Міністерство освіти і науки України Державний університет «Одеська політехніка»

Кафедра прикладної математики

МОДУЛЬНА КОНТРОЛЬНА РОБОТА 1

з дисципліни "Нейроні мережі"

Виконала:
студентка 4 курсу
групи НАВ-191
Матиченко А.Д.

Прийняла:
Доктор техн. наук, доцент.
Полякова М. В.

Одеса 2022

Завдання 1. Задано навчальну вибірку: 1-й клас: $\begin{pmatrix}-1\\-1\end{pmatrix},\begin{pmatrix}0\\0\end{pmatrix}$; 2-й клас: $\begin{pmatrix}-1\\0\end{pmatrix},\begin{pmatrix}0\\-1\end{pmatrix}$.

Визначити нелінійне перетворення векторів-ознак навчальних прикладів так, щоб їх можна було розділити прямою.

Розв'язання.

1. Класи об'єктів нелінійно розділяються і потрібно побудувати поверхню, що розділяє. Визначимо нелінійне перетворення прикладів навчальної вибірки за допомогою двох радіальних базисних функцій $\varphi_1(x)=e^{\left(\|x-t_1\|^2\right)}$, $\varphi_2(x)=e^{\left(\|x-t_2\|^2\right)}$. $t_1(-1,-1)-center\ \varphi_1(x)$, $t_2(0,0)-center\ \varphi_2(x)$:

   $$\varphi_1(-1,-1)=e^{\left(-\left((-1+1)^2+(-1+1)^2\right)\right)}=e^0=1 \qquad \varphi_2(-1,-1)=e^{\left(-\left((-1-0)^2+(-1-0)^2\right)\right)}=e^{-2}=0.135$$

   $$\varphi_1(0,0)=e^{\left(-\left((0+1)^2+(0+1)^2\right)\right)}=e^{-2}=0.135 \quad , \quad \varphi_2(0,0)=e^{\left(-\left((0-0)^2+(0-0)^2\right)\right)}=e^0=1$$

   $$\varphi_1(-1,0)=e^{\left(-\left((-1+1)^2+(0+1)^2\right)\right)}=e^{-1}=0.37 \quad \varphi_2(-1,0)=e^{\left(-\left((-1-0)^2+(0-0)^2\right)\right)}=e^{-1}=0.37$$

   $$\varphi_1(0,-1)=e^{\left(-\left((0+1)^2+(-1+1)^2\right)\right)}=e^{-1}=0.37 \quad \varphi_2(0,-1)=e^{\left(-\left((0-0)^2+(-1-0)^2\right)\right)}=e^{-1}=0.37$$

   $$\begin{pmatrix}-1\\-1\end{pmatrix}\rightarrow\begin{pmatrix}1\\0.135\end{pmatrix},\begin{pmatrix}0\\0\end{pmatrix}\rightarrow\begin{pmatrix}0.135\\1\end{pmatrix},\begin{pmatrix}-1\\0\end{pmatrix}\rightarrow\begin{pmatrix}0.37\\0.37\end{pmatrix},\begin{pmatrix}0\\-1\end{pmatrix}\rightarrow\begin{pmatrix}0.37\\0.37\end{pmatrix}.$$

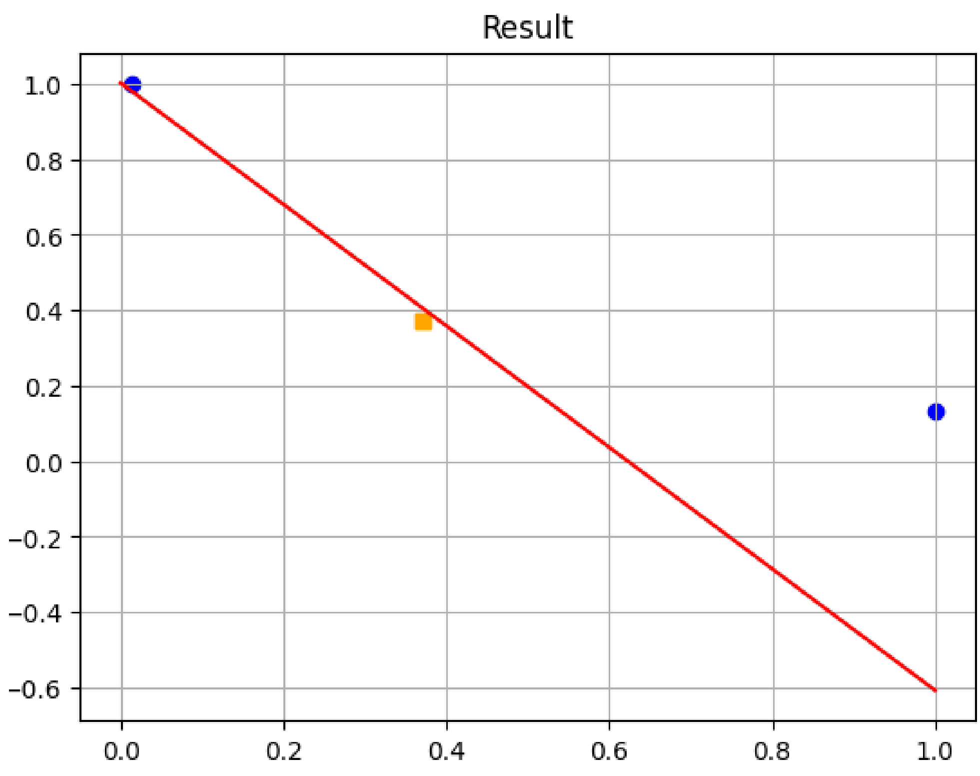
2. За допомогою персептрона побудуємо роздільну полищину. Для цього скористаємося кодом:

```
import numpy as np


class Perceptron():
def __init__(self, num_features = 2):
      self.num_features = num_features
      self.weights = np.zeros((num_features, 1), dtype=float)
      self.bias = np.zeros(1, dtype=np.float)

def forward(self, x):
      linear = np.dot(x, self.weights)+self.bias
      predictions = np.where(linear > 0, 1, 0)
      return predictions

def backward(self, x, y):
      predictions = self.forward(x)
      errors = y - predictions
      return errors

def train(self, x, y, epochs=1000):
      for e in range(epochs):
            for i in range(y.shape[0]):
```

```
                errors = self.backward(x[i].reshape(1, self.num_features),
                y[i]).reshape(-1)
                self.weights += (errors * x[i]).reshape(self.num_features,
                1)
                self.bias += errors

    def evaluate(self, x, y):
        predictions = self.forward(x).reshape(-1)
        accuracy = np.sum(predictions == y)/  y.shape[0]
        return accuracy

data = np.genfromtxt('data/dataset.txt', delimiter=' ')
X_train, y_train = data[:, :2], data[:, 2]
print(data
ppn = Perceptron()
ppn.train(X_train, y_train, epochs=50)
print("Parameters of model:\t")
print('Weights:%s\n' % ppn.weights)
print('Bias: %s\n' % ppn.bias)
```

3. Отримаємо: $-0.9975x_1 - 1.605x_2 + 1 = 0$
4. Скористаємося кодом для побудови гіперплощини:

```
import matplotlib.pyplot as plt

x_min, x_max = 0,1
x=np.linspace(x_min,x_max)
y=np.array([1.002 -1.609*temp for temp in x])
plt.plot(x,y,color='red')
plt.scatter(X_train[y_train  ==  0,  0],  X_train[y_train  ==  0,  1],
label='class 0', marker='o', color='blue')
plt.scatter(X_train[y_train  ==  1,  0],  X_train[y_train  ==  1,  1],
label='class 1', marker='s', color='orange')
plt.title('Result')
plt.grid()
```



Завдання 2. Виконати навчання персептрона для 2-х класів на навчальній вибірці з векторів: 1-й клас: $\begin{pmatrix}1\\0\end{pmatrix}$, 2-й клас: $\begin{pmatrix}1\\1\end{pmatrix}$, $\begin{pmatrix}0\\0\end{pmatrix}$.

Розв'язання.

Розширення вектора функцій$1:\begin{pmatrix}1\\0\\1\end{pmatrix};\ 2:\begin{pmatrix}1\\1\\1\end{pmatrix},\ \begin{pmatrix}0\\0\\1\end{pmatrix}$

Виберіть початковий вектор коефіцієнтів зважування . $w(1)=\begin{pmatrix}1\\-2\\0\end{pmatrix}$

Встановіть коефіцієнт корекції, щоб бути більше нуля. $C=1$

1. Наведемо перший приклад для тренувань. і обчислимо скалярний добуток, оскільки він належить до першого класу, а скалярний добуток дорівнює нулю, то виконуємо корекцію. $x(1)=\begin{pmatrix}1\\0\\1\end{pmatrix}$

   $w(k)^T\,x(k)=(1,-2,0)\begin{pmatrix}1\\0\\1\end{pmatrix}=1\,x(1)\ w=(2)=w(1)$

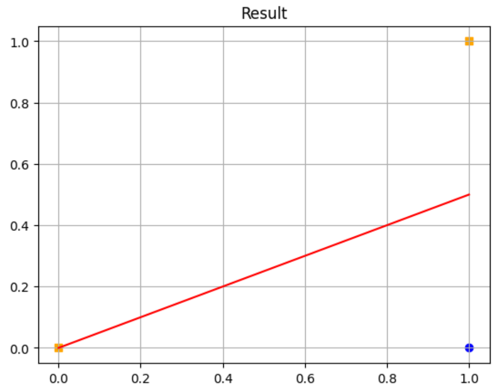
2. Наведемо для навчання другий приклад, належить до класу 2 і векторний добуток менше нуля, корекція не виконується., $x(2)=\begin{pmatrix}1\\1\\1\end{pmatrix}, w(2)^T\,x(2)=(1,-2,0)\begin{pmatrix}1\\1\\1\end{pmatrix}=-1, w(3)=w(2)$.

3. Наведемо для навчання третій приклад, належить до класу 2 і векторний добуток дорівнює нулю, корекція не виконується . $x(3)=\begin{pmatrix}0\\0\\1\end{pmatrix}, w(4)^T\,x(3)=(1,-2,0)\begin{pmatrix}0\\0\\1\end{pmatrix}=0$.

4. Рішення вважається отриманим, якщо помилка класифікації не виникає при пред'явленні всієї вибірки перцептрону, тобто, наприклад, 4 рази

поспіль корекція не проводиться, тобто стільки разів, скільки прикладів в навчальному зразку, що і виконалося.

5. у нас є тільки три вагові фактори, функція прийме таку форму: $d(x) = w_1x_1 + w_2x_2 + w_3$ - ця функція розділяється двома класами. Для того щоб отримати розділову поверхню (це межа двох класів), потрібно прирівняти дискримінанту функцію до нуля. Це межа двох класів. $d(x) = 1x_1 - 2x_2 + 0, x_2 = \frac{1}{2}x_1;$



6. Відподвідь: в нашому випадку, будемо вважати, що точкі які лежать на прямій відповідають классу 1.

Завдання 3. Вхідні вектори лінійного нейрона $\begin{pmatrix}-3\\2\end{pmatrix},\begin{pmatrix}1\\0\end{pmatrix},\begin{pmatrix}-2\\3\end{pmatrix},\begin{pmatrix}0\\1\end{pmatrix}$ вихідні значення: $(-1,1,-1,1)$, Отримати вагові коефіцієнти лінійного нейрона, що перетворюють входи у виходи. Побудувати поверхню, що розділяє.

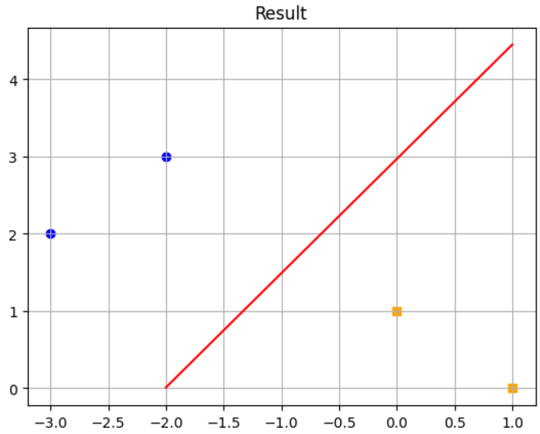
Розв'язання

1. Рішення знаходимо за формулою $\omega = X^+d, \omega = \left(X^TX\right)^{-1}X^Td$. Для цього скористаємося наступним кодом:

```
import numpy as np

#Data
x=np.array([-3,1,-2,0])
y=np.array([2,0,3,1])
z=np.array([1,1,1,1])
d=np.array([-1,1,-1,1])


#Solution
X=np.vstack((x,y,z))
w=np.linalg.pinv(X.T)
res=w.dot(d.T)
print(res)
```

2. Отримаємо поверхню $0.43x_1 - 0.29x_2 + 0.86 = 0$ з наступними ваговими коефіцієнтами $\omega = (0.43; -0.29; 0.86)$

3. Побудувати поверхню, що розділяє наші точки за допомогою наступного коду:




```
import matplotlib.pyplot as plt

x_min, x_max = -2,1
x=np.linspace(x_min,x_max)
y=np.array([1.48*temp+2.97 for temp in x])
plt.plot(x,y,color='red')
plt.scatter(X_train[y_train == 0, 0], X_train[y_train == 0, 1], label='class 0', marker='o', color='blue')
plt.scatter(X_train[y_train == 1, 0], X_train[y_train == 1, 1], label='class 1', marker='s', color='orange')
plt.title('Result')
plt.grid()
```