Prepare a prediction model for profit of 50\_startups data.

Do transformations for getting better predictions of profit and

make a table containing R^2 value for each prepared model.

R&D Spend -- Research and devolop spend in the past few years

Administration -- spend on administration in the past few years

Marketing Spend -- spend on Marketing in the past few years

State -- states from which data is collected

Profit -- profit of each state in the past few years

ANS : import pandas as pd

import numpy as np

import matplotlib.pyplot as plt

import seaborn as sns

import statsmodels.formula.api as smf

import statsmodels.api as sm

from statsmodels.graphics.regressionplots import influence\_plot

sta=pd.read\_csv('E:\\assiment\\50\_Startups.csv')

sta.head()

sta.info()

data=sta.rename({'R&D Spend':'RDS','Administration':'ADMS','Marketing Spend':'MKTS'},axis=1)

data.head()

data[data.duplicated()]

data.describe()

data.corr()

sns.set\_style(style='white')

sns.pairplot(data)

model=smf.ols('Profit~RDS+ADMS+MKTS',data=data).fit()

model.params

model.tvalues , np.round(model.pvalues,4)

model.rsquared , model.rsquared\_adj

slr\_c=smf.ols('Profit~ADMS',data=data).fit()

slr\_c.tvalues , slr\_c.pvalues

slr\_d=smf.ols('Profit~MKTS',data=data).fit()

slr\_d.tvalues , slr\_d.pvalues

slr\_d=smf.ols('Profit~ADMS+MKTS',data=data).fit()

slr\_d.tvalues , slr\_d.pvalues

rsq\_RDS=smf.ols('RDS~ADMS+MKTS',data=data).fit().rsquared

vif\_RDS=1/(1-rsq\_RDS)

rsq\_ADMS=smf.ols('ADMS~RDS+MKTS',data=data).fit().rsquared

vif\_ADMS=1/(1-rsq\_ADMS)

rsq\_MKTS=smf.ols('MKTS~ADMS+RDS',data=data).fit().rsquared

vif\_MKTS=1/(1-rsq\_MKTS)

d={'Variables':['RDS','ADMS','MKTS'],

'Vif':[vif\_RDS,vif\_ADMS,vif\_MKTS]}

Vif\_df=pd.DataFrame(d1)

Vif\_df

sm.qqplot(model.resid,line='q')

plt.title("Normal Q-Q plot of residuals")

plt.show()

list(np.where(model.resid<-30000))

def standard\_values(vals) : return (vals-vals.mean())/vals.std()

plt.scatter(standard\_values(model.fittedvalues),standard\_values(model.resid))

plt.title('Residual Plot')

plt.xlabel('standardized fitted values')

plt.ylabel('standardized residual values')

plt.show()

fig=plt.figure(figsize=(15,8))

sm.graphics.plot\_regress\_exog(model,'RDS',fig=fig)

plt.show()

fig=plt.figure(figsize=(15,8))

sm.graphics.plot\_regress\_exog(model,'ADMS',fig=fig)

plt.show()

fig=plt.figure(figsize=(15,8))

sm.graphics.plot\_regress\_exog(model,'MKTS',fig=fig)

plt.show()

(c,\_)=model.get\_influence().cooks\_distance

c

fig=plt.figure(figsize=(20,7))

plt.stem(np.arange(len(data)),np.round(c,3))

plt.xlabel('Row Index')

plt.ylabel('Cooks Distance')

plt.show()

np.argmax(c) , np.max(c)

influence\_plot(model)

plt.show

k=data.shape[1]

n=data.shape[0]

leverage\_cutoff = (3\*(k+1))/n

leverage\_cutoff

data[data.index.isin([49])]

data\_new=data.copy()

data\_new.head()

data1=data\_new.drop(data\_new.index[[49]],axis=0).reset\_index(drop=True)

data1

new\_data1=pd.DataFrame({'RDS':70000,'ADMS':90000,'MKTS':120000},index=[0])

new\_data1