import numpy as np

import pandas as pd

data=pd.read\_csv("Data-PV1.csv")

data.info()

print(".........................................................")

print(data.isnull().sum())

data.dropna(inplace=True)

print(".........................................................")

data=data.drop\_duplicates()

display(data.describe())

print(".........................................................")

display(data)

#from sklearn.linear\_model import LinearRegression

from sklearn.model\_selection import train\_test\_split

x= data.iloc[:,:-1]

y=data.iloc[:,-1]

x\_train,x\_test,y\_train,y\_test = train\_test\_split(x,y,train\_size=0.75)

#m= LinearRegression()

m.fit(x\_train,y\_train)

print(len(x\_train),"train +",len(x\_test),"test")

print(m.score(x\_train,y\_train))

print(m.score(x\_test,y\_test))

from sklearn.neural\_network import MLPRegressor

model = MLPRegressor(activation="relu",hidden\_layer\_sizes=(150,150,150),solver="adam",shuffle=True,max\_iter=1000,batch\_size="auto")

model.fit(x\_train,y\_train)

model.fit(x\_test,y\_test)

print(model.score(x\_train,y\_train))

print(model.score(x\_test,y\_test))

from sklearn.neighbors import KNeighborsRegressor

model = KNeighborsRegressor(n\_neighbors=3)

model.fit(x\_train,y\_train)

model.fit(x\_test,y\_test)

print(model.score(x\_train,y\_train))

print(model.score(x\_test,y\_test))