**Lab Submission-3**

**Water Quality Prediction**

**Source Code:**

import pandas as pd

import numpy as np

import matplotlib.pyplot as plt

cell\_df = pd.read\_csv('water\_quality1.csv')

cell\_df['class'].value\_counts()

impure\_df = cell\_df[cell\_df['class']==0][0:12]

pure\_df = cell\_df[cell\_df['class']==1][0:12]

axes = pure\_df.plot(kind='scatter', x='temperature', y='city', color='blue', label='pure')

impure\_df.plot(kind='scatter', x='temperature', y='city', color='red', label='impure', ax=axes)

cell\_df.columns

feature\_df = cell\_df[['temperature', 'dissolved oxygen', 'pH', 'conductivity', 'ORP',

       'turbidity']]

X = np.asarray(feature\_df)

# Dependent variable (are those which we have to predict)

Y = np.asarray(cell\_df['class'])

# X[0:5]

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=4)

# returns 4 numpy array as o/p in the above order.

# 23 x 6

X\_train.shape

# 23 x 1

Y\_train.shape

# 6 x 6

X\_test.shape

# 6 x 1

Y\_test.shape

from sklearn import svm

classifier = svm.SVC(kernel='linear', gamma='auto', C=2)

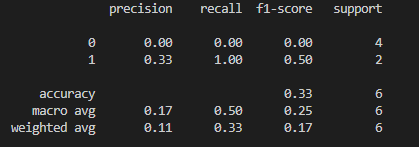
classifier.fit(X\_train, Y\_train)

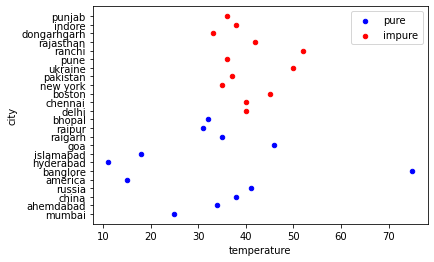
Y\_predict = classifier.predict(X\_test) #returns 1D array.

from sklearn.metrics import classification\_report

print(classification\_report(Y\_test, Y\_predict))

**Output:**

****

****