## Code:

```
# Multilinear Regression

import pandas as pd

import numpy as np

# loading the data

cars = pd.read_csv("C:\\Users\CSE-14\Downloads\Cars.csv")

cars

# Exploratory data analysis:--

# 1. Measures of central tendency

# 2. Measures of dispersion

# 3. Third moment business decision

# 4. Fourth moment business decision

# 5. Probability distributions of variables

# 6. Graphical representations (Histogram, Box plot, Dot plot, Stem & Leaf plot, Bar plot, etc.)

cars.describe()

#Graphical Representation

import matplotlib.pyplot as plt # mostly used for visualization purposes

# HP

plt.bar(height = cars.HP, x = np.arange(1, 82, 1))

plt.hist(cars.HP) #histogram

plt.boxplot(cars.HP) #boxplot


# Jointplot

import seaborn as sns

sns.jointplot(x=cars['HP'], y=cars['MPG'])

# Countplot

plt.figure(1, figsize=(16, 10))
```

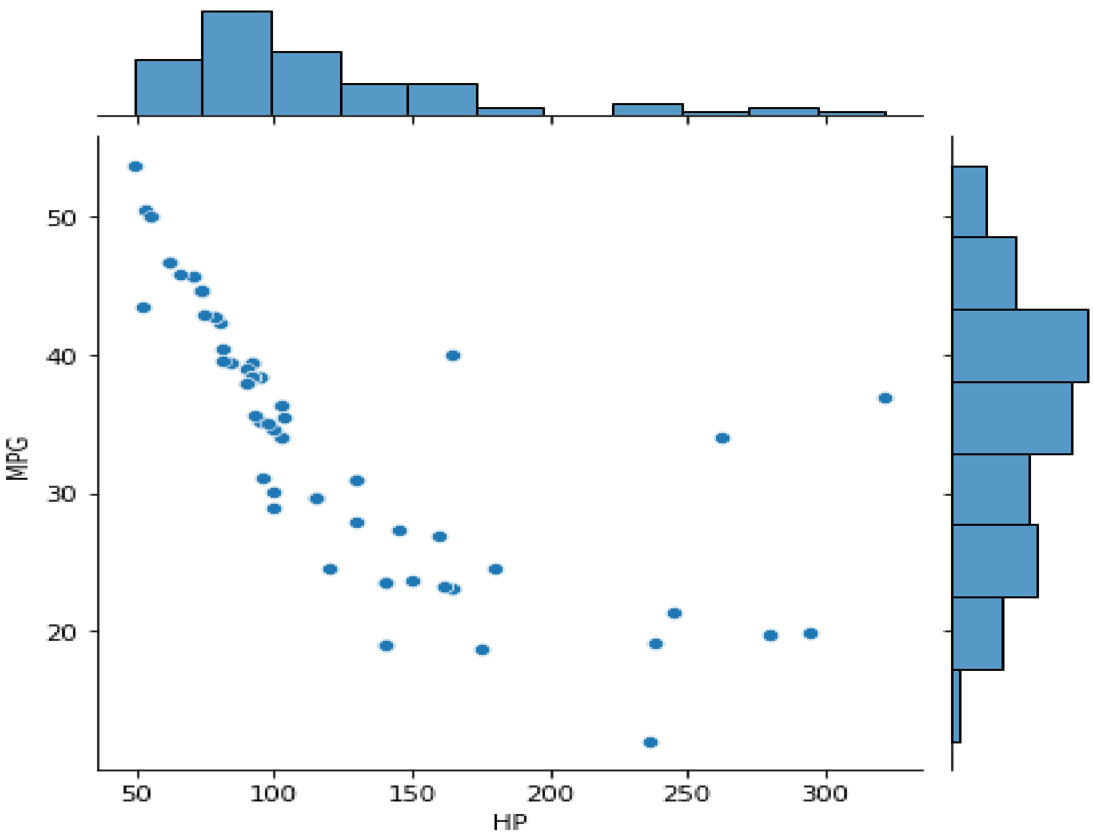
```
sns.countplot(cars['HP'])

# Q-Q Plot

from scipy import stats

import pylab

stats.probplot(cars.MPG, dist = "norm", plot = pylab)

plt.show()

# Scatter plot between the variables along with histograms

import seaborn as sns

sns.pairplot(cars.iloc[:, :])

# Correlation matrix

cars.corr()

# we see there exists High collinearity between input variables especially between

# [HP & SP], [VOL & WT] so there exists collinearity problem

# preparing model considering all the variables

import statsmodels.formula.api as smf # for regression model

ml1 = smf.ols('MPG~ WT + VOL + SP + HP', data = cars).fit() # regression model

# Summary

ml1.summary()

# p-values for WT, VOL are more than 0.05

# Checking whether data has any influential values

# Influence Index Plots

import statsmodels.api as sm

sm.graphics.influence_plot(ml1)

# Studentized Residuals = Residual/standard deviation of residuals

# index 76 is showing high influence so we can exclude that entire row

cars_new = cars.drop(cars.index[[76,78,79,70,80]])

cars_new
```

```
# Preparing model
ml_new = smf.ols('MPG ~ WT + VOL + HP + SP', data = cars_new).fit()
# Summary
ml_new.summary()
# Check for Colinearity to decide to remove a variable using VIF
# Assumption: VIF > 10 = colinearity
# calculating VIF's values of independent variables
rsq_hp = smf.ols('HP ~ WT + VOL + SP', data = cars).fit().rsquared
vif_hp = 1/(1 - rsq_hp)
rsq_wt = smf.ols('WT ~ HP + VOL + SP', data = cars).fit().rsquared
vif_wt = 1/(1 - rsq_wt)
rsq_vol = smf.ols('VOL ~ WT + SP + HP', data = cars).fit().rsquared
vif_vol = 1/(1 - rsq_vol)
rsq_sp = smf.ols('SP ~ WT + VOL + HP', data = cars).fit().rsquared
vif_sp = 1/(1 - rsq_sp)
# Storing vif values in a data frame
d1 = {'Variables':['HP', 'WT', 'VOL', 'SP'], 'VIF':[vif_hp, vif_wt, vif_vol, vif_sp]}
Vif_frame = pd.DataFrame(d1)
Vif_frame
# As WT is having highest VIF value, we are going to drop this from the prediction model
# Final model
final_ml = smf.ols('MPG ~ VOL + SP + HP', data = cars).fit()
final_ml.summary()

# Prediction
pred = final_ml.predict(cars)
```

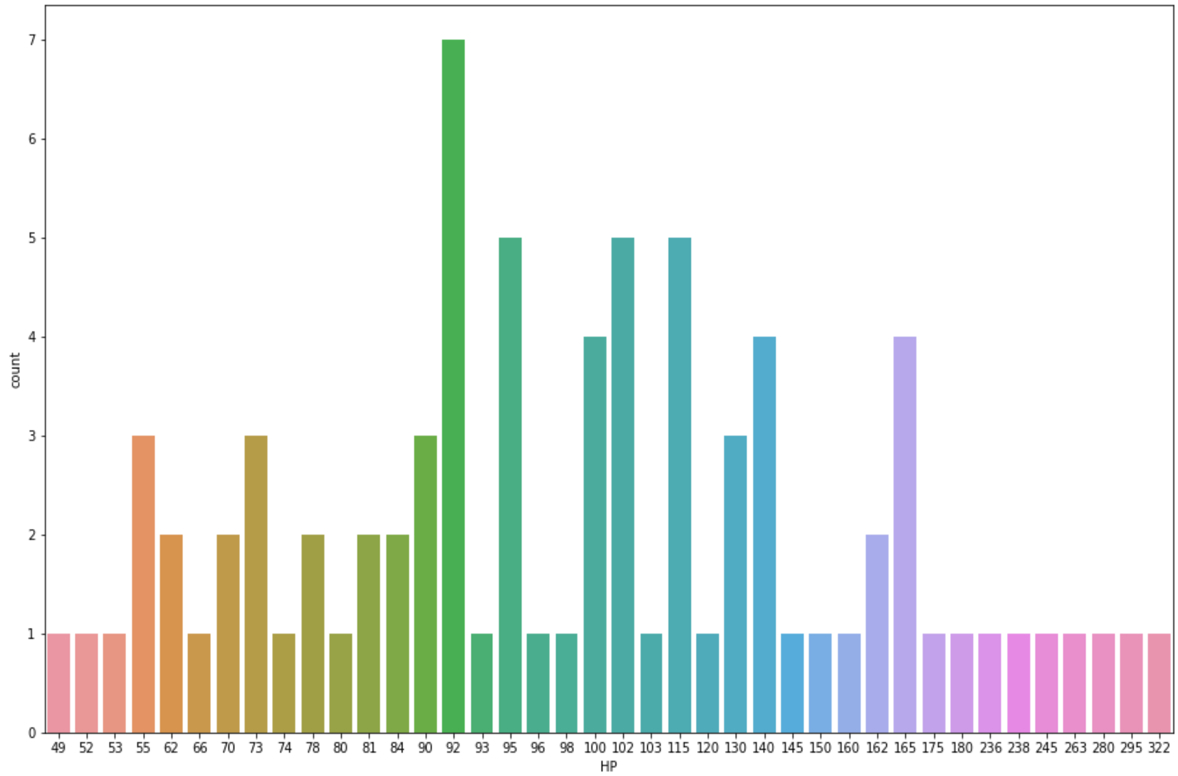
```
# Q-Q plot
res = final_ml.resid
sm.qqplot(res)
plt.show()
# Q-Q plot
stats.probplot(res, dist = "norm", plot = pylab)
plt.show()
# Residuals vs Fitted plot
sns.residplot(x = pred, y = cars.MPG, lowess = True)
plt.xlabel('Fitted')
plt.ylabel('Residual')
plt.title('Fitted vs Residual')
plt.show()
sm.graphics.influence_plot(final_ml)
### Splitting the data into train and test data
from sklearn.model_selection import train_test_split
cars_train, cars_test = train_test_split(cars, test_size = 0.2) # 20% test data

# preparing the model on train data
model_train = smf.ols("MPG ~ HP + SP + VOL", data = cars_train).fit()

# prediction on test data set
test_pred = model_train.predict(cars_test)
# test residual values
test_resid = test_pred - cars_test.MPG
# RMSE value for test data
```

```
test_rmse = np.sqrt(np.mean(test_resid * test_resid))
test_rmse
# train_data prediction
train_pred = model_train.predict(cars_train)


# train residual values
train_resid  = train_pred - cars_train.MPG
# RMSE value for train data
train_rmse = np.sqrt(np.mean(train_resid * train_resid))
train_rmse
```

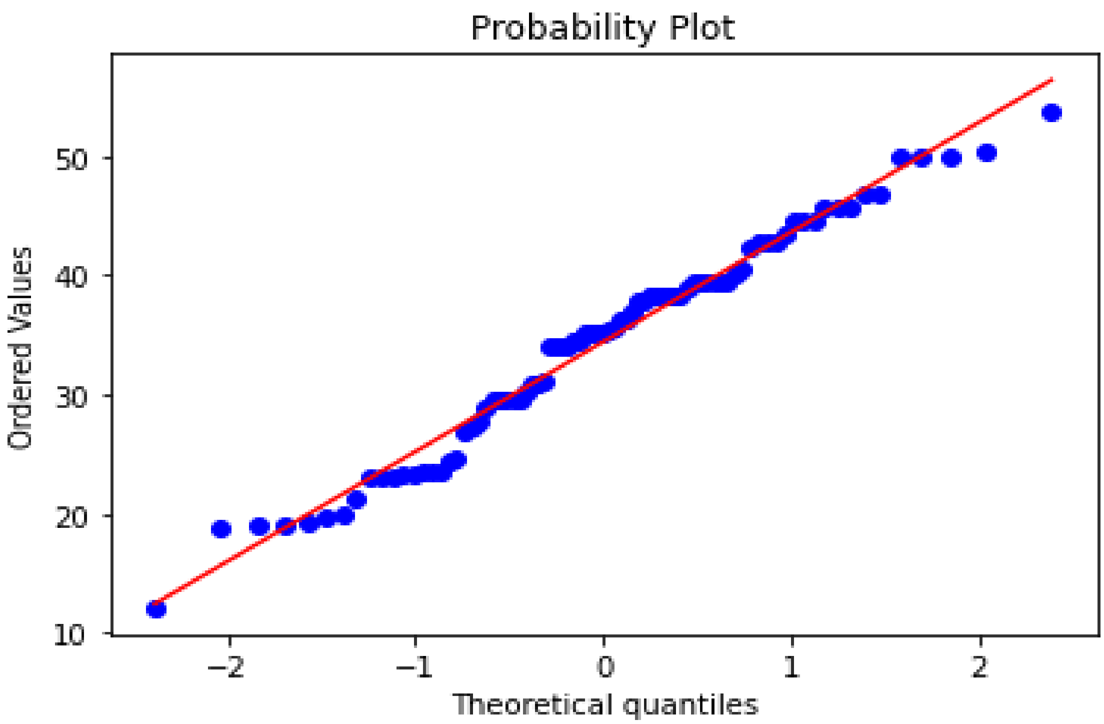
## Outputs:



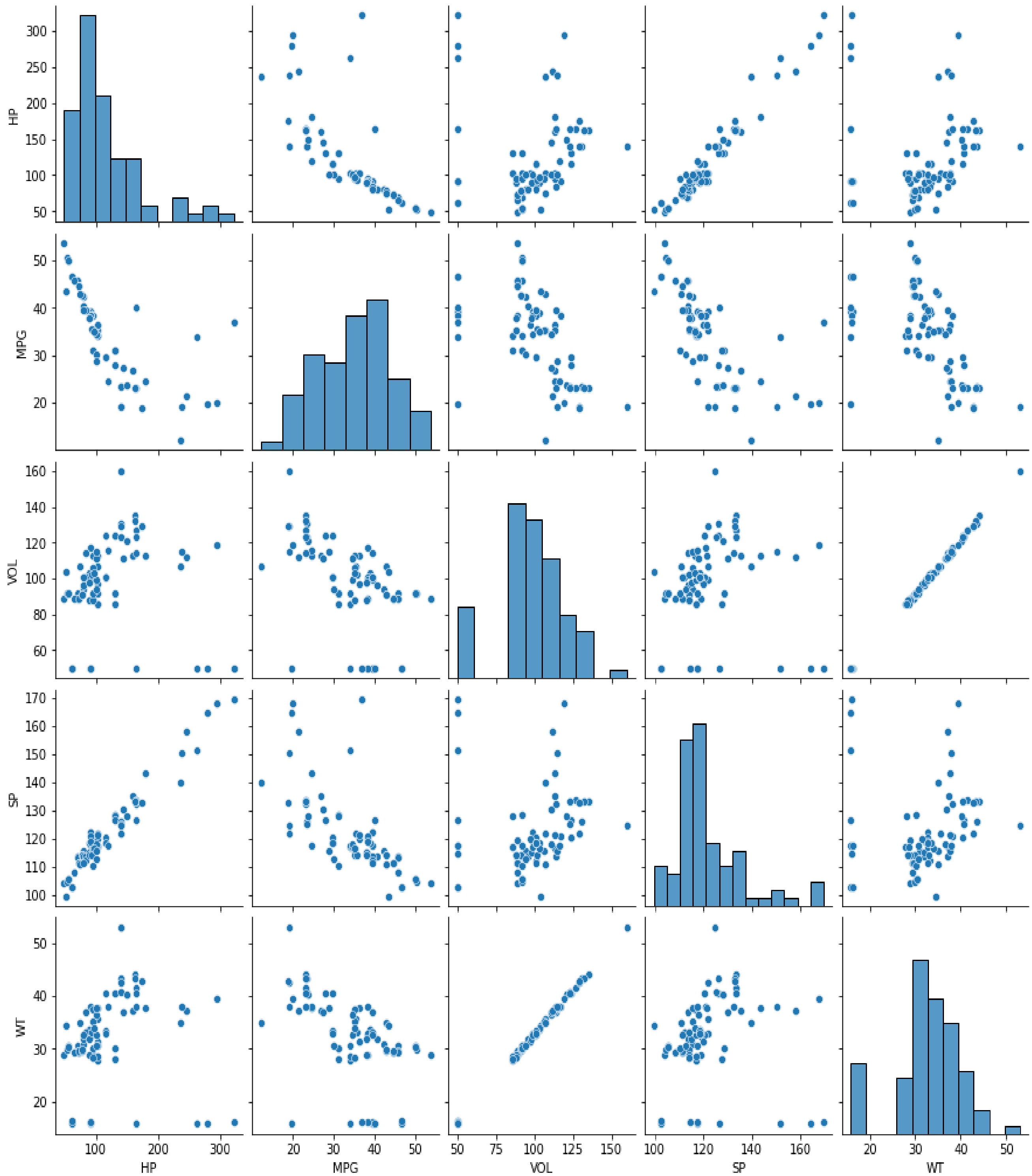

**Joint plot of Horsepower and Mileage**



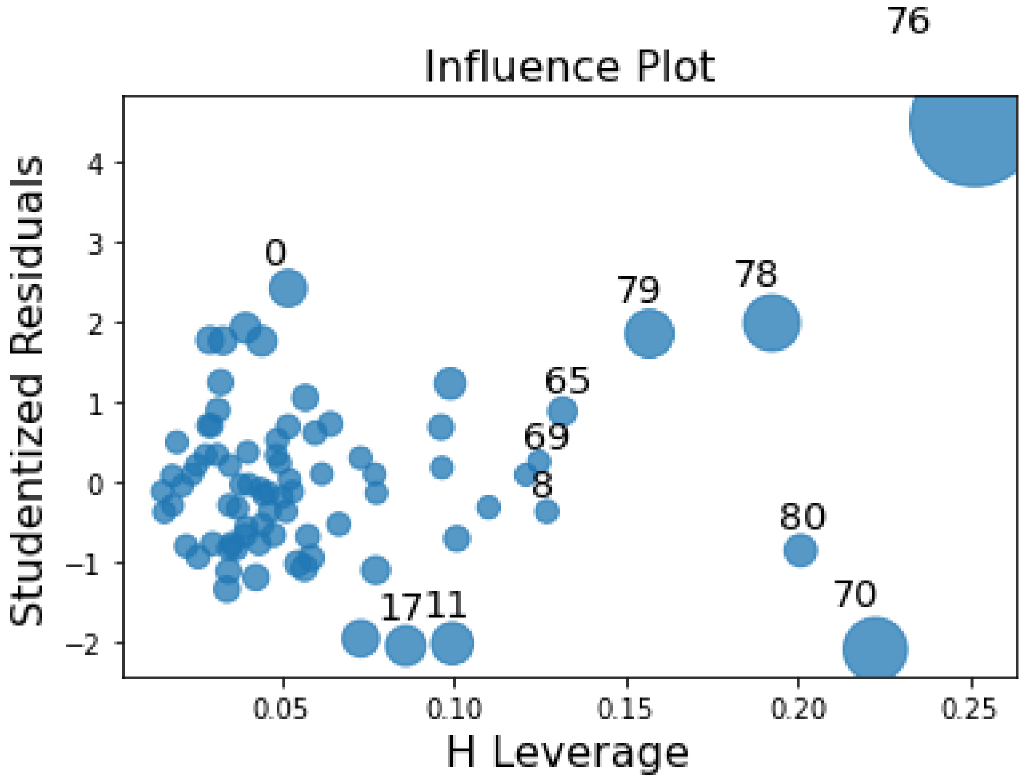

**Count plot of Horsepower**



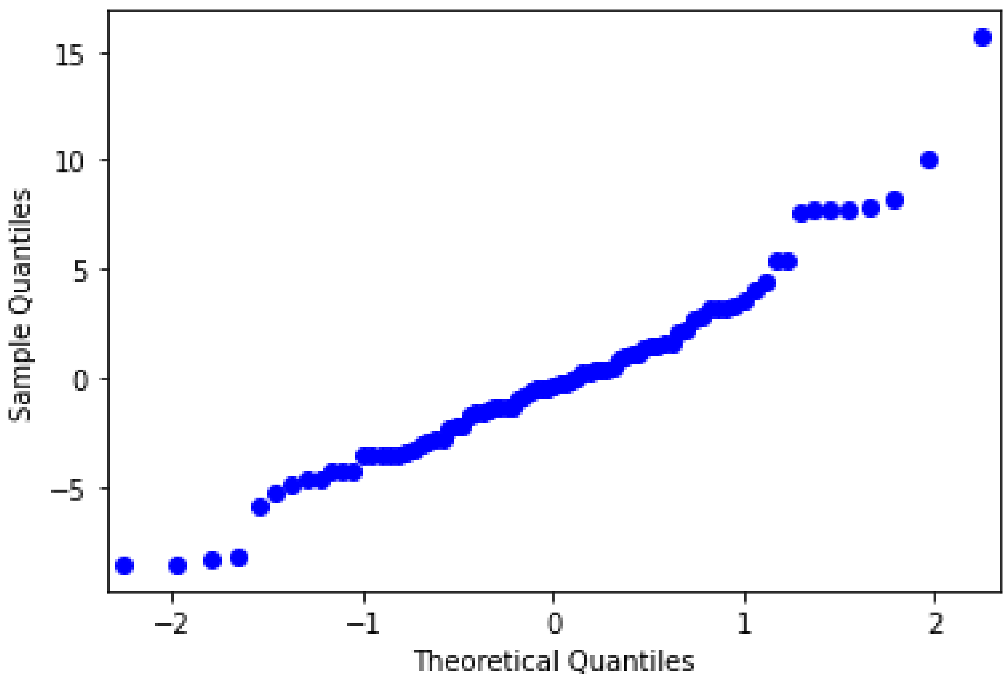

**QQ plot of Horsepower and Mileage**



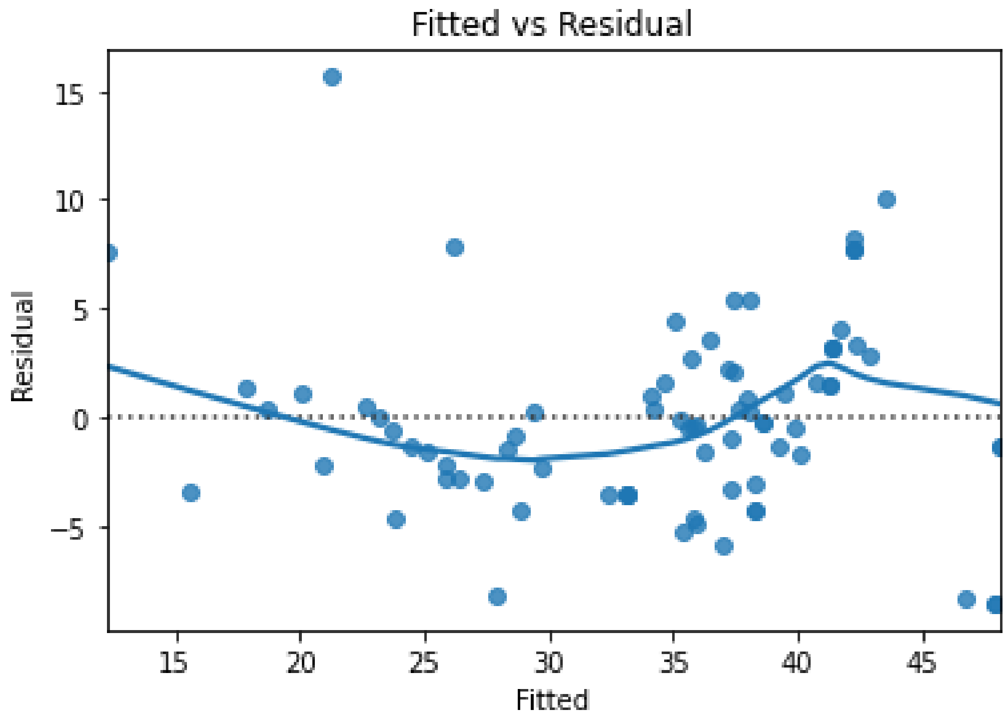

**Pair plot of Cars Dataset**



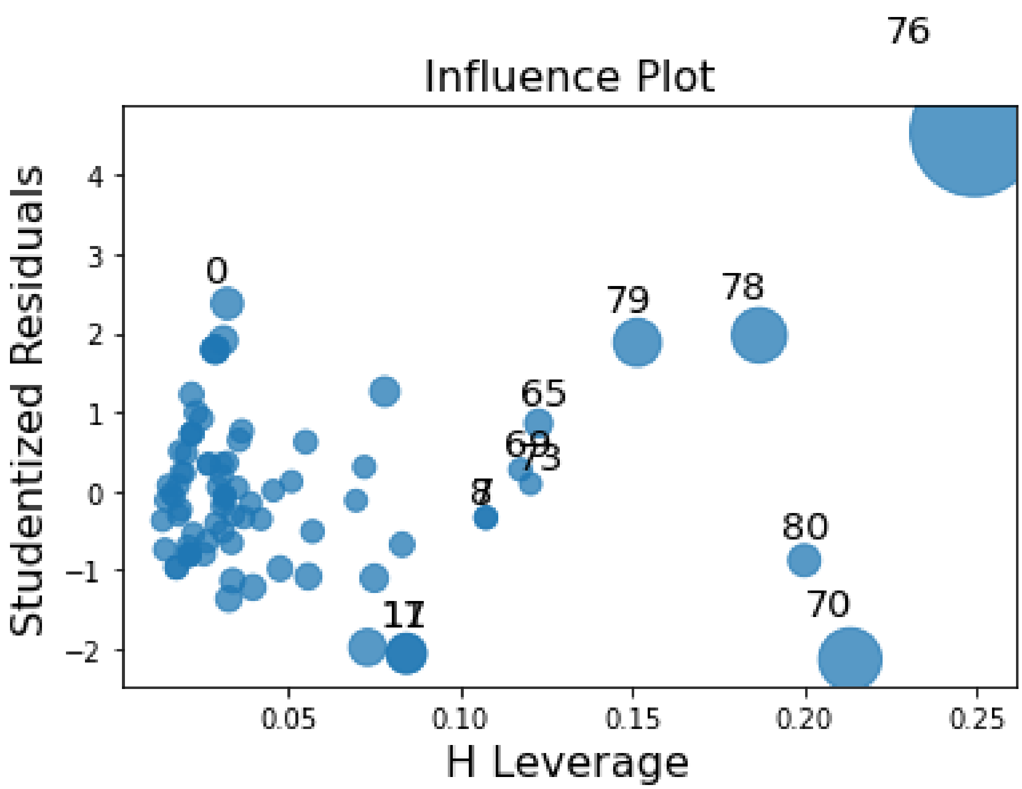

**Influence plot of Cars Dataset**




**QQ plot of Final Data Model**




**Scatter plot of Fitted Vs Residuals**




**QQ plot of Final Data Model**