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

# Introduce one new additional data point into our assigned dataset
one_new <- data.frame("Maruti Swift Dzire VDi", 2009, 270000, 150000,
"Diesel", "Individual", "Manual",
                      "Second Owner", "19.5 kmpl", "1248 CC", "74
bhp", "190Nm@ 2000rpm",5)
names(one_new) <- c("name", "year", "selling_price", "km_driven",
"fuel", "seller_type", "transmission",
                    "owner", "mileage", "engine", "max_power",
"torque", "seats")
Car_details_v3 <- rbind(Car_details_v3, one_new)

# 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)
```

```r
# 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

```r
summary(car$Manufacturer)
```

```
##      Germany        Japan   other Asia other Europe           US
##          501         3420         2916          417          653
```

```
summary(car$fuel)
```

```
##    CNG Diesel    LPG Petrol
##     52   4300     35   3520
```

```
summary(car$seller_type)
```

```
##           Dealer       Individual Trustmark Dealer
##             1107             6564              236
```

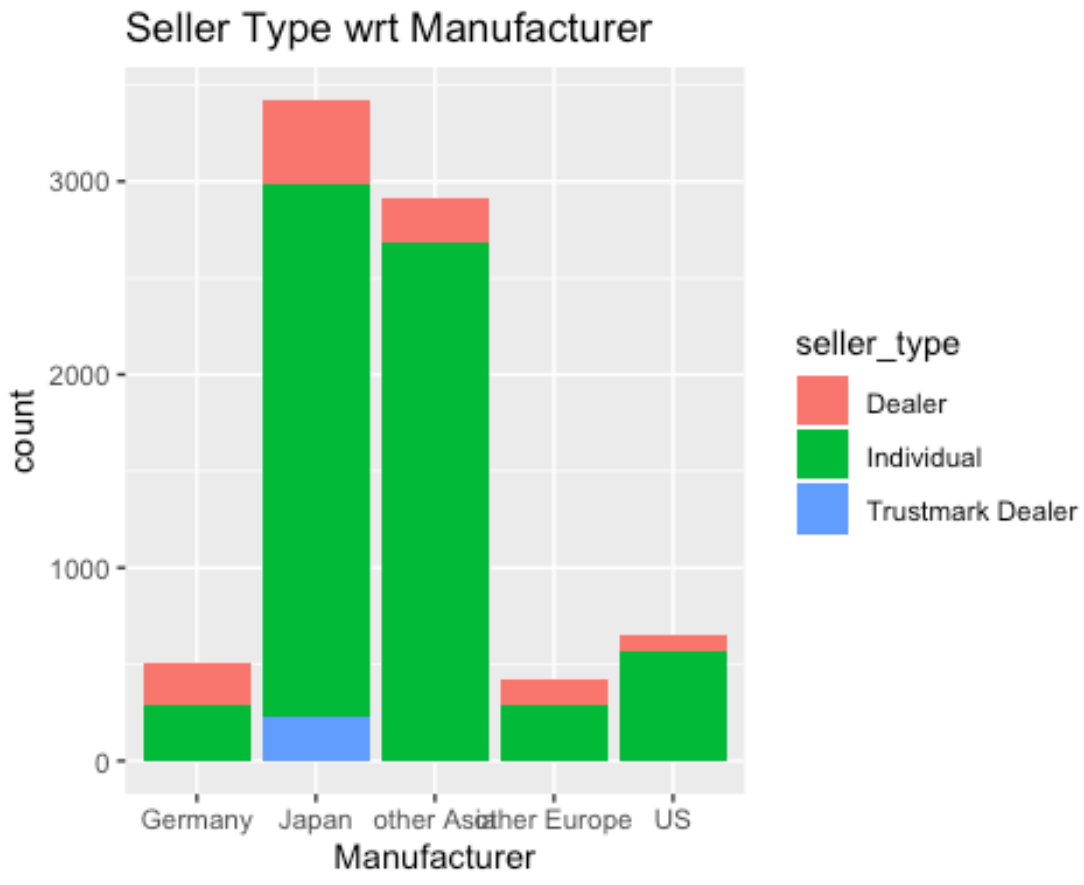
```
summary(car$owner)
```

```
##          First Owner Fourth & Above Owner         Second Owner
##                 5215                  160                 2017
##       Test Drive Car          Third Owner
##                    5                  510
```

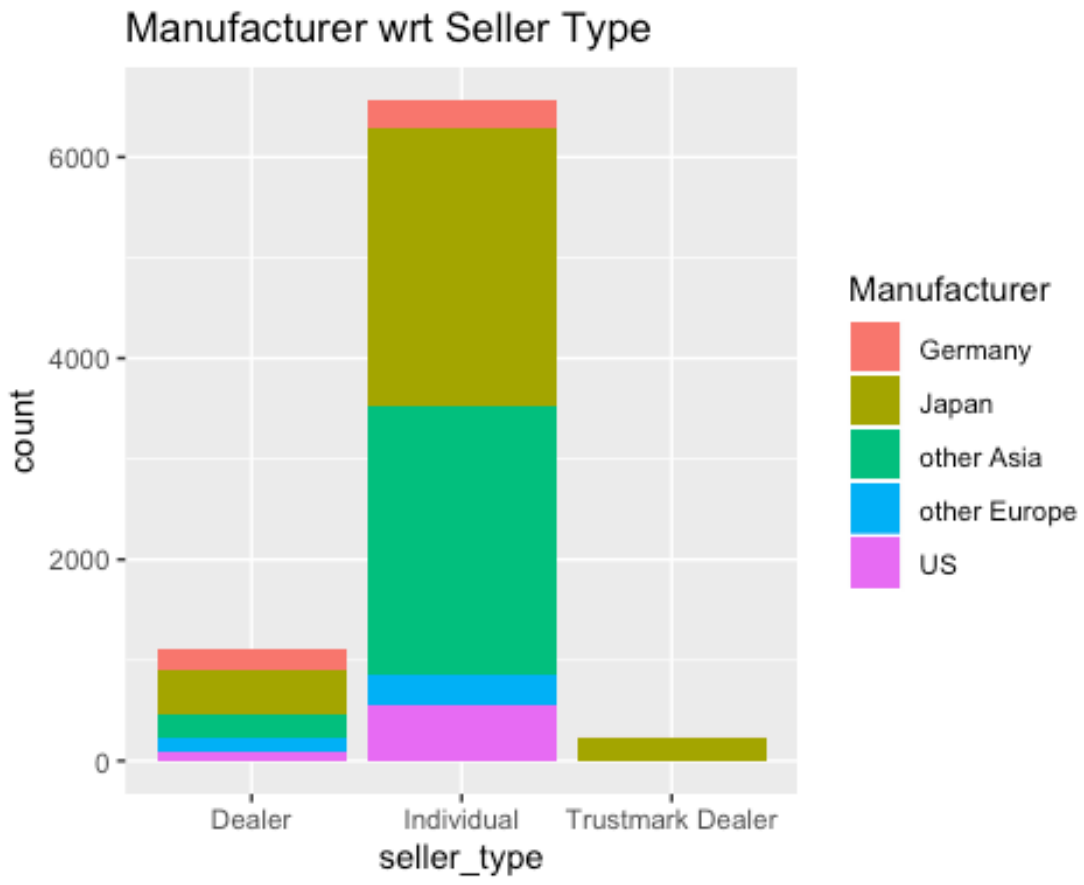
```
summary(car$transmission)
```

```
## Automatic    Manual
##      1041      6866
```

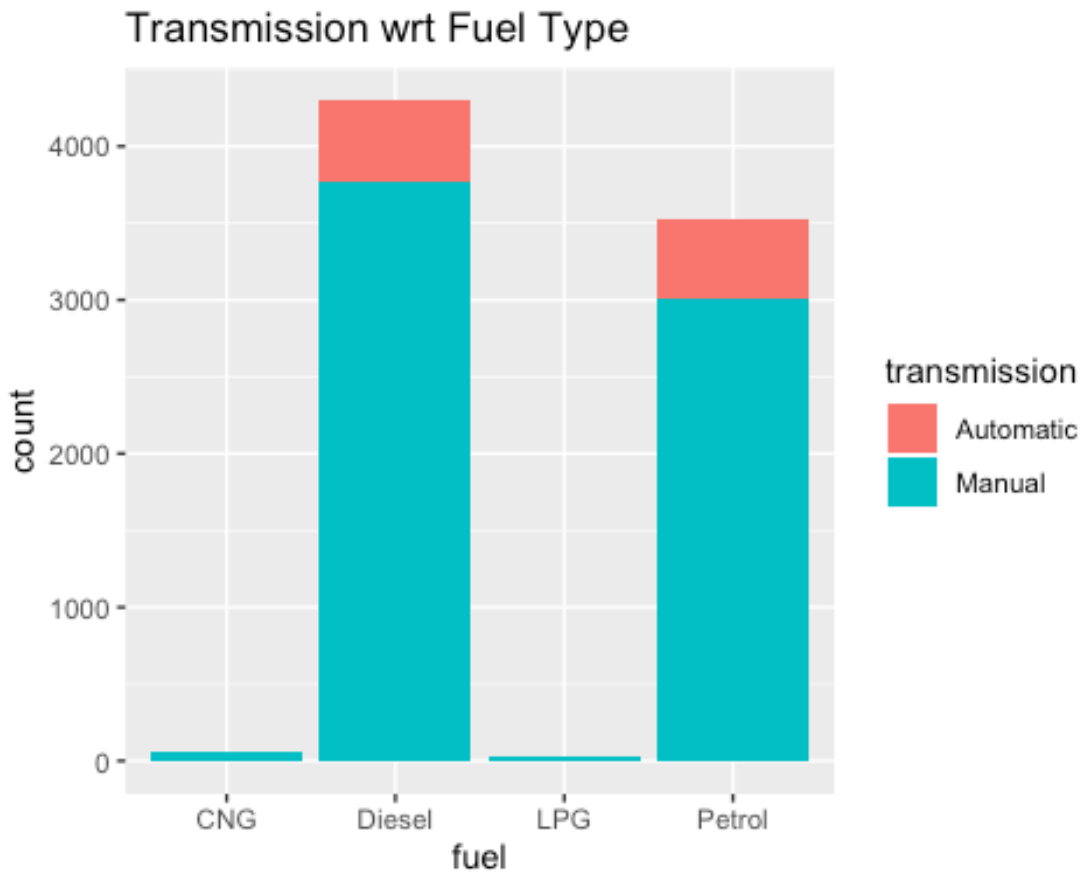
```
ggplot(car) +
  geom_bar(mapping = aes(x = Manufacturer, fill = seller_type)) +
  ggtitle("Seller Type wrt Manufacturer")
```



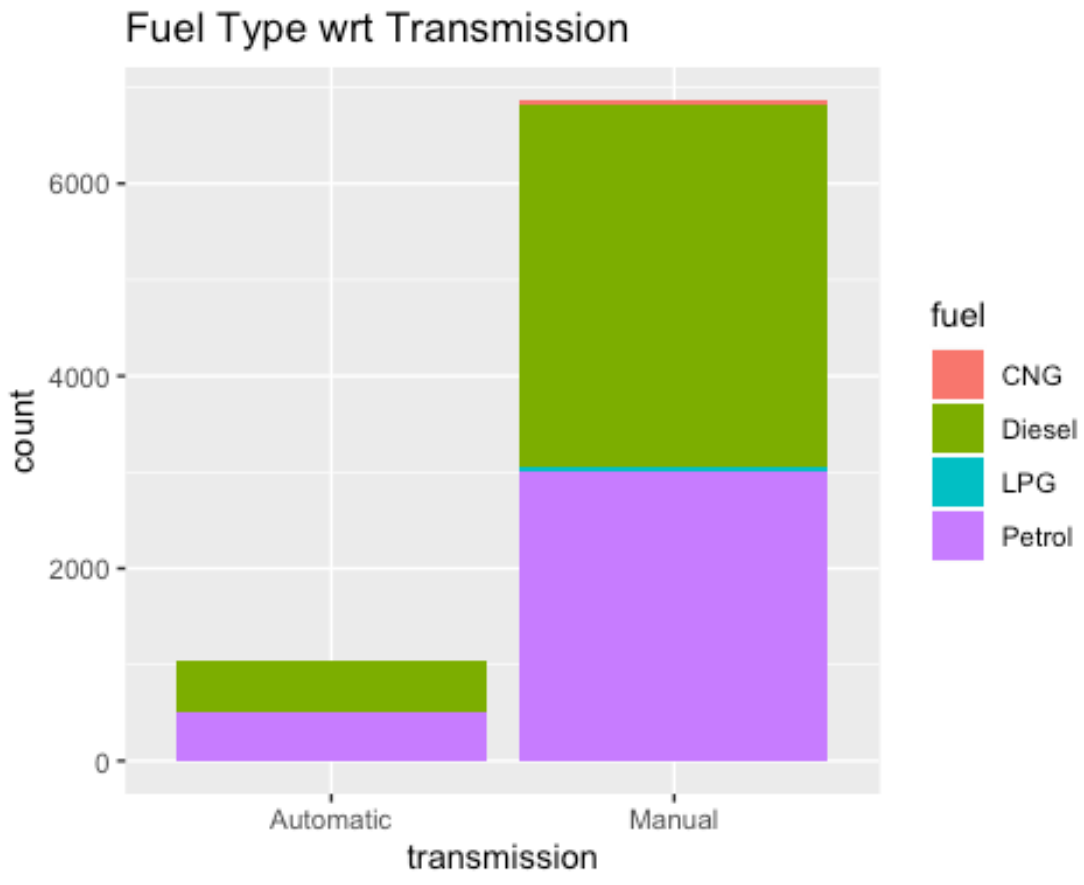

```
ggplot(car) +
  geom_bar(mapping = aes(x = seller_type, fill = Manufacturer)) +
  ggtitle("Manufacturer wrt Seller Type")
```



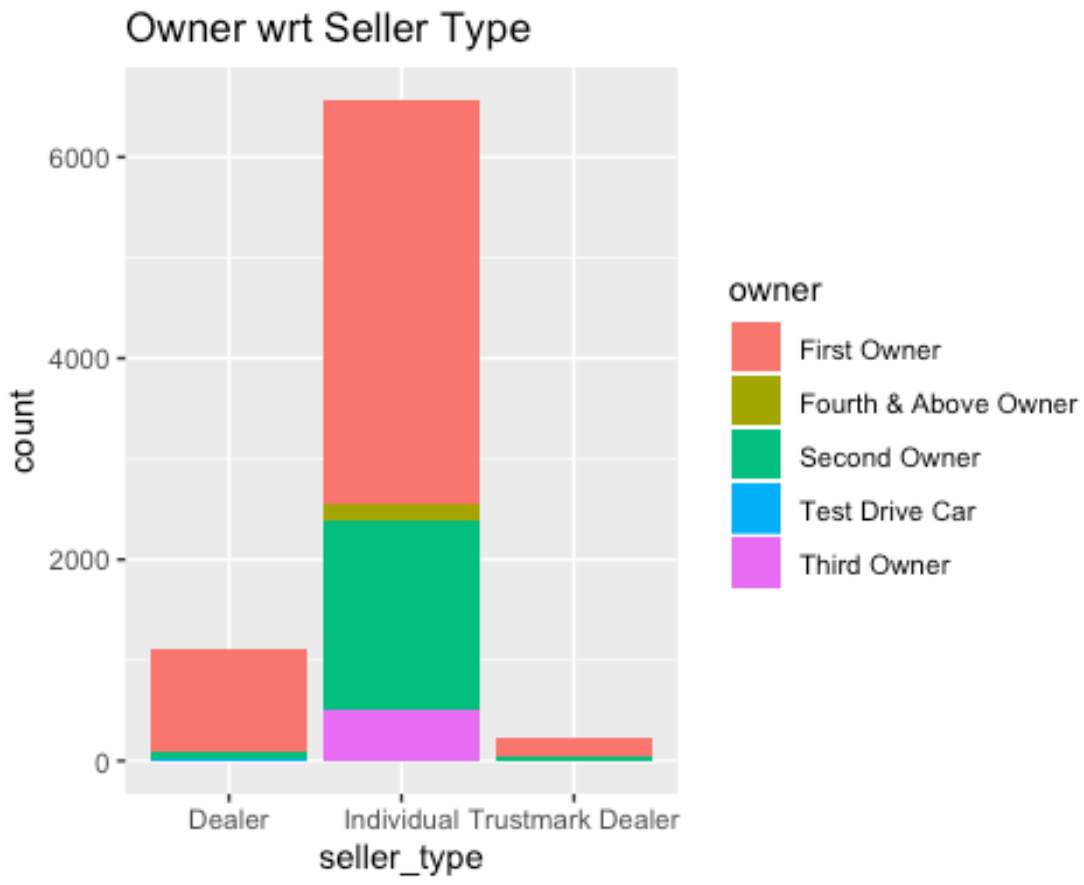

```r
ggplot(car) +
  geom_bar(mapping = aes(x = fuel, fill = transmission)) +
  ggtitle("Transmission wrt Fuel Type")
```



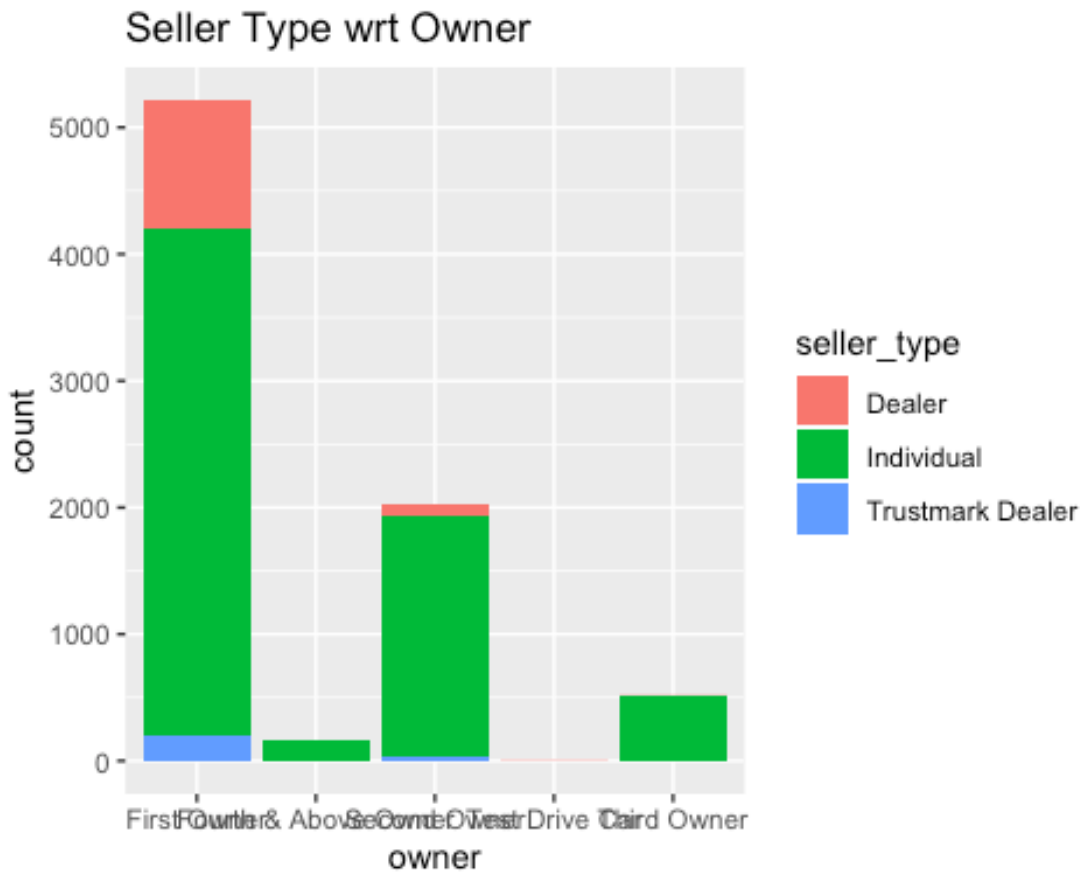

```
ggplot(car) +
  geom_bar(mapping = aes(x = transmission, fill = fuel)) +
  ggtitle("Fuel Type wrt Transmission")
```



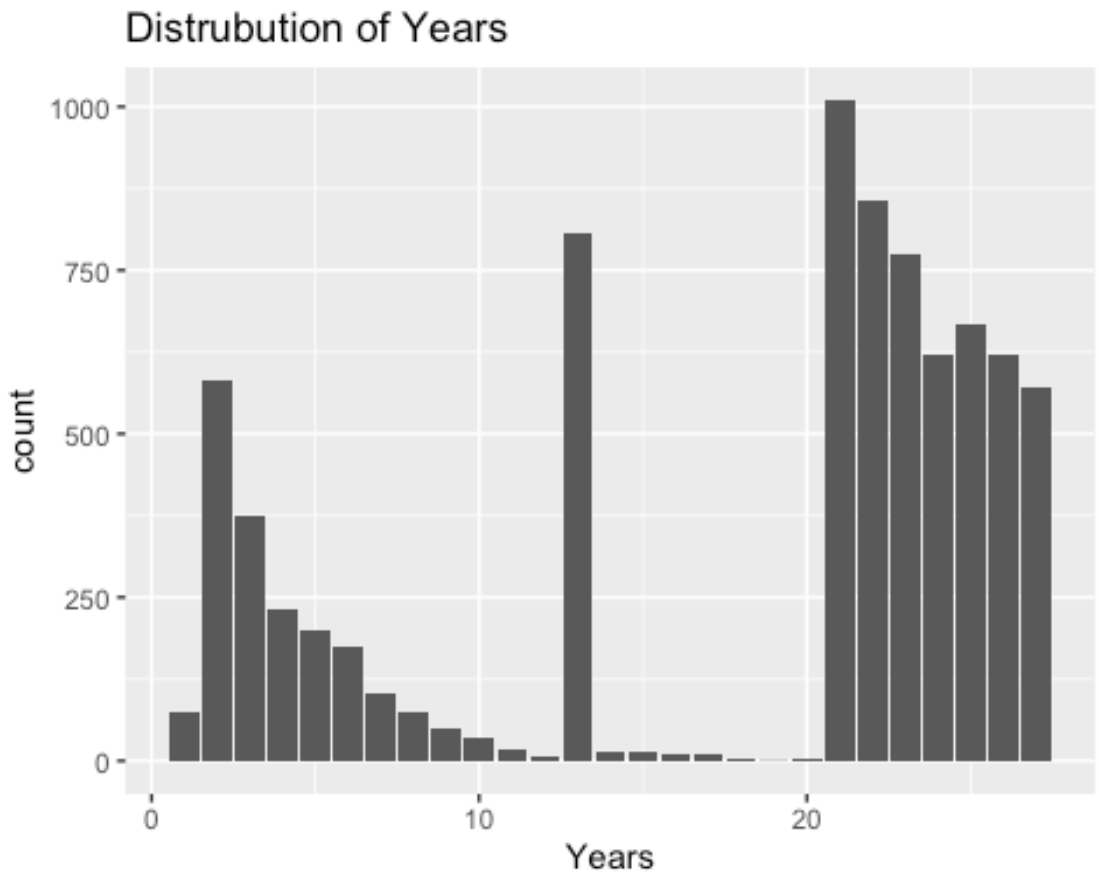

```r
ggplot(car) +
  geom_bar(mapping = aes(x = seller_type, fill = owner)) +
  ggtitle("Owner wrt Seller Type")
```



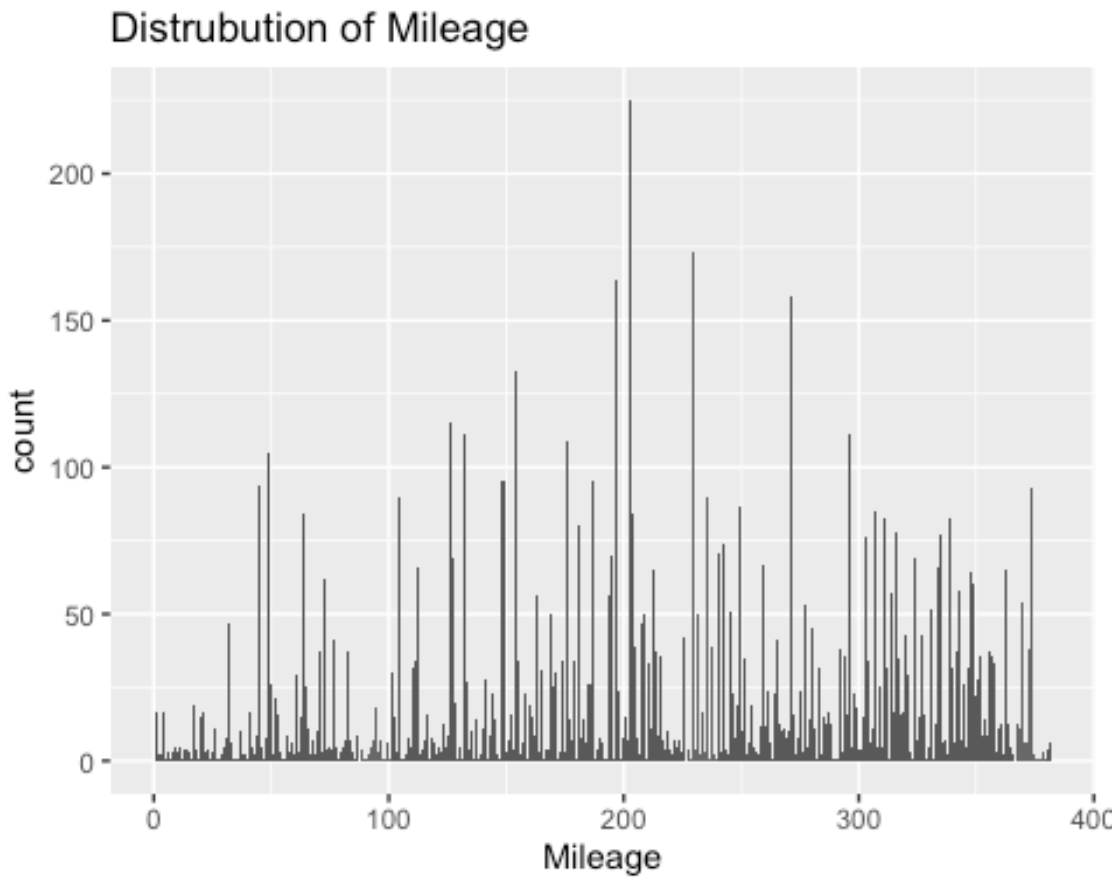

```
ggplot(car) +
  geom_bar(mapping = aes(x = owner, fill = seller_type)) +
  ggtitle("Seller Type wrt Owner")
```



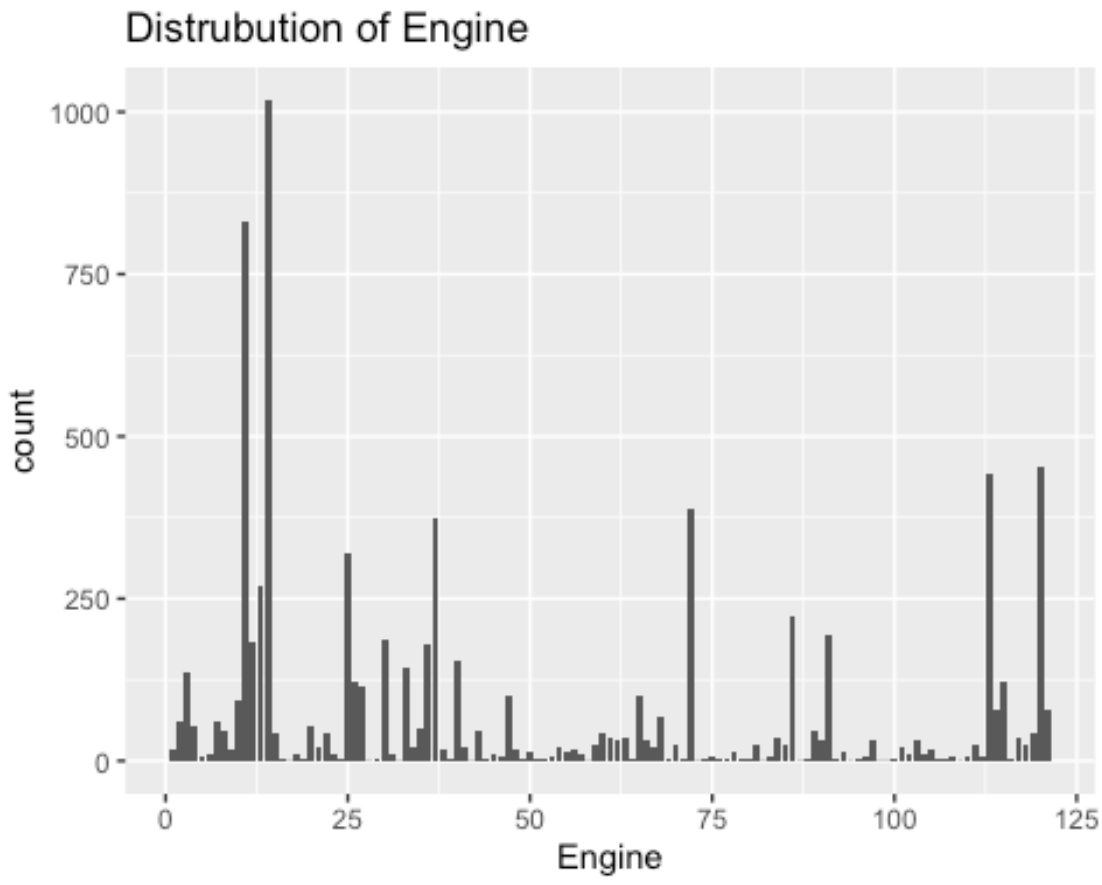

```r
ggplot(car) +
  geom_bar(mapping = aes(x = Years))+
  ggtitle("Distrubution of Years")
```



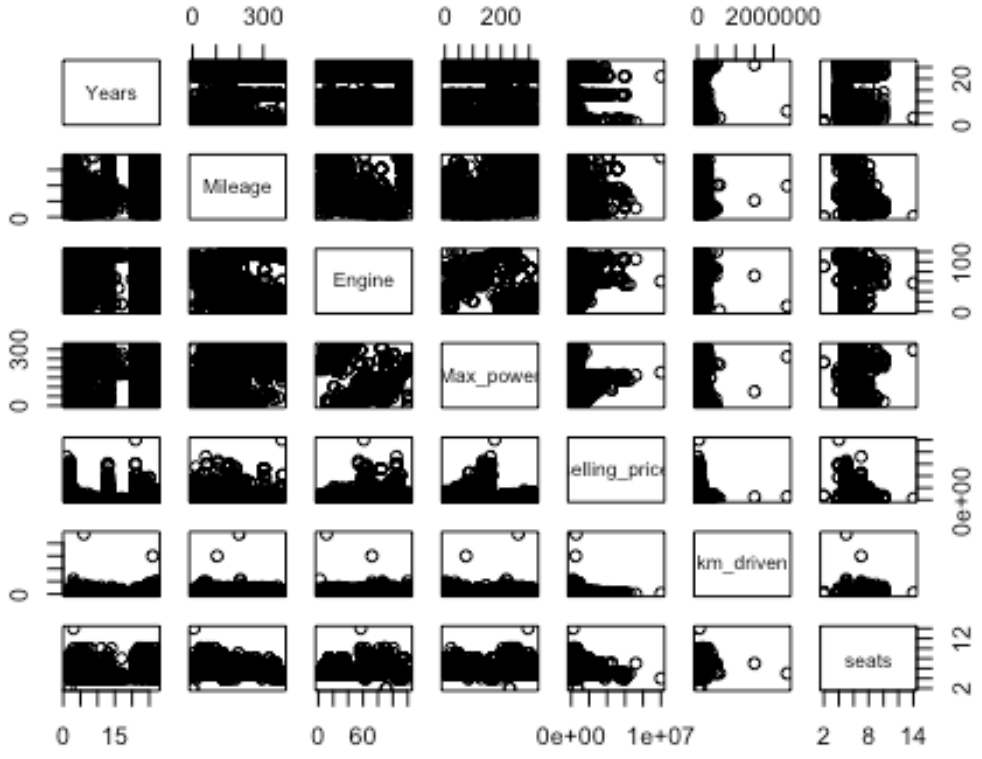

```
ggplot(car) +
  geom_bar(mapping = aes(x = Mileage))+
  ggtitle("Distrubution of Mileage")
```




```r
ggplot(car) +
  geom_bar(mapping = aes(x = Engine))+
  ggtitle("Distrubution of Engine")
```




```
# 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
## -2640476  -252377   -14143   199644  8535393
##
```

```
## Coefficients:
##                                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)                    2.549e+06  9.960e+04  25.589  < 2e-16
***
## ManufacturerJapan             -4.054e+05  3.079e+04 -13.169  < 2e-16
***
## Manufacturerother Asia        -4.947e+05  3.080e+04 -16.062  < 2e-16
***
## Manufacturerother Europe      -1.407e+05  3.812e+04  -3.690 0.000226
***
## ManufacturerUS                -4.940e+05  3.623e+04 -13.635  < 2e-16
***
## Years                         -1.056e+04  7.730e+02 -13.658  < 2e-16
***
## Mileage                       -6.894e+02  9.531e+01  -7.233 5.18e-13
***
## Engine                         1.309e+03  1.851e+02   7.072 1.66e-12
***
## Max_power                     -1.254e+02  9.002e+01  -1.393 0.163544
## km_driven                     -1.975e+00  1.276e-01 -15.484  < 2e-16
***
## fuelDiesel                     3.223e+05  7.955e+04   4.052 5.13e-05
***
## fuelLPG                       -8.412e+04  1.237e+05  -0.680 0.496408
## fuelPetrol                    -1.157e+05  7.933e+04  -1.459 0.144737
## seller_typeIndividual         -3.517e+05  2.032e+04 -17.308  < 2e-16
***
## seller_typeTrustmark Dealer -3.673e+05  4.175e+04  -8.797  < 2e-16
***
## transmissionManual            -9.406e+05  2.268e+04 -41.471  < 2e-16
***
## ownerFourth & Above Owner     -3.484e+05  4.618e+04  -7.543 5.09e-14
***
## ownerSecond Owner             -2.011e+05  1.573e+04 -12.786  < 2e-16
***
## ownerTest Drive Car            2.244e+06  2.533e+05   8.856  < 2e-16
***
## ownerThird Owner              -2.723e+05  2.729e+04  -9.976  < 2e-16
***
## seats                          2.129e+03  9.214e+03   0.231 0.817279
## ---
```

```
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 563200 on 7886 degrees of freedom
## Multiple R-squared:  0.5219, Adjusted R-squared:  0.5207
## F-statistic: 430.4 on 20 and 7886 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=209421.6
## selling_price ~ Manufacturer + Years + Mileage + Engine + Max_power
+
##     km_driven + fuel + seller_type + transmission + owner + seats
##
##                Df  Sum of Sq        RSS    AIC
## - seats         1 1.6936e+10 2.5018e+15 209420
## - Max_power     1 6.1593e+11 2.5024e+15 209422
## <none>                       2.5018e+15 209422
## - Engine        1 1.5867e+13 2.5177e+15 209470
## - Mileage       1 1.6595e+13 2.5184e+15 209472
## - Years         1 5.9180e+13 2.5610e+15 209604
## - km_driven     1 7.6058e+13 2.5779e+15 209656
## - seller_type   2 9.7332e+13 2.5991e+15 209719
## - owner         4 1.0227e+14 2.6041e+15 209730
## - Manufacturer  4 1.0498e+14 2.6068e+15 209739
## - fuel          3 2.2367e+14 2.7255e+15 210093
## - transmission  1 5.4560e+14 3.0474e+15 210979
##
## Step:  AIC=209419.6
## selling_price ~ Manufacturer + Years + Mileage + Engine + Max_power
+
##     km_driven + fuel + seller_type + transmission + owner
##
##                Df  Sum of Sq        RSS    AIC
## - Max_power     1 6.1109e+11 2.5024e+15 209420
## <none>                       2.5018e+15 209420
## - Engine        1 1.7075e+13 2.5189e+15 209471
## - Mileage       1 2.3197e+13 2.5250e+15 209491
## - Years         1 5.9317e+13 2.5611e+15 209603
```

```
## - km_driven     1 7.6065e+13 2.5779e+15 209654
## - seller_type   2 9.7434e+13 2.5992e+15 209718
## - owner         4 1.0439e+14 2.6062e+15 209735
## - Manufacturer  4 1.0986e+14 2.6117e+15 209751
## - fuel          3 2.8871e+14 2.7905e+15 210277
## - transmission  1 5.4909e+14 3.0509e+15 210987
##
## Step:  AIC=209419.6
## selling_price ~ Manufacturer + Years + Mileage + Engine + km_driven
+
##     fuel + seller_type + transmission + owner
##
##                Df  Sum of Sq        RSS    AIC
## <none>                       2.5024e+15 209420
## - Engine        1 2.1835e+13 2.5243e+15 209486
## - Mileage       1 3.2163e+13 2.5346e+15 209519
## - Years         1 5.8994e+13 2.5614e+15 209602
## - km_driven     1 7.6209e+13 2.5786e+15 209655
## - seller_type   2 9.7790e+13 2.6002e+15 209719
## - owner         4 1.0429e+14 2.6067e+15 209734
## - Manufacturer  4 1.1331e+14 2.6157e+15 209762
## - fuel          3 3.0902e+14 2.8114e+15 210334
## - transmission  1 5.5652e+14 3.0589e+15 211005

##
## Call:
## lm(formula = selling_price ~ Manufacturer + Years + Mileage +
##     Engine + km_driven + fuel + seller_type + transmission +
##     owner, data = car3)
##
## Coefficients:
##                 (Intercept)            ManufacturerJapan
##                   2.552e+06                   -4.094e+05
##      Manufacturerother Asia     Manufacturerother Europe
##                  -4.972e+05                   -1.435e+05
##              ManufacturerUS                        Years
##                  -5.023e+05                   -1.051e+04
##                     Mileage                       Engine
##                  -7.485e+02                    1.404e+03
##                   km_driven                   fuelDiesel
```

```
##                   -1.977e+00                    3.227e+05
##                      fuelLPG                   fuelPetrol
##                   -9.040e+04                   -1.216e+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
##                   -2.011e+05                    2.249e+06
##            ownerThird Owner
##                   -2.729e+05
```

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

```
## Start:  AIC=215216.1
## selling_price ~ 1
##
##                Df  Sum of Sq        RSS    AIC
## + transmission  1 1.8232e+15 3.4094e+15 211831
## + Manufacturer  4 9.5123e+14 4.2814e+15 213638
## + seller_type   2 8.6455e+14 4.3680e+15 213792
## + owner         4 3.9028e+14 4.8423e+15 214611
## + km_driven     1 2.5837e+14 4.9742e+15 214818
## + Max_power     1 2.5349e+14 4.9791e+15 214826
## + fuel          3 2.2295e+14 5.0096e+15 214878
## + Years         1 1.3052e+14 5.1021e+15 215018
## + Mileage       1 8.2943e+13 5.1497e+15 215092
## + Engine        1 3.6732e+13 5.1959e+15 215162
## + seats         1 9.0733e+12 5.2235e+15 215204
## <none>                       5.2326e+15 215216
##
## Step:  AIC=211831
## selling_price ~ transmission
##
##                Df  Sum of Sq        RSS    AIC
## + fuel          3 2.5341e+14 3.1560e+15 211226
## + seller_type   2 2.4764e+14 3.1618e+15 211239
## + Manufacturer  4 2.3165e+14 3.1778e+15 211283
## + owner         4 1.6482e+14 3.2446e+15 211447
## + km_driven     1 5.8340e+13 3.3511e+15 211697
## + Max_power     1 5.2309e+13 3.3571e+15 211711
```

```
## + seats         1 3.7608e+13 3.3718e+15 211745
## + Years         1 2.0931e+13 3.3885e+15 211784
## + Engine        1 9.5935e+12 3.3998e+15 211811
## + Mileage       1 4.6084e+12 3.4048e+15 211822
## <none>                       3.4094e+15 211831
##
## Step:  AIC=211226.4
## selling_price ~ transmission + fuel
##
##                Df  Sum of Sq        RSS    AIC
## + seller_type   2 2.0761e+14 2.9484e+15 210692
## + owner         4 1.7973e+14 2.9763e+15 210771
## + Manufacturer  4 1.7401e+14 2.9820e+15 210786
## + km_driven     1 1.5484e+14 3.0012e+15 210831
## + Years         1 5.9660e+13 3.0963e+15 211077
## + Engine        1 2.7345e+13 3.1287e+15 211160
## + Max_power     1 2.1793e+13 3.1342e+15 211174
## + Mileage       1 7.0577e+12 3.1489e+15 211211
## <none>                       3.1560e+15 211226
## + seats         1 2.9026e+11 3.1557e+15 211228
##
## Step:  AIC=210692.3
## selling_price ~ transmission + fuel + seller_type
##
##                Df  Sum of Sq        RSS    AIC
## + Manufacturer  4 1.2869e+14 2.8197e+15 210347
## + owner         4 1.2180e+14 2.8266e+15 210367
## + km_driven     1 1.0910e+14 2.8393e+15 210396
## + Years         1 6.2191e+13 2.8862e+15 210526
## + Engine        1 3.3125e+13 2.9153e+15 210605
## + Max_power     1 2.3247e+13 2.9251e+15 210632
## + Mileage       1 1.2335e+13 2.9361e+15 210661
## + seats         1 3.2268e+12 2.9452e+15 210686
## <none>                       2.9484e+15 210692
##
## Step:  AIC=210347.5
## selling_price ~ transmission + fuel + seller_type + Manufacturer
##
##             Df  Sum of Sq        RSS    AIC
## + owner      4 1.1985e+14 2.6999e+15 210012
```

```
## + km_driven  1 1.0088e+14 2.7188e+15 210061
## + Years      1 6.6268e+13 2.7534e+15 210161
## + Engine     1 2.0821e+13 2.7989e+15 210291
## + seats      1 1.4847e+13 2.8049e+15 210308
## + Mileage    1 1.4105e+13 2.8056e+15 210310
## + Max_power  1 1.1475e+13 2.8082e+15 210317
## <none>                    2.8197e+15 210347
##
## Step:  AIC=210012
## selling_price ~ transmission + fuel + seller_type + Manufacturer +
##     owner
##
##              Df  Sum of Sq        RSS    AIC
## + Years      1 7.3876e+13 2.6260e+15 209795
## + km_driven  1 6.1576e+13 2.6383e+15 209832
## + Mileage    1 3.4690e+13 2.6652e+15 209912
## + Engine     1 2.7859e+13 2.6720e+15 209932
## + Max_power  1 2.0187e+13 2.6797e+15 209955
## + seats      1 1.4986e+13 2.6849e+15 209970
## <none>                    2.6999e+15 210012
##
## Step:  AIC=209794.6
## selling_price ~ transmission + fuel + seller_type + Manufacturer +
##     owner + Years
##
##              Df  Sum of Sq        RSS    AIC
## + km_driven  1 5.4234e+13 2.5717e+15 209632
## + Engine     1 3.1645e+13 2.5943e+15 209701
## + Mileage    1 2.6596e+13 2.5994e+15 209716
## + Max_power  1 2.0410e+13 2.6056e+15 209735
## + seats      1 1.5977e+13 2.6100e+15 209748
## <none>                    2.6260e+15 209795
##
## Step:  AIC=209631.6
## selling_price ~ transmission + fuel + seller_type + Manufacturer +
##     owner + Years + km_driven
##
##              Df  Sum of Sq        RSS    AIC
## + Mileage    1 4.7489e+13 2.5243e+15 209486
## + Engine     1 3.7161e+13 2.5346e+15 209519
```

```
## + Max_power  1 2.7630e+13 2.5441e+15 209548
## + seats      1 2.4541e+13 2.5472e+15 209558
## <none>                    2.5717e+15 209632
##
## Step:  AIC=209486.3
## selling_price ~ transmission + fuel + seller_type + Manufacturer +
##     owner + Years + km_driven + Mileage
##
##             Df  Sum of Sq        RSS    AIC
## + Engine     1 2.1835e+13 2.5024e+15 209420
## + Max_power  1 5.3711e+12 2.5189e+15 209471
## + seats      1 1.5391e+12 2.5227e+15 209483
## <none>                    2.5243e+15 209486
##
## Step:  AIC=209419.6
## selling_price ~ transmission + fuel + seller_type + Manufacturer +
##     owner + Years + km_driven + Mileage + Engine
##
##             Df  Sum of Sq        RSS    AIC
## <none>                    2.5024e+15 209420
## + Max_power  1 6.1109e+11 2.5018e+15 209420
## + seats      1 1.2093e+10 2.5024e+15 209422

##
## 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
##                   -1.216e+05                   -3.521e+05
## seller_typeTrustmark Dealer            ManufacturerJapan
##                   -3.677e+05                   -4.094e+05
##      Manufacturerother Asia     Manufacturerother Europe
##                   -4.972e+05                   -1.435e+05
```

```
##              ManufacturerUS    ownerFourth & Above Owner
##                   -5.023e+05                   -3.489e+05
##           ownerSecond Owner          ownerTest Drive Car
##                   -2.011e+05                    2.249e+06
##            ownerThird Owner                        Years
##                   -2.729e+05                   -1.051e+04
##                    km_driven                      Mileage
##                   -1.977e+00                   -7.485e+02
##                        Engine
##                    1.404e+03
```

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

```
## Start:  AIC=209421.6
## selling_price ~ Manufacturer + Years + Mileage + Engine + Max_power
+
##     km_driven + fuel + seller_type + transmission + owner + seats
##
##                 Df  Sum of Sq        RSS    AIC
## - seats          1 1.6936e+10 2.5018e+15 209420
## - Max_power      1 6.1593e+11 2.5024e+15 209422
## <none>                        2.5018e+15 209422
## - Engine         1 1.5867e+13 2.5177e+15 209470
## - Mileage        1 1.6595e+13 2.5184e+15 209472
## - Years          1 5.9180e+13 2.5610e+15 209604
## - km_driven      1 7.6058e+13 2.5779e+15 209656
## - seller_type    2 9.7332e+13 2.5991e+15 209719
## - owner          4 1.0227e+14 2.6041e+15 209730
## - Manufacturer   4 1.0498e+14 2.6068e+15 209739
## - fuel           3 2.2367e+14 2.7255e+15 210093
## - transmission   1 5.4560e+14 3.0474e+15 210979
##
## Step:  AIC=209419.6
## selling_price ~ Manufacturer + Years + Mileage + Engine + Max_power
+
##     km_driven + fuel + seller_type + transmission + owner
##
##                 Df  Sum of Sq        RSS    AIC
## - Max_power      1 6.1109e+11 2.5024e+15 209420
## <none>                        2.5018e+15 209420
## + seats          1 1.6936e+10 2.5018e+15 209422
```

```
## - Engine        1 1.7075e+13 2.5189e+15 209471
## - Mileage       1 2.3197e+13 2.5250e+15 209491
## - Years         1 5.9317e+13 2.5611e+15 209603
## - km_driven     1 7.6065e+13 2.5779e+15 209654
## - seller_type   2 9.7434e+13 2.5992e+15 209718
## - owner         4 1.0439e+14 2.6062e+15 209735
## - Manufacturer  4 1.0986e+14 2.6117e+15 209751
## - fuel          3 2.8871e+14 2.7905e+15 210277
## - transmission  1 5.4909e+14 3.0509e+15 210987
##
## Step:  AIC=209419.6
## selling_price ~ Manufacturer + Years + Mileage + Engine + km_driven
+
##     fuel + seller_type + transmission + owner
##
##                Df  Sum of Sq        RSS    AIC
## <none>                       2.5024e+15 209420
## + Max_power     1 6.1109e+11 2.5018e+15 209420
## + seats         1 1.2093e+10 2.5024e+15 209422
## - Engine        1 2.1835e+13 2.5243e+15 209486
## - Mileage       1 3.2163e+13 2.5346e+15 209519
## - Years         1 5.8994e+13 2.5614e+15 209602
## - km_driven     1 7.6209e+13 2.5786e+15 209655
## - seller_type   2 9.7790e+13 2.6002e+15 209719
## - owner         4 1.0429e+14 2.6067e+15 209734
## - Manufacturer  4 1.1331e+14 2.6157e+15 209762
## - fuel          3 3.0902e+14 2.8114e+15 210334
## - transmission  1 5.5652e+14 3.0589e+15 211005

##
## Call:
## lm(formula = selling_price ~ Manufacturer + Years + Mileage +
##     Engine + km_driven + fuel + seller_type + transmission +
##     owner, data = car3)
##
## Coefficients:
##                 (Intercept)            ManufacturerJapan
##                   2.552e+06                    -4.094e+05
##      Manufacturerother Asia     Manufacturerother Europe
##                  -4.972e+05                    -1.435e+05
```

```
##              ManufacturerUS                       Years
##                  -5.023e+05                  -1.051e+04
##                     Mileage                      Engine
##                  -7.485e+02                   1.404e+03
##                   km_driven                  fuelDiesel
##                  -1.977e+00                   3.227e+05
##                     fuelLPG                  fuelPetrol
##                  -9.040e+04                  -1.216e+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
##                  -2.011e+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
```

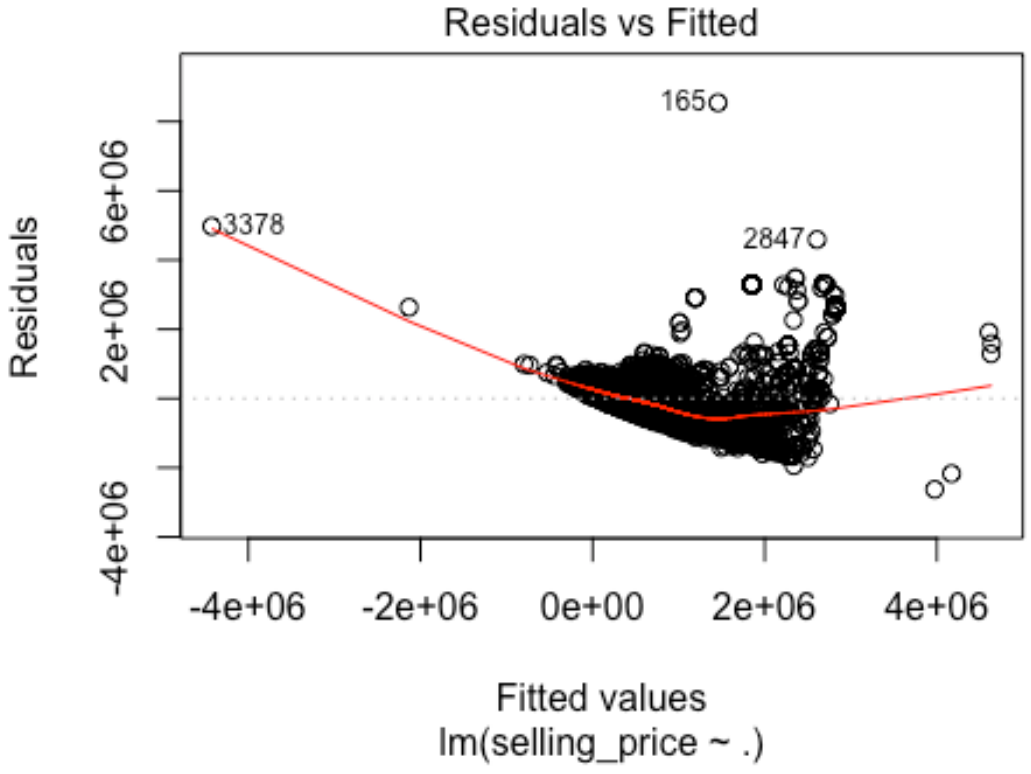
```
## 5       Manual  First Owner
## 6       Manual  First Owner
```

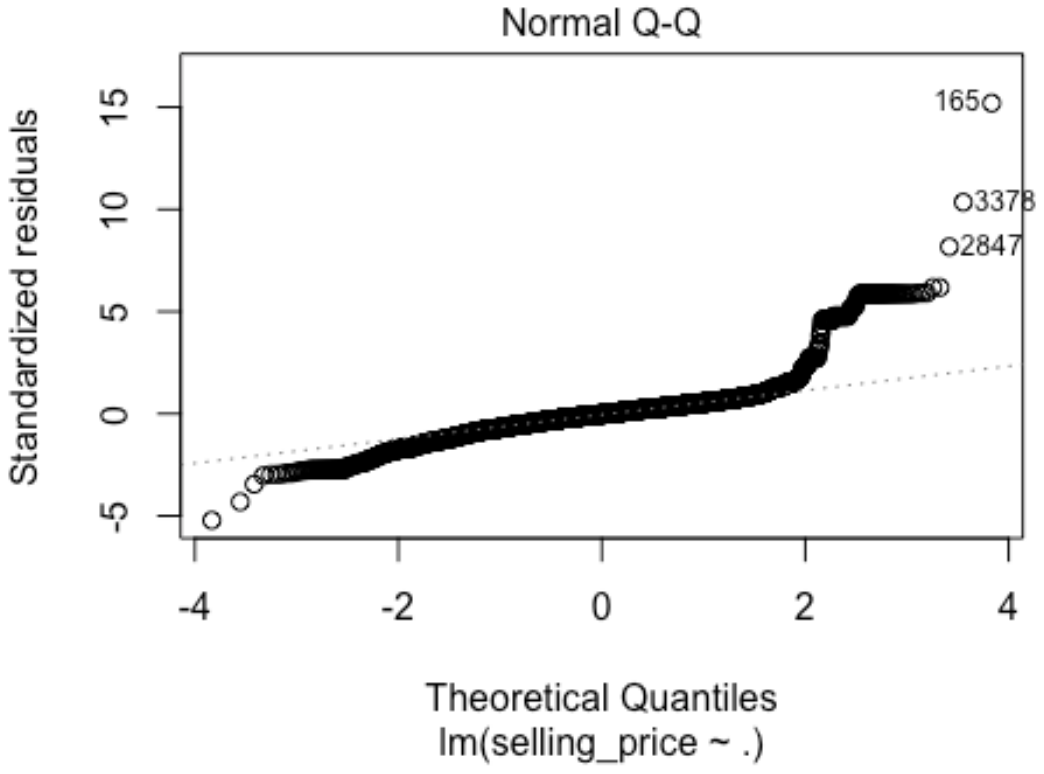
Data Transformation

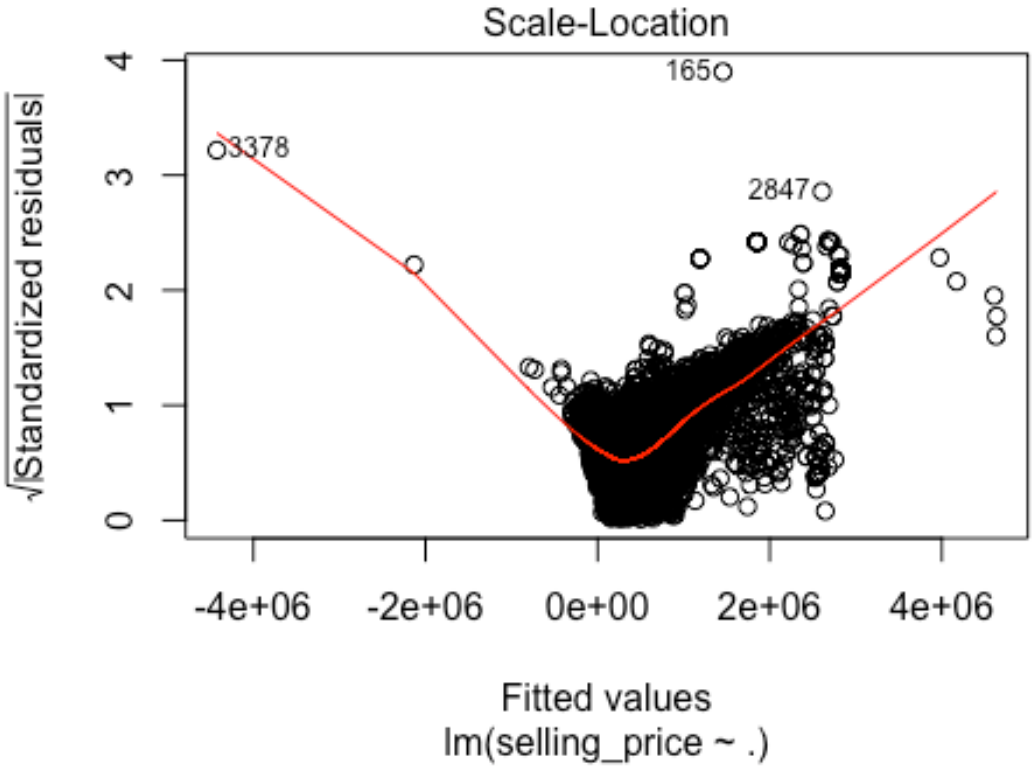
```
lm1 <- lm(selling_price ~ ., data = car4)
summary(lm1)

##
## Call:
## lm(formula = selling_price ~ ., data = car4)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2624986  -251957   -13936   198888  8543437
##
## Coefficients:
##                              Estimate Std. Error t value Pr(>|t|)
## (Intercept)                 2.552e+06  8.758e+04  29.136  < 2e-16
***
## ManufacturerJapan          -4.094e+05  2.966e+04 -13.804  < 2e-16
***
## Manufacturerother Asia     -4.972e+05  2.980e+04 -16.682  < 2e-16
***
## Manufacturerother Europe   -1.435e+05  3.793e+04  -3.782 0.000157
***
## ManufacturerUS             -5.023e+05  3.535e+04 -14.209  < 2e-16
***
## Years                      -1.051e+04  7.708e+02 -13.637  < 2e-16
***
## Mileage                    -7.485e+02  7.433e+01 -10.069  < 2e-16
***
## Engine                      1.404e+03  1.692e+02   8.296  < 2e-16
***
## km_driven                  -1.977e+00  1.275e-01 -15.499  < 2e-16
***
## fuelDiesel                  3.227e+05  7.937e+04   4.065 4.85e-05
***
## fuelLPG                    -9.040e+04  1.235e+05  -0.732 0.464337
## fuelPetrol                 -1.216e+05  7.921e+04  -1.535 0.124909
## seller_typeIndividual      -3.521e+05  2.029e+04 -17.349  < 2e-16
```

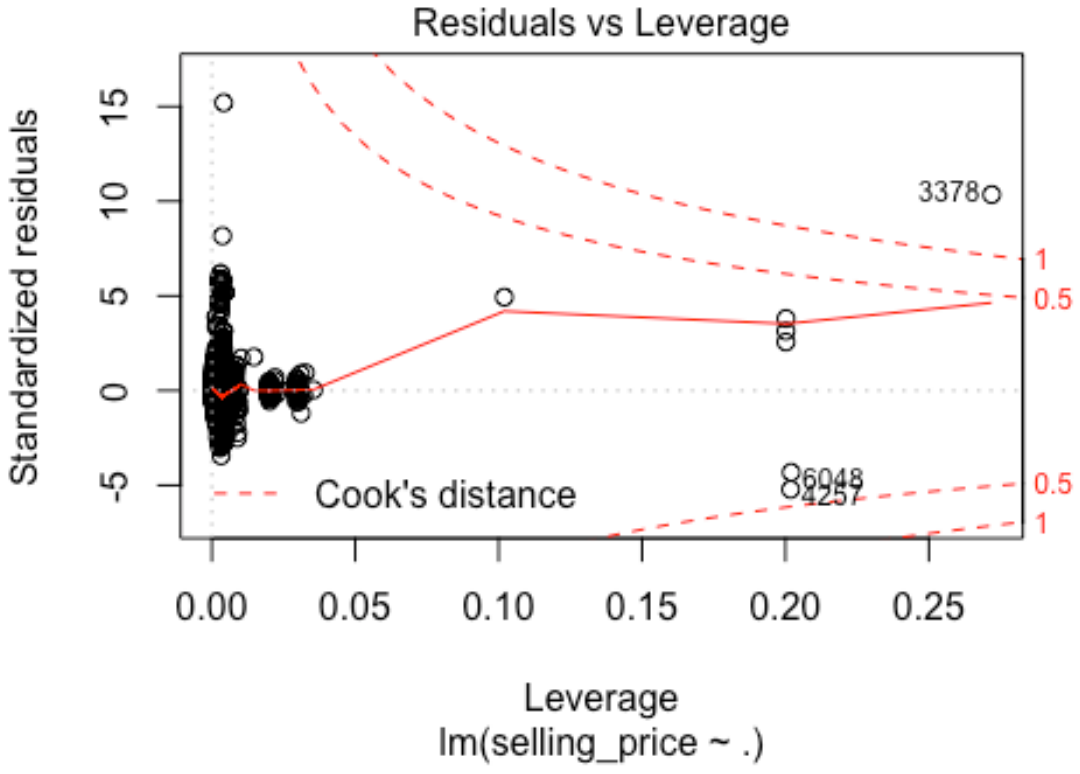
```
***
## seller_typeTrustmark Dealer -3.677e+05  4.174e+04  -8.808  < 2e-16
***
## transmissionManual          -9.429e+05  2.251e+04 -41.884  < 2e-16
***
## ownerFourth & Above Owner   -3.489e+05  4.609e+04  -7.570 4.17e-14
***
## ownerSecond Owner           -2.011e+05  1.562e+04 -12.877  < 2e-16
***
## ownerTest Drive Car          2.249e+06  2.533e+05   8.877  < 2e-16
***
## ownerThird Owner            -2.729e+05  2.717e+04 -10.045  < 2e-16
***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 563200 on 7888 degrees of freedom
## Multiple R-squared:  0.5218, Adjusted R-squared:  0.5207
## F-statistic: 478.1 on 18 and 7888 DF,  p-value: < 2.2e-16
```

```r
plot(lm1)
```


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

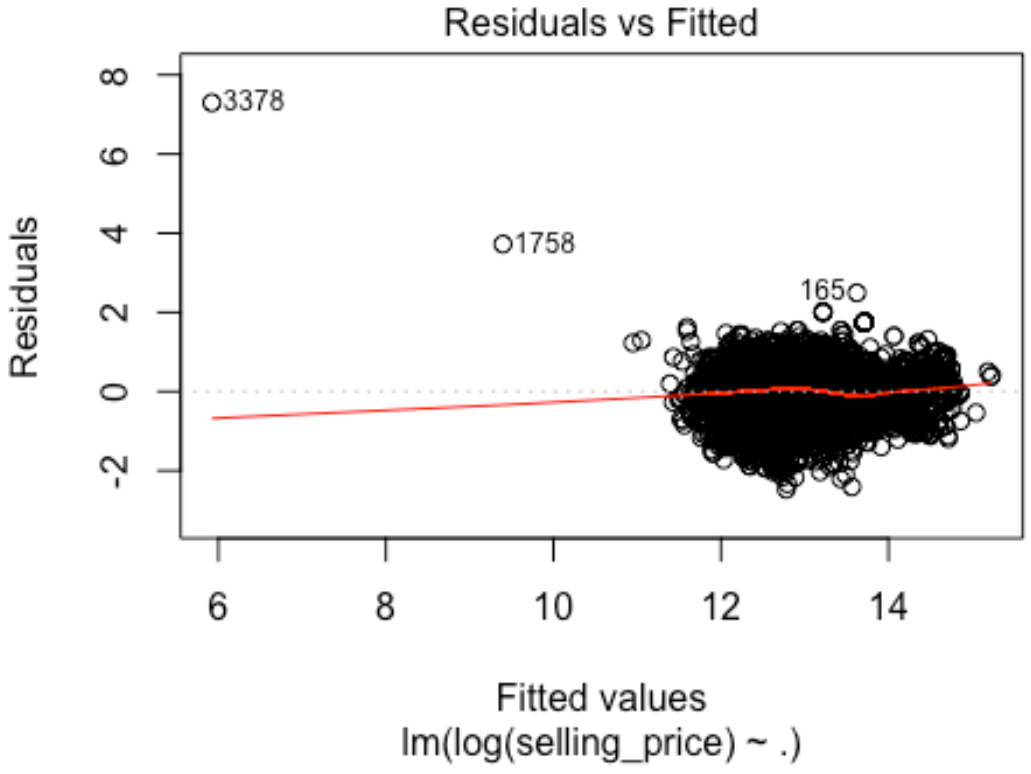

Normal Q-Q
Standardized residuals
15
10
5
0
-5
165
3378
2847
-4
-2
0
2
4
Theoretical Quantiles
lm(selling_price ~ .)

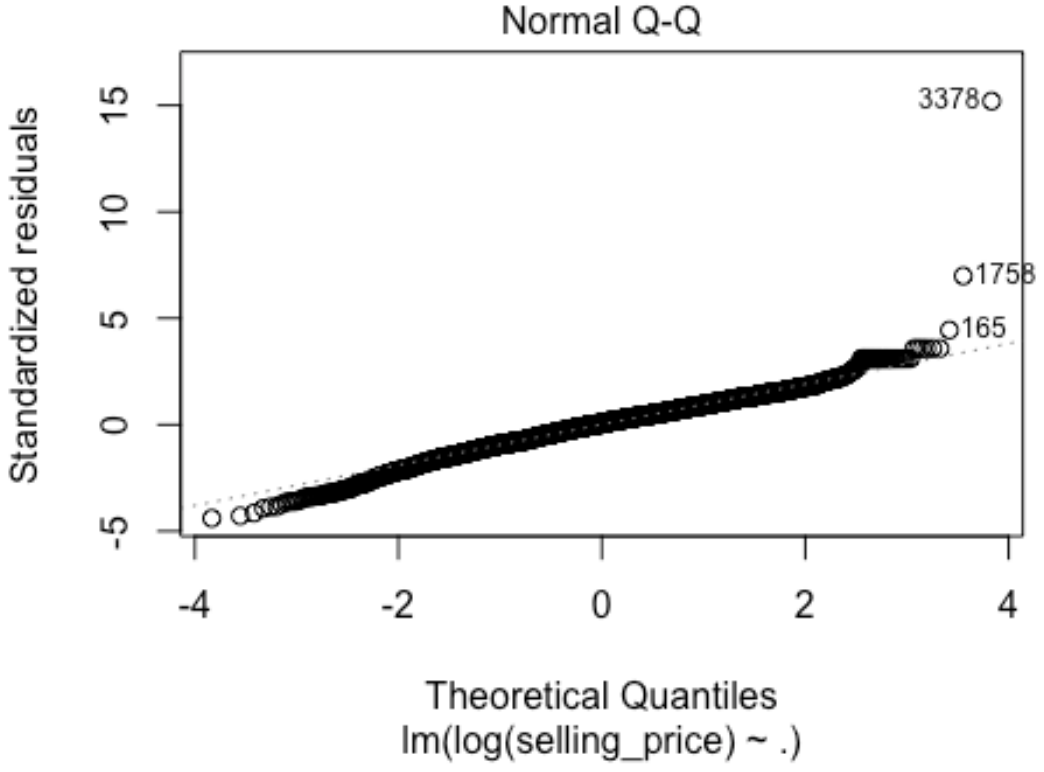

Scale-Location
√|Standardized residuals|
0
1
2
3
4
165
3378
2847
-4e+06
-2e+06
0e+00
2e+06
4e+06
Fitted values
lm(selling_price ~ .)

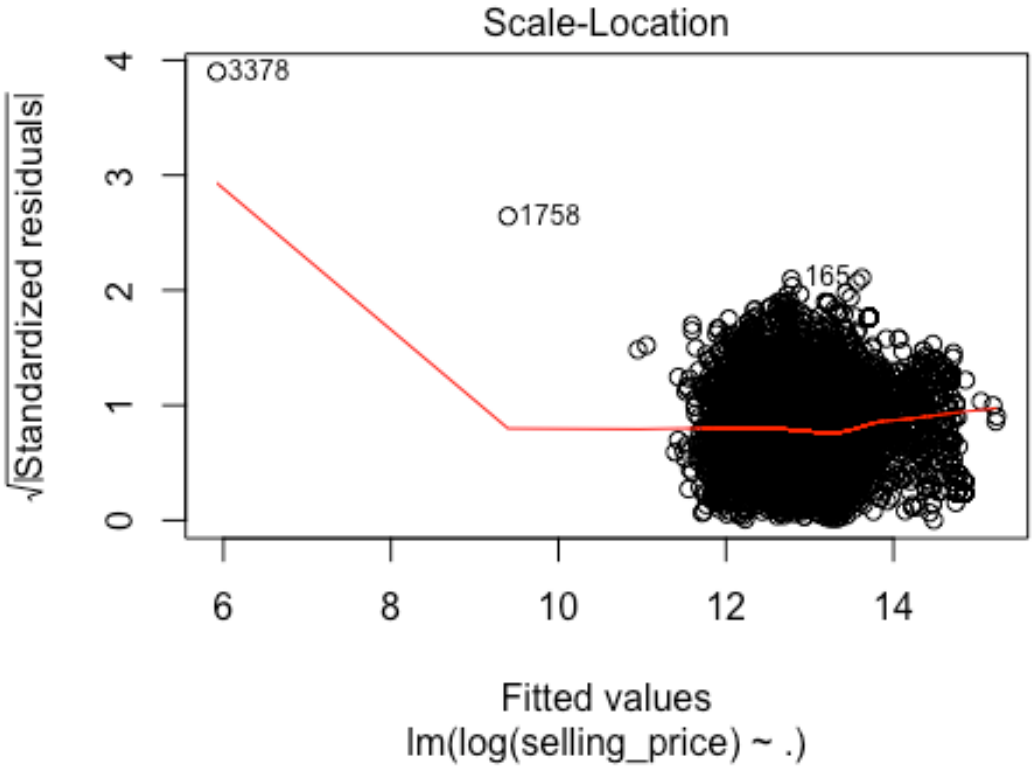


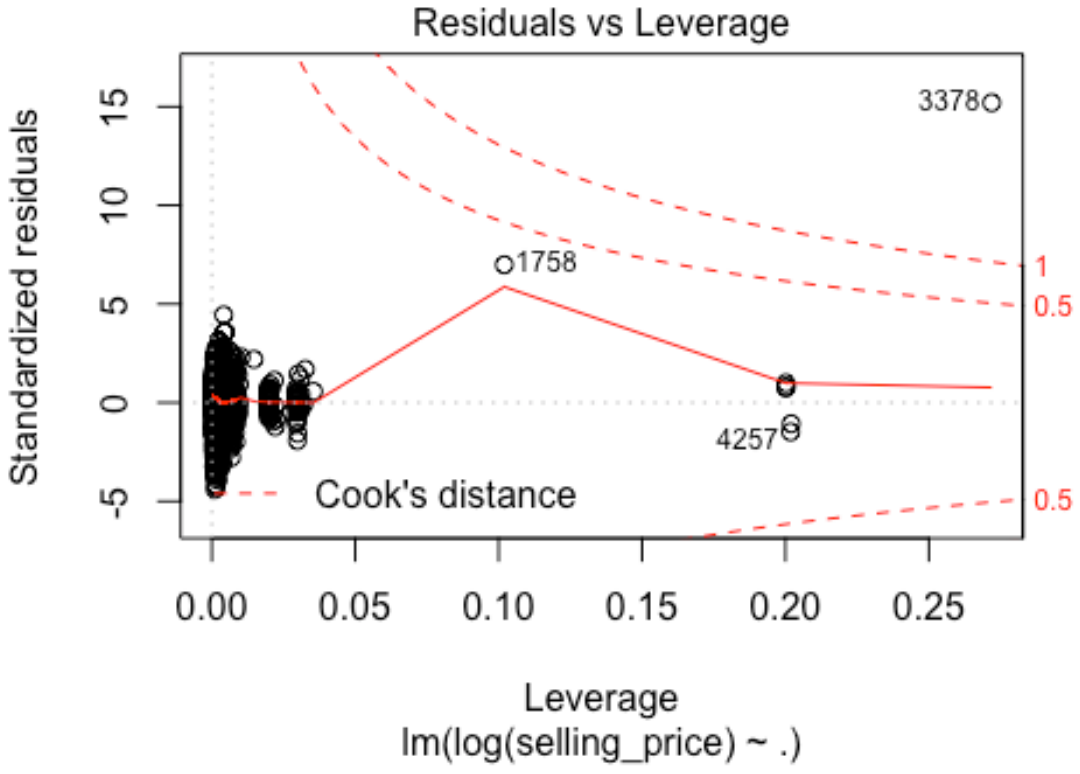

```
# Take the log transformation of response variable: selling price
log1.lm <- lm(log(selling_price) ~ ., data = car4)
summary(log1.lm)

##
## Call:
## lm(formula = log(selling_price) ~ ., data = car4)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -2.4721 -0.3558  0.0351  0.3646  7.2963
##
## Coefficients:
##                               Estimate Std. Error t value Pr(>|t|)
## (Intercept)                  1.451e+01  8.747e-02 165.914  < 2e-16
***
```

```
## ManufacturerJapan            -1.056e-01  2.962e-02  -3.565 0.000366
***
## Manufacturerother Asia       -2.423e-01  2.976e-02  -8.140 4.56e-16
***
## Manufacturerother Europe      8.405e-03  3.788e-02   0.222 0.824416
## ManufacturerUS               -3.406e-01  3.530e-02  -9.648  < 2e-16
***
## Years                         4.450e-03  7.698e-04   5.780 7.73e-09
***
## Mileage                      -9.942e-04  7.424e-05 -13.392  < 2e-16
***
## Engine                       -1.831e-03  1.690e-04 -10.837  < 2e-16
***
## km_driven                    -2.744e-06  1.274e-07 -21.545  < 2e-16
***
## fuelDiesel                    4.438e-01  7.927e-02   5.599 2.22e-08
***
## fuelLPG                      -3.715e-01  1.234e-01  -3.011 0.002611
**
## fuelPetrol                   -1.929e-01  7.910e-02  -2.439 0.014754
*
## seller_typeIndividual        -2.268e-01  2.027e-02 -11.189  < 2e-16
***
## seller_typeTrustmark Dealer   6.910e-03  4.169e-02   0.166 0.868341
## transmissionManual           -8.536e-01  2.248e-02 -37.966  < 2e-16
***
## ownerFourth & Above Owner    -8.027e-01  4.603e-02 -17.440  < 2e-16
***
## ownerSecond Owner            -4.077e-01  1.560e-02 -26.139  < 2e-16
***
## ownerTest Drive Car           1.133e+00  2.530e-01   4.477 7.67e-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 7888 degrees of freedom
## Multiple R-squared:  0.539,  Adjusted R-squared:  0.538
## F-statistic: 512.5 on 18 and 7888 DF,  p-value: < 2.2e-16

plot(log1.lm)
```


Residuals vs Fitted
3378
1758
165
Residuals
8
6
4
2
0
-2
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|
3378
1758
165
0
1
2
3
4
6
8
10
12
14
Fitted values
lm(log(selling_price) ~ .)

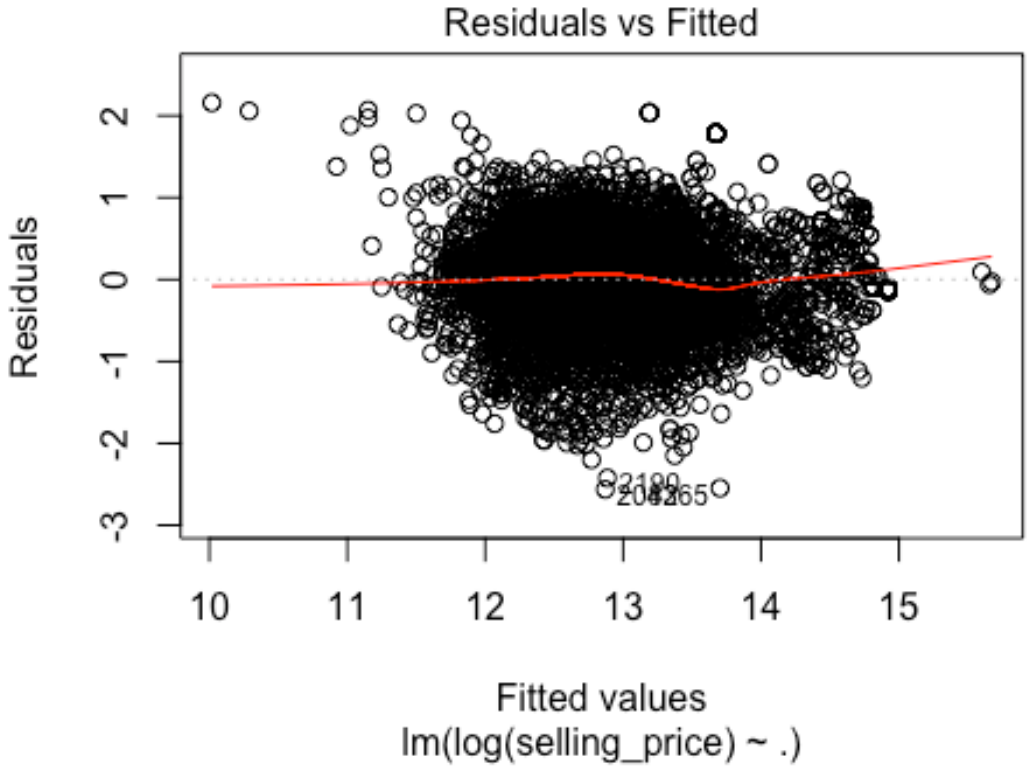


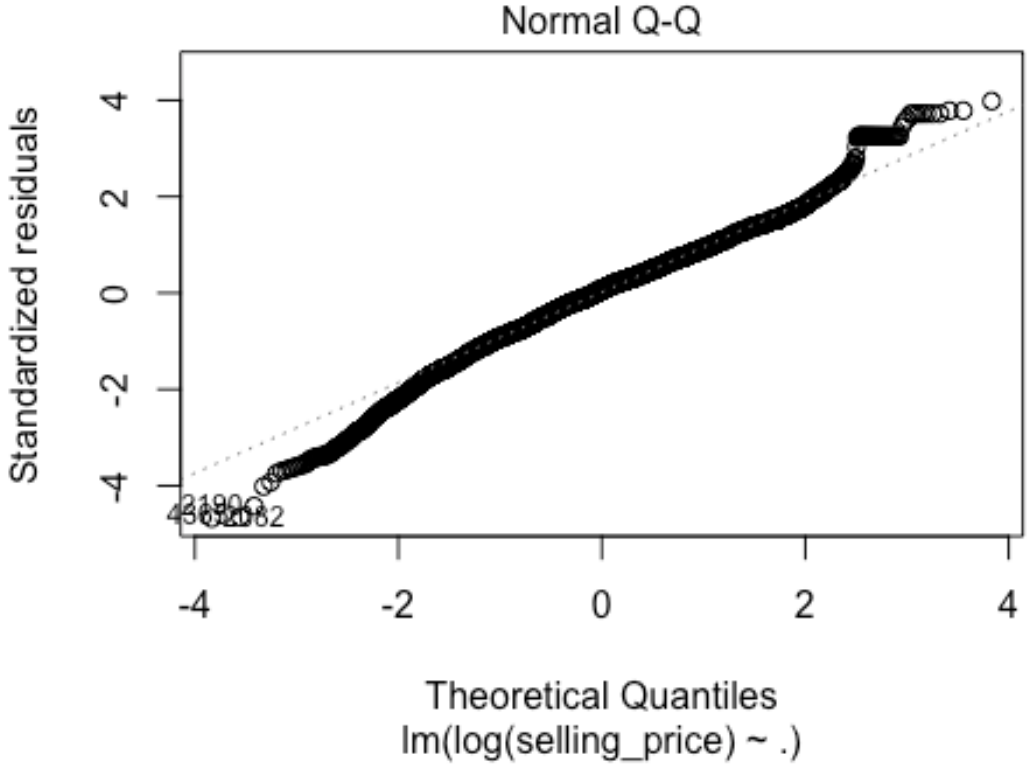

```
# Omit some problematic observations: 165, 1758, 3378, 3898, 4257,
5022, 6048, 6432, 6492, 7154, 7521, 7823
car <- car4[-c(165, 1758, 3378, 3898, 4257, 5022, 6048, 6432, 6492,
7154, 7521, 7823),]
log.lm <- lm(log(selling_price) ~ ., data = car)
summary(log.lm)

##
## Call:
## lm(formula = log(selling_price) ~ ., data = car)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2.5619 -0.3384  0.0371  0.3548  2.1576
##
## Coefficients:
```

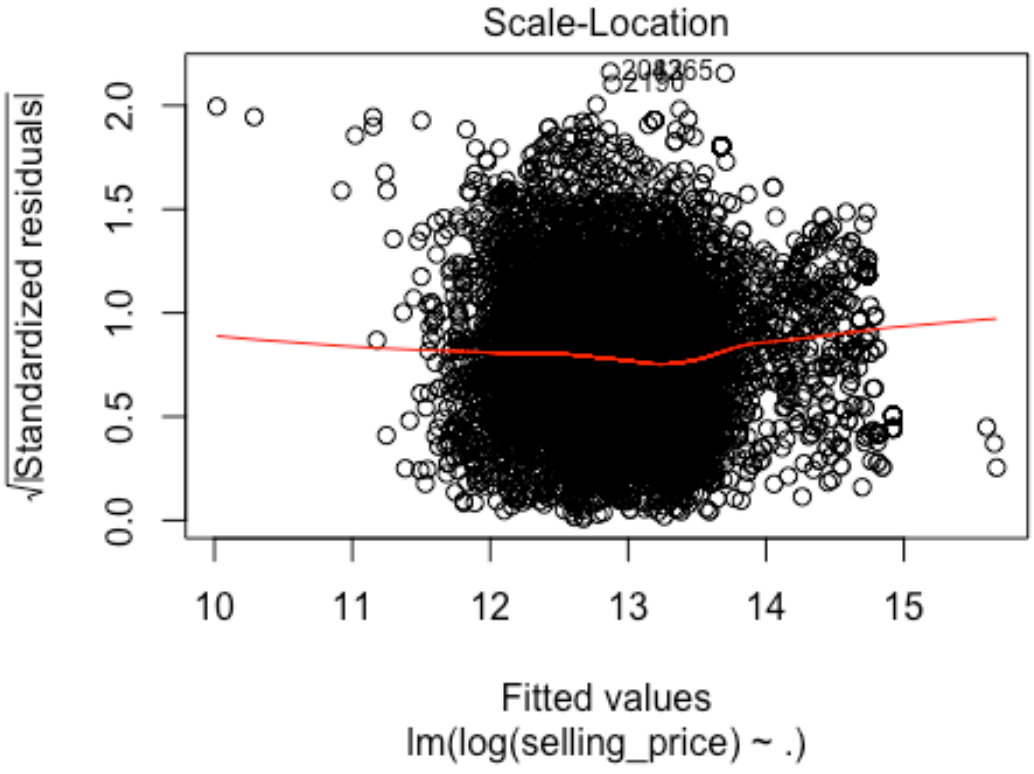
```
##                                Estimate Std. Error t value Pr(>|t|)
## (Intercept)                   1.458e+01  8.531e-02 170.862  < 2e-16
***
## ManufacturerJapan            -8.042e-02  2.896e-02  -2.777  0.00551
**
## Manufacturerother Asia       -2.371e-01  2.907e-02  -8.155 4.05e-16
***
## Manufacturerother Europe     -5.318e-03  3.698e-02  -0.144  0.88565
## ManufacturerUS               -3.343e-01  3.445e-02  -9.702  < 2e-16
***
## Years                         5.242e-03  7.528e-04   6.963 3.60e-12
***
## Mileage                      -1.215e-03  7.326e-05 -16.588  < 2e-16
***
## Engine                       -1.770e-03  1.648e-04 -10.740  < 2e-16
***
## km_driven                    -4.565e-06  1.567e-07 -29.122  < 2e-16
***
## fuelDiesel                    4.789e-01  7.726e-02   6.199 5.98e-10
***
## fuelLPG                      -3.545e-01  1.202e-01  -2.949  0.00320
**
## fuelPetrol                   -2.231e-01  7.710e-02  -2.894  0.00382
**
## seller_typeIndividual        -2.022e-01  1.982e-02 -10.203  < 2e-16
***
## seller_typeTrustmark Dealer   1.271e-02  4.063e-02   0.313  0.75439
## transmissionManual           -8.099e-01  2.203e-02 -36.758  < 2e-16
***
## ownerFourth & Above Owner    -7.375e-01  4.500e-02 -16.389  < 2e-16
***
## ownerSecond Owner            -3.724e-01  1.532e-02 -24.308  < 2e-16
***
## ownerTest Drive Car           1.572e+00  3.179e-01   4.943 7.86e-07
***
## ownerThird Owner             -5.561e-01  2.666e-02 -20.862  < 2e-16
***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.5482 on 7876 degrees of freedom
```

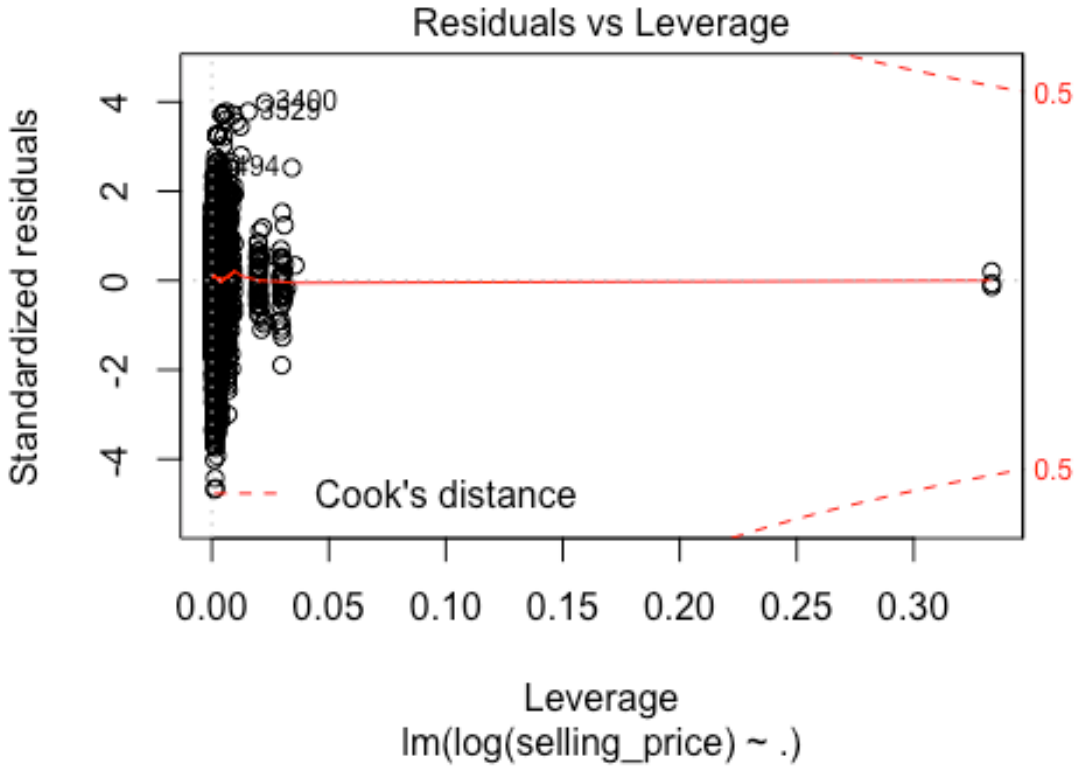
```
## Multiple R-squared:  0.5607, Adjusted R-squared:  0.5597
## F-statistic: 558.4 on 18 and 7876 DF,  p-value: < 2.2e-16
```

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


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


Normal Q-Q
Standardized residuals
4
2
0
-2
-4
-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
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  553.49  138.37  460.503 < 2.2e-16 ***
## Years            1    6.81    6.81   22.654 1.974e-06 ***
## Mileage          1    3.99    3.99   13.283 0.0002696 ***
## Engine           1  107.17  107.17  356.671 < 2.2e-16 ***
## km_driven        1  438.61  438.61 1459.694 < 2.2e-16 ***
## fuel             3  908.28  302.76 1007.585 < 2.2e-16 ***
## seller_type      2  238.09  119.05  396.185 < 2.2e-16 ***
## transmission     1  463.72  463.72 1543.263 < 2.2e-16 ***
## owner            4  300.12   75.03  249.700 < 2.2e-16 ***
## Residuals     7876 2366.60    0.30
```

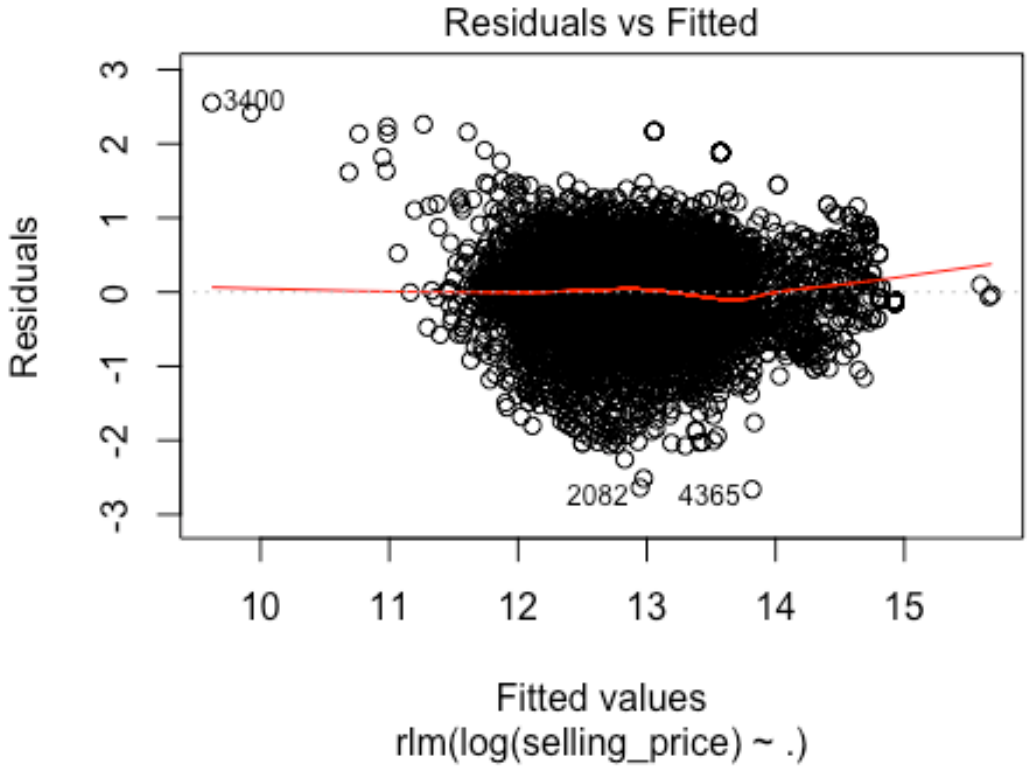
```
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

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

##                   GVIF Df GVIF^(1/(2*Df))
## Manufacturer 1.542866  4        1.055701
## Years        1.109696  1        1.053421
## Mileage      1.312655  1        1.145712
## Engine       1.117918  1        1.057316
## km_driven    1.486285  1        1.219133
## fuel         1.333576  3        1.049147
## seller_type  1.349528  2        1.077818
## transmission 1.457660  1        1.207336
## owner        1.247930  4        1.028073

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

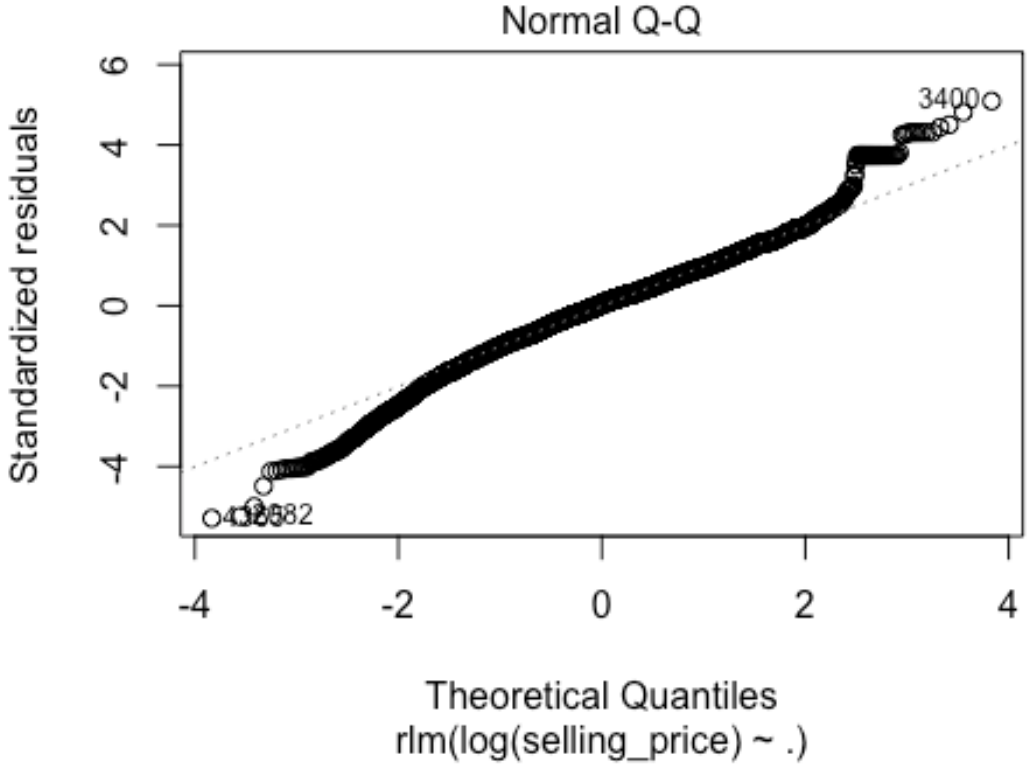
Robust Regression

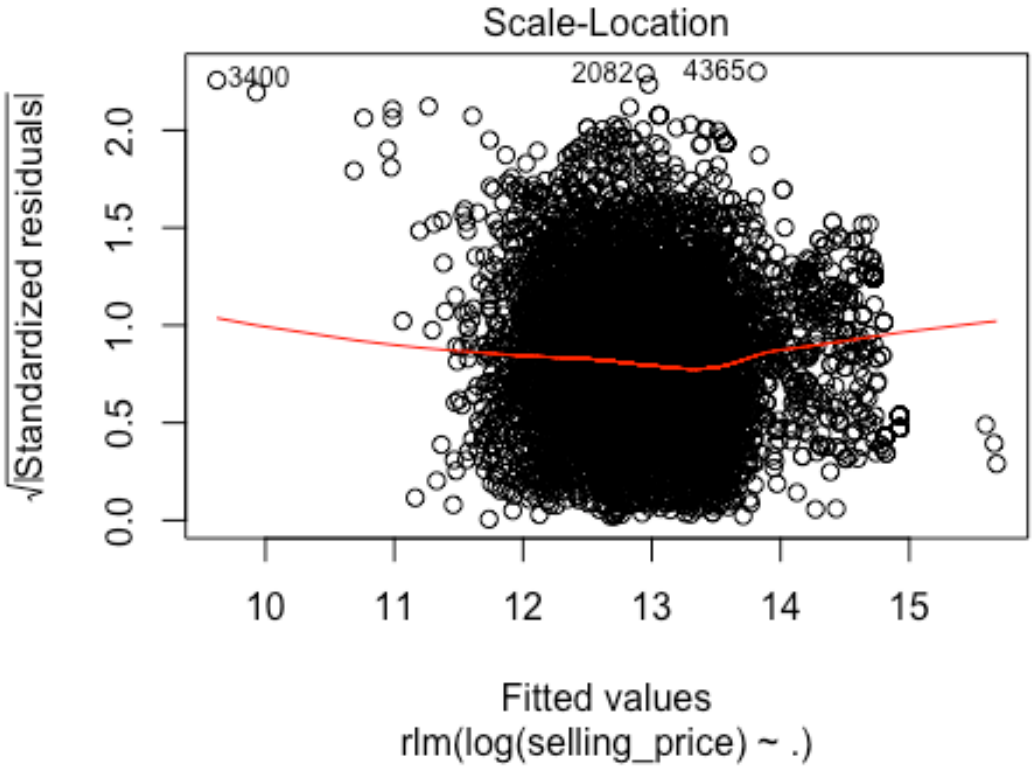
```
# Huber's t Function
robust_huber.lm <- rlm(log(selling_price) ~., data = car, psi =
psi.huber)
summary(robust_huber.lm)

##
## Call: rlm(formula = log(selling_price) ~ ., data = car, psi =
psi.huber)
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2.66062 -0.34493  0.02083  0.33195  2.55343
##
## Coefficients:
##                                Value    Std. Error t value
## (Intercept)                     14.5929   0.0808   180.6960
## ManufacturerJapan               -0.0903   0.0274    -3.2949
## Manufacturerother Asia          -0.2444   0.0275    -8.8792
## Manufacturerother Europe         0.0059   0.0350     0.1699
## ManufacturerUS                  -0.3845   0.0326   -11.7887
## Years                            0.0045   0.0007     6.2733
```

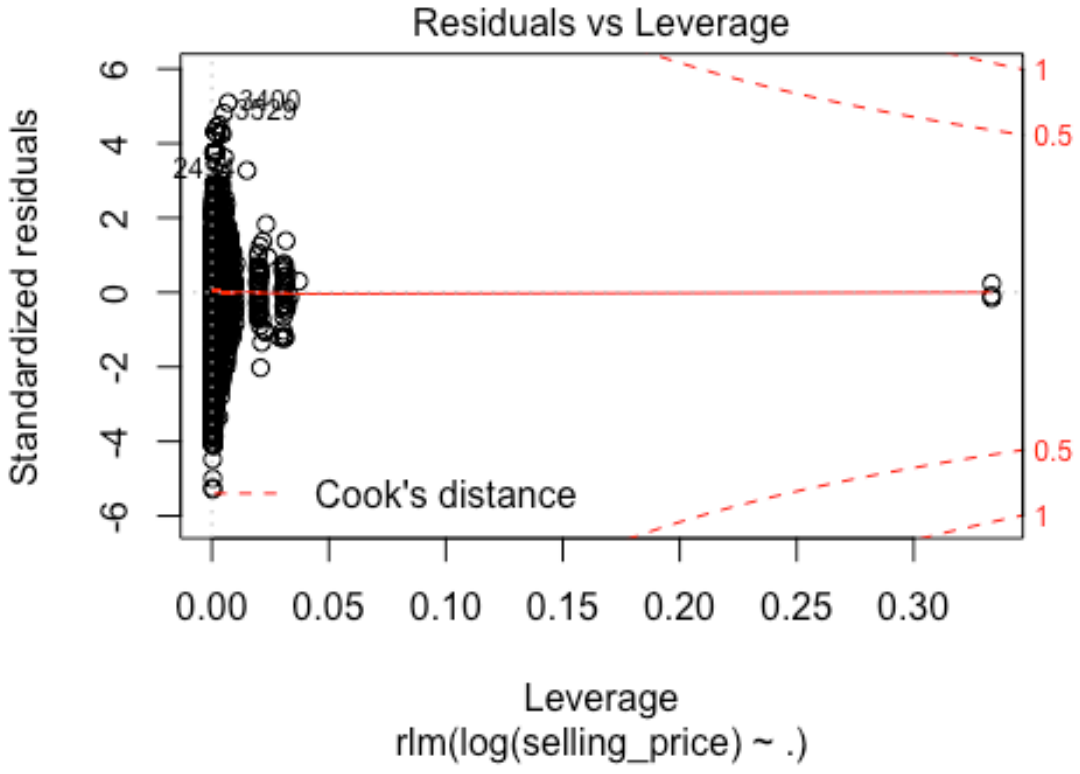
```
## Mileage                         -0.0015   0.0001   -21.2459
## Engine                          -0.0019   0.0002   -12.0151
## km_driven                        0.0000   0.0000   -36.6568
## fuelDiesel                       0.5032   0.0731     6.8792
## fuelLPG                         -0.3931   0.1138    -3.4533
## fuelPetrol                      -0.2252   0.0730    -3.0853
## seller_typeIndividual           -0.1594   0.0188    -8.4932
## seller_typeTrustmark Dealer      0.0105   0.0385     0.2718
## transmissionManual              -0.7082   0.0209   -33.9533
## ownerFourth & Above Owner       -0.7361   0.0426   -17.2814
## ownerSecond Owner               -0.3576   0.0145   -24.6563
## ownerTest Drive Car              1.6044   0.3010     5.3305
## ownerThird Owner                -0.5327   0.0252   -21.1096
##
## Residual standard error: 0.5033 on 7876 degrees of freedom
```

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

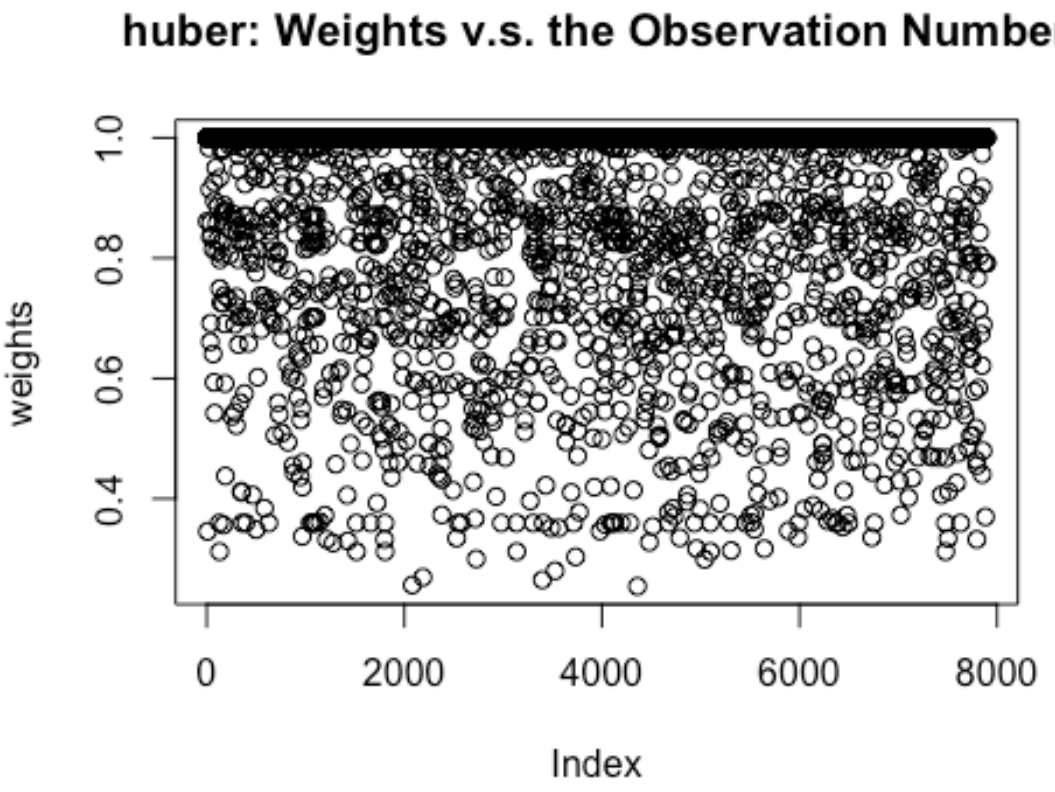

Residuals vs Fitted
Residuals
Fitted values
rlm(log(selling_price) ~ .)
3400
2082
4365
-3
-2
-1
0
1
2
3
10
11
12
13
14
15


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


Scale-Location
3400
2082
4365
√|Standardized residuals|
0.0
0.5
1.0
1.5
2.0
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")
```

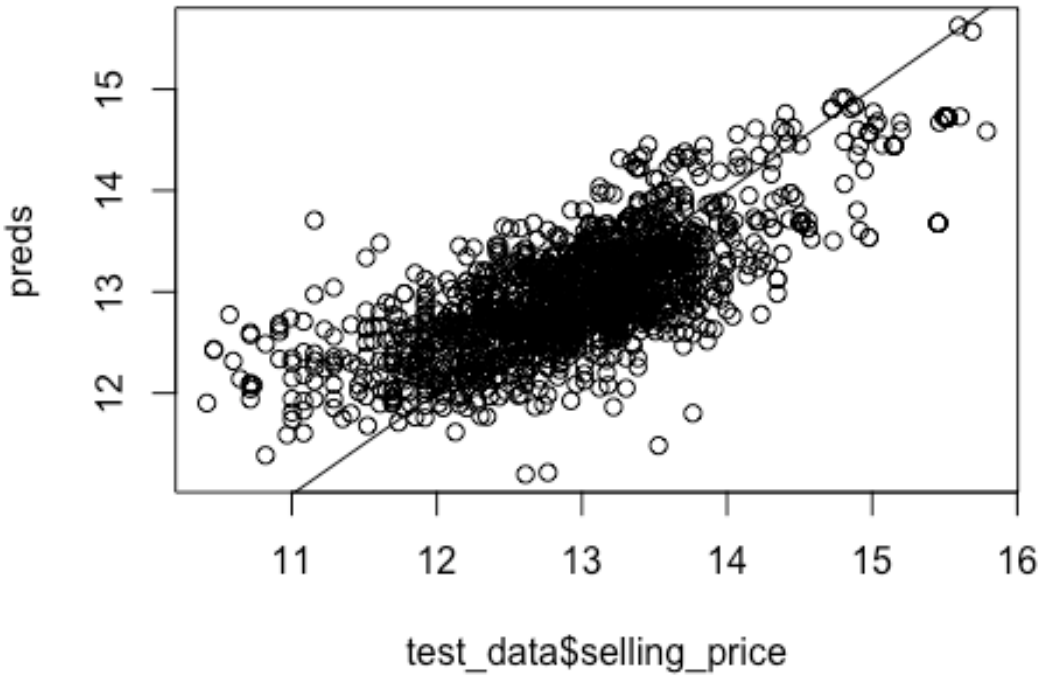


Prediction: Cross Validation

```
# 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
## -2.56918 -0.33887  0.03491  0.35093  2.19138
##
## Coefficients:
##                                Estimate Std. Error t value Pr(>|t|)
## (Intercept)                   1.456e+01  9.459e-02 153.903  < 2e-16
***
## ManufacturerJapan            -6.712e-02  3.232e-02  -2.077  0.03784
*
## Manufacturerother Asia       -2.321e-01  3.243e-02  -7.159 9.05e-13
***
## Manufacturerother Europe     -7.057e-03  4.059e-02  -0.174  0.86198
## ManufacturerUS               -3.228e-01  3.823e-02  -8.444  < 2e-16
***
## Years                         5.599e-03  8.344e-04   6.710 2.12e-11
***
## Mileage                      -1.176e-03  8.177e-05 -14.384  < 2e-16
***
## Engine                       -1.745e-03  1.828e-04  -9.543  < 2e-16
***
## km_driven                    -4.628e-06  1.759e-07 -26.311  < 2e-16
***
## fuelDiesel                    4.785e-01  8.573e-02   5.581 2.48e-08
***
## fuelLPG                      -4.124e-01  1.310e-01  -3.147  0.00166
**
## fuelPetrol                   -2.260e-01  8.551e-02  -2.643  0.00823
**
## seller_typeIndividual        -1.844e-01  2.197e-02  -8.392  < 2e-16
***
## seller_typeTrustmark Dealer  8.921e-03  4.476e-02   0.199  0.84203
## transmissionManual           -8.258e-01  2.463e-02 -33.531  < 2e-16
***
## ownerFourth & Above Owner    -7.162e-01  4.953e-02 -14.460  < 2e-16
***
## ownerSecond Owner            -3.727e-01  1.706e-02 -21.843  < 2e-16
```

```
***
## ownerTest Drive Car          1.558e+00  5.472e-01   2.847  0.00443
**
## ownerThird Owner            -5.385e-01  3.007e-02 -17.910  < 2e-16
***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.5461 on 6297 degrees of freedom
## Multiple R-squared:  0.5642, Adjusted R-squared:  0.563
## F-statistic: 452.9 on 18 and 6297 DF,  p-value: < 2.2e-16

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 0.556752 0.427820
```