深度学习

构造神经网络解决XOR问题

Author:WangQi
Email:wangqikaixin@gmail.com

概览

1. 感知器与线性可分问题。

2. 激活函数的进化。

3. 线性不可分问题。

4. 构造逻辑运算神经元。

5. 构造神经网络解决XOR问题。

6. Python实现神经网络。

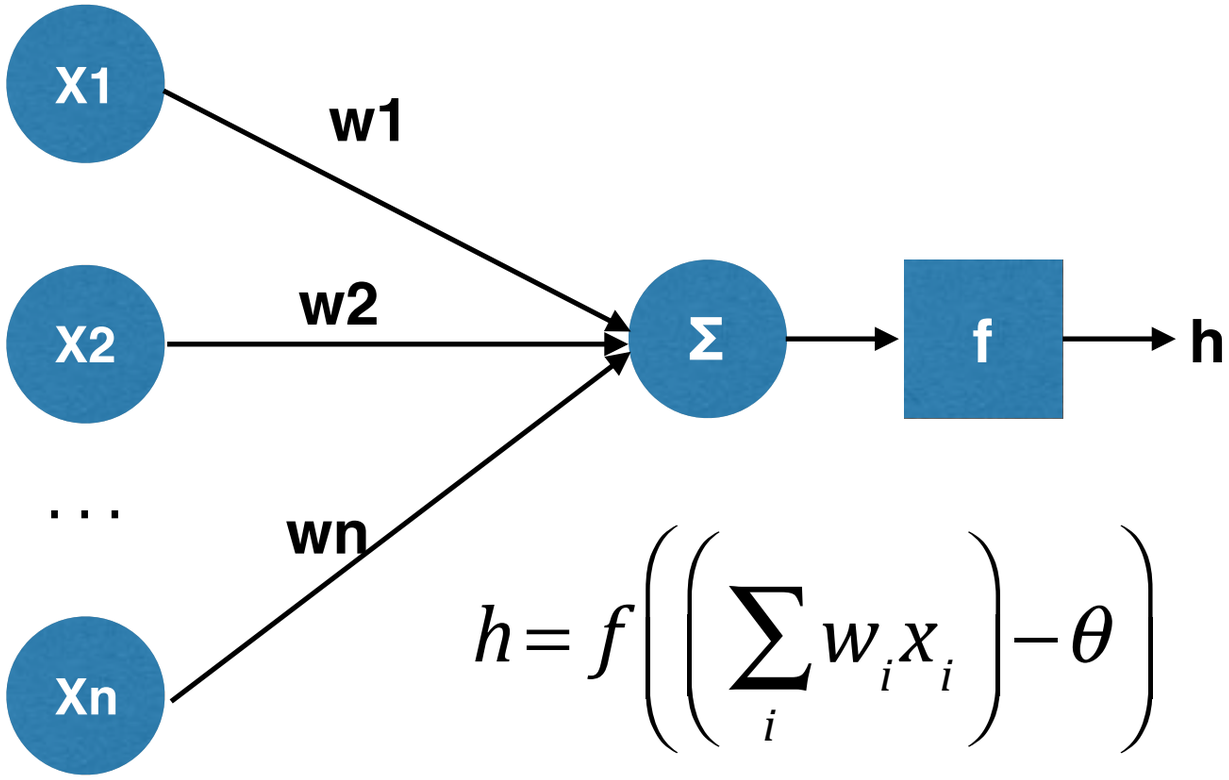
Author:WangQi
Email:wangqikaixin@gmail.com

1. 感知器与线性可分问题

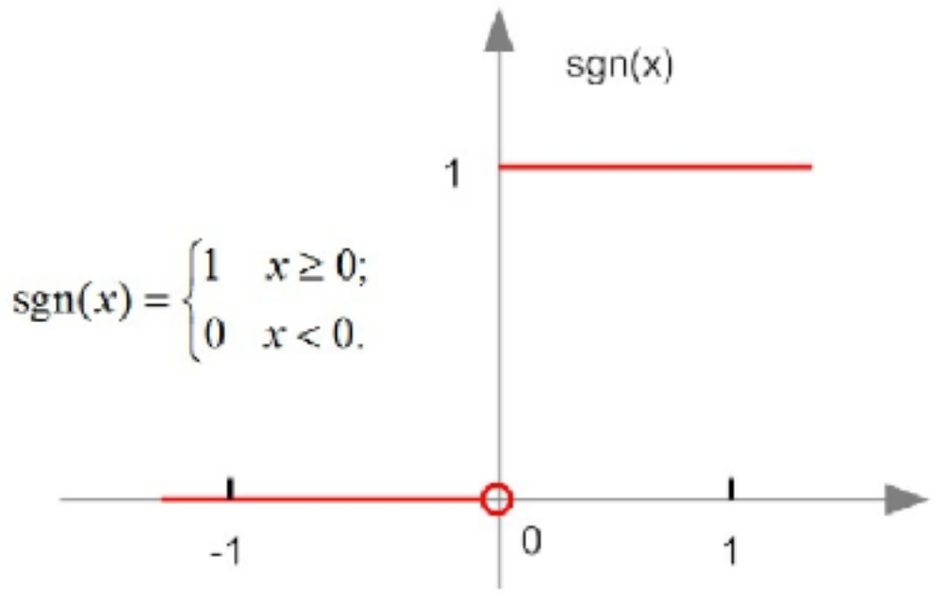
Author:WangQi
Email:wangqikaixin@gmail.com

M-P模型

1943年，McCulloch与Pitts提出了M-P模型（McCulloch－Pitts’ neuron model）。



$$h = f\left(\left(\sum_i w_i x_i\right) - \theta\right)$$

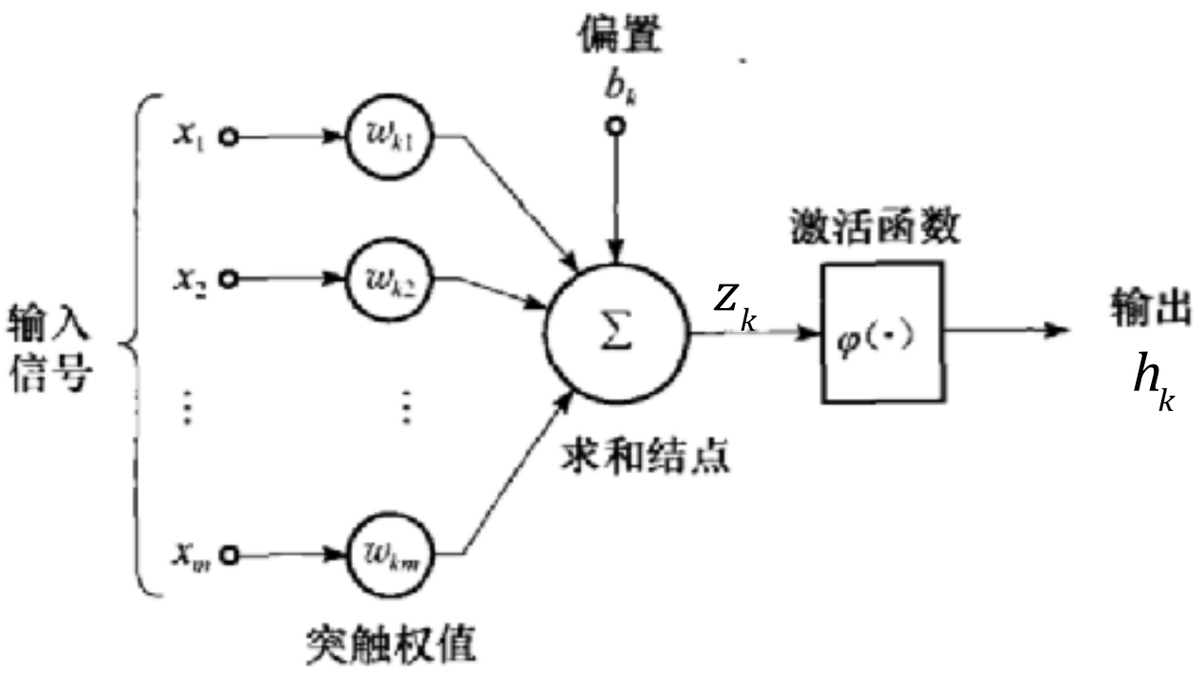


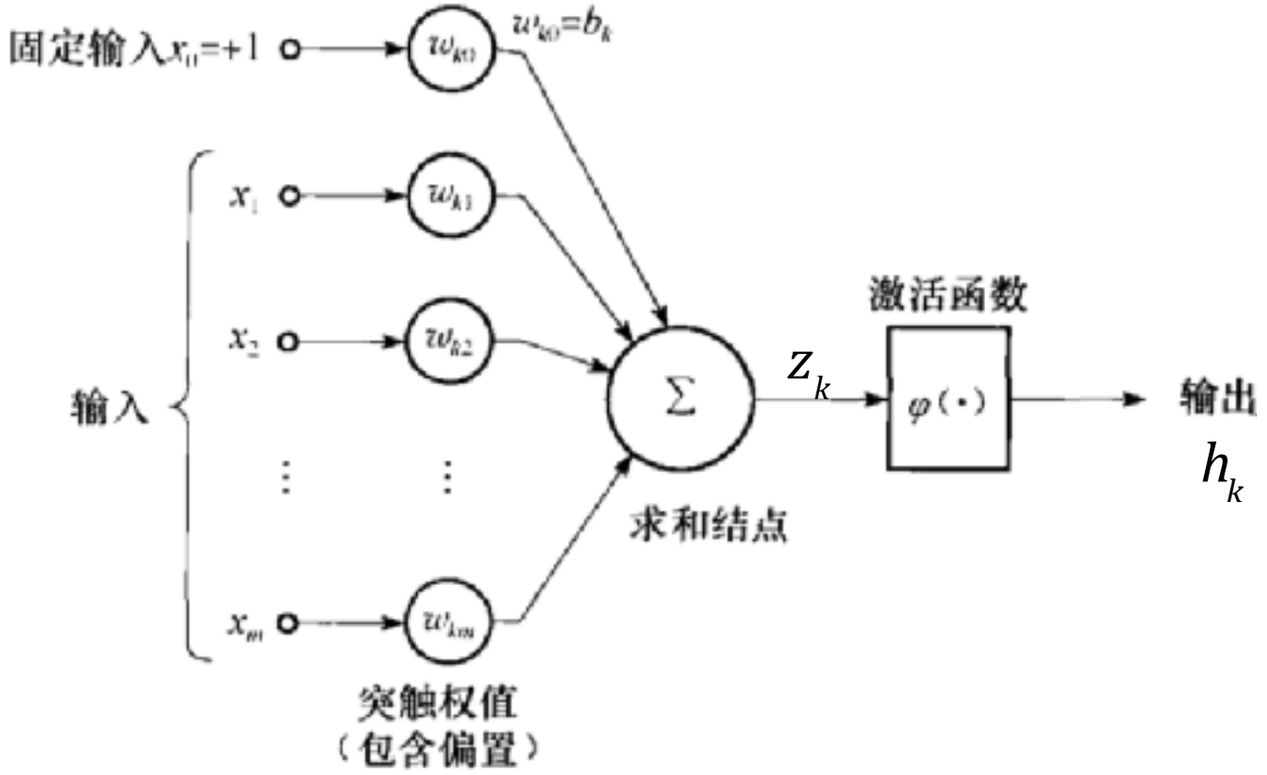
阶跃函数

Author:WangQi
Email:wangqikaixin@gmail.com

M-P模型

$$h=f\left(\left(\sum_i w_i x_i\right)-\theta\right) \quad \text{等价于} \quad h=f\left(\left(\sum_i w_i x_i\right)+b\right) \quad \text{等价于} \quad h=f\left(\sum_i w_i x_i\right)$$





Author:WangQi
Email:wangqikaixin@gmail.com

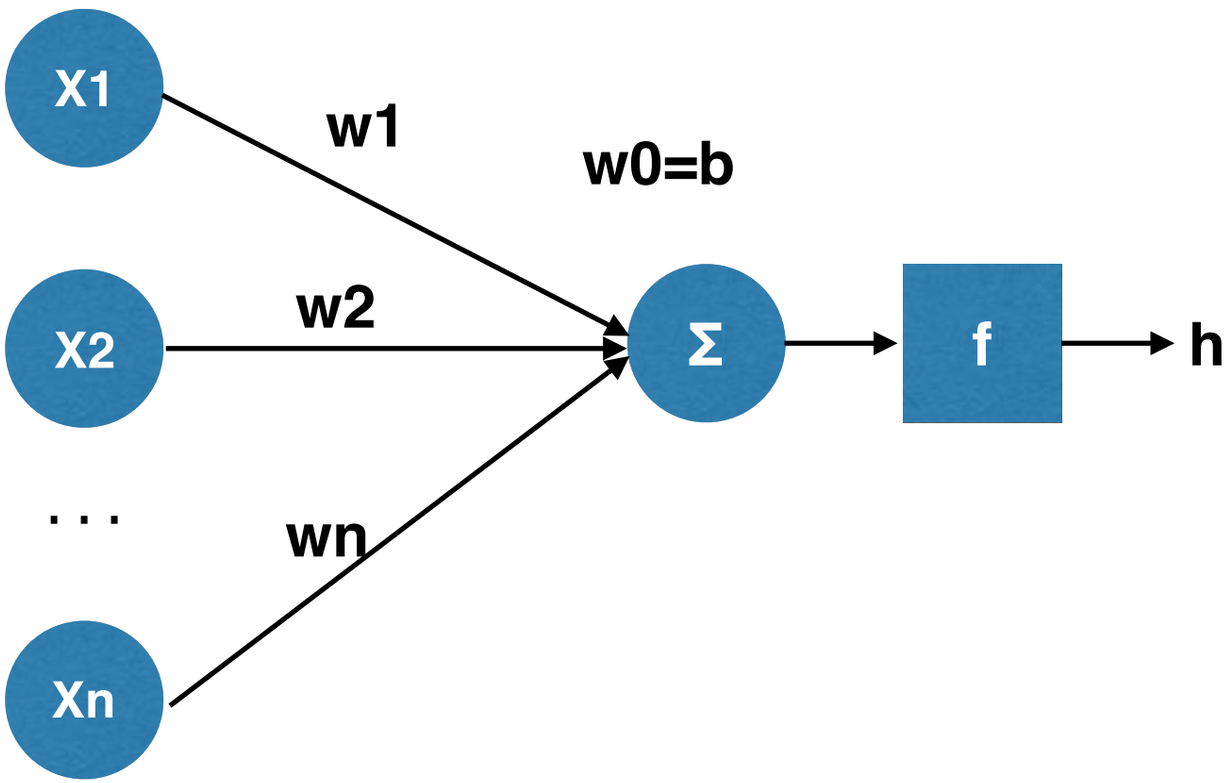
M-P模型的特点

- 模拟了单个生物神经元的电流传导过程。
- 忽略整合作用。
- 忽略神经元不应期。
- 忽略了突触时延。
- 假定突触连接强度是固定的。

Author:WangQi
Email:wangqikaixin@gmail.com

Rosenblatt感知器

1958年，Rosenblatt提出感知器作为第一个监督学习的ANN。Rosenblatt感知器建立在非线性神经元M-P模型之上。

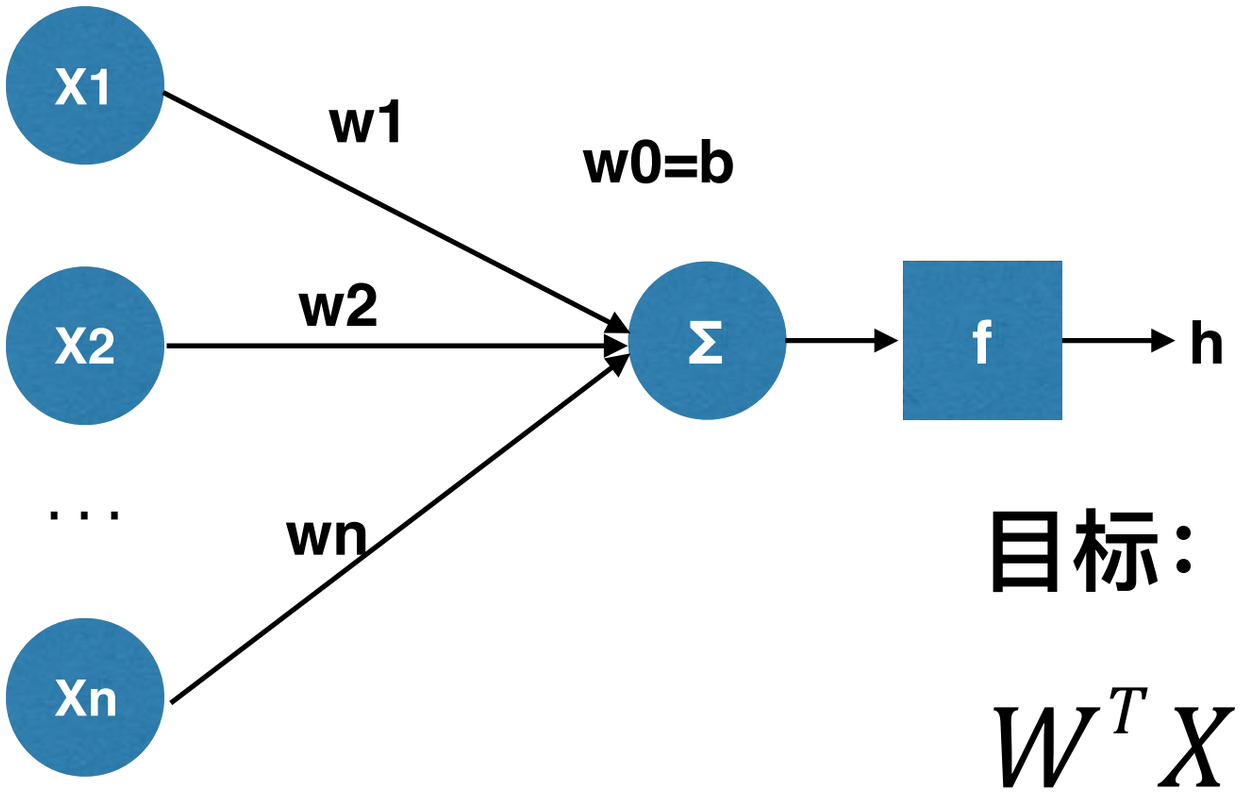


特点：

1. **基于M-P模型构建，可用于二分类。**
2. **可以对线性可分数据建模。**
3. **可以进行监督训练。**

Author:WangQi
Email:wangqikaixin@gmail.com

Rosenblatt感知器的训练



$$h = f\left(\sum_i w_i x_i\right)$$

注意：默认的所有的向量均为列向量

目标：找到一个权值向量W使得：

$W^T X > 0$ **表示一类数据**

$W^T X \leq 0$ **表示另一类数据**

Author:WangQi
Email:wangqikaixin@gmail.com

Rosenblatt感知器的训练

变量和参数：

$\mathbf{x}(n) = m+1$ 维输入向量

$= [+1, x_1(n), x_2(n), \cdots, x_m(n)]^T$

$\mathbf{w}(n) = m+1$ 维权值向量

$= [b, w_1(n), w_2(n), \cdots, w_m(n)]^T$

$b =$ 偏置

$h(n) =$ 实际响应(量化的)

$y(n) =$ 期望响应

$\eta =$ 学习率参数，一个比1小的正常数

1. 初始化。设 $\mathbf{w}(0)=\mathbf{0}$。对时间步 $n=1, 2, \cdots$ 执行下列计算。
2. 激活。在时间步 n，通过提供连续值输入向量 $\mathbf{x}(n)$ 和期望响应 $y(n)$ 来激活感知器。
3. 计算实际响应。计算感知器的实际响应：

$$h(n) = \mathrm{sgn}[\mathbf{w}^T(n)\mathbf{x}(n)]$$

这里 $\mathrm{sgn}(\cdot)$ 是符号函数。

4. 权值向量的自适应。更新感知器的权值向量：

$$\mathbf{w}(n+1) = \mathbf{w}(n) + \eta[y(n) - h(n)]\mathbf{x}(n)$$

这里

$$y(n) = \begin{cases} +1 & \text{若 } \mathbf{x}(n) \text{ 属于类 } \mathscr{C}_1 \\ -1 & \text{若 } \mathbf{x}(n) \text{ 属于类 } \mathscr{C}_2 \end{cases}$$

5. 继续。时间步 n 增加1，返回第2步。

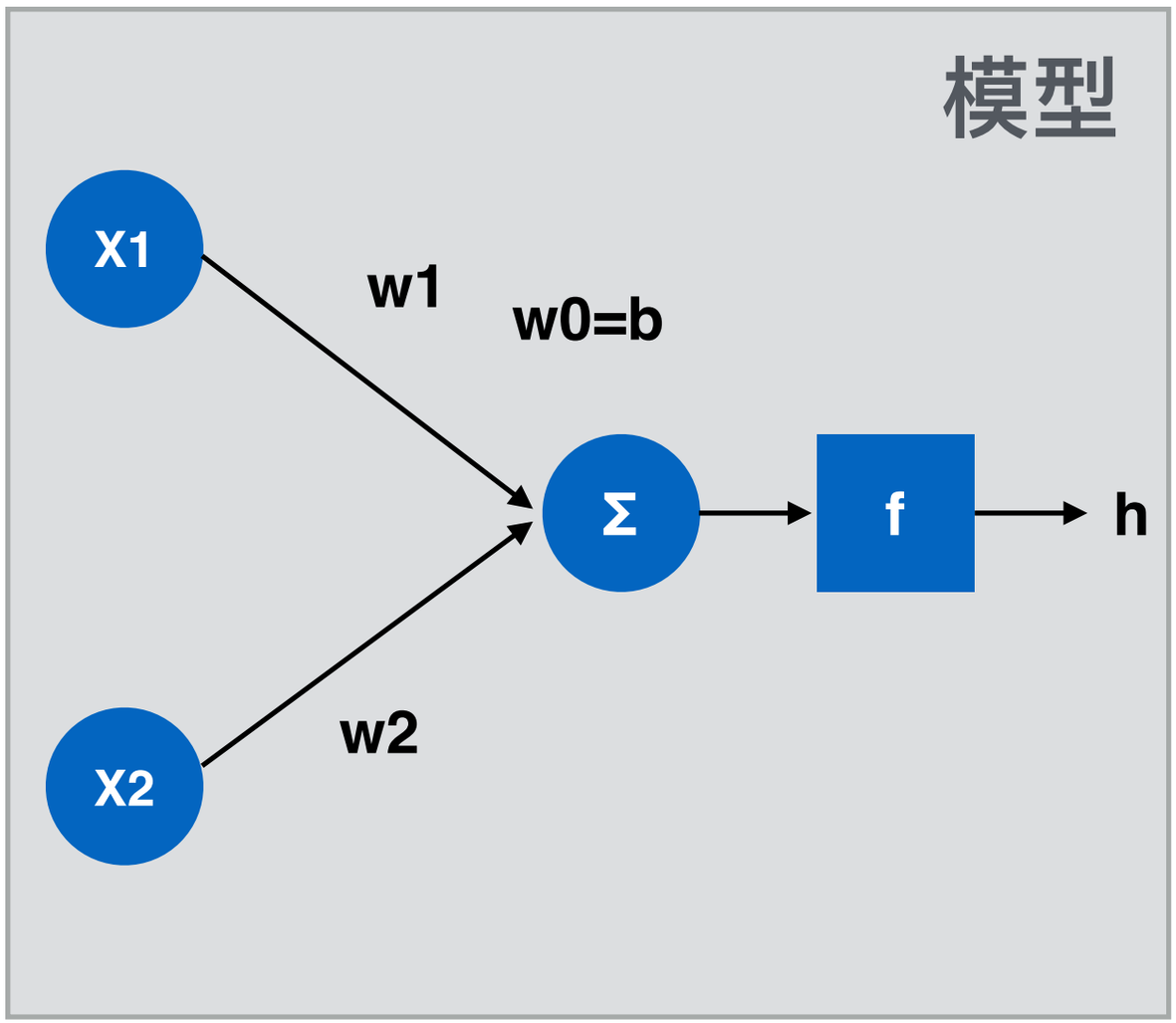
Author:WangQi
Email:wangqikaixin@gmail.com

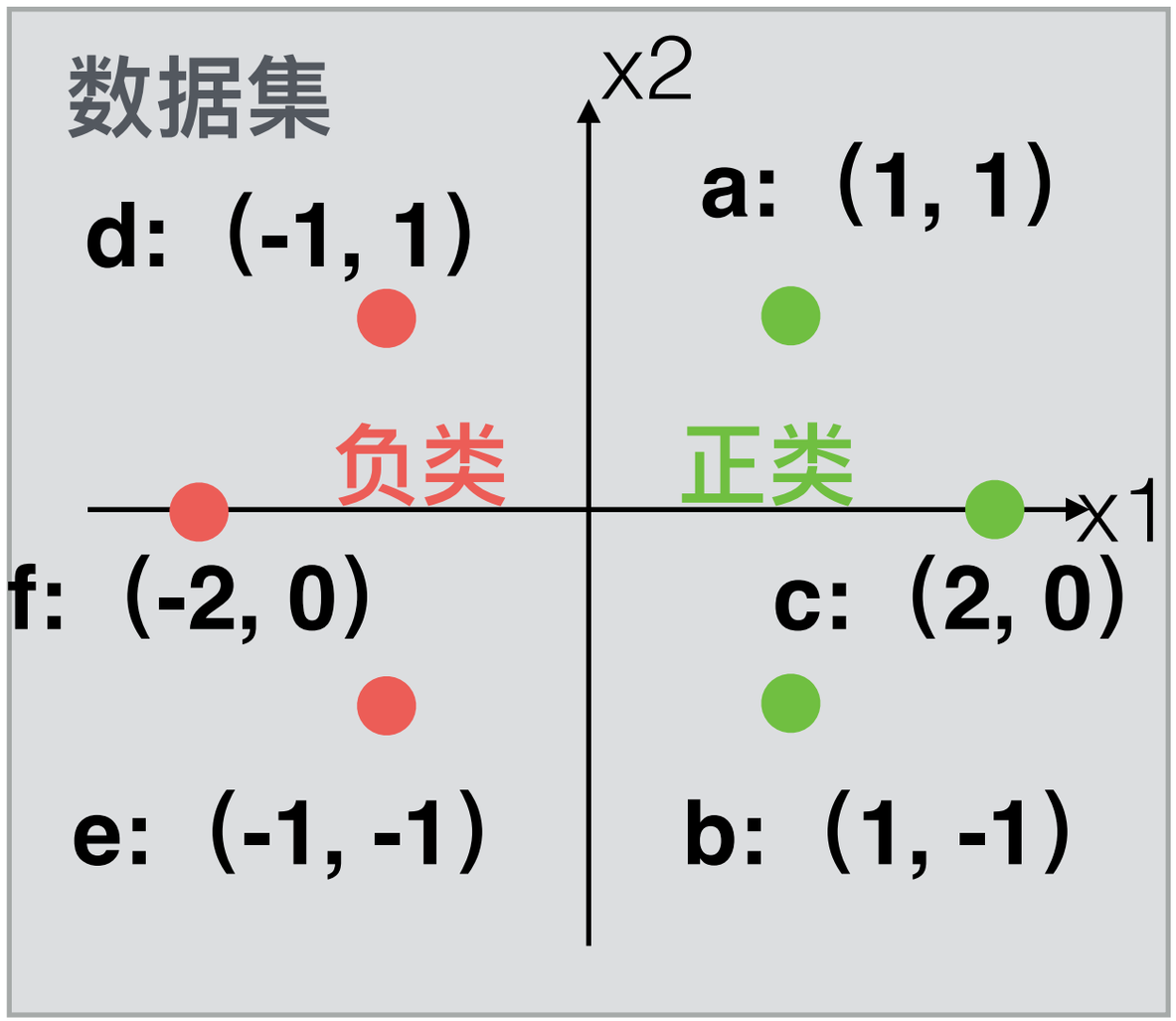
感知器的训练实例

以包含两个输入的神经元为例：

a、b、d、e训练集

c、f测试集





Author:WangQi
Email:wangqikaixin@gmail.com

感知器的训练实例

1. 初始化。

模型： $h=\mathrm{sgn}\left(z\right)=\mathrm{sgn}\left(w_0+x_1w_1+x_2w_2\right)$

参数初始化： $w_0=w_1=w_2=0$

学习率： $lr=1$

$$h=\mathrm{sgn}\left(0+0*x_1+0*x_2\right)$$

Author:WangQi
Email:wangqikaixin@gmail.com

感知器的训练实例

2. 激活。

随机选择一个样本，使用模型处理，得到输出值。

此处选择点“a”，有：

$$
\begin{aligned}
z &= w_0 + x_1 w_1 + x_2 w_2 \\
&= 0 + 1*0 + 1*0 \\
&= 0
\end{aligned}
$$

Author:WangQi
Email:wangqikaixin@gmail.com

感知器的训练实例

3. 计算响应（输出）。

输出： $h=\mathrm{sgn}\left(z\right)=0$

$$\mathrm{sgn}(x)=\begin{cases}1 & x>0;\\ 0 & x\le 0.\end{cases}$$

Author:WangQi
Email:wangqikaixin@gmail.com

感知器的训练实例

4. 更新权值。

标记： **正类** $y=1$

输出： $h=\mathrm{sgn}\left(z\right)=0$

更新： $w=w+lr\left(y-h\right)x$ → $w_0=0+1=1$, $w_1=0+1=1$, $w_2=0+1=1$

$$h=\mathrm{sgn}\left(1+1*x_1+1*x_2\right)$$

Author:WangQi
Email:wangqikaixin@gmail.com

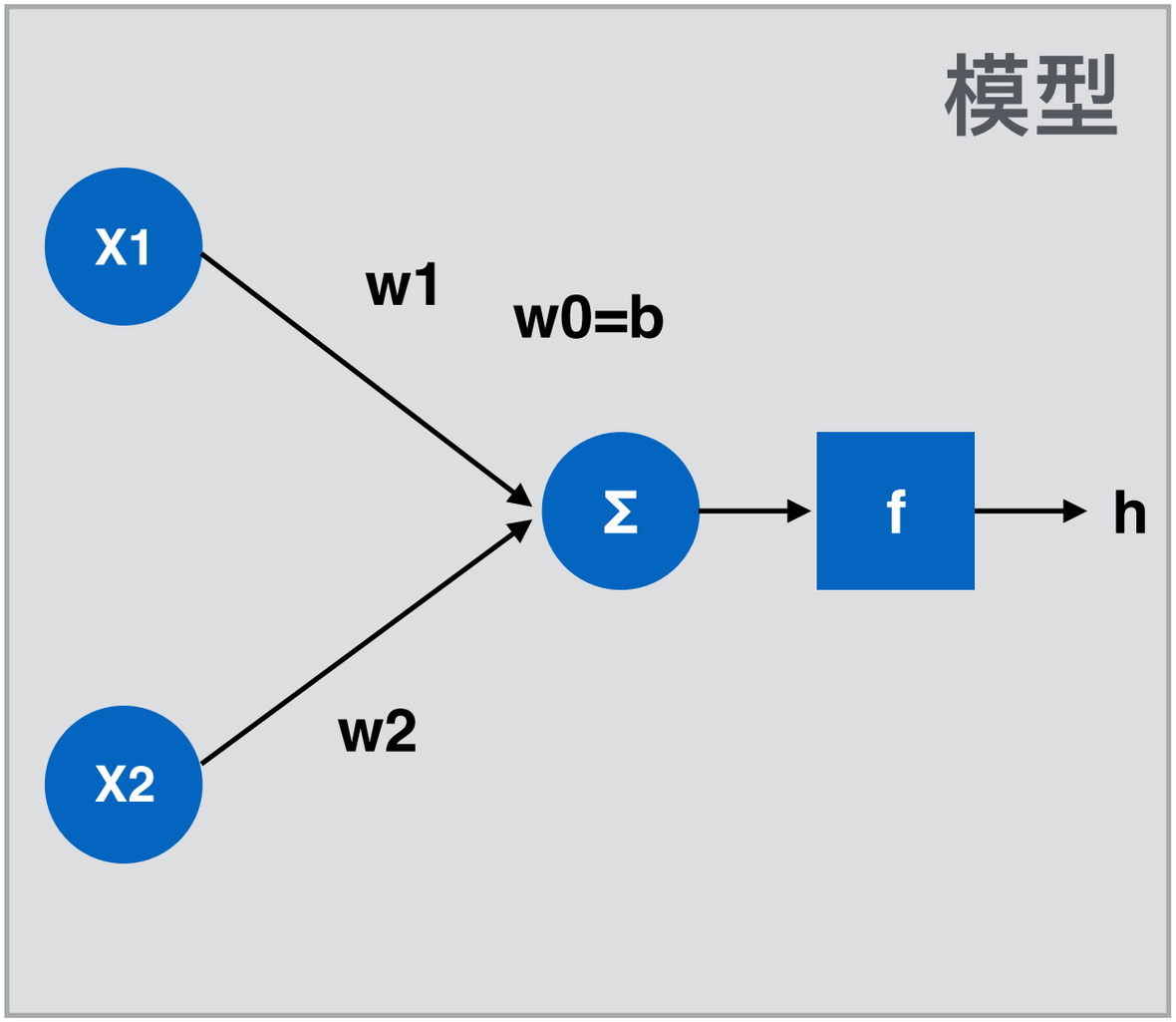
感知器的训练实例

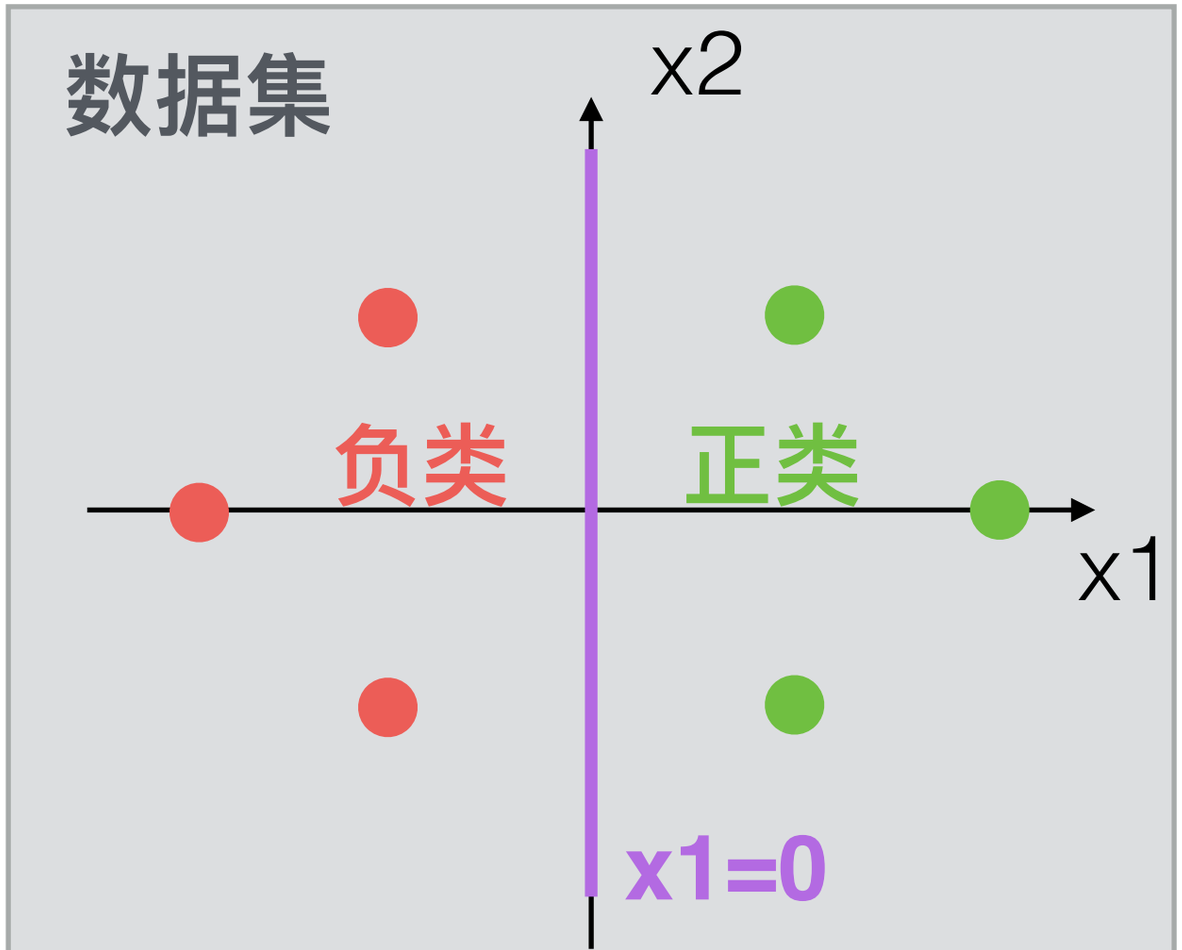
5. 重复2-4步骤。

分别带入b、d、e点进行训练。也可以重复使用训练集，直到算法收敛。

Author:WangQi
Email:wangqikaixin@gmail.com

感知器的训练实例





分界线不止一条

Author:WangQi
Email:wangqikaixin@gmail.com

Rosenblatt感知器

感知器是用于线性可分模式分类的最简单的神经网络模型。它由一个可以连接权重的神经元组成。Rosenblatt证明了“**感知器收敛定理**”，即训练集线性可分时，感知器算法收敛。通过扩展感知器的输出层神经元数量其可以完成多分类任务。

Author:WangQi
Email:wangqikaixin@gmail.com

实现：感知机

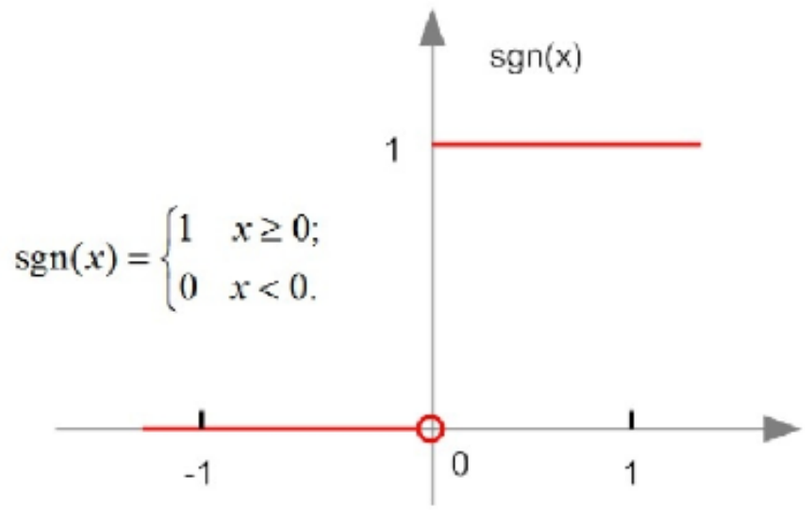
Author:WangQi
Email:wangqikaixin@gmail.com

2. 激活函数的进化

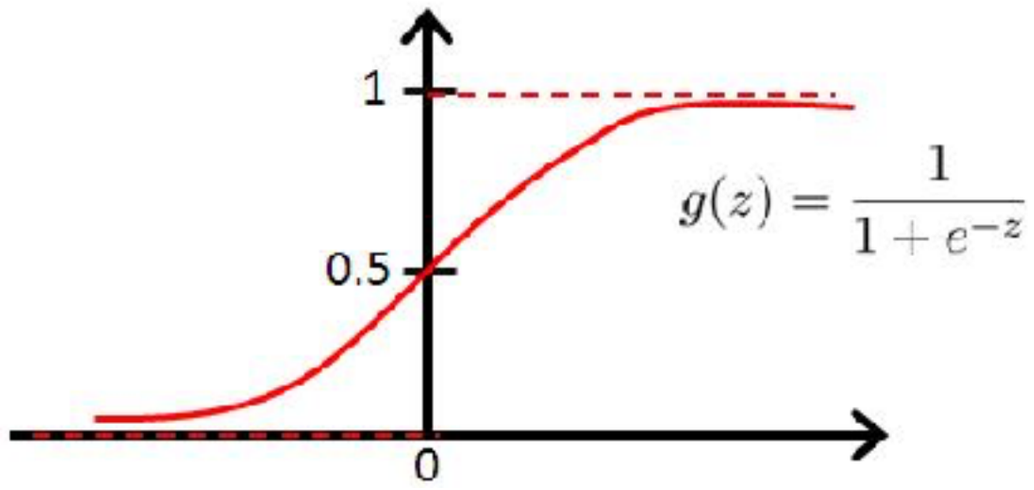
Author:WangQi
Email:wangqikaixin@gmail.com

激活函数分类

- 硬限幅器



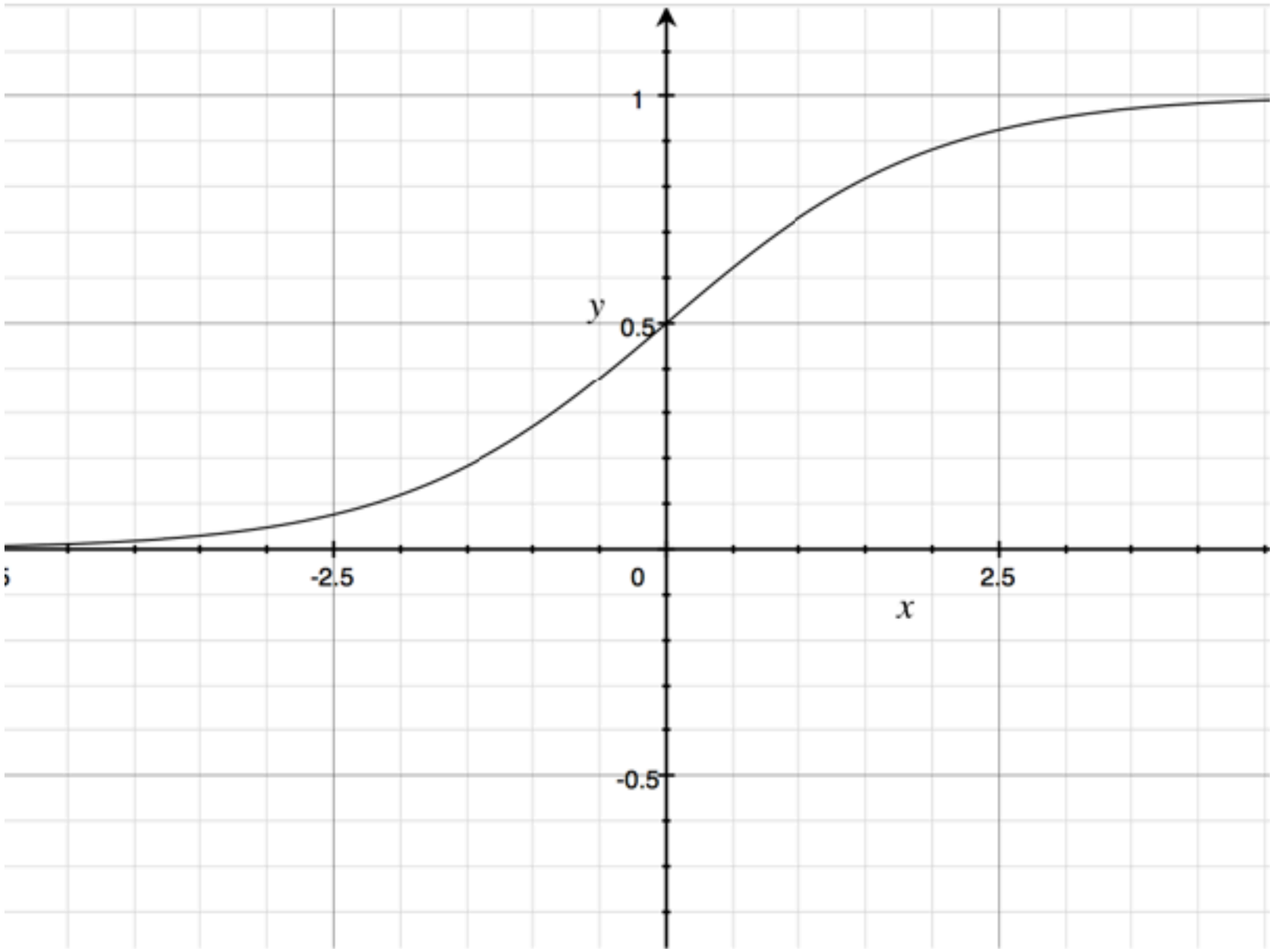
- 软限幅器



Author:WangQi
Email:wangqikaixin@gmail.com

logistic激活函数

$$y=\frac{1}{1+e^{-x}}$$



优点：
定义域为R，值域为(0,1)。
可以用来做二分类。
类似于生物神经元。

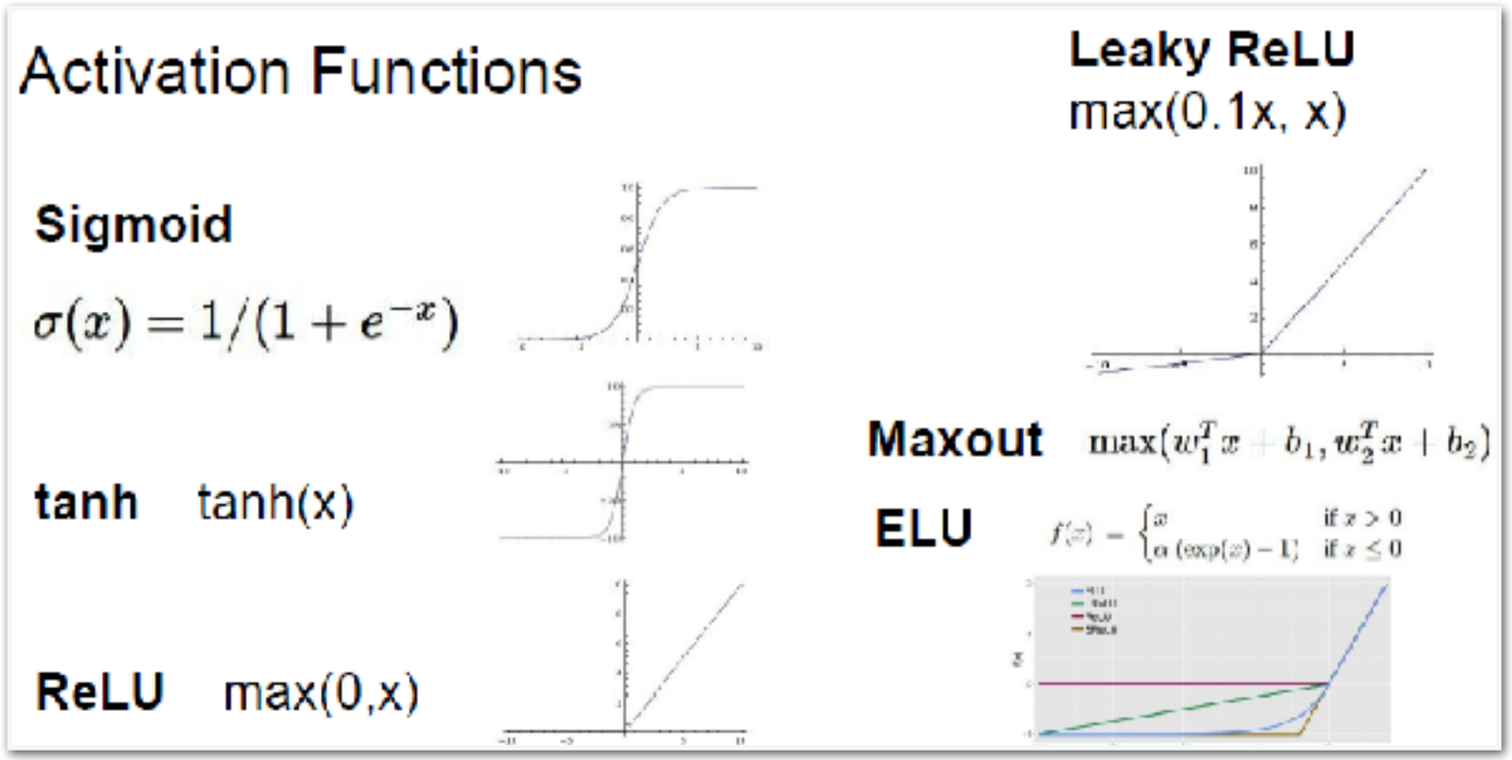
缺点：
计算量比较大。
求导涉及除法。
会导致梯度消失。

logistic神经元可以模拟逻辑回归。

Author:WangQi
Email:wangqikaixin@gmail.com

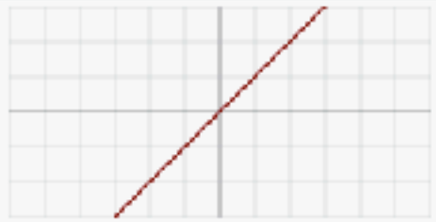
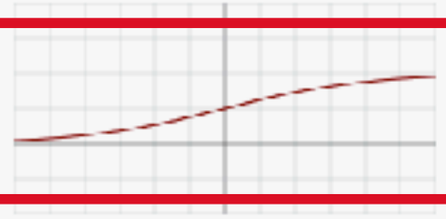
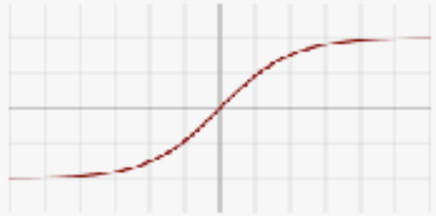
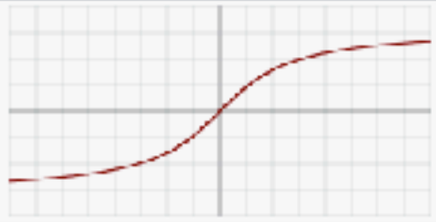
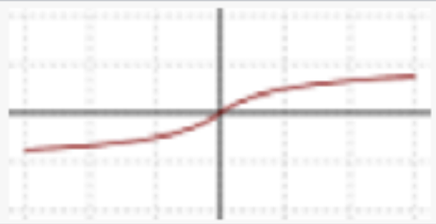
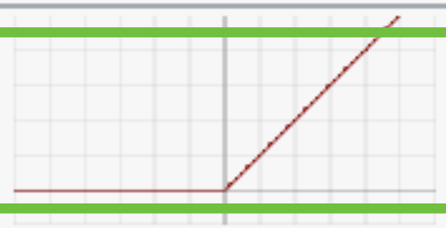
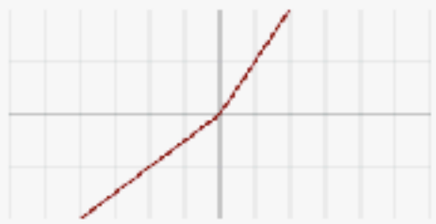
常见神经元的激活函数

- 阶梯函数
- 线性函数
- 饱和线性函数
- 对数S形函数
- 强制非负校正函数
- ……



优秀的激活函数可以使神经网络更好的工作。设置线性激活函数或者不设置激活函数，会导致输出永远是输入的线性组合。

Author:WangQi
Email:wangqikaixin@gmail.com

Identity		$f(x) = x$
Binary step		$f(x) = \begin{cases} 0 & \text{for} \quad x < 0 \\ 1 & \text{for} \quad x \geq 0 \end{cases}$
Logistic (a.k.a. Soft step)		$f(x) = \frac{1}{1 + e^{-x}}$
TanH		$f(x) = \tanh(x) = \frac{2}{1 + e^{-2x}} - 1$
ArcTan		$f(x) = \tan^{-1}(x)$
Softsign [7][8]		$f(x) = \frac{x}{1 + \lvert x \rvert}$
Rectified linear unit (ReLU)[9]		$f(x) = \begin{cases} 0 & \text{for} \quad x < 0 \\ x & \text{for} \quad x \geq 0 \end{cases}$
Leaky rectified linear unit (Leaky ReLU)[10]		$f(x) = \begin{cases} 0.01x & \text{for} \quad x < 0 \\ x & \text{for} \quad x \geq 0 \end{cases}$

早期感知机激活函数

早期常用激活函数

注意：一个神经网络中不一定只有一种激活函数。

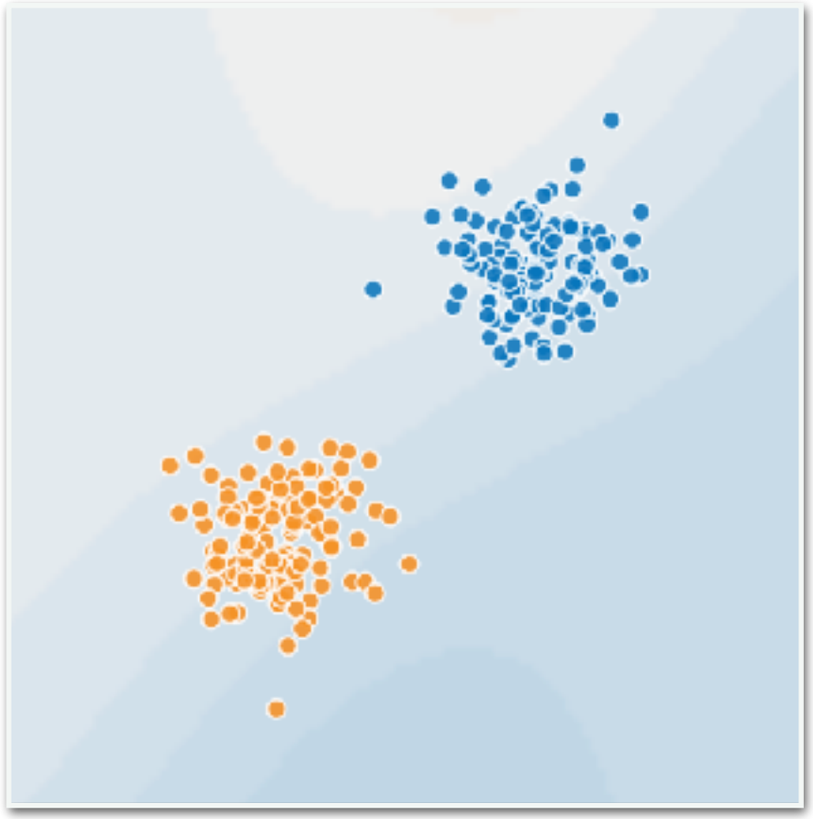
现在常用激活函数

Author:WangQi
Email:wangqikaixin@gmail.com

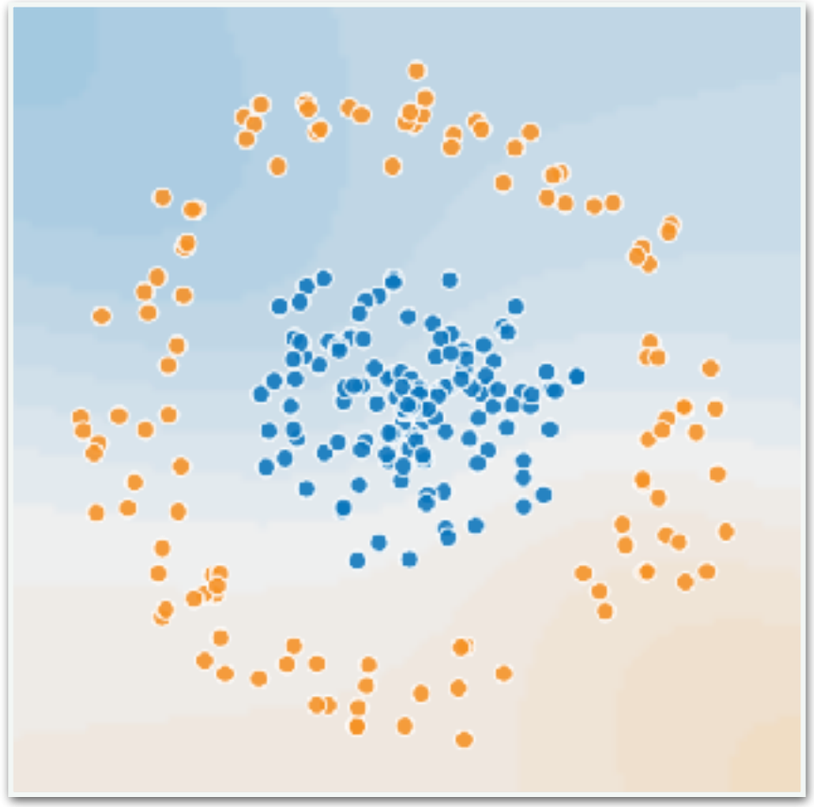
3. 线性不可分问题

Author:WangQi
Email:wangqikaixin@gmail.com

线性不可分



线性可分



线性不可分

Author:WangQi
Email:wangqikaixin@gmail.com

请问：以下数据集是否线性可分?

a. 手写数字数据集MNIST

b. 近五年中国房屋价格与面积的数据集

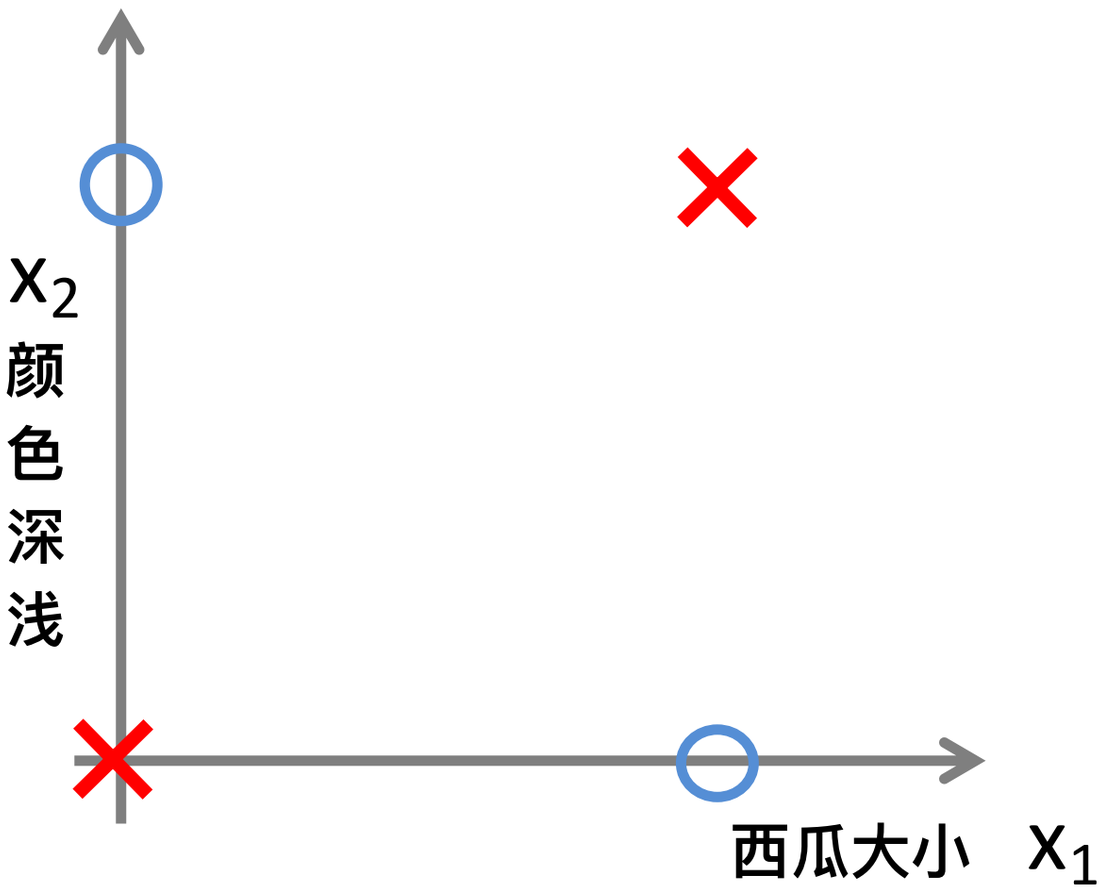
c. 用于做人脸验证的数据集YouTube Video Faces

大部分的数据是线性不可分的，线性不可分的数据需要非线性模型进行拟合。

Author:WangQi
Email:wangqikaixin@gmail.com

线性不可分与异或（XOR）

西瓜好坏与西瓜大小、西瓜颜色深浅的关系可由四个样本表示如下：



圈代表好瓜，叉代表坏瓜

用x1=1表示x1的输入为真

用x1=0表示x1的输入为假

用x2=1表示x2的输入为真

用x2=0表示x2的输入为假

XOR输入不同输出1

XOR即异或门，也是一个半加器。在图像上表现出线性不可分。

Author:WangQi
Email:wangqikaixin@gmail.com

4. 构造模拟逻辑运算的神经元

Author:WangQi
Email:wangqikaixin@gmail.com

前提

1. 使用单个神经元。

2. 使用logistic神经元。

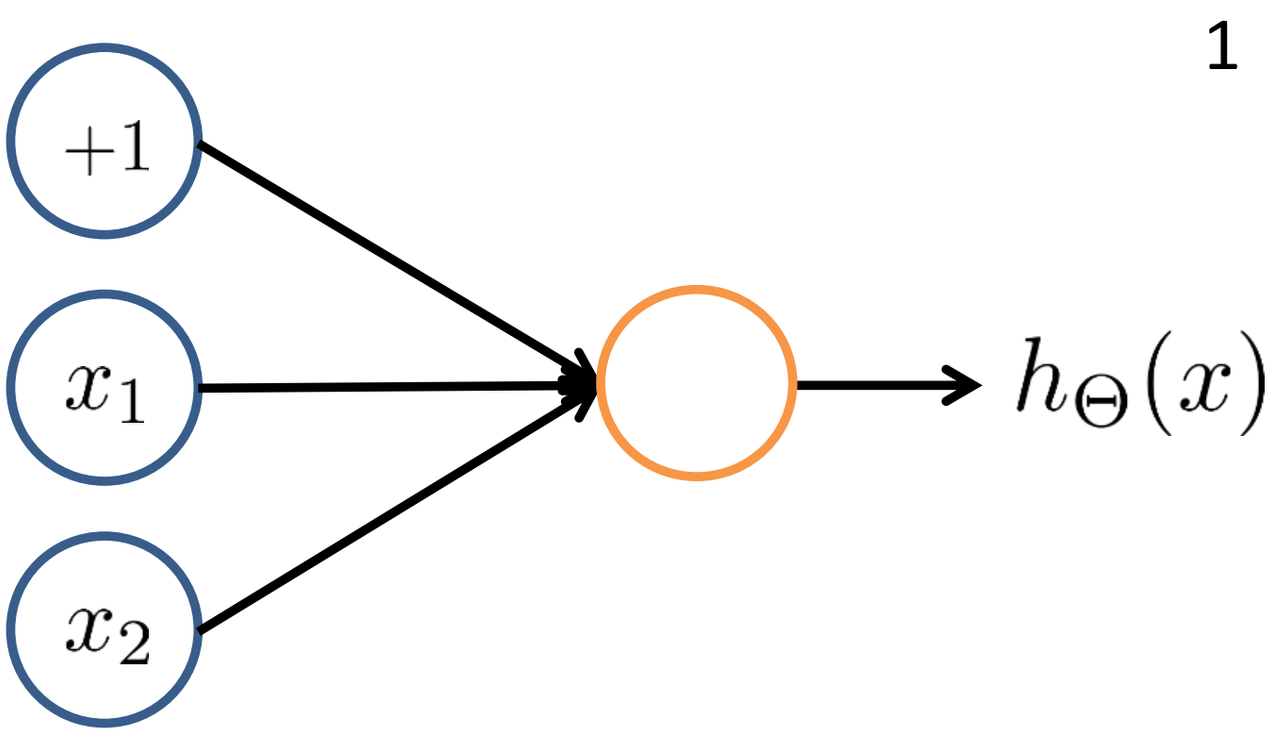
Author:WangQi
Email:wangqikaixin@gmail.com

构造AND逻辑运算单元

$x_1, x_2 \in \{0, 1\}$

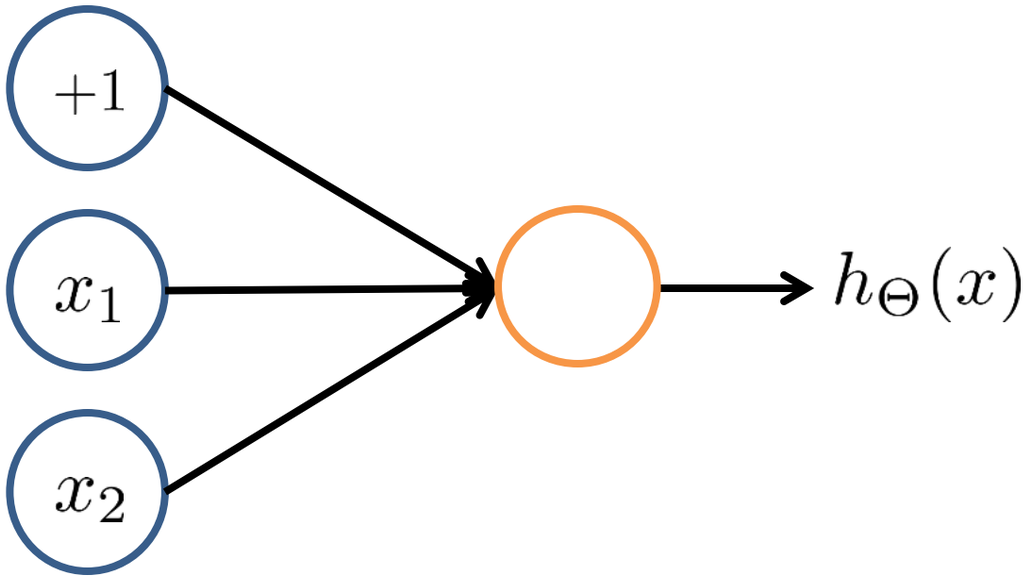
$y = x_1 \text{ AND } x_2$

x_1	x_2	$h_\Theta(x)$
0	0	
0	1	
1	0	
1	1	



Author:WangQi
Email:wangqikaixin@gmail.com

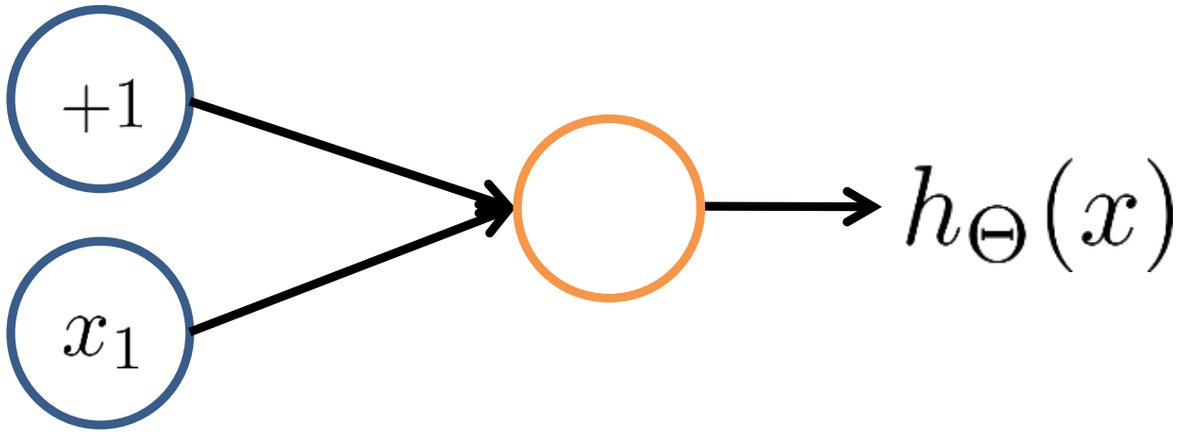
构造逻辑运算单元OR



x_1	x_2	$h_\Theta(x)$
0	0	
0	1	
1	0	
1	1	

Author:WangQi
Email:wangqikaixin@gmail.com

构造逻辑运算单元NOT



x_1	$h_\Theta(x)$
0	
1	

Author:WangQi
Email:wangqikaixin@gmail.com

思考：使用感知器能否构造基础逻辑运算？

Author:WangQi
Email:wangqikaixin@gmail.com

5. 构造神经网络解决XOR问题

Author:WangQi
Email:wangqikaixin@gmail.com

使用基础逻辑运算单元可以构建XOR单元。

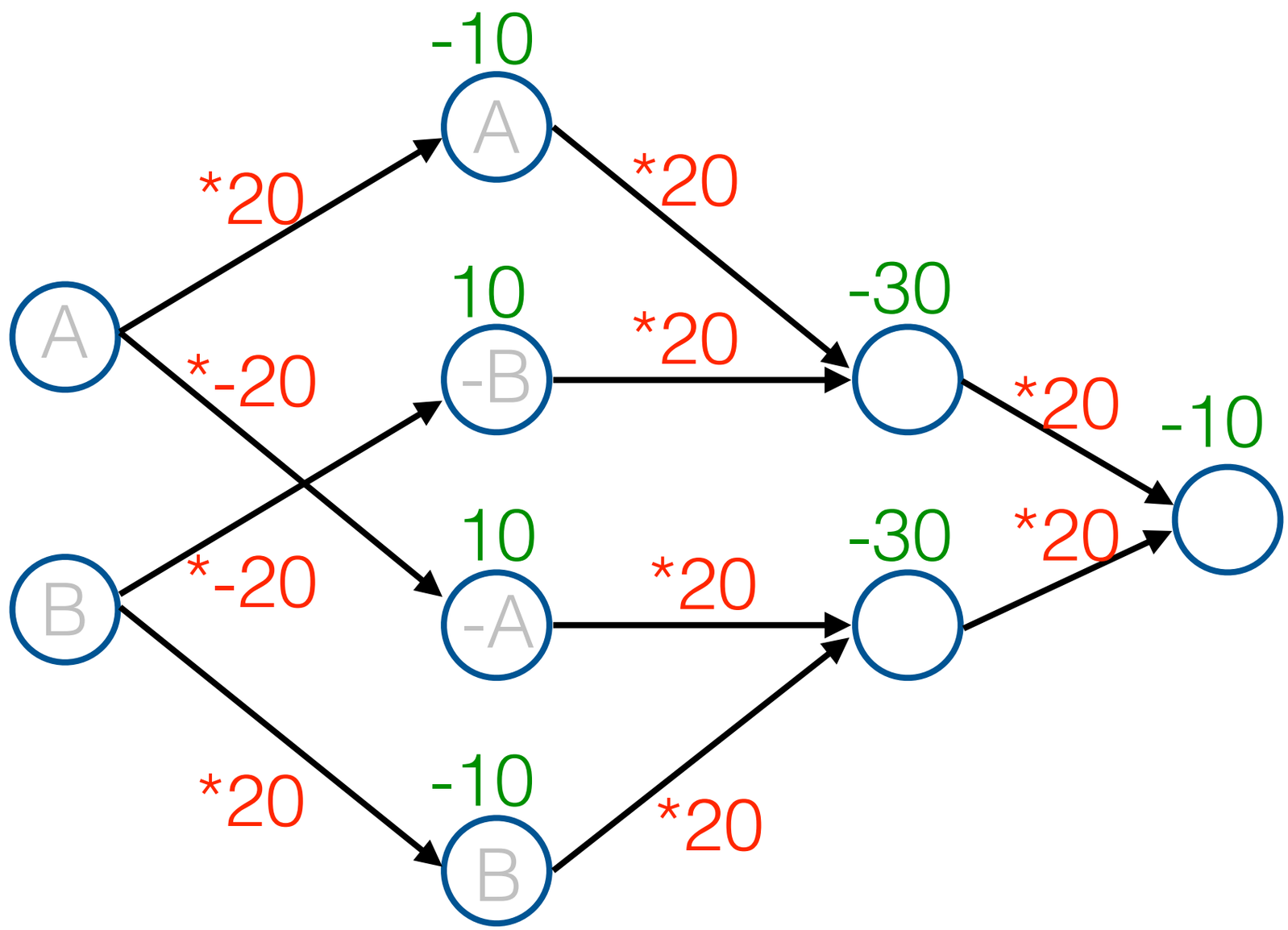
A xor B = (A and (not B)) or ((not A) and B)

x_1	x_2	$h_\Theta(x)$
0	0	
0	1	
1	0	
1	1	

输入不同输出1

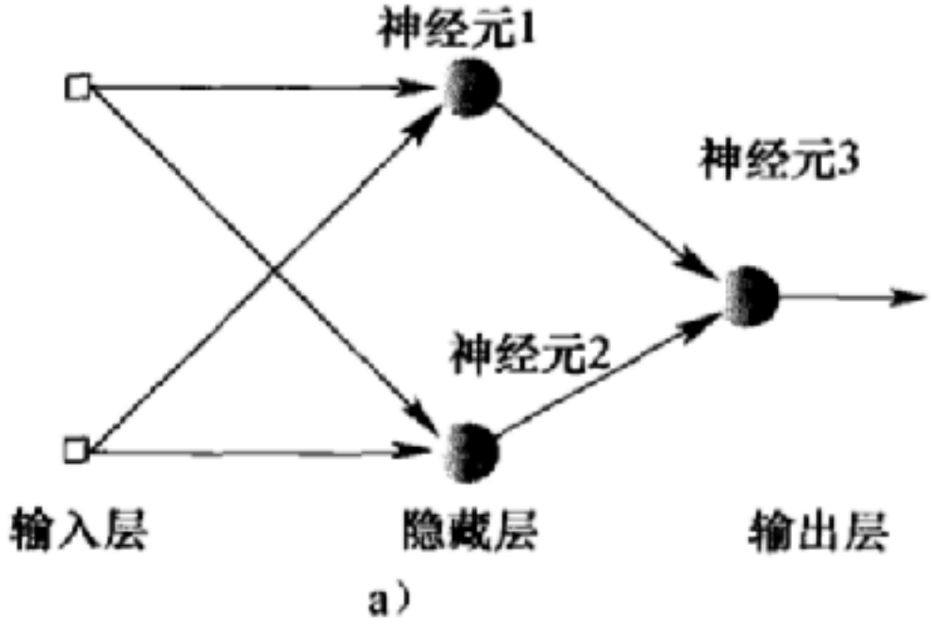
Author:WangQi
Email:wangqikaixin@gmail.com

利用逻辑神经元构建XOR神经网络

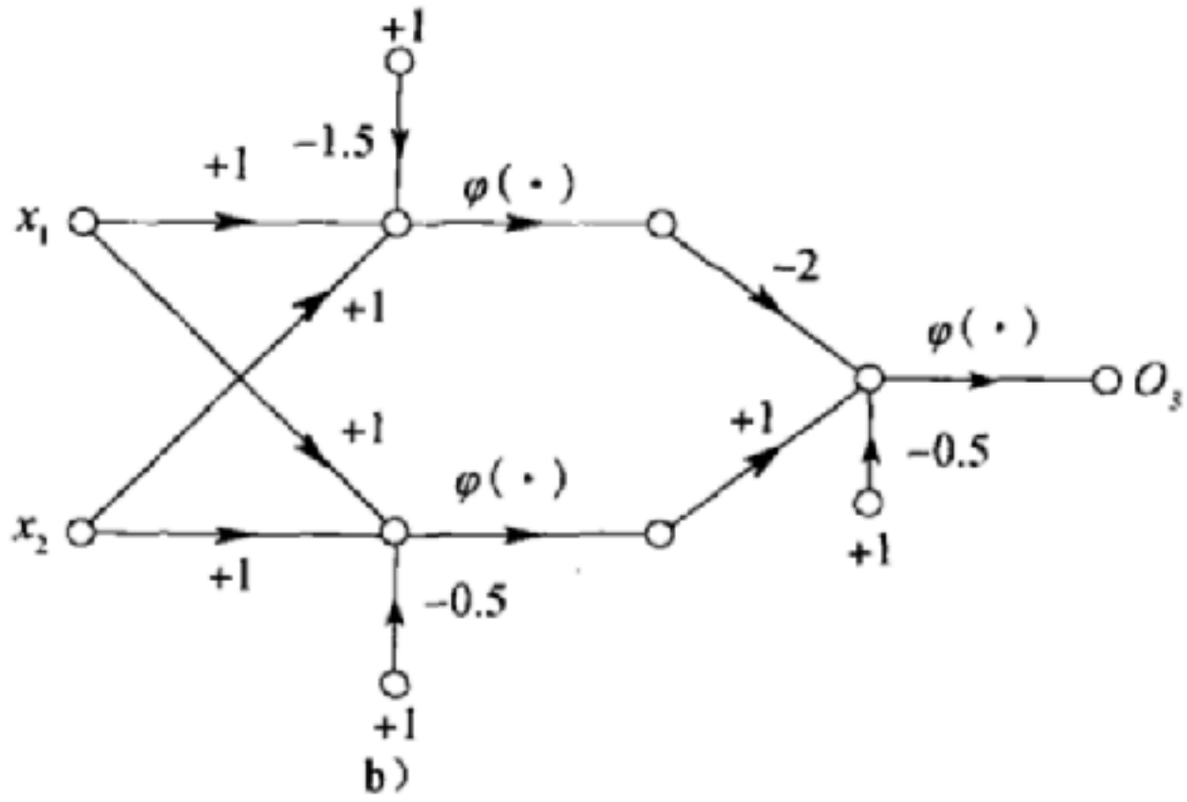


Author:WangQi
Email:wangqikaixin@gmail.com

最简单的XOR神经网络



结构图

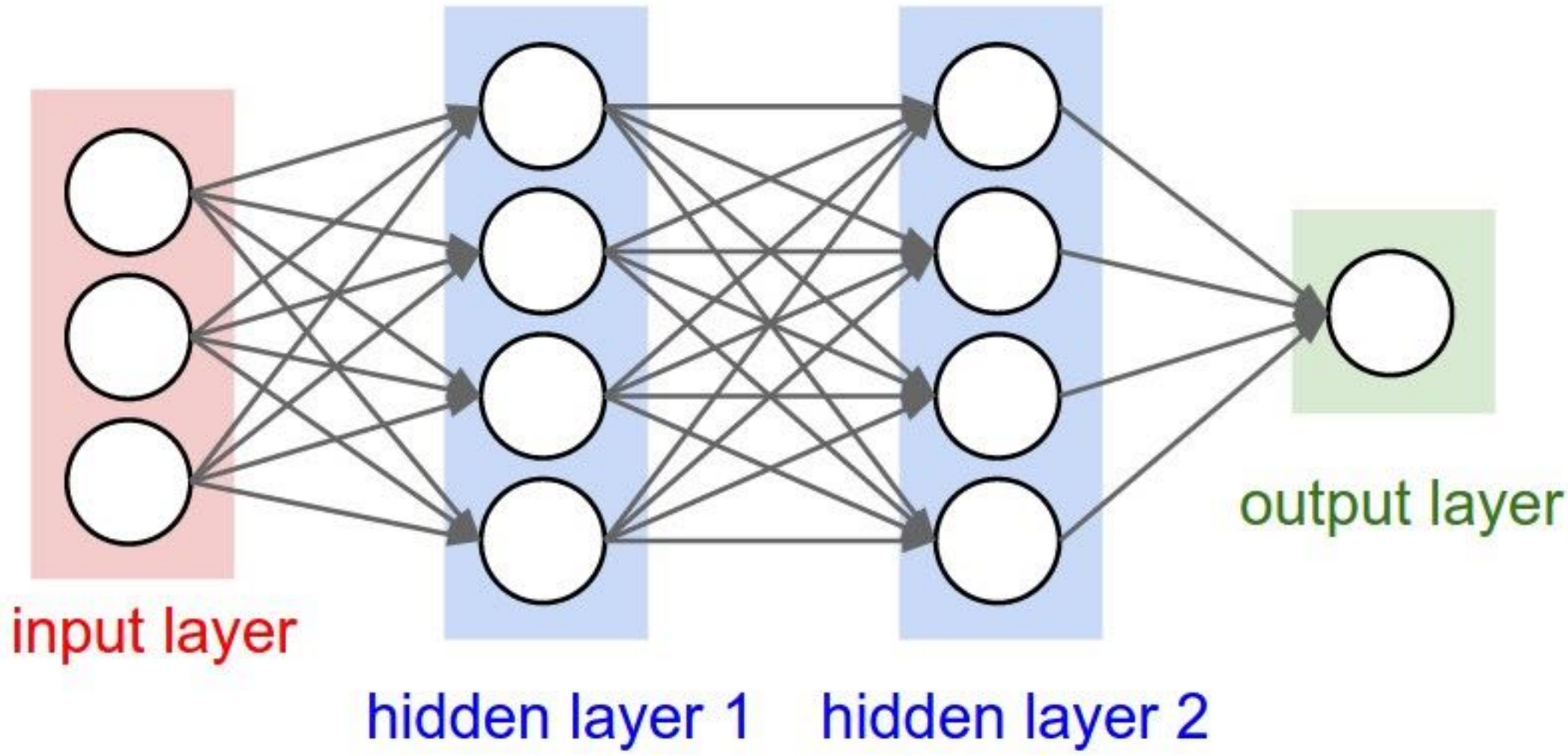


信号流图

注意：此处使用的是感知器神经元。

Author:WangQi
Email:wangqikaixin@gmail.com

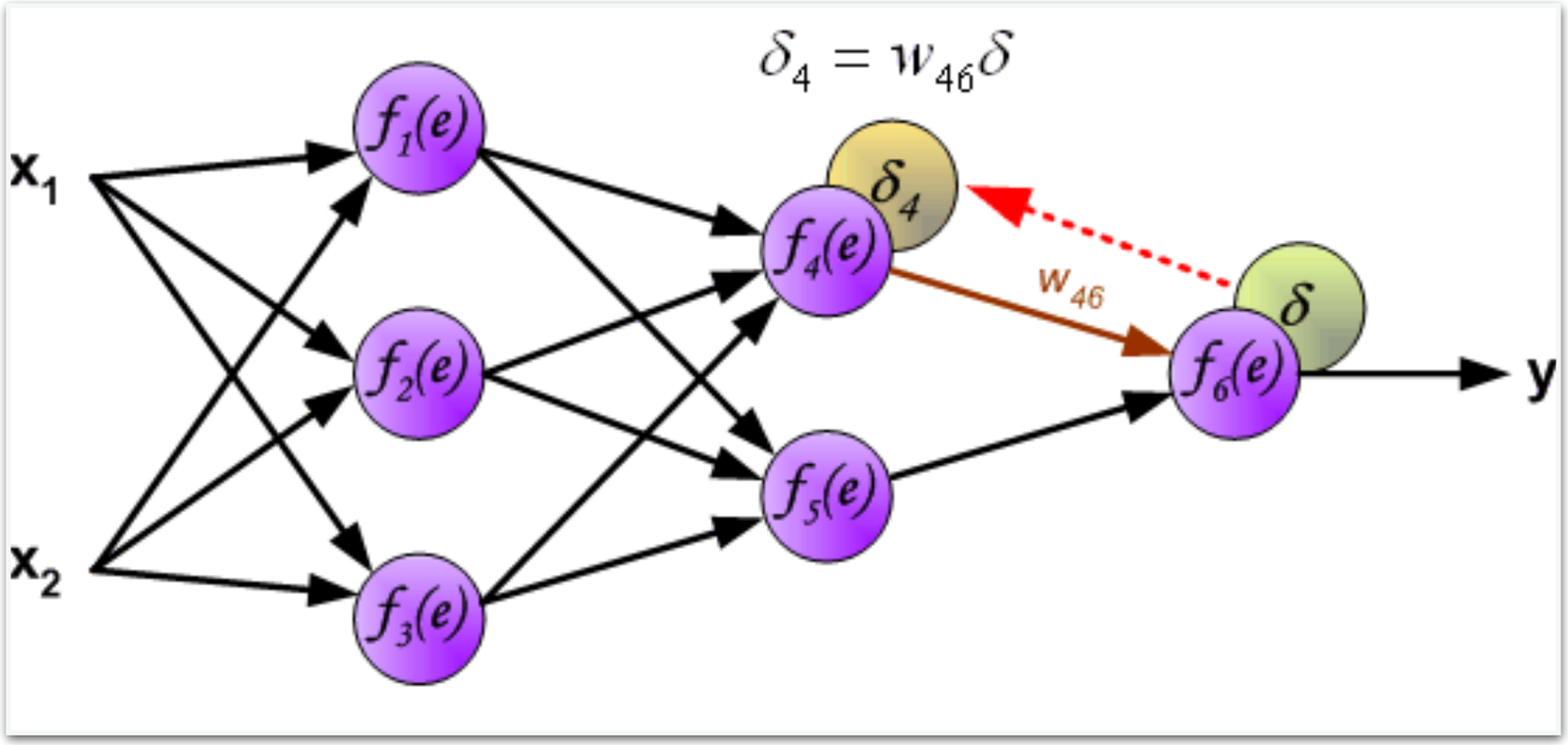
构造神经网络



图中每一个圆圈代表一个神经元(unit)。多个神经（一竖行）元组成一个层（layer）。层与层的神经元之间全连接。第一层我们称之为输入层，最后一层为输出层，其余为隐藏层。神经网络是一个有向无环图。

Author:WangQi
Email:wangqikaixin@gmail.com

如何参数设置



实际中，我们并不手动设计神经网络的结构和参数。而是使用反向传播算法自动求参数。

Author:WangQi
Email:wangqikaixin@gmail.com

playground演示

Author:WangQi
Email:wangqikaixin@gmail.com

6. Python实现神经网络

Author:WangQi
Email:wangqikaixin@gmail.com

小节

- 感知器的形态与性质。
- 感知器的训练过程。
- 常用激活函数logistic函数、双曲正切函数、修正线性单元、softmax函数。
- Sigmoid函数的值域决定了其可以做二分类。
- 线性不可分是常见的情形，需要使用非线性模型拟合。
- 神经网络构造逻辑单元并解决XOR问题。

Author:WangQi
Email:wangqikaixin@gmail.com

THANKS

Author:WangQi
Email:wangqikaixin@gmail.com