**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

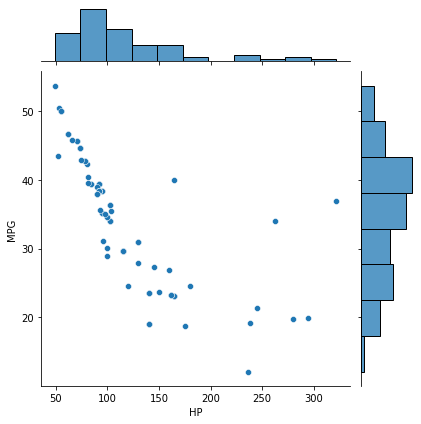
train\_resid = train\_pred - cars\_train.MPG

# RMSE value for train data

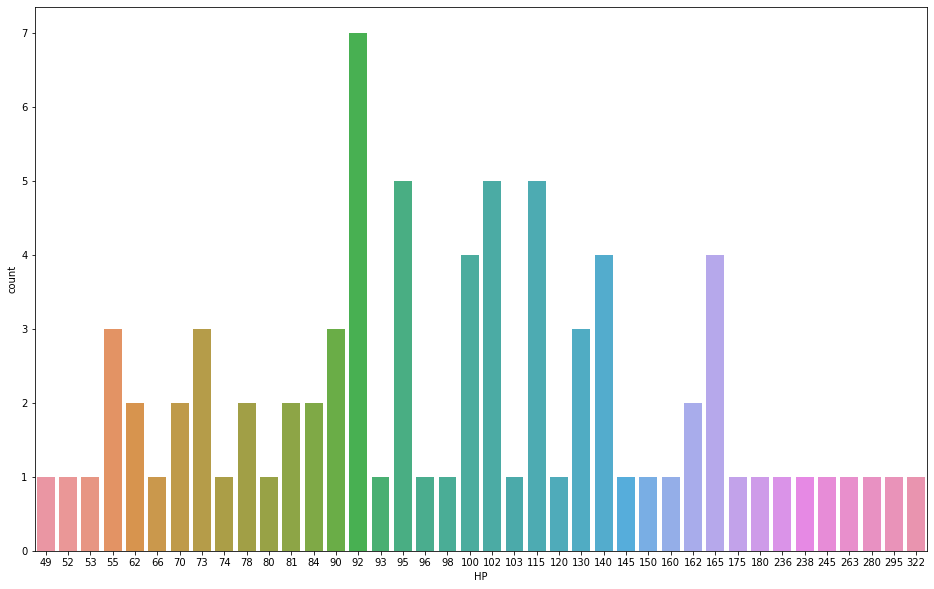
train\_rmse = np.sqrt(np.mean(train\_resid \* train\_resid))

train\_rmse

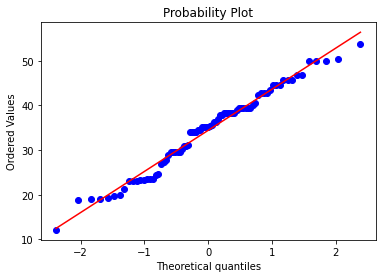
**Outputs:**

****

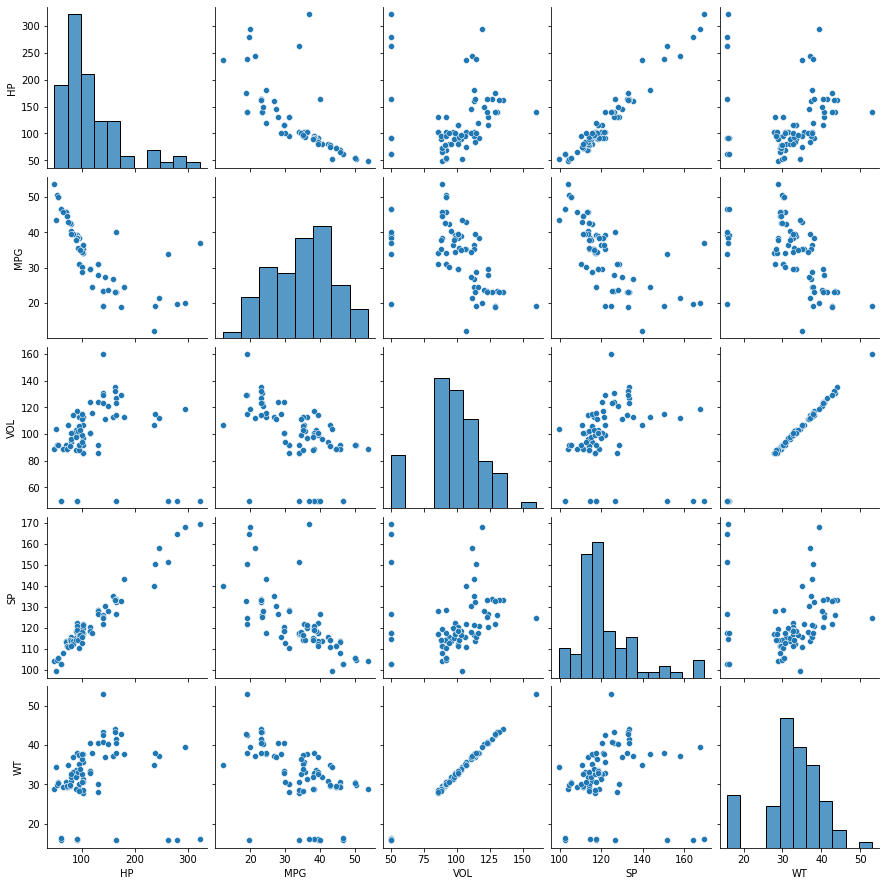
**Joint plot of Horsepower and Mileage**



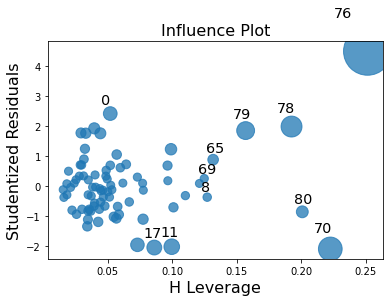
**Count plot of Horsepower**



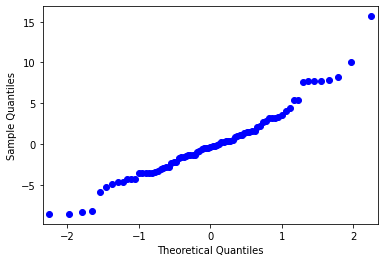
**QQ plot of Horsepower and Mileage**



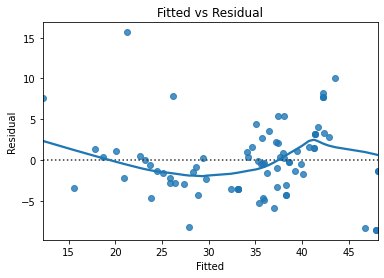
**Pair plot of Cars Dataset**



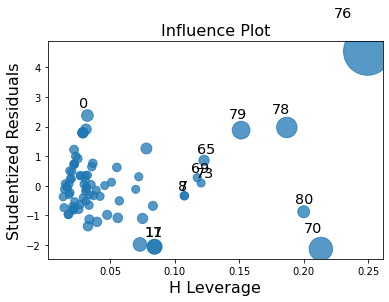
**Influence plot of Cars Dataset**

****

**QQ plot of Final Data Model**



**Scatter plot of Fitted Vs Residuals**



**QQ plot of Final Data Model**