

ANN(BP)-线性不可分问题

ANN(BP)-线性不可分问题

算法设计

网络结构

训练数据

参数规定

输入

标签

学习速率

中间层 $a_{n(1)}$ 的求解

结果层 $a_{n(2)}$ 的求解

激活函数

误差函数

初值设定

算法目标

算法实现

1、前向传播

输入->隐层

隐层->输出

2、反向传播（优化 $w'_{0,0}$ ）

2.1、计算误差

2.2、隐含层->输出层

误差对 $w'_{0,0}$ 求导

计算 $\frac{\partial E}{\partial neto_1}$

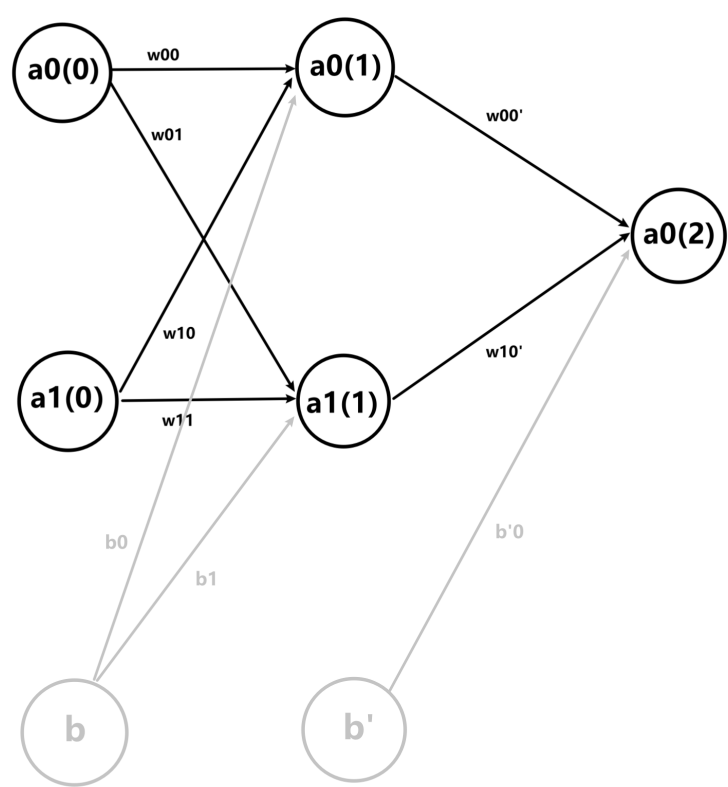
计算 $\frac{\partial neto_1}{\partial w'_{0,0}}$

合并

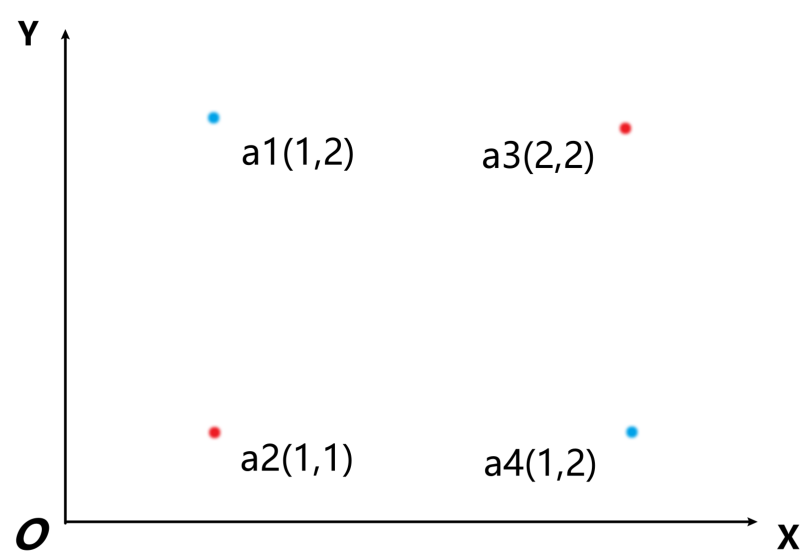
2.3、更新 $w'_{0,0}$

算法设计

网络结构



训练数据



参数规定

输入

- $a_0(0)$: X值
- $a_1(0)$: Y值

标签

- 蓝色: 尽量接近0 (大概率负样本)
- 红色: 尽量接近1 (大概率正样本)

学习速率

- $\eta = 0.8$

中间层 $a_{n(1)}$ 的求解

$$a_{n(1)} = g \left(\begin{bmatrix} w_{0,0} & w_{1,0} \\ w_{0,1} & w_{1,1} \end{bmatrix} \begin{bmatrix} a_{0(0)} \\ a_{1(0)} \end{bmatrix} + \begin{bmatrix} b_0 \\ b_1 \end{bmatrix} \right) \quad (1)$$

结果层 $a_{n(2)}$ 的求解

$$a_{n(2)} = g \left(\begin{bmatrix} a_{0(1)} & a_{1(1)} \end{bmatrix} \begin{bmatrix} w'_{0,0} \\ w'_{1,0} \end{bmatrix} + \begin{bmatrix} b'_0 \end{bmatrix} \right) \quad (2)$$

激活函数

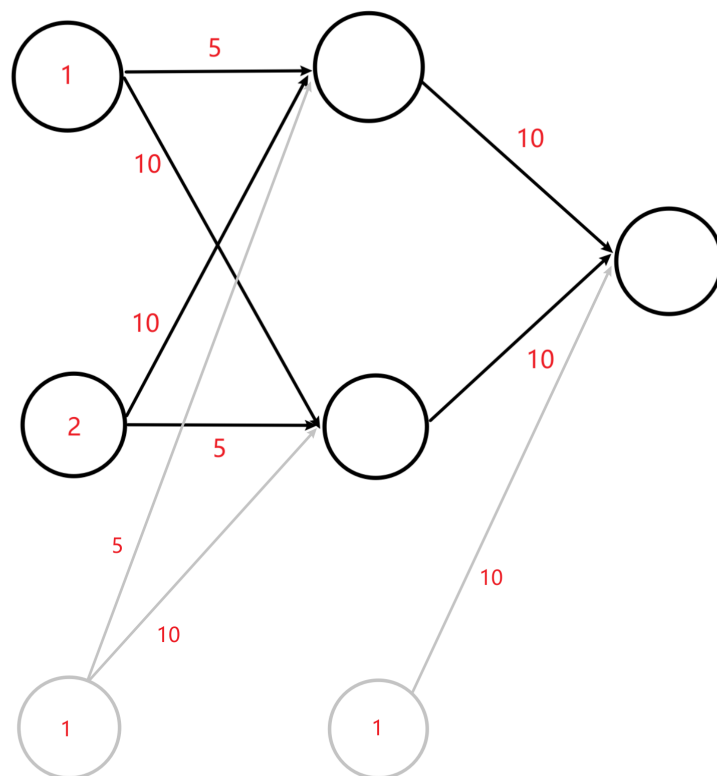
sigmoid函数: $g(z) = \frac{1}{1+e^{-z}}$

即将结果 z 压缩到空间 $[0,1]$ 中, 数值表示**正样本**的可能性。

误差函数

$$E = \frac{1}{2} (target - output)^2$$

初值设定



算法目标

输入数据 $a_1(1, 2)$, 更新 $w'_{0,0}$ 使输出数值接近0 (大概率负样本)

算法实现

1、前向传播

输入->隐层

$$\begin{aligned}
 a_{n(1)} &= g \left(\begin{bmatrix} 5 & 10 \\ 10 & 5 \end{bmatrix} \begin{bmatrix} 1 \\ 2 \end{bmatrix} + \begin{bmatrix} 5 \\ 10 \end{bmatrix} \right) \\
 &= g \left(\begin{bmatrix} 25 \\ 10 \end{bmatrix} + \begin{bmatrix} 5 \\ 10 \end{bmatrix} \right) \\
 &= g \left(\begin{bmatrix} 30 \\ 20 \end{bmatrix} \right) \\
 &= \begin{bmatrix} 0.9999999999999999065 \\ 0.99999999979388463 \end{bmatrix}
 \end{aligned}$$

```
>>> print(sigmoid(30))
0.999999999999999065
>>> print(sigmoid(20))
0.9999999979388463
```

隐层->输出

$$\begin{aligned} a_{n(2)} &= g \left([0.999999999999999065 \quad 0.9999999979388463] \begin{bmatrix} 10 \\ 10 \end{bmatrix} + [10] \right) \\ &= g(29.999999979387528) \\ &= 0.999999999999999065 \end{aligned}$$

2、反向传播（优化 $w'_{0,0}$ ）

2.1、计算误差

$$\begin{aligned} E &= \frac{1}{2}(\text{target} - \text{output})^2 \\ &= 0.499999999999999065 \end{aligned}$$

2.2、隐含层->输出层

误差对 $w'_{0,0}$ 求导

$$\frac{\partial E}{\partial w_{0,0}} = \frac{\partial E}{\partial \text{neto}_1} \times \frac{\partial \text{neto}_1}{\partial w_{0,0}}$$

计算 $\frac{\partial E}{\partial \text{neto}_1}$

$$E = g(\text{neto}_1) \quad (3)$$

$$\begin{aligned} \frac{\partial E}{\partial \text{neto}_1} &= \text{neto}_1(1 - \text{neto}_1) = 9.99999999999999065 \times (1 - 9.99999999999999065) \\ &= -89.999999999998225 \end{aligned}$$

计算 $\frac{\partial \text{neto}_1}{\partial w'_{0,0}}$

$$\text{neto}_1 = a_{0(0)} \times w'_{0,0} + a_{1(0)} \times w'_{1,0} + b'_0 \quad (4)$$

$$\frac{\partial out_1}{\partial w'_{0,0}} = a_{0,0} = 1 \quad (5)$$

合并

$$\frac{\partial E}{\partial w_{0,0}} = -89.99999999998225 + 1 = -88.99999999998225 \quad (6)$$

2.3、更新 $w'_{0,0}$

$$\begin{aligned} new_w'_{0,0} &= w'_{0,0} - \eta \times \frac{\partial E}{\partial w_{0,0}} \\ &= 10 - 0.8 \times -88.99999999998225 \\ &= 81.1999999999858 \end{aligned}$$