

单层神经网络案例实践

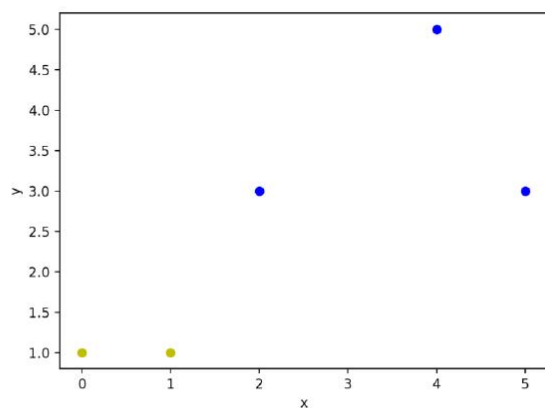
目录

- ◆ 线性分类问题
- ◆ 单层感知器求解

线性分类问题

二维线性分类问题

◆ 单层感知器作为线性分类器被广泛应用



输入数据X, 各维度分别表示偏置, x坐标, y坐标

```
X = np.array ([[1,2,3],
```

```
               [1,4,5],
```

```
               [1,1,1],
```

```
               [1,5,3],
```

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

标签Y

```
Y = np.array([1,1,-1,1,-1])
```

二维线性分类问题

◆ 问题分析

```
# 输入数据X, 各维度分别表示偏置, x坐标, y坐标
X = np.array ([[1,2,3],
               [1,4,5],
               [1,1,1],
               [1,5,3],
               [1,0,1]])

# 标签Y
Y = np.array([1,1,-1,1,-1])
```

$$y = f\left(\sum_{i=1}^n w_i x_i + b\right)$$

$$\begin{aligned} f(WX + b) &= f(w[0] * X[0] + w[1] * X[1] + w[2] * X[2]) \\ &= f(w[0] + w[1]x + w[2]y) \end{aligned}$$

x, y 表示实际的二维坐标

单层感知器求解

◆ 令误差函数为 L , 预测值为 y , 阈值函数 f 为符号函数

- 则 $L = \frac{1}{2}(Y - y)^2$, $y = f(wx + b) = \text{sign}(wx + b)$

基于梯度下降法的参数更新为:

$$\begin{aligned} w_{t+1} &= w_t + \Delta w = w_t - \tau \frac{\partial L}{\partial w} = w_t - \tau \frac{\partial L}{\partial y} \frac{\partial y}{\partial w} = \\ &= w_t + \tau(Y - y) \frac{\partial y}{\partial w} \approx w_t + \tau(Y - y)x \end{aligned}$$

单层感知器求解

基于Python求解单层感知器算法

◆ 代码实现与结果示意

```
#coding:utf8
import numpy as np
import matplotlib.pyplot as plt

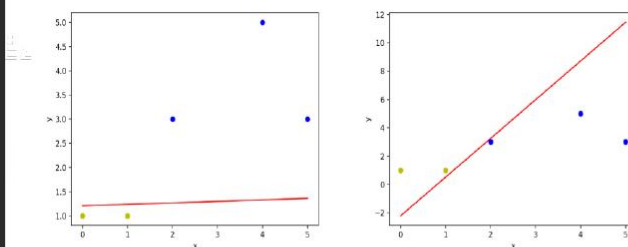
n = 0          #迭代次数
lr = 0.10      #学习速率

# 输入数据，三个维度分别表示偏置，x坐标，y坐标
X = np.array([[1,2,3],
               [1,4,5],
               [1,1,1],
               [1,5,3],
               [1,0,1]])

# 标签
Y = np.array([1,1,-1,1,-1])

# 要学习的模型， $f(WX+b)$ ，对于正样本， $f(WX+b)>0$ ，对于负样本， $f(WX+b)<0$ 

# 权重W初始化，取值范围-1到1
W = (np.random.random(X.shape[1])-0.5)*2
```



多个正确答案

下次预告：多层感知器与反向传播算法