(selling_price ~ .)

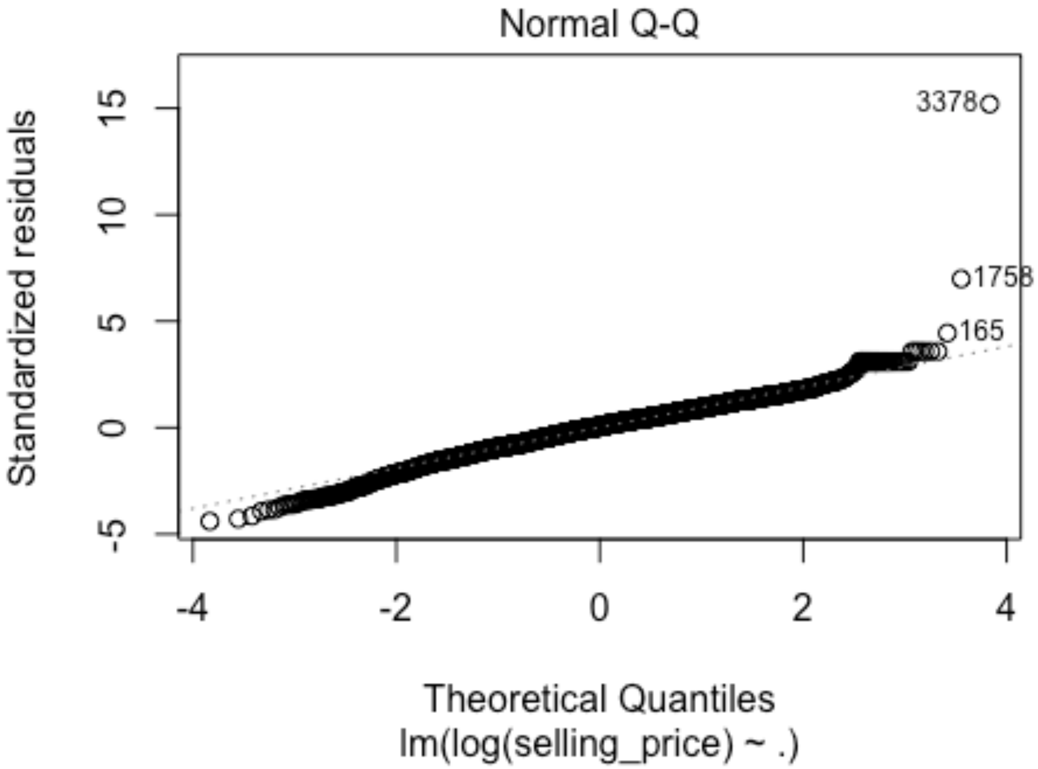


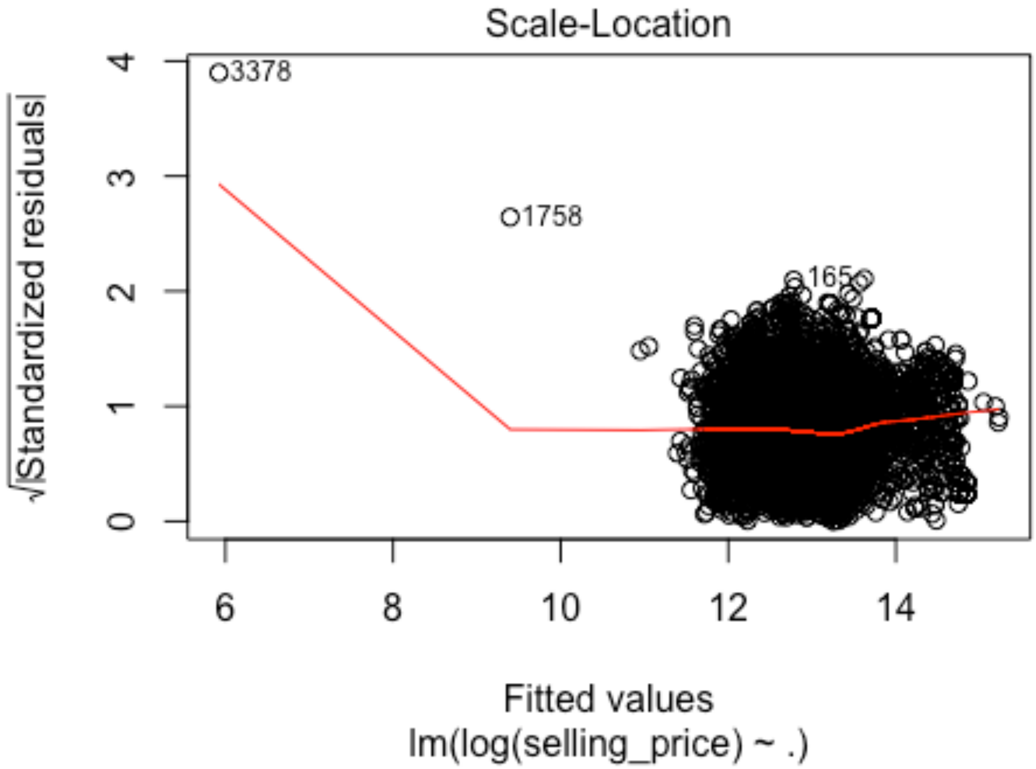
```
# Take the log transformation of response variable: selling price
log1.lm <- lm(log(selling_price) ~ ., data = car4)
summary(log1.lm)

##
## Call:
## lm(formula = log(selling_price) ~ ., data = car4)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -2.4721 -0.3560  0.0351  0.3649  7.2949
##
## Coefficients:
##                                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)                    1.451e+01  8.747e-02 165.905  < 2e-16
***
```

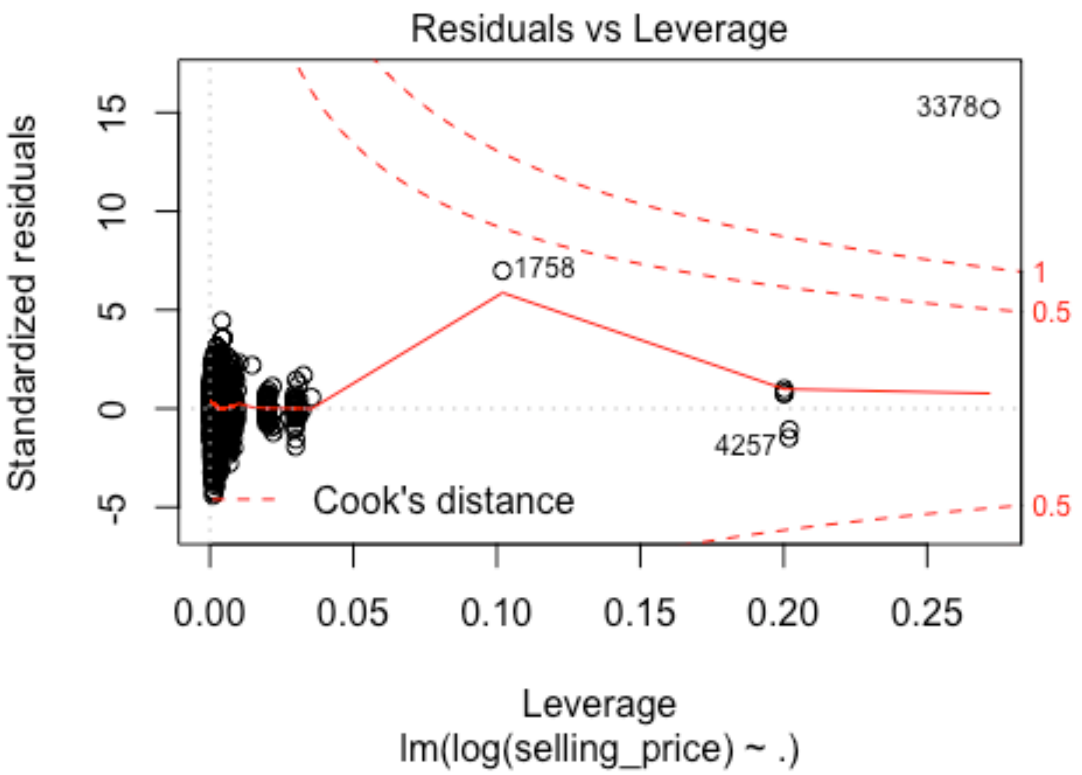
```
## ManufacturerJapan           -1.055e-01  2.962e-02  -3.562  0.00037
***
## Manufacturerother Asia      -2.423e-01  2.977e-02  -8.140 4.55e-16
***
## Manufacturerother Europe     8.429e-03  3.788e-02   0.222  0.82394
## ManufacturerUS              -3.406e-01  3.530e-02  -9.648  < 2e-16
***
## Years                        4.444e-03  7.700e-04   5.770 8.20e-09
***
## Mileage                     -9.942e-04  7.424e-05 -13.391  < 2e-16
***
## Engine                      -1.832e-03  1.690e-04 -10.839  < 2e-16
***
## km_driven                   -2.744e-06  1.274e-07 -21.538  < 2e-16
***
## fuelDiesel                   4.439e-01  7.927e-02   5.600 2.21e-08
***
## fuelLPG                     -3.715e-01  1.234e-01  -3.011  0.00261
**
## fuelPetrol                  -1.929e-01  7.911e-02  -2.438  0.01478
*
## seller_typeIndividual       -2.268e-01  2.027e-02 -11.188  < 2e-16
***
## seller_typeTrustmark Dealer  6.896e-03  4.169e-02   0.165  0.86862
## transmissionManual          -8.536e-01  2.248e-02 -37.962  < 2e-16
***
## ownerFourth & Above Owner   -8.028e-01  4.603e-02 -17.440  < 2e-16
***
## ownerSecond Owner           -4.077e-01  1.560e-02 -26.129  < 2e-16
***
## ownerTest Drive Car          1.133e+00  2.530e-01   4.477 7.69e-06
***
## ownerThird Owner            -6.175e-01  2.713e-02 -22.758  < 2e-16
***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.5625 on 7887 degrees of freedom
## Multiple R-squared:  0.539,  Adjusted R-squared:  0.538
## F-statistic: 512.4 on 18 and 7887 DF,  p-value: < 2.2e-16
```

```
plot(log1.lm)
```


Residuals vs Fitted
Residuals
8
6
4
2
0
-2
3378
1758
165
6
8
10
12
14
Fitted values
lm(log(selling_price) ~ .)


Normal Q-Q
Standardized residuals
15
10
5
0
-5
3378
1758
165
-4
-2
0
2
4
Theoretical Quantiles
lm(log(selling_price) ~ .)

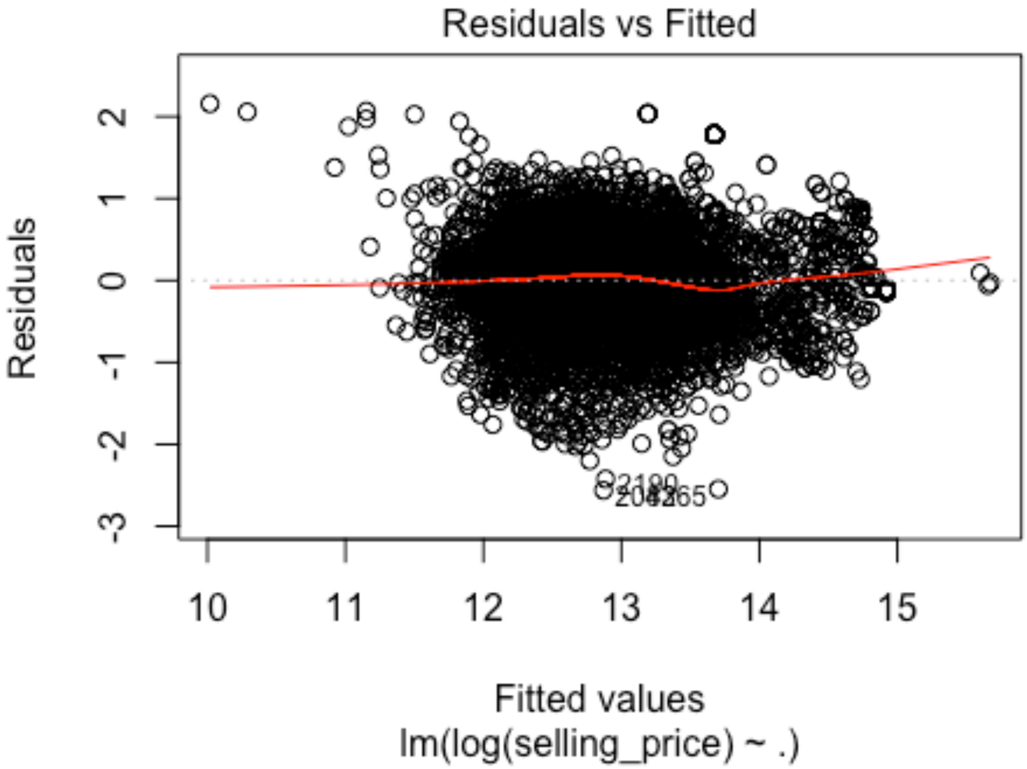

Scale-Location
√|Standardized residuals|
0
1
2
3
4
3378
1758
165
6
8
10
12
14
Fitted values
lm(log(selling_price) ~ .)



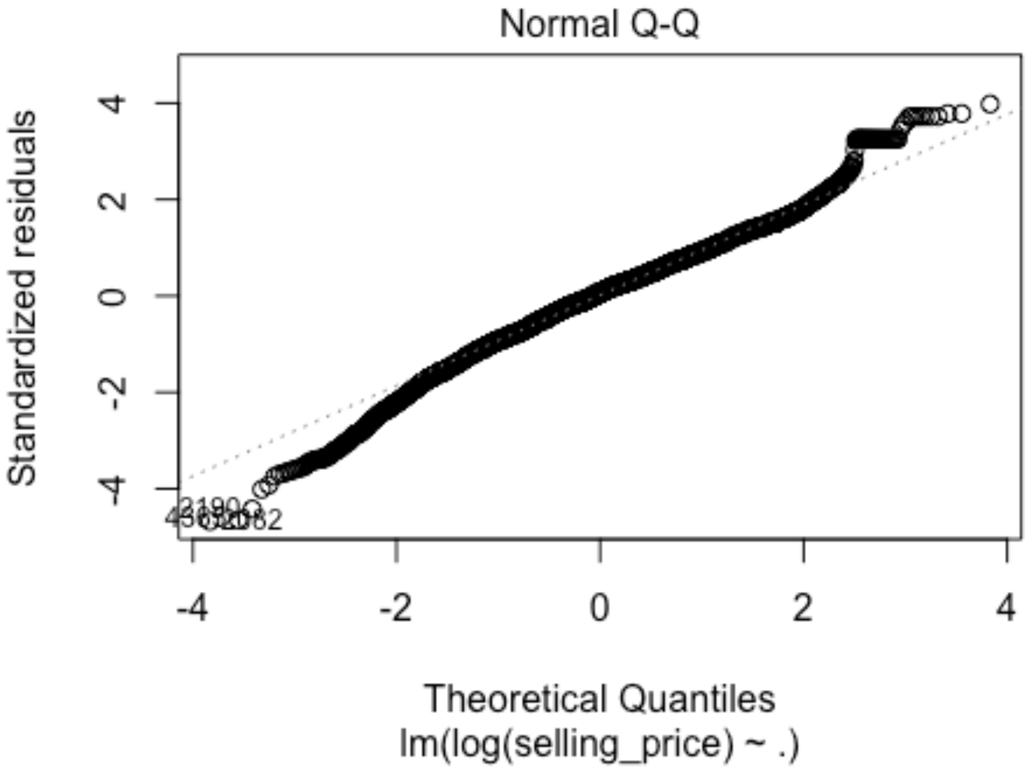
```
# Omit some problematic observations: 165, 1758, 3378, 3898, 4257,
5022, 6048, 6432, 6492, 7154, 7521, 7823
car <- car4[-c(165, 1758, 3378, 3898, 4257, 5022, 6048, 6432, 6492,
7154, 7521, 7823),]
log.lm <- lm(log(selling_price) ~ ., data = car)
summary(log.lm)

##
## Call:
## lm(formula = log(selling_price) ~ ., data = car)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2.56192 -0.33847  0.03713  0.35488  2.15722
##
## Coefficients:
```

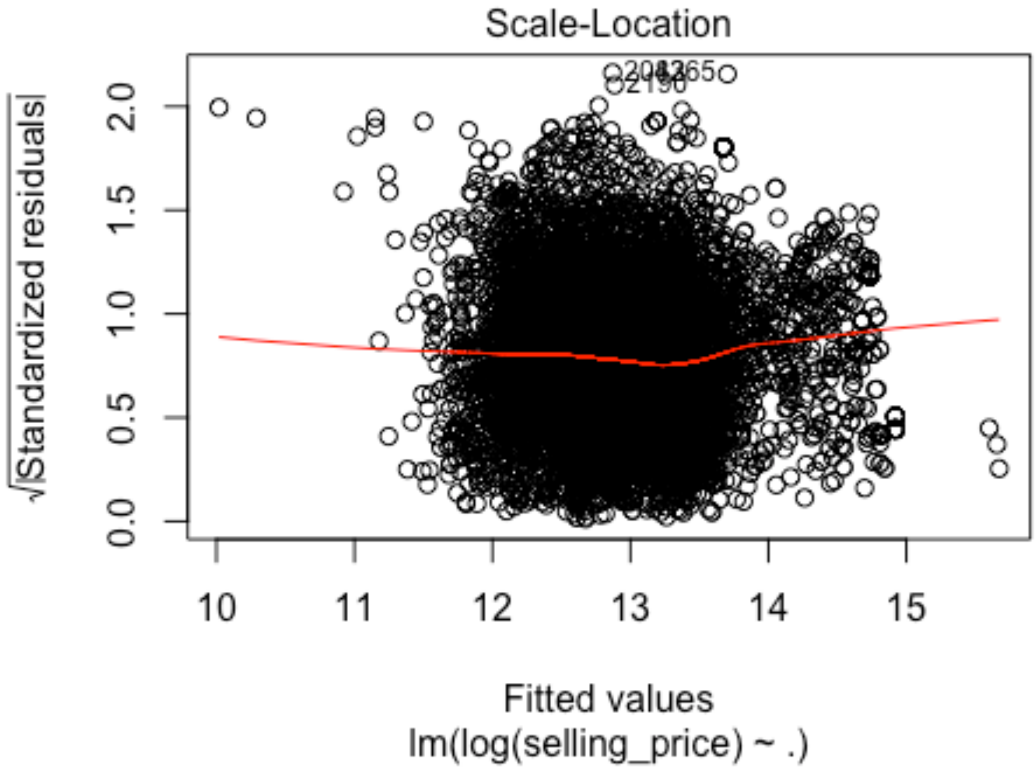
```
##                               Estimate Std. Error t value Pr(>|t|)
## (Intercept)                  1.458e+01  8.531e-02 170.852  < 2e-16
***
## ManufacturerJapan           -8.038e-02  2.897e-02  -2.775  0.00553
**
## Manufacturerother Asia      -2.371e-01  2.907e-02  -8.154 4.05e-16
***
## Manufacturerother Europe    -5.303e-03  3.698e-02  -0.143  0.88597
## ManufacturerUS              -3.343e-01  3.446e-02  -9.702  < 2e-16
***
## Years                        5.238e-03  7.530e-04   6.955 3.80e-12
***
## Mileage                     -1.215e-03  7.326e-05 -16.587  < 2e-16
***
## Engine                      -1.770e-03  1.648e-04 -10.742  < 2e-16
***
## km_driven                   -4.564e-06  1.568e-07 -29.114  < 2e-16
***
## fuelDiesel                   4.790e-01  7.727e-02   6.199 5.97e-10
***
## fuelLPG                     -3.545e-01  1.202e-01  -2.948  0.00320
**
## fuelPetrol                  -2.231e-01  7.710e-02  -2.893  0.00383
**
## seller_typeIndividual       -2.022e-01  1.982e-02 -10.202  < 2e-16
***
## seller_typeTrustmark Dealer  1.270e-02  4.063e-02   0.313  0.75458
## transmissionManual          -8.099e-01  2.203e-02 -36.756  < 2e-16
***
## ownerFourth & Above Owner   -7.375e-01  4.500e-02 -16.389  < 2e-16
***
## ownerSecond Owner           -3.724e-01  1.532e-02 -24.301  < 2e-16
***
## ownerTest Drive Car          1.572e+00  3.180e-01   4.942 7.88e-07
***
## ownerThird Owner            -5.561e-01  2.666e-02 -20.861  < 2e-16
***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.5482 on 7875 degrees of freedom
## Multiple R-squared:  0.5607, Adjusted R-squared:  0.5597
## F-statistic: 558.3 on 18 and 7875 DF,  p-value: < 2.2e-16
```

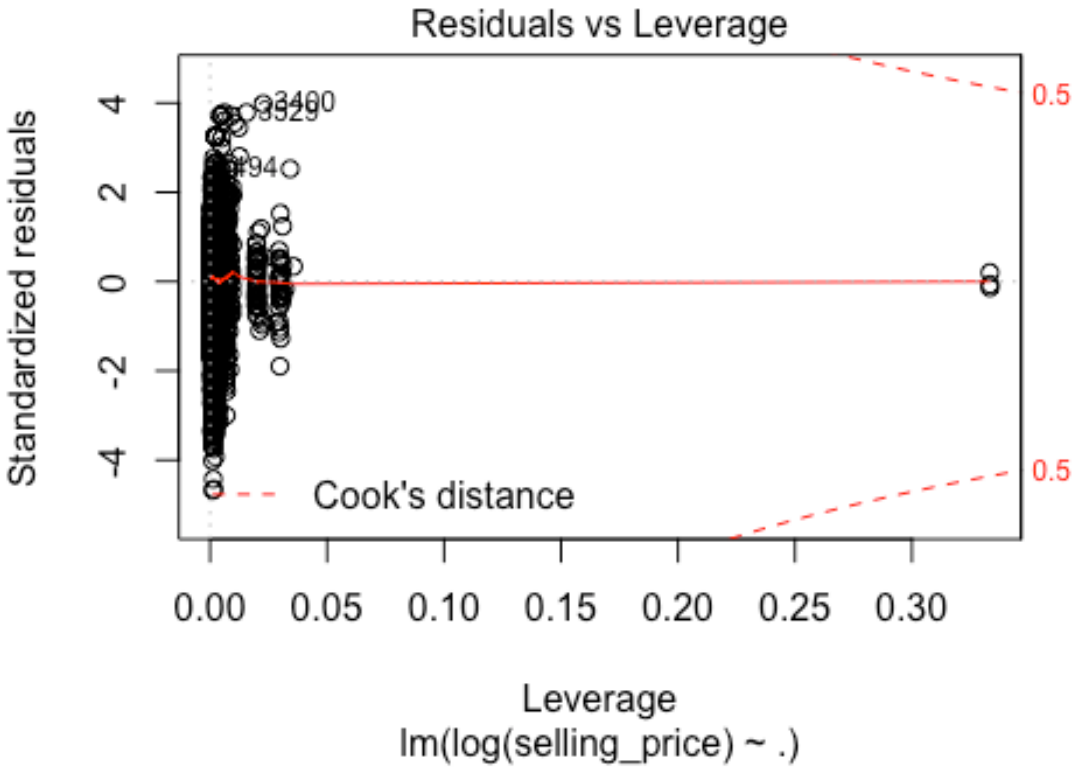
```
plot(log.lm)
```



Residuals vs Fitted


Normal Q-Q
Standardized residuals
4
2
0
-2
-4
2190
2082
-4
-2
0
2
4
Theoretical Quantiles
lm(log(selling_price) ~ .)


Scale-Location
√|Standardized residuals|
0.0
0.5
1.0
1.5
2.0
208265
2190
10
11
12
13
14
15
Fitted values
lm(log(selling_price) ~ .)

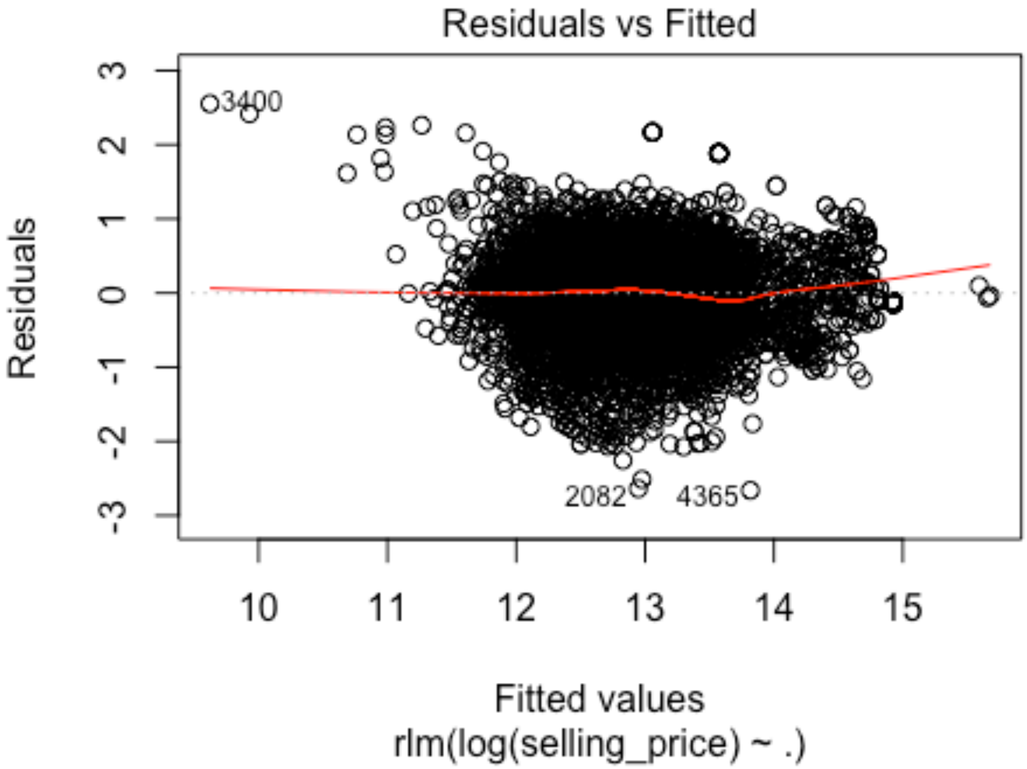


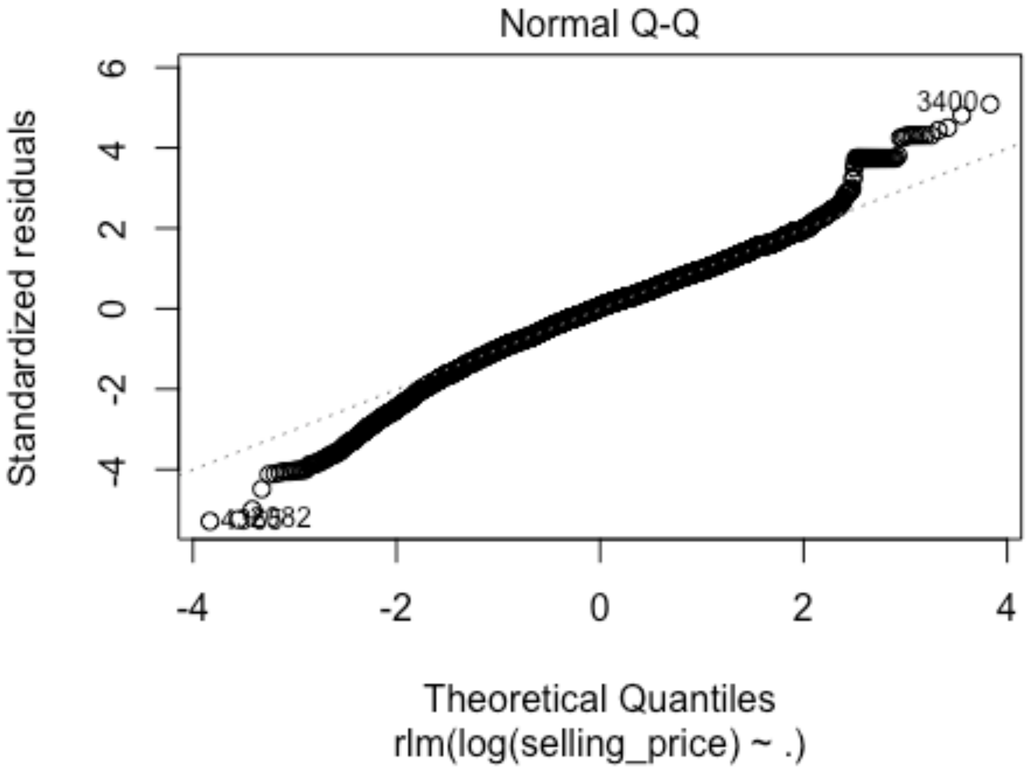
```
anova(log.lm)

## Analysis of Variance Table
##
## Response: log(selling_price)
##                 Df  Sum Sq Mean Sq  F value    Pr(>F)
## Manufacturer     4  553.43  138.36  460.396 < 2.2e-16 ***
## Years            1    6.77    6.77   22.531 2.104e-06 ***
## Mileage          1    3.99    3.99   13.281 0.0002698 ***
## Engine           1  107.28  107.28  356.971 < 2.2e-16 ***
## km_driven        1  438.36  438.36 1458.695 < 2.2e-16 ***
## fuel             3  908.37  302.79 1007.557 < 2.2e-16 ***
## seller_type      2  238.07  119.03  396.098 < 2.2e-16 ***
## transmission     1  463.70  463.70 1543.014 < 2.2e-16 ***
## owner            4  300.10   75.02  249.649 < 2.2e-16 ***
## Residuals     7875 2366.58    0.30
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

#vcov(log.lm)
vif(log.lm)

##                   GVIF Df GVIF^(1/(2*Df))
## Manufacturer 1.542927  4        1.055706
## Years        1.109882  1        1.053509
## Mileage      1.312657  1        1.145713
## Engine       1.117929  1        1.057322
## km_driven    1.486053  1        1.219038
## fuel         1.333536  3        1.049142
## seller_type  1.349496  2        1.077812
## transmission 1.457635  1        1.207326
## owner        1.247841  4        1.028063

#confint(log.lm, level = 0.95)
```

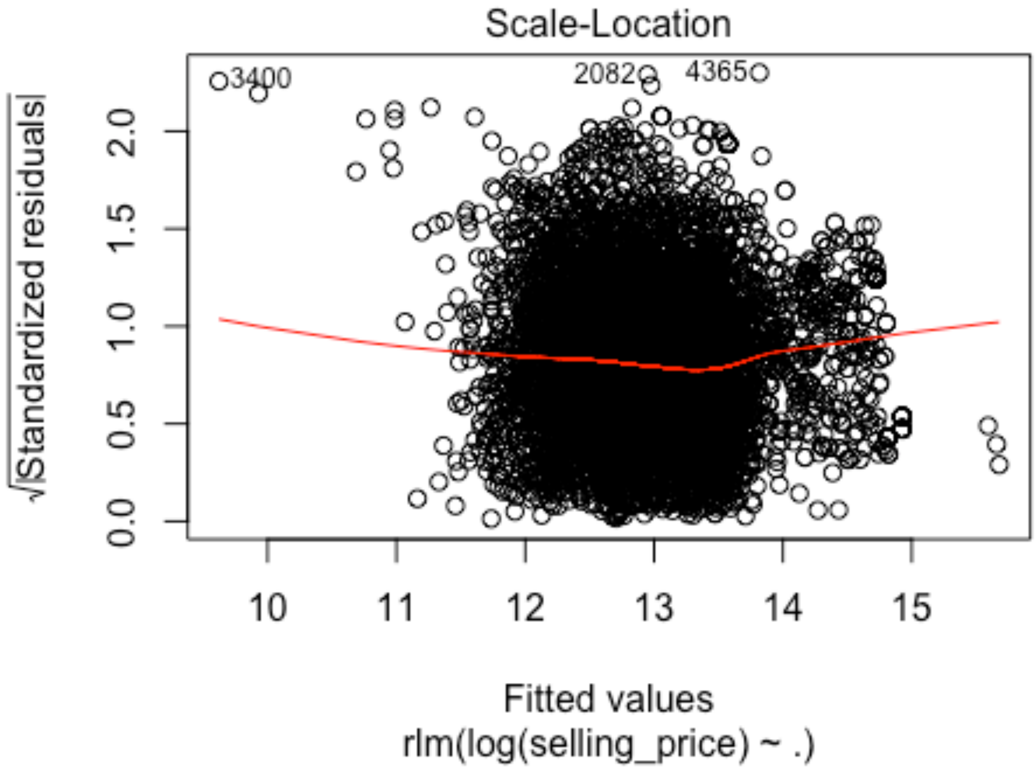
Robust Regression

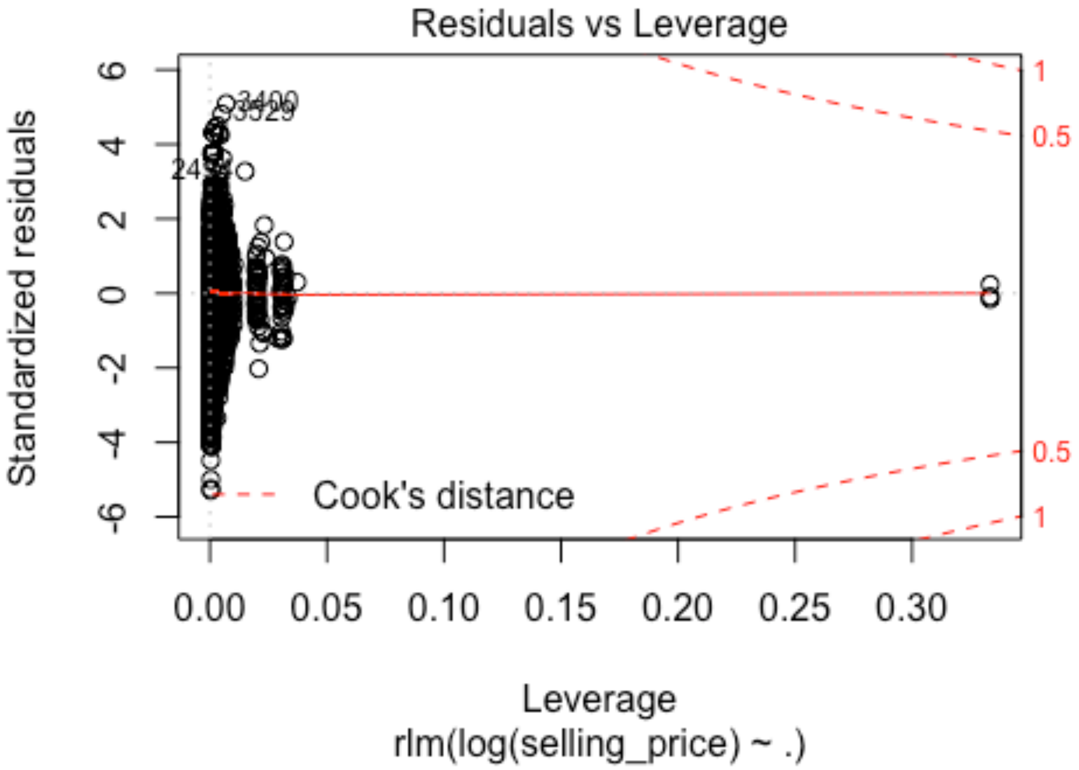
```
# Huber’s t Function
robust_huber.lm <- rlm(log(selling_price) ~., data = car, psi =
psi.huber)
summary(robust_huber.lm)

##
## Call: rlm(formula = log(selling_price) ~ ., data = car, psi =
psi.huber)
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2.66069 -0.34494  0.02092  0.33193  2.55281
##
## Coefficients:
##                               Value    Std. Error t value
## (Intercept)                    14.5929   0.0808   180.6820
## ManufacturerJapan              -0.0903   0.0274    -3.2927
## Manufacturerother Asia         -0.2444   0.0275    -8.8787
## Manufacturerother Europe        0.0060   0.0350     0.1708
## ManufacturerUS                 -0.3845   0.0326   -11.7882
## Years                           0.0045   0.0007     6.2638
## Mileage                          -0.0015   0.0001   -21.2444
## Engine                           -0.0019   0.0002   -12.0162
## km_driven                         0.0000   0.0000   -36.6445
## fuelDiesel                        0.5032   0.0731     6.8796
## fuelLPG                          -0.3931   0.1138    -3.4530
## fuelPetrol                       -0.2252   0.0730    -3.0845
## seller_typeIndividual            -0.1594   0.0188    -8.4924
## seller_typeTrustmark Dealer       0.0104   0.0385     0.2713
## transmissionManual               -0.7082   0.0209   -33.9499
## ownerFourth & Above Owner        -0.7362   0.0426   -17.2815
## ownerSecond Owner                -0.3575   0.0145   -24.6479
## ownerTest Drive Car               1.6044   0.3010     5.3301
## ownerThird Owner                 -0.5328   0.0252   -21.1097
##
## Residual standard error: 0.5033 on 7875 degrees of freedom
```

```r
plot(robust_huber.lm)
```

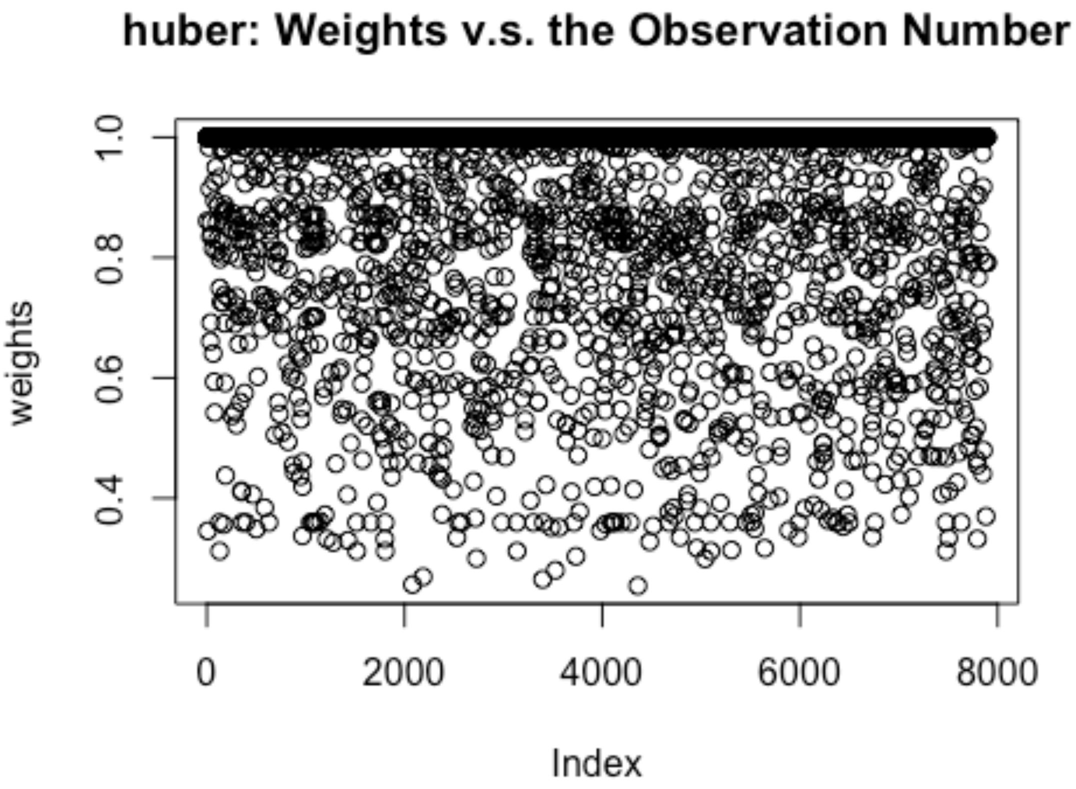

Residuals vs Fitted
Residuals
3
2
1
0
-1
-2
-3
3400
2082
4365
10
11
12
13
14
15
Fitted values
rlm(log(selling_price) ~ .)


Normal Q-Q
Standardized residuals
6
4
2
0
-2
-4
3400
2582
-4
-2
0
2
4
Theoretical Quantiles
rlm(log(selling_price) ~ .)


Scale-Location
√|Standardized residuals|
0.0
0.5
1.0
1.5
2.0
3400
2082
4365
10
11
12
13
14
15
Fitted values
rlm(log(selling_price) ~ .)



```r
weights <- robust_huber.lm$w
plot(weights, main = "huber: Weights v.s. the Observation Number")
```



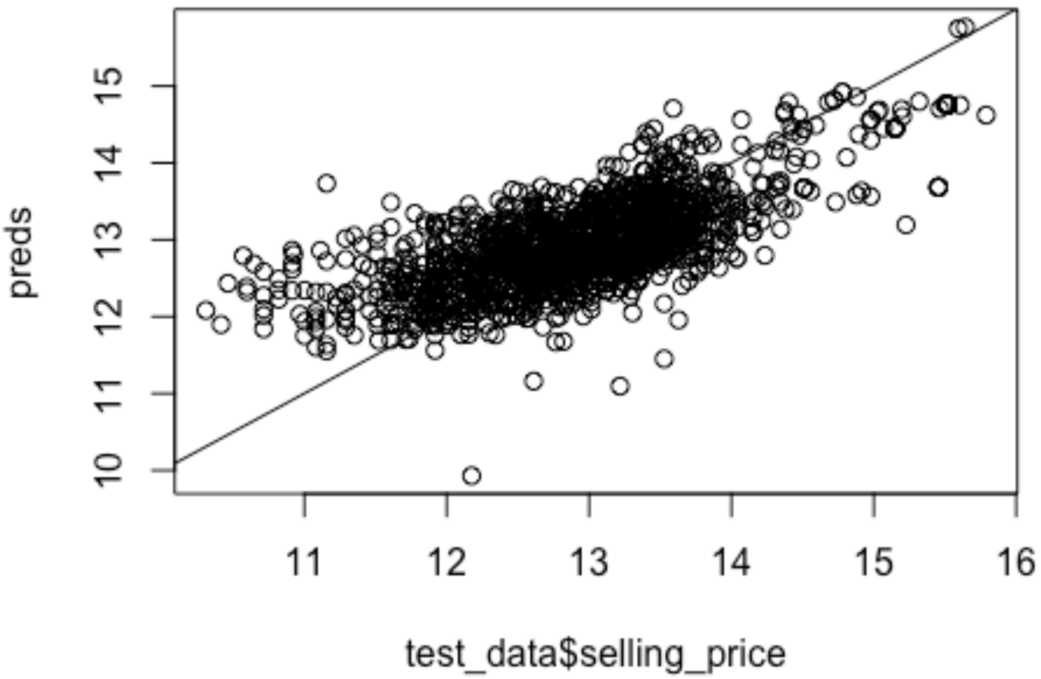
Prediction: Cross Validation

```
# Split data into 80% for training the model and 20% of the data for
testing the model
set.seed(1168)
nsamp = ceiling(0.8 * length(car$selling_price))
training_samps = sample(c(1:length(car$selling_price)), nsamp)
training_samps = sort(training_samps)
train_data <- car[training_samps, ]
test_data <- car[-training_samps, ]

# Fit the log model using the training data
train.lm <- lm(log(selling_price) ~ ., data = train_data)
summary(train.lm)
```

```
##
## Call:
## lm(formula = log(selling_price) ~ ., data = train_data)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2.58555 -0.32670  0.03861  0.34589  2.13617
##
## Coefficients:
##                               Estimate Std. Error t value Pr(>|t|)
## (Intercept)                  1.463e+01  9.208e-02 158.902  < 2e-16
***
## ManufacturerJapan           -7.483e-02  3.184e-02  -2.350 0.018797
*
## Manufacturerother Asia      -2.396e-01  3.195e-02  -7.499 7.33e-14
***
## Manufacturerother Europe    -2.182e-02  4.041e-02  -0.540 0.589311
## ManufacturerUS              -3.327e-01  3.790e-02  -8.780  < 2e-16
***
## Years                        4.638e-03  8.295e-04   5.591 2.36e-08
***
## Mileage                     -1.249e-03  8.093e-05 -15.432  < 2e-16
***
## Engine                      -1.674e-03  1.821e-04  -9.192  < 2e-16
***
## km_driven                   -4.737e-06  1.758e-07 -26.942  < 2e-16
***
## fuelDiesel                   4.497e-01  8.308e-02   5.414 6.41e-08
***
## fuelLPG                     -4.468e-01  1.274e-01  -3.507 0.000456
***
## fuelPetrol                  -2.668e-01  8.287e-02  -3.219 0.001292
**
## seller_typeIndividual       -1.946e-01  2.179e-02  -8.929  < 2e-16
***
## seller_typeTrustmark Dealer  1.119e-02  4.425e-02   0.253 0.800368
## transmissionManual          -8.079e-01  2.433e-02 -33.206  < 2e-16
***
## ownerFourth & Above Owner   -7.136e-01  4.946e-02 -14.428  < 2e-16
***
## ownerSecond Owner           -3.799e-01  1.687e-02 -22.511  < 2e-16
```

```
***
## ownerTest Drive Car          1.654e+00  5.417e-01   3.053 0.002272
**
## ownerThird Owner            -5.513e-01  2.969e-02 -18.571  < 2e-16
***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.5407 on 6297 degrees of freedom
## Multiple R-squared:  0.5726, Adjusted R-squared:  0.5714
## F-statistic: 468.8 on 18 and 6297 DF,  p-value: < 2.2e-16
```

```r
test_data$selling_price = log(test_data$selling_price)

# Predict the selling price using the testing data
preds <- predict(train.lm, test_data)
plot(test_data$selling_price, preds)
abline(c(0,1))
```



```
# Evaluate the quality of our prediction
R.sq = r2(preds, test_data$selling_price)

## 'r2()' does not support models of class 'numeric'.

RMSPE = rmse(preds, test_data$selling_price)
MAPE = mae(preds, test_data$selling_price)
print(c(R.sq, RMSPE, MAPE))

## [1]       NA 0.578001 0.443486
```