Machine learning and vision laboratory

Dr H.Mohammad Zadeh

دانشگاه صنعتی شریف

department: Electrical Engineering

Amirreza Velae 400102222
github repository

Pre-Lab 3

November 17, 2024

Machine learning and vision laboratory

Pre-Lab 3

Amirreza Velae 400102222
github repository

We have two classes, 1 and -1, and we want to find a method to classify their data.

- **Class 1**: $(1, 1), (-1, -1)$
- **Class -1**: $(1, -1), (-1, 1)$

1. Can these data points be linearly separated?

Soloution

If we visualize the data points, we can see that they can not be linearly separated. Because the data points of each class are located in the opposite corners of the square. So, we can not draw a line to separate them. Here is the python code to visualize the data points:

```python
import matplotlib.pyplot as plt

class1 = [(1, 1), (-1, -1)]
class2 = [(1, -1), (-1, 1)]

plt.scatter(*zip(*class1), c='r', label='class 1')
plt.scatter(*zip(*class2), c='b', label='class -1')
plt.legend()
plt.show()
```

Here is the result of the code:

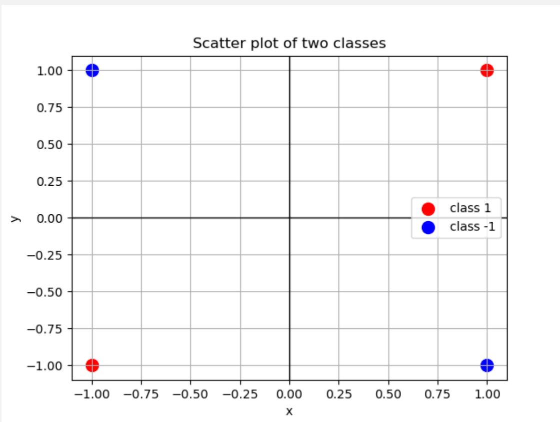

Figure 1: Data points

Obviously, we can not draw a line to separate the data points. So, the data points can not be linearly separated.

2. Define $y(x) = W^T \phi(x)$, where x is a two-dimensional vector (x_1 as the first dimension and x_2 as the second dimension), and $\phi(x) = (1, x_1, x_2, x_1 x_2)$. Determine W such that in this new space, the data points can be linearly separated (i.e., y can classify the data points).

Soloution

We can write the $y(x)$ as follows:

$$\begin{aligned} y(x) &= W^T \phi(x) \\ &= w_0 + w_1 x_1 + w_2 x_2 + w_3 x_1 x_2 \end{aligned}$$

First, lets find $\phi(x)$ for each data point:

$$\begin{aligned} \texttt{class 1:} \quad & \phi(1,1) = (1,1,1,1) \\ & \phi(-1,-1) = (1,-1,-1,1) \\ \texttt{class -1:} \quad & \phi(1,-1) = (1,1,-1,-1) \\ & \phi(-1,1) = (1,-1,1,-1) \end{aligned}$$

Now, we can write the $y(x)$ for each data point:

$$\begin{aligned} \texttt{class 1:} \quad & y(1,1) = w_0 + w_1 + w_2 + w_3 = 1 \\ & y(-1,-1) = w_0 - w_1 - w_2 + w_3 = 1 \\ \texttt{class -1:} \quad & y(1,-1) = w_0 + w_1 - w_2 - w_3 = -1 \\ & y(-1,1) = w_0 - w_1 + w_2 - w_3 = -1 \end{aligned}$$

Now we have four equations with four unknowns. We can solve this linear system of equations to find W:

$$Ax = b$$

$$\begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -1 & -1 & 1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & 1 & -1 \end{bmatrix} \begin{bmatrix} w_0 \\ w_1 \\ w_2 \\ w_3 \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \\ -1 \\ -1 \end{bmatrix}$$

We can solve this linear system of equations using python:

```python
        import numpy as np

        A = np.array([[1, 1, 1, 1],
                      [1, -1, -1, 1],
                      [1, 1, -1, -1],
                      [1, -1, 1, -1]])
        b = np.array([1, 1, -1, -1])

        W = np.linalg.solve(A, b)
        print(W)

```

The result of the code is:

$$W = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix}$$

and yes, I'm an very absentminded, because it was so obvious from the beginning.

3. Find a kernel function $K(x, x')$ such that $K(x, x') = \phi(x)^T \phi(x')$ or determine if it exists. Explain why this kernel function can distinguish between the two classes.

Soloution

We can write the $K(x, x')$ as follows:

$$\begin{aligned} K(x, x') &= \phi(x)^T \phi(x') \\ &= (1, x_1, x_2, x_1x_2)^T (1, x'_1, x'_2, x'_1x'_2) \\ &= 1 + x_1x'_1 + x_2x'_2 + x_1x_2x'_1x'_2 \end{aligned}$$

Now, we can write the $K(x, x')$ for each data point:

`class 1:` $K(x_{1,1}, x_{-1,-1}) = 1 + 1 \times -1 + 1 \times -1 + 1 \times -1 \times 1 \times -1 = 4$

`class -1:` $K(x_{1,-1}, x_{-1,1}) = 1 + 1 \times -1 + -1 \times 1 + 1 \times 1 \times -1 \times 1 = 0$

As we can see, the $K(x, x')$ can distinguish between the two classes. Because the $K(x, x')$ for the data points of class 1 is different from the $K(x, x')$ for the data points of class -1. So, the $K(x, x')$ can distinguish between the two classes. But this was arbitrary from beginning, because the $K(x, x')$ has the term $x_1x_2x'_1x'_2$ which considers sign of the data points.