感知器

传奇是从这儿开始的

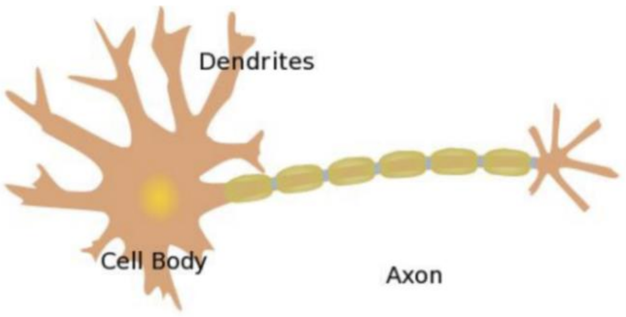
神经网络的起源

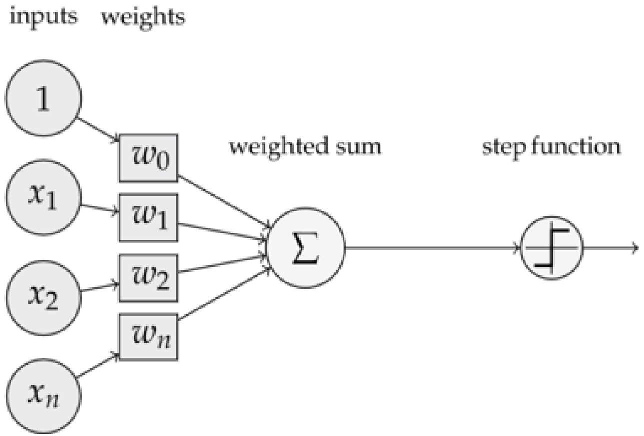
- 感知器（Perceptron）
  - Frank Rosenblatt 于20世纪50年代提出
  - 前馈计算
  - 激活函数
  - 感知器求解规则
- 感知器的局限
  - 异或（XOR）问题
  - 线性分割
- 多层感知机（Multi-layer perceptron）
  - S型激活函数
  - 反向传播
  - 梯度下降

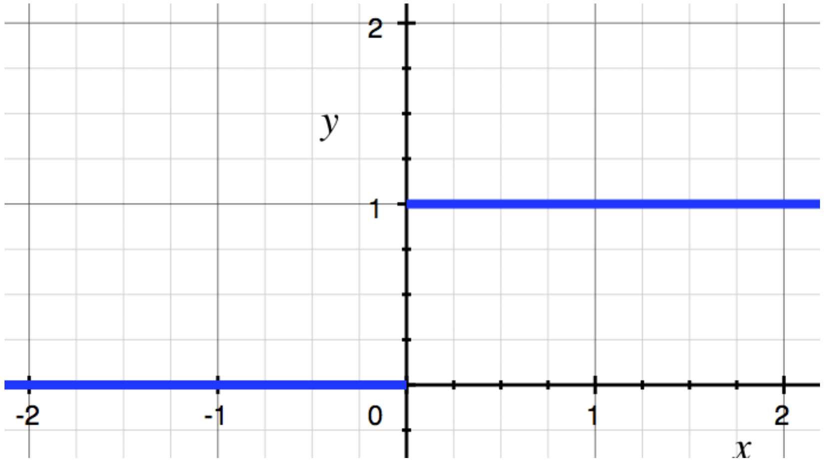
▶ 感知器（perceptron）

$$
\begin{aligned}
y &= f(w \cdot x + b) \\
&= f(w_1x_1 + w_2x_2 + w_3x_3 + bias)
\end{aligned}
$$

$$f(x)=\begin{cases}1 & x>0 \\ 0 & x\le 0\end{cases}$$



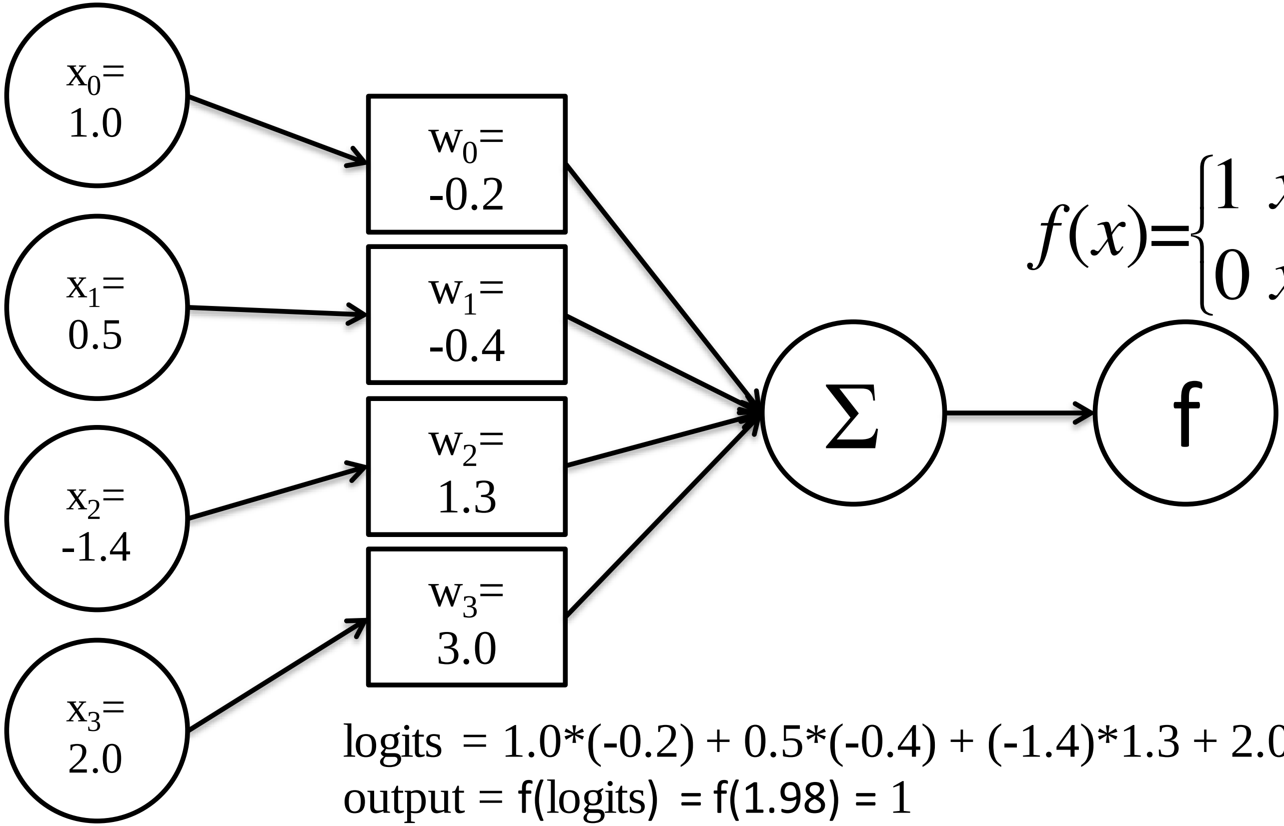




感知器的前馈计算

- $logit = w_0 x_0 + w_1 x_1 + w_2 x_2 + w_3 x_3 + \ldots + w_n x_n$
- w_0 = b (bias，偏置)， $x_0 = 1$
- w = [w_0, w_1, w_2, … , wn], x = [x_0, x_1, x_2, … x_n]
  则logit = w · x
- output = f(logit), $f(x)=\begin{cases} 1 & x>0 \\ 0 & x\le 0 \end{cases}$

感知器的前馈计算



$$f(x)=\begin{cases}1 & x>0\\ 0 & x\leq 0\end{cases}$$

logits = 1.0*(-0.2) + 0.5*(-0.4) + (-1.4)*1.3 + 2.0*3.0 = 1.98

output = f(logits) = f(1.98) = 1

感知器的前馈计算

- 向量化
  例如
  x_1=[-1.0, 3.0, 2.0]　　w = [4.0, -3.0, 5.0]
  x_2=[2.0, -1.0, 5.0]　　b = 2.0
  x_3=[-2.0, 0.0, 3.0]
  x_4=[4.0, 1.0, 6.0]

- 则X = $\begin{bmatrix} -1.0 & 3.0 & 2.0 \\ 2.0 & -1.0 & 5.0 \\ -2.0 & 0.0 & 3.0 \\ 4.0 & 1.0 & 6.0 \end{bmatrix}$

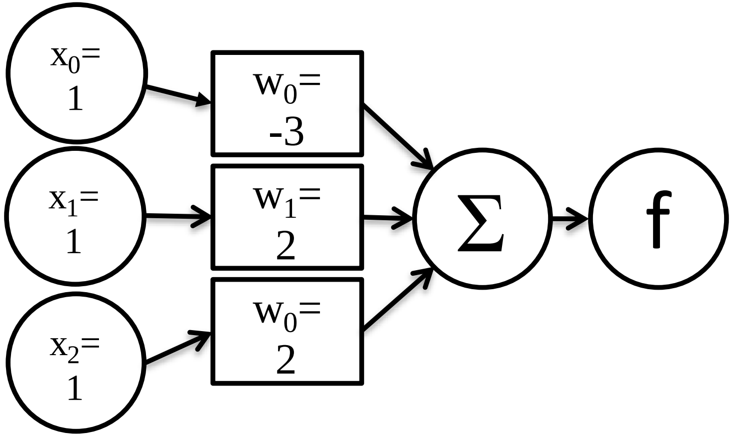
$$logits = \begin{bmatrix} -1.0 & 3.0 & 2.0 \\ 2.0 & -1.0 & 5.0 \\ -2.0 & 0.0 & 3.0 \\ 4.0 & 1.0 & 6.0 \end{bmatrix} \cdot \begin{bmatrix} 4.0 \\ -3.0 \\ 5.0 \end{bmatrix} + 2.0$$

$$= \begin{bmatrix} -1.0 & 38.0 & 7.0 & 43.0 \end{bmatrix}$$

- 则 $output = f(x) = \begin{bmatrix} 0 & 1 & 1 & 1 \end{bmatrix}$

感知器的运用

- 使用感知器可以完成一些基础逻辑操作
  - 例如：逻辑与

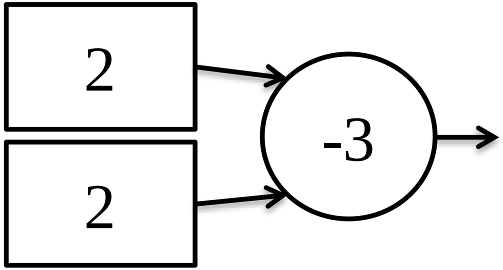


x1	x2	output
1	1	1
1	0	0
0	1	0
0	0	0

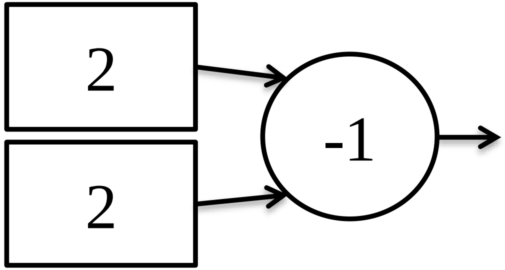
感知器实现逻辑运算

不 止 于 代 码

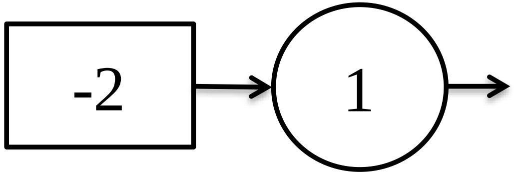
逻辑与



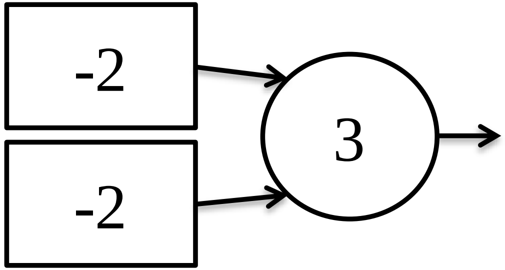
逻辑或



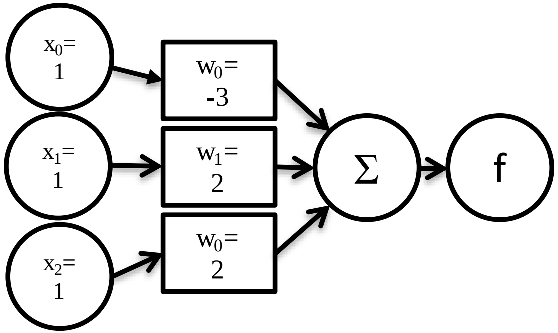
逻辑非



逻辑与非



感知器实现逻辑与



x1	x2	output
1	1	1
1	0	0
0	1	0
0	0	0

$$w_1x_1 + w_2x_2 + b = 0$$

- - - - $x_1 + x_2 - 1 = 0$

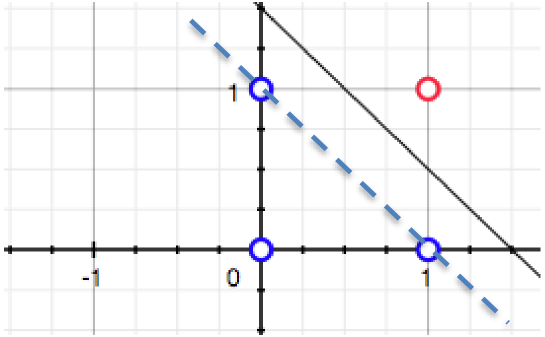
——— $2x_1 + 2x_2 - 3 = 0$

通过真值表求解

$$\begin{cases} 1 \times w_1 + 1 \times w_2 + b & > 0 \\ 1 \times w_1 + 0 \times w_2 + b & \le 0 \\ 0 \times w_1 + 1 \times w_2 + b & \le 0 \\ 0 \times w_1 + 0 \times w_2 + b & \le 0 \end{cases} \rightarrow$$

可能的一些解：

$$\begin{cases} w_1 = & 2 \\ w_2 = & 2 \\ b = & -3 \end{cases} \text{或} \begin{cases} w_1 = & 1 \\ w_2 = & 1 \\ b = & -1 \end{cases}$$



数值求解

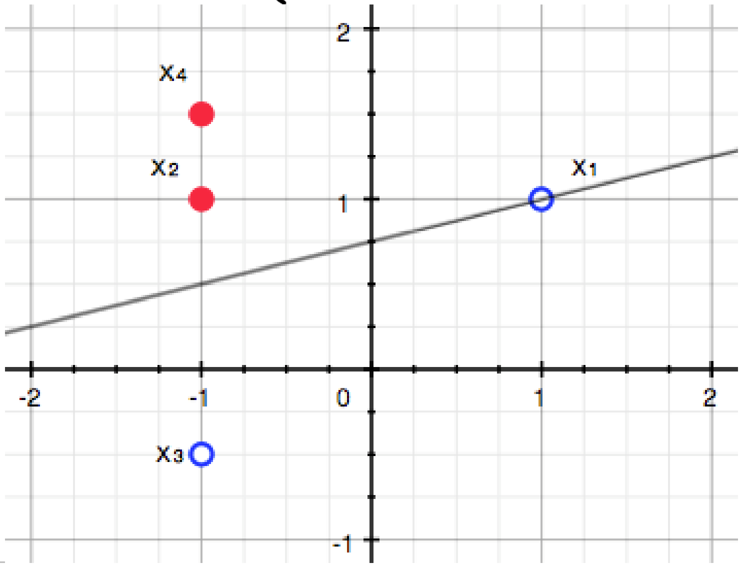
$$\mathbf{X} = \begin{bmatrix} 1 & 1 \\ -1 & 1 \\ -1 & -0.5 \\ -1 & 1.5 \end{bmatrix} \qquad \mathbf{Y} = \begin{bmatrix} 0 \\ 1 \\ 0 \\ 1 \end{bmatrix}$$

$$\begin{cases} w_1 + w_2 + b & \leq 0 \\ -w_1 + w2 + b & > 0 \\ -w_1 - 0.5w_2 + b & \leq 0 \\ -w_1 + 1.5w_2 + b & > 0 \end{cases}$$

直接进行数值求解

一组可能的解：

$$\begin{cases} w_1 = & -1 \\ w_2 = & 4 \\ b = & -3 \end{cases}$$



▶ 感知器的学习规则

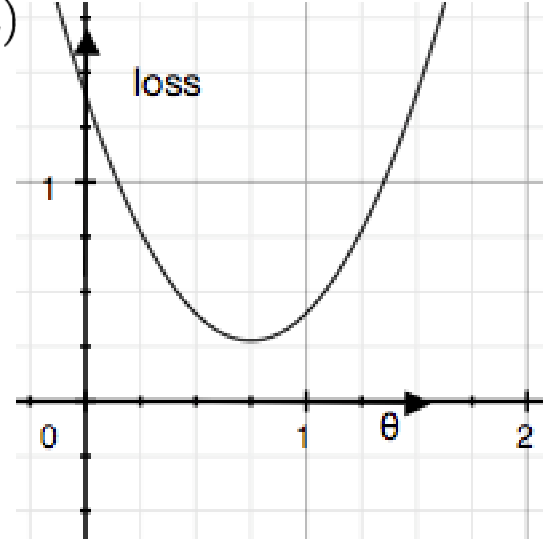
- 回顾一下梯度下降算法
  - 监督学习
  - 误差和损失函数（loss/cost function）

损失函数的导函数称为梯度函数，代表着损失函数在θ轴各点的斜率

$$y = \theta x \quad t = (ground \quad truth)$$

$$\begin{aligned} loss &= \frac{1}{2}(t-y)^2 \\ &= \frac{1}{2}t^2 - ty + \frac{1}{2}y^2 \\ &= \frac{1}{2}t^2 - t\theta x + \frac{1}{2}\theta^2 x^2 \end{aligned}$$

$$L(\theta) = \frac{1}{2}x^2 \cdot \theta^2 - tx \cdot \theta + \frac{1}{2}t^2$$



$$\begin{aligned} grad_\theta &= \frac{dL}{d\theta} \\ &= \frac{dL}{d(t-y)} \frac{d(t-y)}{d\theta} \\ &= (t-y) \cdot (-x) \\ &= (y-t)x \end{aligned}$$

$$\begin{aligned} grad_\theta &= \frac{dL}{d\theta} \\ &= x^2\theta - tx \\ &= x(\theta x - t) \\ &= (y-t)x \end{aligned}$$

$$\theta_{new} = \theta_{old} - \eta \cdot grad_\theta$$

学习率η

$$y = \theta x$$

$$loss = \frac{1}{2}(t-y)^2$$

$$grad_\theta = (y-t)x$$

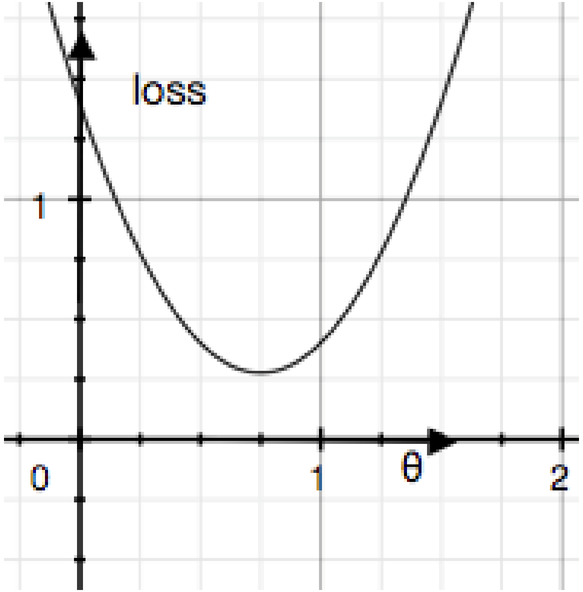
$$\theta_{new} = \theta_{old} - \eta \cdot grad_\theta$$

若η=1

1)
$$\theta = 3, x = 2, t = 8$$
$$y = 3 \times 2 = 6$$
$$loss = \frac{1}{2}(8-6)^2 = 2$$
$$grad_\theta = (6-8) \times 2 = -4$$
$$\theta_{new} = 3 - (-4) = 7$$

2)
$$\theta = 7, x = 2, t = 8$$
$$y = 7 \times 2 = 14$$
$$loss = \frac{1}{2}(8-14)^2 = 18$$
$$grad_\theta = (14-8) \times 2 = 12$$
$$\theta_{new} = 7 - 12 = -5$$

3)
$$\theta = -5, x = 2, t = 8$$
$$y = -5 \times 2 = -10$$
$$loss = \frac{1}{2}(8-(-10))^2 = 162$$
$$grad_\theta = (-10-8) \times 2 = -36$$
$$\theta_{new} = -5 - (-36) = 31$$



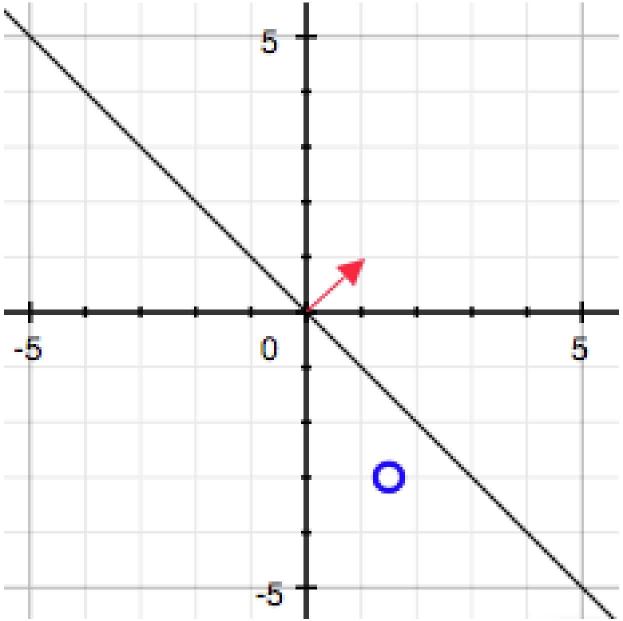
▶ 感知器学习规则

设 $x_1 = 1.5, \quad x_2 = -3$

$w_1 = 1, \quad w_2 = 1$

$b = 0$

此时 $$\begin{aligned} output &= f(w_1 x_1 + w_2 x_2 + b) \\ &= f(1 \times 1.5 + 1 \times -3 + 0) \\ &= f(-1.5) \\ &= 0 \end{aligned}$$



约定 $y = output$，若t=1，那么 $$\begin{aligned} \Delta &= t - y \\ &= 1 - 0 \\ &= 1 \end{aligned}$$

- 定义损失函数如下：

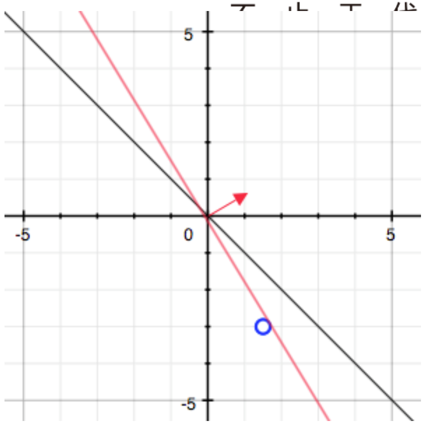
$$L(w,b) = -(t-y)(x \cdot w + b)$$

- 其对w和b的梯度为：

$$grad_w = \frac{dL}{dw} = -(t-y)x, \quad grad_b = \frac{dL}{db} = -(t-y)$$

- 对权重更新时使用：

$$w_{new} = w_{old} + \eta(t-y)x, \quad b_{new} = b_{old} + \eta(t-y)$$



- 对之前的例子应用权重更新：

$$\begin{aligned} W &= W + \eta(t-y)x \\ &= \begin{bmatrix}1 & 1\end{bmatrix} + 0.1 \times (1-0) \times \begin{bmatrix}1.5 & -3\end{bmatrix} \\ &= \begin{bmatrix}1.15 & 0.7\end{bmatrix} \end{aligned}$$

$$\begin{aligned} b &= b + \eta(t-y) \\ &= 0 + 0.1 \times (1-0) \\ &= 0.1 \end{aligned}$$

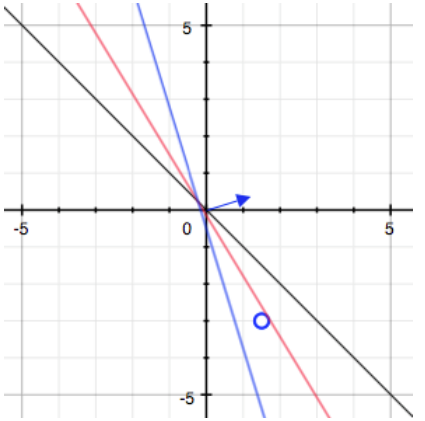
- 更新后的

$$\begin{aligned} output &= f(Wx+b) \\ &= f(1.15 \times 1.5 + 0.7 \times (-3) + 0.1) \\ &= f(-0.274) \\ &= 0 \end{aligned}$$

- 再次更新后

$$W = \begin{bmatrix}1.3, 0.4\end{bmatrix}, b = 0.2$$

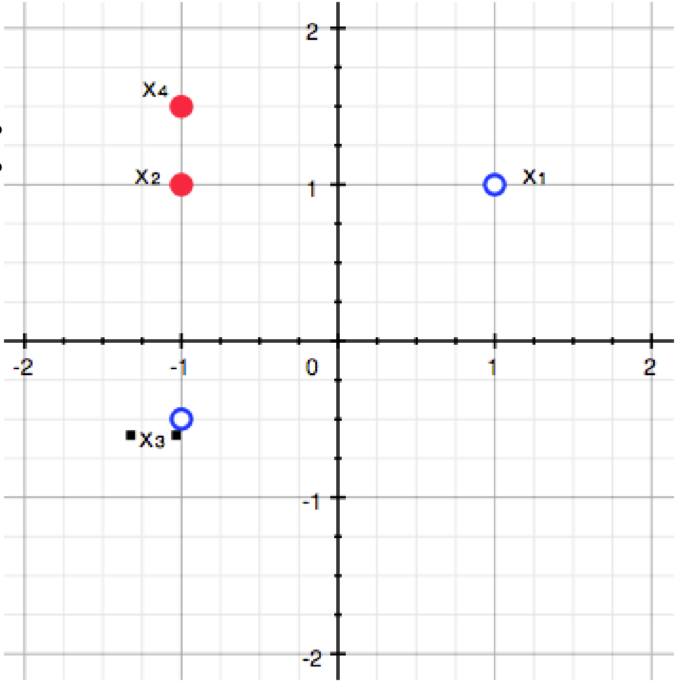
$$\begin{aligned} output &= f(1.3.5 + 0.4 + 0.2) \\ &= f(0.95) \\ &= 1 \end{aligned}$$



▶ 感知器学习规则

- 多条数据情况
- 假定我们有这样几条输入：

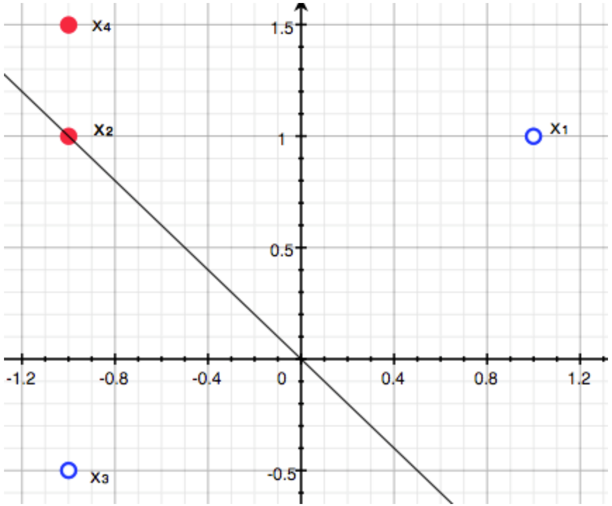
$$X = \begin{bmatrix} 1 & 1 \\ -1 & 1 \\ -1 & -0.5 \\ -1 & 1.5 \end{bmatrix} \quad t = \begin{bmatrix} 0 \\ 1 \\ 0 \\ 1 \end{bmatrix}$$



初始化设置w=[1, 1] ， b=0，则

$$logits = X \cdot w + b = \begin{bmatrix} 1 & 1 \\ -1 & 1 \\ -1 & -0.5 \\ -1 & 1 \end{bmatrix} \cdot \begin{bmatrix} 1 \\ 1 \end{bmatrix} + 0 = \begin{bmatrix} 1 \\ 0 \\ -1.5 \\ 0.5 \end{bmatrix}$$

$$output = f(logits) = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 1 \end{bmatrix}, \qquad f(x) = \begin{cases} 1, x > 0 \\ 0, x \le 0 \end{cases}$$



现在， $t = \begin{bmatrix} 0 \\ 1 \\ 0 \\ 1 \end{bmatrix}, y = output = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 1 \end{bmatrix}$

$$grad_w = -\sum_i (t-y)x_i$$

$$= -\left(\begin{bmatrix}0\\1\\0\\1\end{bmatrix} - \begin{bmatrix}1\\0\\0\\1\end{bmatrix}\right)\cdot\begin{bmatrix}1 & 1\\-1 & 1\\-1 & -0.5\\-1 & 1\end{bmatrix}$$

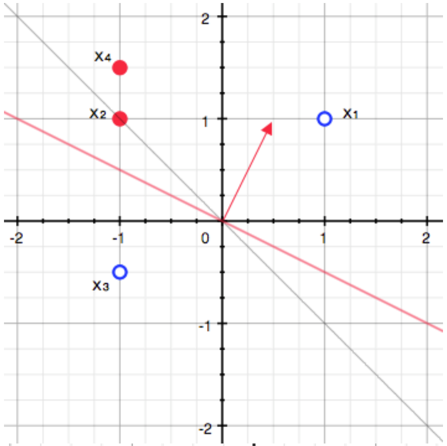
$$= -\begin{bmatrix}-1\\1\\0\\0\end{bmatrix}\cdot\begin{bmatrix}1 & 1\\-1 & 1\\-1 & -0.5\\-1 & 1\end{bmatrix}$$

$$= -\begin{bmatrix}-2 & 0\end{bmatrix}$$

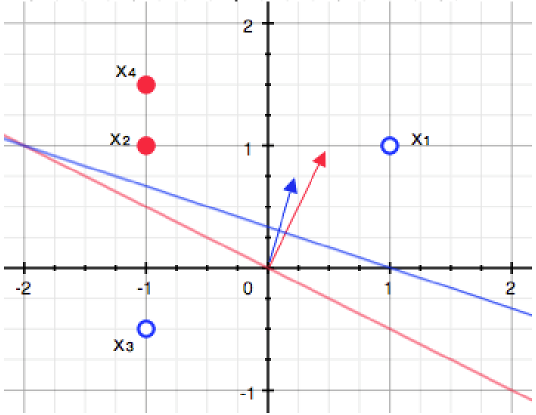
$$grad_b = -\sum(t-y)$$

$$= -(-1+1+0+0)$$

$$= 0$$



设η=0.25，则 $$W_{new} = W_{old} - \eta grad_w = \begin{bmatrix}0.5 & 1\end{bmatrix}$$ $$b = 0$$



重复这个过程，得到新的

$$grad_w = \sum_i -(t-y)x_i$$

$$= -\begin{bmatrix}-1 & -1\end{bmatrix}$$

$$grad_b = \sum -(t-y)$$

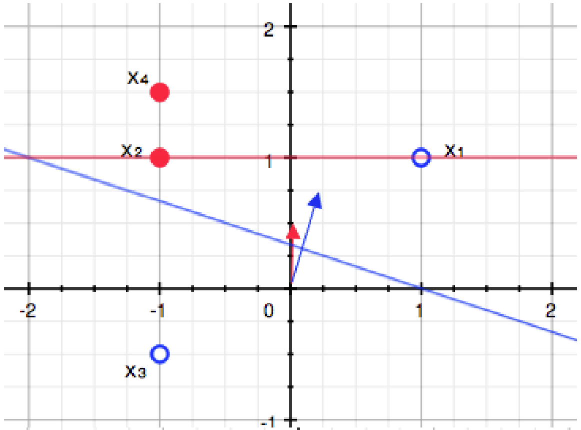
$$= 1$$

$$W_{new} = W_{old} - \eta grad_w = \begin{bmatrix}0.25 & 0.75\end{bmatrix}$$

$$b = -0.25$$

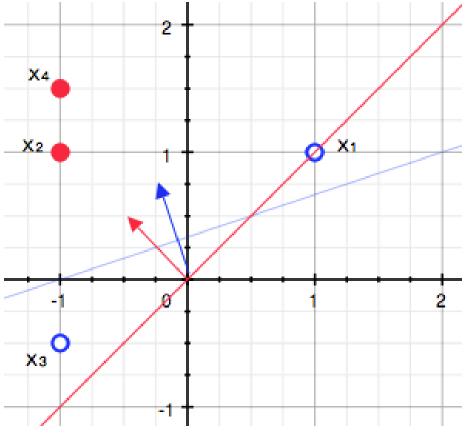
3) $W_{new} = W_{old} - \eta grad_w = \begin{bmatrix} 0 & 0.5 \end{bmatrix}$

$b = -0.5$

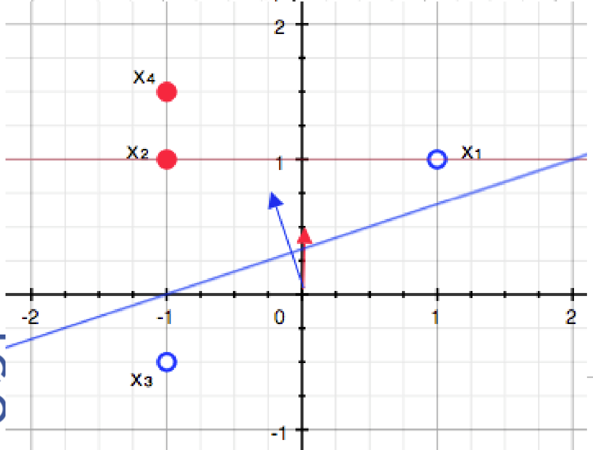


4) $W_{new} = W_{old} - \eta grad_w = \begin{bmatrix} 0.25 & 0.75 \end{bmatrix}$

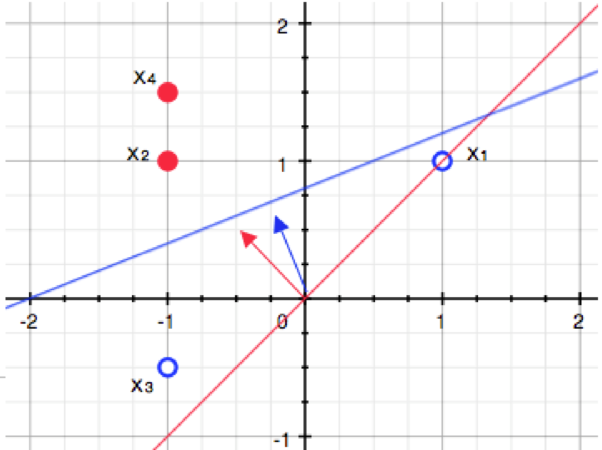
$b = -0.25$



5)

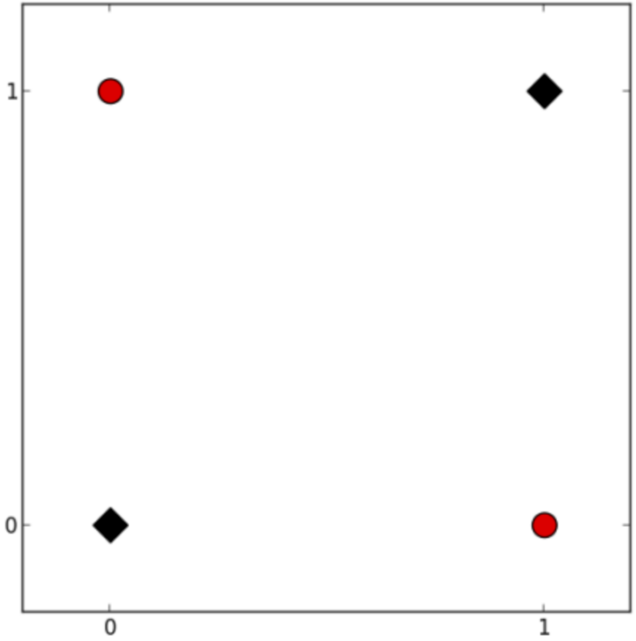


6)



感知器的局限

- 仅能做0-1输出
- 仅能处理线性分类问题
  - 无法处理XOR问题



- 如何解决？
  - 20年后，多层感知机解决了这些问题

THANK YOU