**Data Preprocessing**

**Importing Libraries**

import numpy as np

import pandas as pd

import matplotlib.pyplot as plt

**Importing Dataset**

dataset =pd.read\_csv('water\_data.csv',encoding='unicode\_escape')

dataset

**Data Visualization and Missing Values**

dataset.info()

**Convert to numeric**

dataset.iloc[:,3:]=dataset.iloc[:,3:].applymap(lambda x: pd.to\_numeric(x,errors= "coerce"))

dataset.info()

**Drop Columns**

dataset.drop(['STATE','STATION CODE',"LOCATIONS","TOTAL COLIFORM (MPN/100ml)Mean"],axis=1,inplace=True)

**Check for Null Values**

print(dataset.isnull().any())

**Removing Outliners**

dataset.describe()

plt.scatter(range(1991),dataset["PH"])

plt.scatter(range(1991),dataset["D.O. (mg/l)"])

plt.scatter(range(1991),dataset["CONDUCTIVITY (µmhos/cm)"])

plt.scatter(range(1991),dataset["B.O.D. (mg/l)"])

plt.scatter(range(1991),dataset["NITRATENAN N+ NITRITENANN (mg/l)"])

plt.scatter(range(1991),dataset["FECAL COLIFORM (MPN/100ml)"])

dataset=dataset[dataset["PH"]<14]

dataset=dataset[dataset["PH"]>4]

dataset=dataset[dataset["B.O.D. (mg/l)"]<190]

dataset=dataset[dataset["FECAL COLIFORM (MPN/100ml)"]<1000000000]

print(dataset.info())

dataset.describe()

**Fill NaN**

dataset['Temp']=dataset['Temp'].replace(np.NaN,dataset['Temp'].mean())#26.318446

dataset['D.O. (mg/l)']=dataset['D.O. (mg/l)'].replace(np.NaN,dataset['D.O. (mg/l)'].mean())

dataset['CONDUCTIVITY (µmhos/cm)']=dataset['CONDUCTIVITY (µmhos/cm)'].replace(np.NaN,dataset['CONDUCTIVITY (µmhos/cm)'].mean())

dataset['NITRATENAN N+ NITRITENANN (mg/l)']=dataset['NITRATENAN N+ NITRITENANN (mg/l)'].replace(np.NaN,dataset['NITRATENAN N+ NITRITENANN (mg/l)'].mean())

dataset.info()

**Organize Dataset**

df=dataset.groupby(by=["year"],sort=True,as\_index=True).mean()

df

df.describe()

y=pd.Series()

yy=pd.DataFrame()

y=df["PH"].apply(lambda x: (0 if (8>=x>=7)

else (0.028 if (8.5>=x>=8) or (7>=x>=6.5)

else (0.084 if (9>=x>=8.8) or (6.5>=x>=6.3)

else (0.112 if (10>=x>=9) or (6.3>=x>=6)

else 0.14)))))

yy["PH"]=df["PH"].apply(lambda x: (0 if (8>=x>=7)

else (0.028 if (8.5>=x>=8) or (7>=x>=6.5)

else (0.084 if (9>=x>=8.8) or (6.5>=x>=6.3)

else (0.112 if (10>=x>=9) or (6.3>=x>=6)

else 0.14)))))

yy["D.O. (mg/l)"]=df["D.O. (mg/l)"].apply(lambda x: (0 if (8>=x>=6.5)

else (0.04 if (6.5>=x>=6)

else 0.2)))

y=y+yy["D.O. (mg/l)"]

yy["CONDUCTIVITY (µmhos/cm)"]=df["CONDUCTIVITY (µmhos/cm)"].apply(lambda x: (0 if (1500>=x>=50)

else (0.012 if (2000>=x>=1500)

else (0.048 if (2500>=x>=2000)

else 0.06))))

y=y+yy["CONDUCTIVITY (µmhos/cm)"]

yy["NITRATENAN N+ NITRITENANN (mg/l)"]=df["NITRATENAN N+ NITRITENANN (mg/l)"].apply(lambda x: (0 if (1>=x)

else (0.036 if (1.5>=x>=1)

else (0.144 if (2>=x>=1.5)

else 0.18))))

y=y+yy["NITRATENAN N+ NITRITENANN (mg/l)"]

yy["B.O.D. (mg/l)"]=df["B.O.D. (mg/l)"].apply(lambda x: (0 if (3>=x>=0)

else (0.024 if (5>=x>=3)

else (0.072 if (10>=x>=5)

else 0.12))))

y=y+yy["B.O.D. (mg/l)"]

yy["FECAL COLIFORM (MPN/100ml)"]=df["FECAL COLIFORM (MPN/100ml)"].apply(lambda x: (0 if (5000>=x>=0)

else (0.04 if (10000>=x>=5000)

else (0.12 if (100000>=x>=10000)

else 0.2))))

y=y+yy["FECAL COLIFORM (MPN/100ml)"]

y=y\*100

yy["y"]=y

yy

x=df.index.tolist()

x=list(map(lambda z:[z,],x))

x

y=list(y)

y

**Splitting Data**

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

x\_train,x\_test,y\_train,y\_test=train\_test\_split(x,y,test\_size=0.2,random\_state=0)

**Model Training**

from sklearn.linear\_model import LinearRegression

lr= LinearRegression()

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

**Predicting for Test Data**

y\_pre=lr.predict(x\_test)

y\_pre

plt.scatter(x\_test,y\_test)

plt.plot(x\_test,y\_pre,"r")

lr.predict([[2025]])

from sklearn.metrics import r2\_score

r2\_score(y\_test,y\_pre)

from sklearn.metrics import mean\_squared\_error

mean\_squared\_error(y\_test,y\_pre)

from joblib import dump

dump(lr,'model.save')