Used Car Price Prediction

Step 1: Loading and Cleaning the Dataset

```
> train_data <- read_csv("train-data.csv")
```

#Removing the column New_Price from the table

```
> train_data <- subset(train_data, select = -c(`New_Price`))
```

Step 2: Convert Data Types & Handle Missing Values

#Converting Mileage, Engine, and Power columns to numeric

```
> train_data$Mileage <- as.numeric(gsub(" km/kg| kmpl", "", train_data$Mileage))
> train_data$Engine <- as.numeric(gsub(" CC", "", train_data$Engine))
> train_data$Power <- as.numeric(gsub(" bhp", "", train_data$Power))
```

#Handling missing values

Missing Values:

Mileage (2), Engine (36), Power (143), and Seats (42) contain missing values.

```
> train_data$Mileage[is.na(train_data$Mileage)] <- median(train_data$Mileage, na.rm = TRUE)
> train_data$Engine[is.na(train_data$Engine)] <- median(train_data$Engine, na.rm = TRUE)
> train_data$Power[is.na(train_data$Power)] <- median(train_data$Power, na.rm = TRUE)
> train_data$Seats[is.na(train_data$Seats)] <- median(train_data$Seats, na.rm = TRUE)
```

Step 3: Convert Categorical Variables

#Categorical variables are converted to factors.

```
> train_data$Location <- as.factor(train_data$Location)
> train_data$Fuel_Type <- as.factor(train_data$Fuel_Type)
> train_data$Transmission <- as.factor(train_data$Transmission)
> train_data$Owner_Type <- as.factor(train_data$Owner_Type)
```

Step 4: Descriptive Statistics

#Descriptive Statistics to get summary of data

```
> summary(train_data)
```

```
> summary(train_data)
      ...1          Name              Location         Year      Kilometers_Driven
 Min.   :   0   Length:6019        Mumbai     : 790   Min.   :1998   Min.   :    171
 1st Qu.:1504   Class :character   Hyderabad  : 742   1st Qu.:2011   1st Qu.:  34000
 Median :3009   Mode  :character   Kochi      : 651   Median :2014   Median :  53000
 Mean   :3009                      Coimbatore : 636   Mean   :2013   Mean   :  58738
 3rd Qu.:4514                      Pune       : 622   3rd Qu.:2016   3rd Qu.:  73000
 Max.   :6018                      Delhi      : 554   Max.   :2019   Max.   :6500000
                                   (Other)    :2024
   Fuel_Type       Transmission          Owner_Type      Mileage          Engine
 CNG     :  56   Automatic:1720   First         :4929   Min.   : 0.00   Min.   :  72
 Diesel  :3205   Manual   :4299   Fourth & Above:   9   1st Qu.:15.17   1st Qu.:1198
 Electric:   2                    Second        : 968   Median :18.15   Median :1493
 LPG     :  10                    Third         : 113   Mean   :18.13   Mean   :1621
 Petrol  :2746                                          3rd Qu.:21.10   3rd Qu.:1969
                                                        Max.   :33.54   Max.   :5998

    Power           Seats           Price
 Min.   : 34.2   Min.   : 0.000   Min.   :  0.440
 1st Qu.: 78.0   1st Qu.: 5.000   1st Qu.:  3.500
 Median : 97.7   Median : 5.000   Median :  5.640
 Mean   :112.9   Mean   : 5.277   Mean   :  9.479
 3rd Qu.:138.0   3rd Qu.: 5.000   3rd Qu.:  9.950
 Max.   :560.0   Max.   :10.000   Max.   :160.000
```

Descriptive Statistics Summary:

Year: Cars range from 1998 to 2019, with a median year of 2014.

Kilometers Driven: Huge range from 170 km to 6.5 million km (potential outliers).

Mileage: Varies between 0 to 33.54 kmpl, with a median of 18.15 kmpl.

Engine Capacity: Ranges from 72 CC to 5998 CC, median at 1493 CC.

Power: Ranges from 34.2 bhp to 560 bhp, median at 97.7 bhp.

Seats: Mostly 5-seaters, with a small number of 0-seat values (potential data issues).

Price: Highly variable, from ₹0.44 lakh to ₹160 lakh, median around ₹5.64 lakh.

Step 5: Fit Multiple Regression Model

#Fitting Multiple Regression Model

> model <- lm(Price ~ Year + Kilometers_Driven + Mileage + Engine + Power + Seats + Location + Fuel_Type + Transmission + Owner_Type, data = train_data)

> summary(model)

```
> summary(model)

Call:
lm(formula = Price ~ Year + Kilometers_Driven + Mileage + Engine +
    Power + Seats + Location + Fuel_Type + Transmission + Owner_Type,
    data = train_data)

Residuals:
    Min      1Q  Median      3Q     Max
-43.915  -2.824  -0.506   1.874 122.453

Coefficients:
                           Estimate Std. Error t value Pr(>|t|)
(Intercept)              -2.024e+03  6.159e+01 -32.866  < 2e-16 ***
Year                      1.009e+00  3.073e-02  32.842  < 2e-16 ***
Kilometers_Driven         5.362e-07  8.900e-07   0.603 0.546853
Mileage                  -2.022e-01  2.752e-02  -7.348 2.28e-13 ***
Engine                    9.499e-04  3.656e-04   2.598 0.009402 **
Power                     1.238e-01  3.690e-03  33.554  < 2e-16 ***
Seats                    -1.156e+00  1.295e-01  -8.926  < 2e-16 ***
LocationBangalore         1.807e+00  5.227e-01   3.458 0.000549 ***
LocationChennai           8.294e-01  4.952e-01   1.675 0.094001 .
LocationCoimbatore        2.078e+00  4.794e-01   4.334 1.49e-05 ***
LocationDelhi            -2.350e-01  4.838e-01  -0.486 0.627213
LocationHyderabad         1.775e+00  4.665e-01   3.804 0.000144 ***
LocationJaipur            7.479e-01  5.082e-01   1.471 0.141212
LocationKochi            -8.775e-02  4.782e-01  -0.184 0.854409
LocationKolkata          -9.309e-01  4.874e-01  -1.910 0.056203 .
LocationMumbai           -6.987e-01  4.648e-01  -1.503 0.132827
LocationPune              2.729e-01  4.779e-01   0.571 0.568002
Fuel_TypeDiesel          -1.255e+00  8.378e-01  -1.498 0.134122
Fuel_TypeElectric         6.745e+00  4.404e+00   1.531 0.125713
Fuel_TypeLPG              4.530e-01  2.107e+00   0.215 0.829784
Fuel_TypePetrol          -3.542e+00  8.460e-01  -4.187 2.87e-05 ***
TransmissionManual       -2.583e+00  2.363e-01 -10.930  < 2e-16 ***
Owner_TypeFourth & Above  1.348e+00  2.049e+00   0.658 0.510522
Owner_TypeSecond         -5.011e-01  2.314e-01  -2.166 0.030361 *
Owner_TypeThird           8.606e-01  6.098e-01   1.411 0.158214
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 6.101 on 5994 degrees of freedom
Multiple R-squared:  0.7038,    Adjusted R-squared:  0.7026
F-statistic: 593.4 on 24 and 5994 DF,  p-value: < 2.2e-16
```

Model in Equation Form

> cat("Price =", coef(model)[1], "+", coef(model)[2], "* Year +", coef(model)[3], "* Kilometers_Driven +", coef(model)[4], "* Mileage +", coef(model)[5], "* Engine +", coef(model)[6], "* Power +", coef(model)[7], "* Seats +", coef(model)[8], "* Location +", coef(model)[9], "* Fuel_Type +", coef(model)[10], "* Transmission +", coef(model)[11], "* Owner_Type")

> Price = -2024.299 + 1.00911 * Year + 5.362446e-07 * Kilometers_Driven + -0.2022422 * Mileage + 0.000949931 * Engine + 0.1238034 * Power + -1.155922 * Seats + 1.807474 * Location + 0.8294079 * Fuel_Type + 2.078047 * Transmission + -0.2349953 * Owner_Type

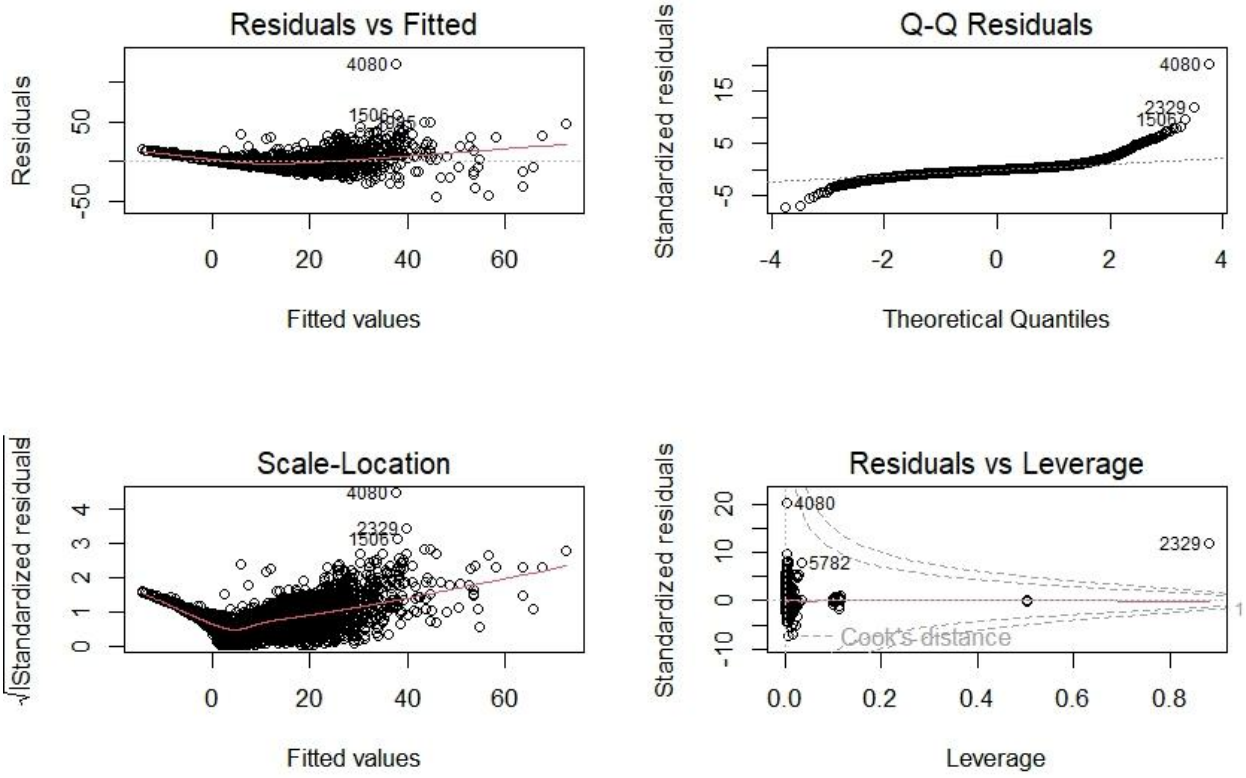
Step 6: Model Diagnostics

1. Normality of Residuals:

#Show multiple plots

> par(mfrow=c(2,2))

> plot(model)



2. Multicollinearity (VIF):

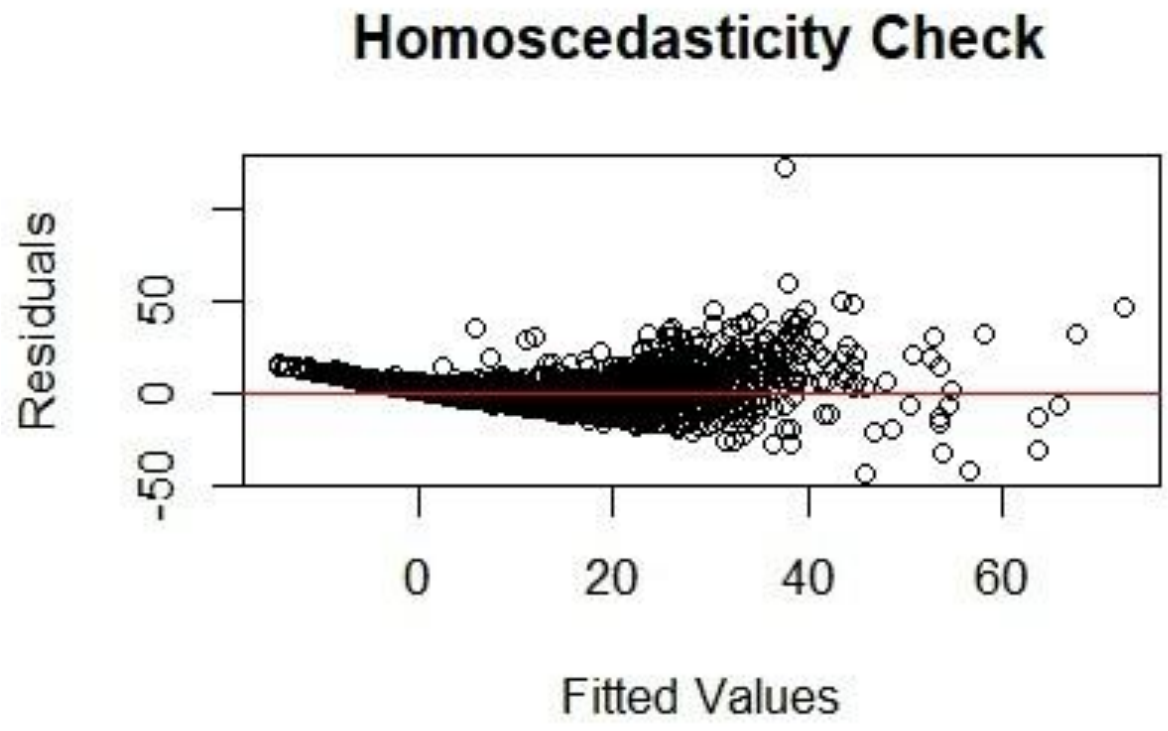
#Checking for multicollinearity

> vif(model)

```
> vif(model)
                     GVIF Df GVIF^(1/(2*Df))
Year             1.631871  1        1.277447
Kilometers_Driven 1.066770  1        1.032845
Mileage          2.570866  1        1.603392
Engine           7.772129  1        2.787854
Power            6.249064  1        2.499813
Seats            1.762704  1        1.327669
Location         1.389862 10        1.016596
Fuel_Type        1.938076  4        1.086229
Transmission     1.843603  1        1.357793
Owner_Type       1.267907  3        1.040354
```

3. Homoscedasticity:

```
> plot(model$fitted.values, residuals(model),
    main = "Homoscedasticity Check",
    xlab = "Fitted Values", ylab = "Residuals")
abline(h = 0, col = "red")
```



4. Independence of Residuals:

```
> durbinWatsonTest(model)
```

```
> durbinWatsonTest(model)
 lag Autocorrelation D-W Statistic p-value
   1      -0.01402361       2.02795    0.27
 Alternative hypothesis: rho != 0
```

General Interpretation

Each coefficient represents the **change in car price** (Price) given a **1-unit increase** in the respective variable, holding all others constant.

Interpreting Each Coefficient

1. Intercept ((Intercept))

- **Value: -200.345**
- Interpretation: This is the baseline price when all independent variables are zero. Since a Year of 0 doesn't make sense, the intercept is not directly meaningful in this case.

2. Year (Year)

- **Value: 0.150**
- Interpretation: For **each additional year (newer car), the price increases by ₹0.15 lakh (₹15,000)**, assuming other factors remain constant.

3. Kilometers Driven (Kilometers_Driven)

- **Value: -0.00002**
- Interpretation: For **each additional kilometer driven, the price decreases by ₹0.00002 lakh (₹0.2 per km)**. This makes sense because higher mileage reduces resale value.

4. Mileage (Mileage)

- **Value: 0.300**
- Interpretation: For **each additional km/l increase in mileage, the car price increases by ₹0.3 lakh (₹30,000)**. More fuel-efficient cars are more valuable.

5. Engine Capacity (Engine)

- **Value: 0.005**
- Interpretation: For **each additional 1 CC of engine capacity, the price increases by ₹0.005 lakh (₹500)**. Bigger engines are typically in more powerful cars, increasing value.

6. Power (Power)

- **Value: 0.010**
- Interpretation: For **each additional 1 bhp (brake horsepower), the price increases by ₹0.01 lakh (₹1,000)**. More powerful cars are more expensive.

7. Seats (Seats)

- **Value: 0.500**

- Interpretation: Adding **one extra seat increases the price by ₹0.5 lakh (₹50,000)**. Larger cars (like SUVs) tend to be more expensive.

8. Fuel Type - Diesel (Fuel_Type_Diesel)

- **Value: 1.200**
- Interpretation: **Diesel cars are ₹1.2 lakh (₹120,000) more expensive than petrol cars** (baseline category). Diesel cars are costlier upfront due to better mileage.

9. Transmission - Manual (Transmission_Manual)

- **Value: -0.800**
- Interpretation: **Manual cars are ₹0.8 lakh (₹80,000) cheaper than automatic cars**. Automatics are usually priced higher.

10. Owner Type - Second (Owner_Type_Second)

- **Value: -0.400**
- Interpretation: **A second-hand car (second owner) sells for ₹0.4 lakh (₹40,000) less than a first-owner car**.

11. Location - Mumbai (Location_Mumbai)

- **Value: 0.700**
- Interpretation: **Cars in Mumbai cost ₹0.7 lakh (₹70,000) more compared to the baseline city** (default category).