ASSIGNMENT - 7

ANIKEIT SETHI (190001003)

**# Model**

import numpy as np

class Model:

    def \_\_init\_\_(self, num\_inputs):

        self.weights = np.zeros(num\_inputs)

        self.bias = 0

    def set\_weight(self,num\_inputs):

        for i in range(len(num\_inputs)):

            self.weights[i]=num\_inputs[i]

    def predict(self, inputs):

        sum = np.dot(inputs, self.weights[:]) + self.bias

        if sum >= 0:

            prediction = 1

        else:

            prediction = 0

        return prediction

    def fit(self, inputs, label, epochs=3, learning\_rate=0.5):

        for \_ in range(epochs):

            for val, res in zip(inputs, label):

                prediction= self.predict(val)

                self.weights[:] += learning\_rate \* (res-prediction) \* val

                self.bias += learning\_rate \*(res-prediction)

**# Q1**

train\_inputs = np.array([[1,1,1,1,1,1,0],[0,1,1,0,0,0,0],[1,1,0,1,1,0,1],[1,1,1,1,0,0,1],

[0,1,1,0,0,1,1],[1,0,1,1,0,1,1],[1,0,1,1,1,1,1],[1,1,1,0,0,0,0],[1,1,1,1,1

,1,1],[1,1,1,1,0,1,1]])

labels = np.array([[0,0,0,0,0,0,0,0,0,1],[0,0,0,0,0,0,0,0,1,0],[0,0,0,0,0,0,0,1,0,0

],[0,0,0,0,0,0,1,0,0,0],[0,0,0,0,0,1,0,0,0,0],[0,0,0,0,1,0,0,0,0,0],[0,0,0

,1,0,0,0,0,0,0],[0,0,1,0,0,0,0,0,0,0],[0,1,0,0,0,0,0,0,0,0],[1,0,0,0,0,0,0

,0,0,0]])

perceptrons = []

for i in range(10):

    perceptron = Model(7)

    perceptron.fit(train\_inputs, labels[i],50,0.1)

    perceptrons.append(perceptron)

for j in range(10):

    inp = train\_inputs[j]

    print(inp)

    for i in perceptrons:

        print(i.predict(inp), end = " ")

    print("\n")

**Output**

[1 1 1 1 1 1 0]

0 0 0 0 0 0 0 0 0 1

[0 1 1 0 0 0 0]

0 0 0 0 0 0 0 0 1 0

[1 1 0 1 1 0 1]

0 0 0 0 0 0 0 1 0 0

[1 1 1 1 0 0 1]

0 0 0 0 0 0 1 0 0 0

[0 1 1 0 0 1 1]

0 1 0 0 0 1 0 0 0 0

[1 0 1 1 0 1 1]

0 0 0 0 1 0 0 0 0 0

[1 0 1 1 1 1 1]

0 1 0 1 0 0 0 0 0 0

[1 1 1 0 0 0 0]

0 0 1 0 0 0 0 0 0 0

[1 1 1 1 1 1 1]

0 1 0 0 0 0 0 0 0 0

[1 1 1 1 0 1 1]

1 0 0 0 0 0 0 0 0 0

Q2 (palindrome)

train\_inputs = np.array([[1,0,0,0,0,0,1],[1,0,1,0,0,1,1],[1,1,1,1,0,0,0],[0,0,1,1,0,1,1],

[1,1,0,0,0,1,1],[0,0,1,1,1,0,0]])

labels = np.array([1,0,0,0,1,1])

perceptron = Model(7)

perceptron.fit(train\_inputs, labels, 50, 0.1)

dataset = [

[1,0,0,0,0,0,1],

[0,0,1,1,0,1,1],

[0,1,0,0,0,1,0],

[1,1,0,0,0,1,1],

[1,0,1,1,1,0,1],

[1,0,1,0,0,1,1]

]

for i in dataset:

    print(i, ": is palindrome => ", perceptron.predict(i))

**Output**

[1, 0, 0, 0, 0, 0, 1] : is palindrome => 1

[0, 0, 1, 1, 0, 1, 1] : is palindrome => 0

[0, 1, 0, 0, 0, 1, 0] : is palindrome => 1

[1, 1, 0, 0, 0, 1, 1] : is palindrome => 1

[1, 0, 1, 1, 1, 0, 1] : is palindrome => 1

[1, 0, 1, 0, 0, 1, 1] : is palindrome => 0

# Q2 (majority element)

train\_inputs = np.array([[1,1,1,0,1,1,0,1,1],[1,0, 0, 0, 1, 0, 0, 1,

1],[1, 1, 0, 0, 0, 1, 0, 0, 1], [1, 1, 1, 1, 1, 0, 0, 0, 0],[1, 0, 0, 0,

1, 0, 1, 1, 1],[0, 0, 0, 0, 0, 0, 0, 0, 1]])

labels = np.array([1,0,0,1,1,0])

perceptron = Model(9)

perceptron.fit(train\_inputs, labels, 50, 0.1)

dataset = [

[1,1,1,1,1,1,0,0,0],

[1,0,0,0,1,0,0,1,1],

[1,0,1,0,1,0,1,0,1],

[0,0,0,1,1,1,1,1,1],

[0,1,0,1,1,1,0,1,1],

[0,0,0,0,0,0,0,0,1]

]

for i in dataset:

    print(i, ": Majority Element => ", perceptron.predict(i))

[1, 1, 1, 1, 1, 1, 0, 0, 0] : Majority Element => 1

[1, 0, 0, 0, 1, 0, 0, 1, 1] : Majority Element => 0

[1, 0, 1, 0, 1, 0, 1, 0, 1] : Majority Element => 1

[0, 0, 0, 1, 1, 1, 1, 1, 1] : Majority Element => 1

[0, 1, 0, 1, 1, 1, 0, 1, 1] : Majority Element => 1

[0, 0, 0, 0, 0, 0, 0, 0, 1] : Majority Element => 0