

МИНОБРНАУКИ РОССИИ
САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ
ЭЛЕКТРОТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ
«ЛЭТИ» ИМ. В. И. УЛЬЯНОВА (ЛЕНИНА)

Кафедра математического обеспечения и применения ЭВМ

ОТЧЕТ
по практической работе №8
по дисциплине «Машинное обучение»

Студент гр. 1310

Комаров Д. Е.

Преподаватель

Жангиров Т.Р.

Санкт-Петербург

2025

Задание 1

Постановка задачи

Дан набор данных, представленный в таблице 1.

Таблица 1 – Данные для 1 задания

i	\mathbf{x}_i	y_i
\mathbf{x}_1	(4,2.9)	1
\mathbf{x}_2	(3.5,4)	1
\mathbf{x}_3	(2.5,1)	-1
\mathbf{x}_4	(2,2.1)	-1

Необходимо:

- 1) рассчитать μ_{+1} и μ_{-1} , а так матрицу B - матрица межклассового разброса;
- 2) рассчитать $S+1$ и $S-1$, а также матрицу S – матрица внутриклассового разброса;
- 3) найти направление w , которое лучше всего дискриминирует классы;
- 4) на направлении w , найти точку, которая лучше всего делит классы.

Код программы

```
import numpy as np

D = [
    [4,2.9,1],
    [3.5,4,1],
    [2.5,1,-1],
    [2,2.1,-1]
]

Dp=np.array([d[:-1] for d in D if d[-1]==1]).transpose()
Dn=np.array([d[:-1] for d in D if d[-1]==-1]).transpose()
mup=np.mean(Dp, axis=1)
mun=np.mean(Dn, axis=1)
print(mup)
print(mun)
mudelt=np.matrix(mup-mun)
B=np.dot(mudelt.transpose(), mudelt)
print(B)
Sp=len(Dp)*np.cov(Dp, ddof=0)
```

```

Sn=len(Dn)*np.cov(Dn, ddof=0)
S=Sp+Sn
print(Sp)
print(Sn)
print(S)
S = S + 1e-9 * np.eye(S.shape[0])
SiB=np.dot(np.linalg.inv(S), B)
eig=np.linalg.eigh(SiB)
print(eig.eigenvalues)
print(eig.eigenvectors)
w=eig.eigenvectors[1]
t=np.dot(w, (mup+mun)/2)
print(t*w)

```

Результат выполнения

1) Были получены

$$\mu_{+1} = \begin{pmatrix} 3.75 \\ 3.45 \end{pmatrix},$$

$$\mu_{-1} = \begin{pmatrix} 2.25 \\ 1.55 \end{pmatrix},$$

$$B = \begin{pmatrix} 2.25 & 2.85 \\ 2.85 & 3.61 \end{pmatrix}.$$

2) Были получены

$$S_{+1} = \begin{pmatrix} 0.13 & -0.28 \\ -0.28 & 0.61 \end{pmatrix};$$

$$S_{-1} = \begin{pmatrix} 0.13 & -0.28 \\ -0.28 & 0.61 \end{pmatrix};$$

$$S = \begin{pmatrix} 0.25 & -0.55 \\ -0.55 & 1.21 \end{pmatrix}.$$

3) Так как матрицы S вырождена, было добавлено значение 10^{-9} к главной диагонали. Направление, которое лучше всего дискриминирует классы

$$w = \begin{pmatrix} -0.84 \\ -0.53 \end{pmatrix},$$

4) Точка на направлении w , которая лучше всего делит классы

$$\mu = \begin{pmatrix} 3.27 \\ 2.09 \end{pmatrix}.$$

Задание 2

Постановка задачи

Дан набор данных, представленный в таблице 2.

Таблица 2 – Данные для 2 задания

i	x_{i1}	x_{i2}	y_i	α_i
\mathbf{x}_1	4	2.9	1	0.414
\mathbf{x}_2	4	4	1	0
\mathbf{x}_3	1	2.5	-1	0
\mathbf{x}_4	2.5	1	-1	0.018
\mathbf{x}_5	4.9	4.5	1	0
\mathbf{x}_6	1.9	1.9	-1	0
\mathbf{x}_7	3.5	4	1	0.018
\mathbf{x}_8	0.5	1.5	-1	0
\mathbf{x}_9	2	2.1	-1	0.414
\mathbf{x}_{10}	4.5	2.5	1	0

Необходимо:

- 1) рассчитать уравнение гиперплоскости $h(x)$ для метода опорных векторов;
- 2) вычислить расстояние от точки x_6 до гиперплоскости $h(x)$, указать, находится ли эта точка в границах опорных векторов;
- 3) классифицировать точку (3, 3) используя гиперплоскость $h(x)$.

Код программы

```

import numpy as np
D = np.array([
    [4, 2.9, 1, 0.414],
    [4, 4, 1, 0],
    [1, 2.5, -1, 0],
    [2.5, 1, -1, 0.018],
    [4.9, 4.5, -1, 0],
    [1.9, 1.9, -1, 0],
    [3.5, 4, 1, 0.018],
    [0.5, 1.5, -1, 0],
    [2, 2.1, -1, 0.414],
    [4.5, 2.5, 1, 0]
])
w=np.sum([x[:2]*x[2]*x[3] for x in D],axis=0)
b=1/D[0][2]-np.dot(w,D[0][:2])
print(w[0],w[1],b)
dist=abs(w[0]*D[5][0]+w[1]*D[5][1]+b)/np.sqrt(w[0]**2+w[1]**2)
print(dist)
print(abs(w[0]*D[5][0]+w[1]*D[5][1]+b))
point=[3,3]
h=w[0]*point[0]+w[1]*point[1]+b
print(h)
if h>0:

```

```
    print('class +1')
else:
    print('class -1')
```

Результат выполнения

1) Уравнение гиперплоскости

$$h(x_i) = 0.85x_{i1} + 0.39x_{i2} - 3.5.$$

2) Расстояние от точки x_6 до гиперплоскости $h(x)$ равно 1.25. Точка не находится в границах опорных векторов, так как $h(x_6) = 1.16$.

3) Точка (3,3) принадлежит к классу +1, так как $h(3,3) = 0.19$ больше 0.