Multiple Linear Regression Model

Sachin Sharma

12/27/2021

Installing the libraries

```
library(tinytex)
library(ggplot2)
library(tidyverse)
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.1 --
## v tibble  3.1.6     v dplyr   1.0.7
## v tidyr   1.1.4     v stringr 1.4.0
## v readr   2.0.1     v forcats 0.5.1
## v purrr   0.3.4
## Warning: package 'tibble' was built under R version 4.1.2
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
library(rvest)
```

```
## Warning: package 'rvest' was built under R version 4.1.2
##
## Attaching package: 'rvest'
## The following object is masked from 'package:readr':
##
##     guess_encoding
```

```
library(naniar)
library(corrplot)
```

```
## corrplot 0.90 loaded
```

In multiple linear regression the equation is like :

$y = a_0 + a_1x_1 + a_2x_2 + \cdots$

where a_0 is the y intercept, and a_1 is slope , which can be compared with linear regression : $y = mx + c$ where c is y intercept and m is slope

In multiple linear regression model, we will plot scatter plot first to understand the relation between variables , whether or not the variables are linearly correlated :

Loading data :

You can download the dataset from kaggle : https://www.kaggle.com/nehalbirla/vehicle-dataset-from-cardekho

```r
vehicle <- read.csv("car.csv")

head(vehicle)
```

```
##                         name year selling_price km_driven   fuel seller_type
## 1              Maruti 800 AC 2007         60000     70000 Petrol  Individual
## 2 Maruti Wagon R LXI Minor 2007        135000     50000 Petrol  Individual
## 3     Hyundai Verna 1.6 SX 2012        600000    100000 Diesel  Individual
## 4   Datsun RediGO T Option 2017        250000     46000 Petrol  Individual
## 5    Honda Amaze VX i-DTEC 2014        450000    141000 Diesel  Individual
## 6     Maruti Alto LX BSIII 2007        140000    125000 Petrol  Individual
##   transmission        owner
## 1       Manual  First Owner
## 2       Manual  First Owner
## 3       Manual  First Owner
## 4       Manual  First Owner
## 5       Manual Second Owner
## 6       Manual  First Owner
```

```r
colnames(vehicle)
```

```
## [1] "name"          "year"          "selling_price" "km_driven"
## [5] "fuel"          "seller_type"   "transmission"  "owner"
```

Lets make scatter plot to see how strongly variables are correlated, we are interested in Mileage,lh and lc

```r
str(vehicle)
```

```
## 'data.frame':    4340 obs. of  8 variables:
##  $ name         : chr  "Maruti 800 AC" "Maruti Wagon R LXI Minor" "Hyundai Verna 1.6 SX" "Datsun Redi
##  $ year         : int  2007 2007 2012 2017 2014 2007 2016 2014 2015 2017 ...
##  $ selling_price: int  60000 135000 600000 250000 450000 140000 550000 240000 850000 365000 ...
##  $ km_driven    : int  70000 50000 100000 46000 141000 125000 25000 60000 25000 78000 ...
##  $ fuel         : chr  "Petrol" "Petrol" "Diesel" "Petrol" ...
##  $ seller_type  : chr  "Individual" "Individual" "Individual" "Individual" ...
##  $ transmission : chr  "Manual" "Manual" "Manual" "Manual" ...
##  $ owner        : chr  "First Owner" "First Owner" "First Owner" "First Owner" ...
```

```r
# we will first convert fuel,seller_type,transmission,owner in factor variable

vehicle$fuel <- as.factor(vehicle$fuel)
vehicle$seller_type <- as.factor(vehicle$seller_type)
```

```
vehicle$transmission <- as.factor(vehicle$transmission)
vehicle$owner <- as.factor(vehicle$owner)
```

Now converting the factor to numeric to check the correlation between variables

```
vehicle$fuel <- as.numeric(vehicle$fuel)
vehicle$seller_type <- as.numeric(vehicle$seller_type)
vehicle$transmission <- as.numeric(vehicle$transmission)
vehicle$owner <- as.numeric(vehicle$owner)

vehicle$name <- as.factor(vehicle$name)

vehicle$name <- as.numeric(vehicle$name)
```

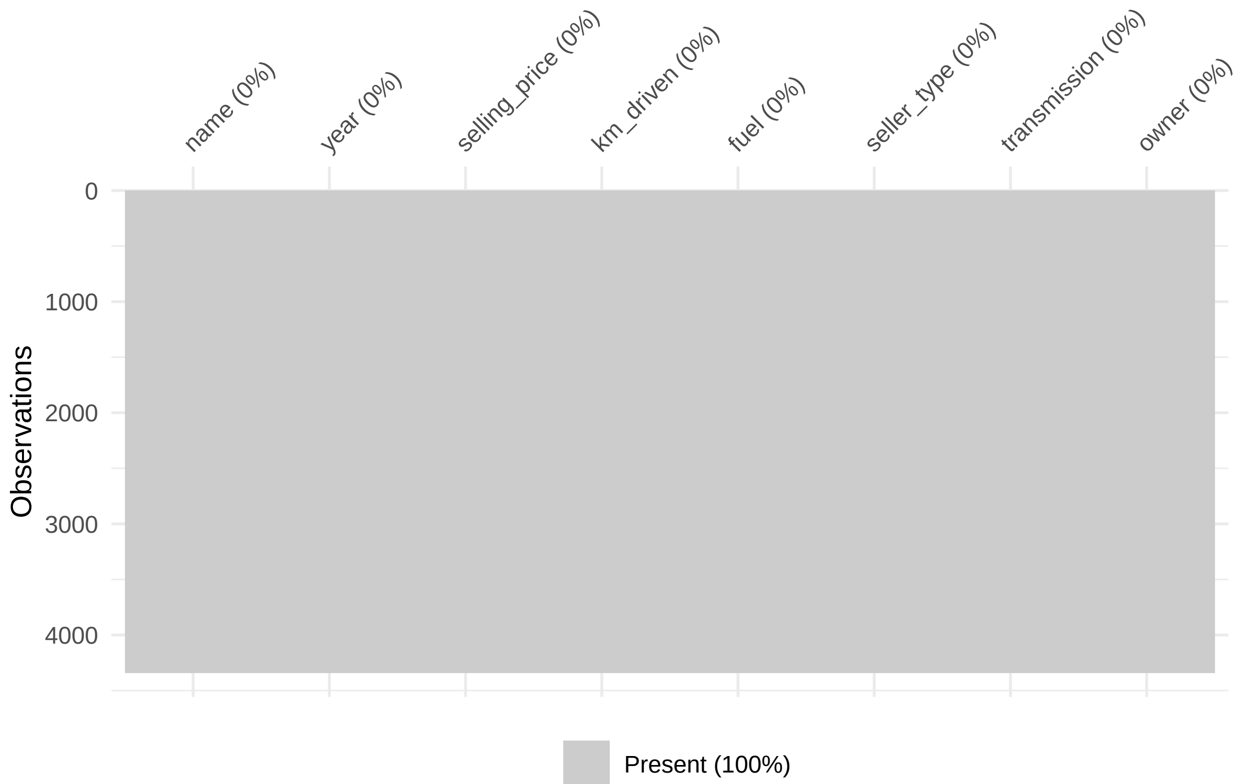
Now checking the missing values in the data if any :

```
sum(is.na(vehicle))
```

```
## [1] 0
```

Visualizing the missing values if any :

```
vis_miss(vehicle)
```

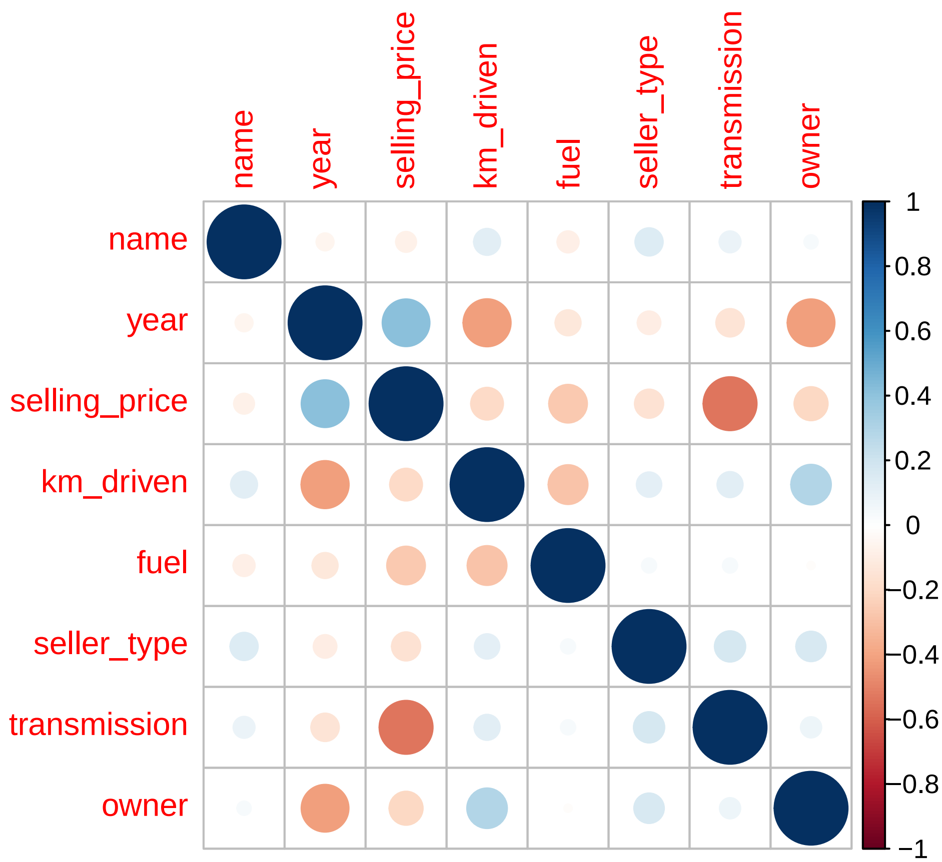


#Here we can see that there is no missing values in the data :

Lets draw a corrplot to see how varibales are related to each other :

```
cort <- cor(vehicle)


corrplot(cort)
```




```r
colnames(vehicle)
```

```
## [1] "name"          "year"          "selling_price" "km_driven"
## [5] "fuel"          "seller_type"   "transmission"  "owner"
```

```r
model_lm_1 <- lm(selling_price~.,data = vehicle)

summary(model_lm_1)
```

```
##
## Call:
## lm(formula = selling_price ~ ., data = vehicle)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1149547  -163275   -27635   115782  7527991
##
## Coefficients:
##                Estimate Std. Error t value Pr(>|t|)
## (Intercept)  -7.106e+07  3.803e+06 -18.687  < 2e-16 ***
## name         -4.301e+01  1.664e+01  -2.585  0.00976 **
## year          3.663e+04  1.881e+03  19.467  < 2e-16 ***
## km_driven    -9.648e-01  1.689e-01  -5.712 1.19e-08 ***
## fuel         -9.358e+04  4.717e+03 -19.837  < 2e-16 ***
## seller_type  -1.947e+04  1.477e+04  -1.318  0.18753
## transmission -8.838e+05  2.203e+04 -40.118  < 2e-16 ***
## owner        -1.710e+04  5.926e+03  -2.886  0.00392 **
```

```
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 429500 on 4332 degrees of freedom
## Multiple R-squared:  0.4498, Adjusted R-squared:  0.449
## F-statistic:   506 on 7 and 4332 DF,  p-value: < 2.2e-16
```

```
model_lm <- lm(selling_price~name+year+km_driven+fuel+seller_type+transmission+owner,data = vehicle)

summary(model_lm)
```

```
##
## Call:
## lm(formula = selling_price ~ name + year + km_driven + fuel +
##     seller_type + transmission + owner, data = vehicle)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1149547  -163275   -27635   115782  7527991
##
## Coefficients:
##                Estimate Std. Error t value Pr(>|t|)
## (Intercept)  -7.106e+07  3.803e+06 -18.687  < 2e-16 ***
## name         -4.301e+01  1.664e+01  -2.585  0.00976 **
## year          3.663e+04  1.881e+03  19.467  < 2e-16 ***
## km_driven    -9.648e-01  1.689e-01  -5.712 1.19e-08 ***
## fuel         -9.358e+04  4.717e+03 -19.837  < 2e-16 ***
## seller_type  -1.947e+04  1.477e+04  -1.318  0.18753
## transmission -8.838e+05  2.203e+04 -40.118  < 2e-16 ***
## owner        -1.710e+04  5.926e+03  -2.886  0.00392 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 429500 on 4332 degrees of freedom
## Multiple R-squared:  0.4498, Adjusted R-squared:  0.449
## F-statistic:   506 on 7 and 4332 DF,  p-value: < 2.2e-16
```

```
gc()
```

```
##          used (Mb) gc trigger  (Mb) max used  (Mb)
## Ncells 1214115 64.9    2027917 108.4  2027917 108.4
## Vcells 2391468 18.3    8388608  64.0  8379655  64.0
```

* represents the significance of variable in the model.

##*** represent highly significant##** represent significant##** represent less significant## no star : represent no significance

Here we can see that seller_type is not significant for the model, we will remove this and update our model as follows :

```
model_lm1 <- lm(selling_price~name+year+km_driven+fuel+transmission+owner,data = vehicle)

summary(model_lm1)
```

```
##
## Call:
## lm(formula = selling_price ~ name + year + km_driven + fuel +
##     transmission + owner, data = vehicle)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1140306  -165511   -24002   114758  7536861
##
## Coefficients:
##                Estimate Std. Error t value Pr(>|t|)
## (Intercept)  -7.098e+07  3.803e+06 -18.666  < 2e-16 ***
## name         -4.575e+01  1.651e+01  -2.772  0.00560 **
## year          3.657e+04  1.881e+03  19.442  < 2e-16 ***
## km_driven    -9.778e-01  1.686e-01  -5.799 7.15e-09 ***
## fuel         -9.398e+04  4.708e+03 -19.962  < 2e-16 ***
## transmission -8.881e+05  2.179e+04 -40.756  < 2e-16 ***
## owner        -1.812e+04  5.876e+03  -3.084  0.00205 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 429500 on 4333 degrees of freedom
## Multiple R-squared:  0.4496, Adjusted R-squared:  0.4489
## F-statistic:   590 on 6 and 4333 DF,  p-value: < 2.2e-16
```

```r
model_lm2 <- lm(selling_price~name+year+km_driven+fuel+transmission,data = vehicle)

summary(model_lm2)
```

```
##
## Call:
## lm(formula = selling_price ~ name + year + km_driven + fuel +
##     transmission, data = vehicle)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1139061  -165137   -27287   113555  7541757
##
## Coefficients:
##                Estimate Std. Error t value Pr(>|t|)
## (Intercept)  -7.479e+07  3.600e+06 -20.777  < 2e-16 ***
## name         -4.543e+01  1.652e+01  -2.749    0.006 **
## year          3.845e+04  1.781e+03  21.586  < 2e-16 ***
## km_driven    -1.047e+00  1.673e-01  -6.257  4.3e-10 ***
## fuel         -9.378e+04  4.712e+03 -19.902  < 2e-16 ***
## transmission -8.889e+05  2.181e+04 -40.759  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 429900 on 4334 degrees of freedom
## Multiple R-squared:  0.4484, Adjusted R-squared:  0.4478
## F-statistic: 704.7 on 5 and 4334 DF,  p-value: < 2.2e-16
```

```r
model_lm3 <- lm(selling_price~name+year+km_driven+transmission+fuel,data = vehicle)
```

```r
summary(model_lm3)
```

```
##
## Call:
## lm(formula = selling_price ~ name + year + km_driven + transmission +
##     fuel, data = vehicle)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1139061  -165137   -27287   113555  7541757
##
## Coefficients:
##                Estimate Std. Error t value Pr(>|t|)
## (Intercept)  -7.479e+07  3.600e+06 -20.777  < 2e-16 ***
## name         -4.543e+01  1.652e+01  -2.749    0.006 **
## year          3.845e+04  1.781e+03  21.586  < 2e-16 ***
## km_driven    -1.047e+00  1.673e-01  -6.257  4.3e-10 ***
## transmission -8.889e+05  2.181e+04 -40.759  < 2e-16 ***
## fuel         -9.378e+04  4.712e+03 -19.902  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 429900 on 4334 degrees of freedom
## Multiple R-squared:  0.4484, Adjusted R-squared:  0.4478
## F-statistic: 704.7 on 5 and 4334 DF,  p-value: < 2.2e-16
```

We will do anova testing : anova stands for analysis of variance

```r
anova(model_lm1,model_lm2)
```

```
## Analysis of Variance Table
##
## Model 1: selling_price ~ name + year + km_driven + fuel + transmission +
##     owner
## Model 2: selling_price ~ name + year + km_driven + fuel + transmission
##   Res.Df        RSS Df   Sum of Sq      F   Pr(>F)
## 1   4333 7.9933e+14
## 2   4334 8.0109e+14 -1 -1.7547e+12 9.5117 0.002055 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
anova(model_lm,model_lm_1)
```

```
## Analysis of Variance Table
##
## Model 1: selling_price ~ name + year + km_driven + fuel + seller_type +
##     transmission + owner
## Model 2: selling_price ~ name + year + km_driven + fuel + seller_type +
##     transmission + owner
##   Res.Df        RSS Df Sum of Sq F Pr(>F)
## 1   4332 7.9901e+14
## 2   4332 7.9901e+14  0         0
```

```r
anova(model_lm_1,model_lm3)
```

```
## Analysis of Variance Table
##
## Model 1: selling_price ~ name + year + km_driven + fuel + seller_type +
##     transmission + owner
## Model 2: selling_price ~ name + year + km_driven + transmission + fuel
##   Res.Df        RSS Df   Sum of Sq      F   Pr(>F)
## 1   4332 7.9901e+14
## 2   4334 8.0109e+14 -2 -2.0751e+12 5.6254 0.003632 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
anova(model_lm_1,model_lm2)
```

```
## Analysis of Variance Table
##
## Model 1: selling_price ~ name + year + km_driven + fuel + seller_type +
##     transmission + owner
## Model 2: selling_price ~ name + year + km_driven + fuel + transmission
##   Res.Df        RSS Df   Sum of Sq      F   Pr(>F)
## 1   4332 7.9901e+14
## 2   4334 8.0109e+14 -2 -2.0751e+12 5.6254 0.003632 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

After comparing the above models , we can see the variation in the p value between our first model and third model , which shows improvement in the results of the model.. or performance of the model.

Now lets do prediction

```r
predict(model_lm3,data.frame(name= 774,km_driven=46000,year = 2007,transmission=2,fuel=5),interval = "c
```

```
##        fit      lwr      upr
## 1 54706.25 27099.85 82312.66
```

```r
head(vehicle)
```

```
##   name year selling_price km_driven fuel seller_type transmission owner
## 1  774 2007         60000     70000    5           2            2     1
## 2 1040 2007        135000     50000    5           2            2     1
## 3  566 2012        600000    100000    2           2            2     1
## 4  120 2017        250000     46000    5           2            2     1
## 5  278 2014        450000    141000    2           2            2     3
## 6  811 2007        140000    125000    5           2            2     1
```