人工神经网络-2

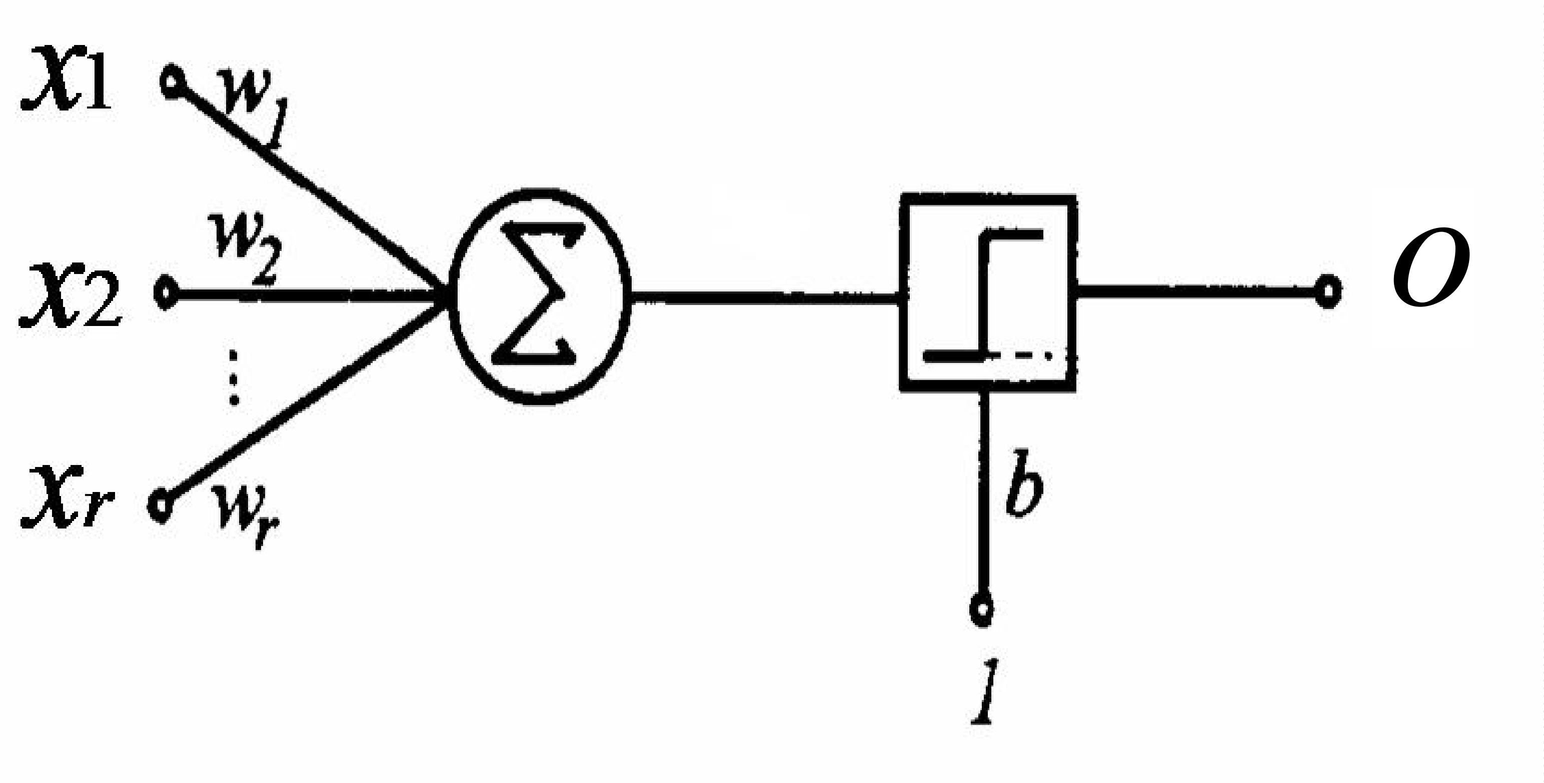
主要内容

- 1. 感知器的模型
- 2. 双输入感知器的设计
- 3. 多输入感知器的设计
- 4. 感知器的学习规则
- 5. Delta法则简介

前言

- 感知器是由美国计算机科学家罗森布拉特（F.Roseblatt）于1957年提出的
- 感知器特别适用于简单的模式分类问题，也可用于基于模式分类的学习控制
- 着重讨论简单感知器（单神经元）的设计

1. 感知器的模型



1. 感知器的模型

$$O=\begin{cases}1, \text{当} y=\sum_{i=1}^{r}\omega_i x_i+b>0 \\ 0, \text{当} y=\sum_{i=1}^{r}\omega_i x_i+b\le 0\end{cases}$$

其中， O是输出， b是常量输入或外界控制信号， x_i和ω_i分别是第 i个输入及其权重

1. 感知器的模型

- 感知器的设计，就是确定各个输入的权重的过程
  - 利用训练数据训练一个感知器,实际上就是让感知器通过学习,自己确定权重的过程

2. 双输入感知器的设计

- 利用感知器实现布尔函数AND

p	q	$p \wedge q$
0	0	0
0	1	0
1	0	0
1	1	1

分析

- 利用感知器实现AND，就是确定一组权值
  - 使得它在前三组的输入下，输出为0，即感知器的函数值y小于等于0
  - 在最后一组的输入下，输出为1，即感知器的函数值y大于0

AND运算的实现

实现AND的问题即可转化为设定感知器的参数ω_1,ω_2和b的值，使得下式成立：

$$\begin{cases} y=\omega_1 x_1+\omega_2 x_2+b>0, 当x_1=x_2=1 \\ y=\omega_1 x_1+\omega_2 x_2+b\le 0, 其它 \end{cases}$$

- 尝试找一组权值?

AND运算的实现

实现*AND*的问题即可转化为设定感知器的参数ω_1,ω_2和b的值，使得下式成立：

$$\begin{cases} y=\omega_1x_1+\omega_2x_2+b>0, \text{当} x_1=x_2=1 \\ y=\omega_1x_1+\omega_2x_2+b\le 0, \text{其它} \end{cases}$$

例如：设定

$$\begin{cases} \omega_1=0.5 \\ \omega_2=0.5 \\ b=-0.8 \end{cases}$$

AND运算的实现

ω_1, ω_2和b的通用求法：

考虑$y = \omega_1 x_1 + \omega_2 x_2 + b = 0$，表示输入变量构成的$x_1 - x_2$坐标系中的一条直线；

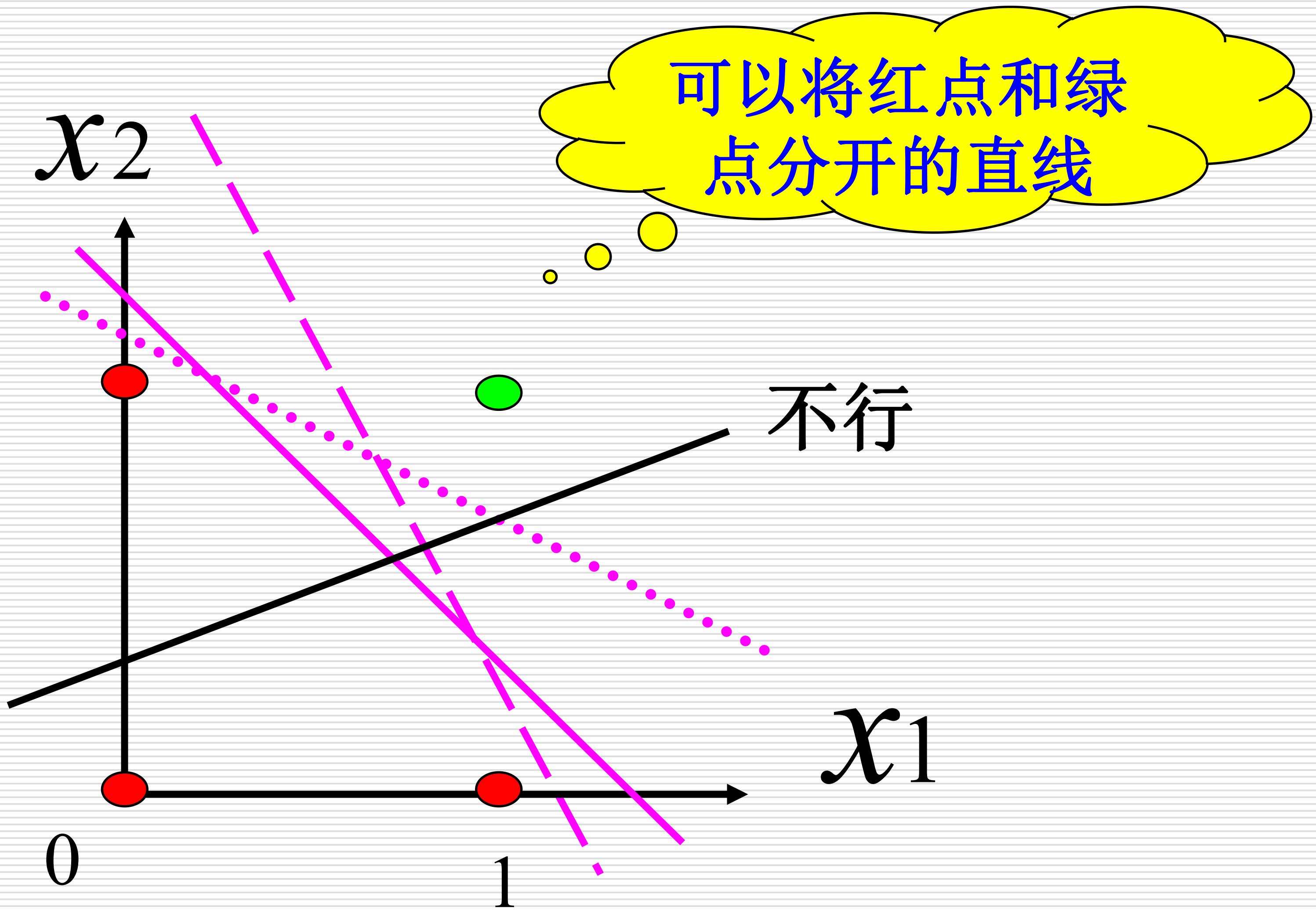
当$\omega_2 > 0$时,直线上方代表$y > 0$的区域；

直线下方代表$y < 0$的区域；

一组输入值则对应$x_1 - x_2$坐标系中的一个点；

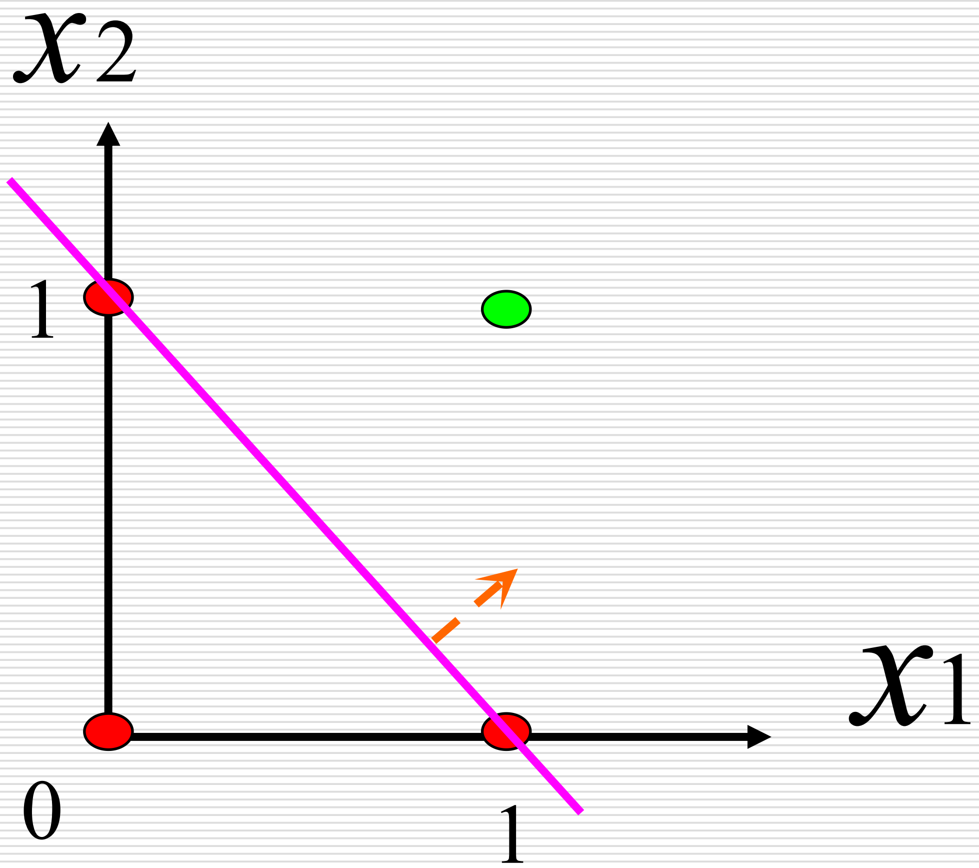
画出这条直线，使得前 三组输入对应的3个点都落在直线下方，最后 一组对应的点落在直线上方，则这条直线满 足AND的真值表，直线对应的参数就确定了 感知器

AND运算的实现



感知器的参数设计

AND运算的实现



为简便，可选一条特殊曲线，标出正方向

写出直线方程---可用点斜式/截距式/两点式

AND运算的实现

相应的直线方程为：$y = x_1 + x_2 - 1 = 0$

则感知器的一组参数可取为：

$$\begin{cases} \omega_1 = 1 \\ \omega_2 = 1 \\ b = -1 \end{cases}$$

注：

1.上述参数取值乘以一个正的常数，也符合要求；

2.上述参数取值如乘以一个负数，则改变了输入点对应的输出值，故不符合要求。

AND运算的实现

由直线写出对应的直线方程的通用方法

1.画出直线与两个坐标轴的交点：假设

交点分别为$a(X1,0), b(0,X2)$

2.代入点斜式方程：

$$x_2 = -\frac{X2}{X1}x_1 + X2$$

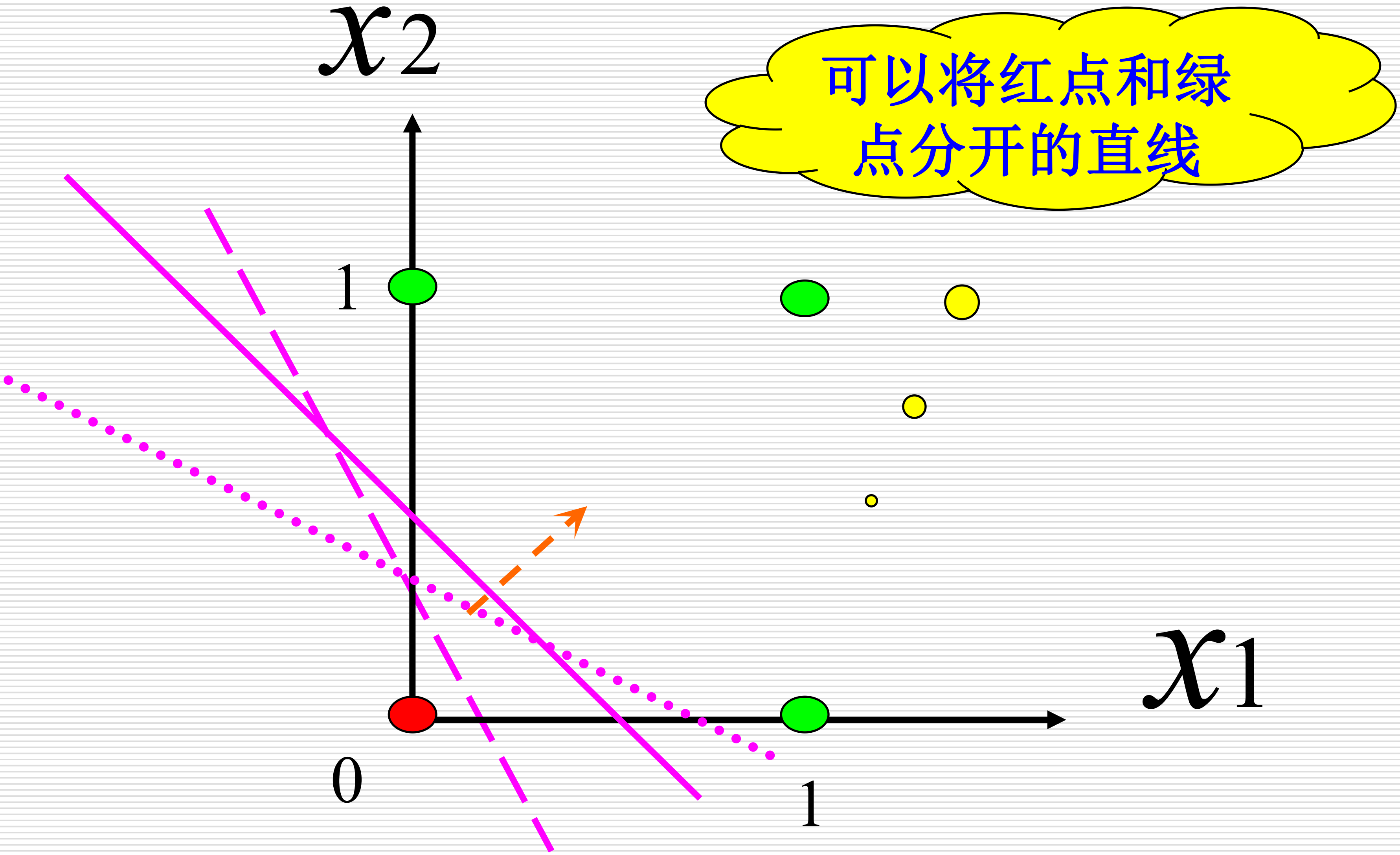
3.整理：

$$\frac{X2}{X1}x_1 + x_2 - X2 = 0 \text{或} -\frac{X2}{X1}x_1 - x_2 + X2 = 0$$

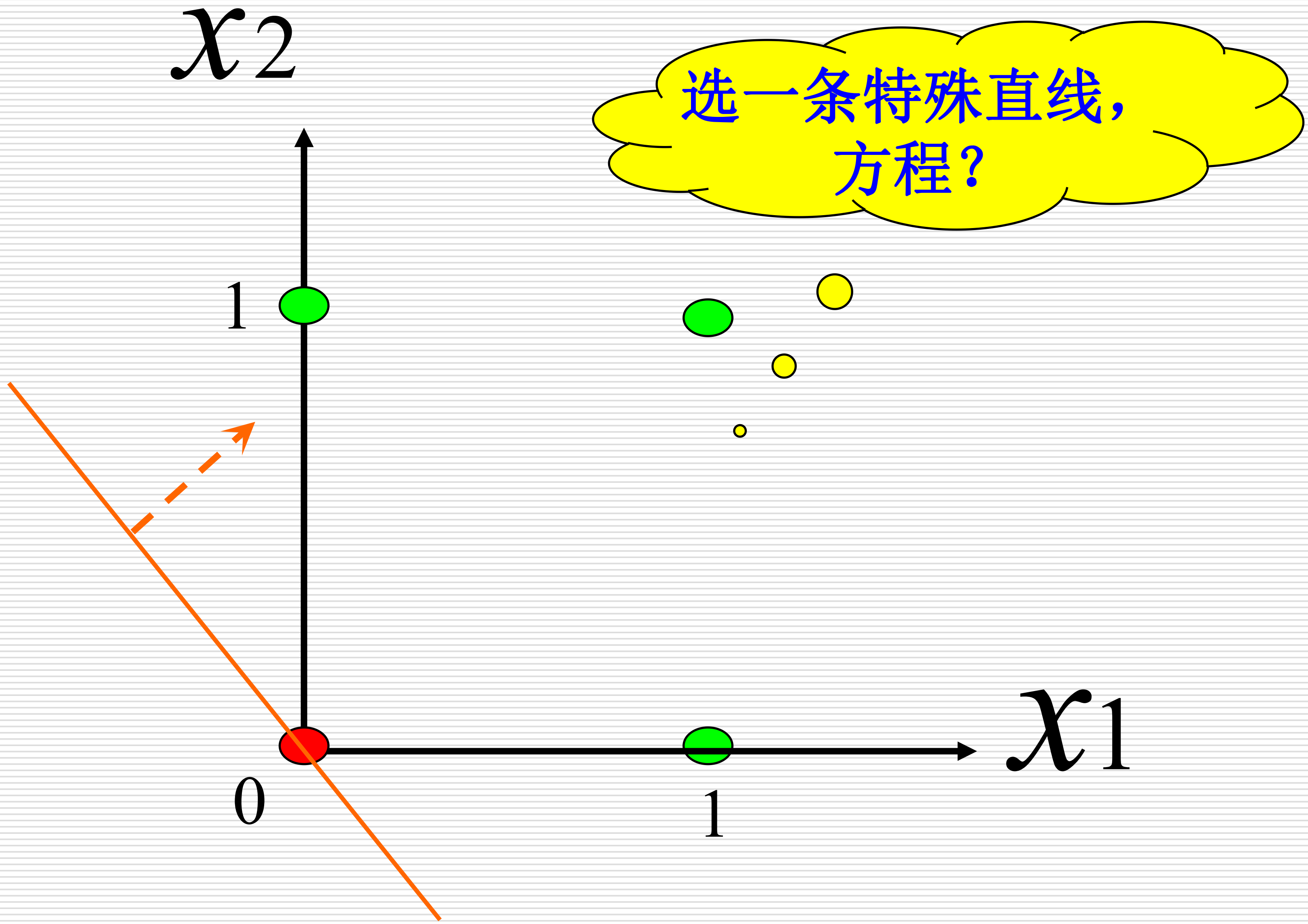
4.根据直线的正方向选择其中一个直线方程;

若直线正方向和坐标轴x_1方向一致，则$\omega_1 > 0$

OR运算的实现



OR运算的实现



2. 双输入感知器的设计

- 结论
  - 只要能画出一条直线，将输入点按照要求分成两类，那么就可以设计出相应的感知器，实现该功能
  - 确定感知器的参数，可以通过获取这条直线的方程得到
- 直线方程，可以通过两点式或点斜式来得到

感知器设计的步骤

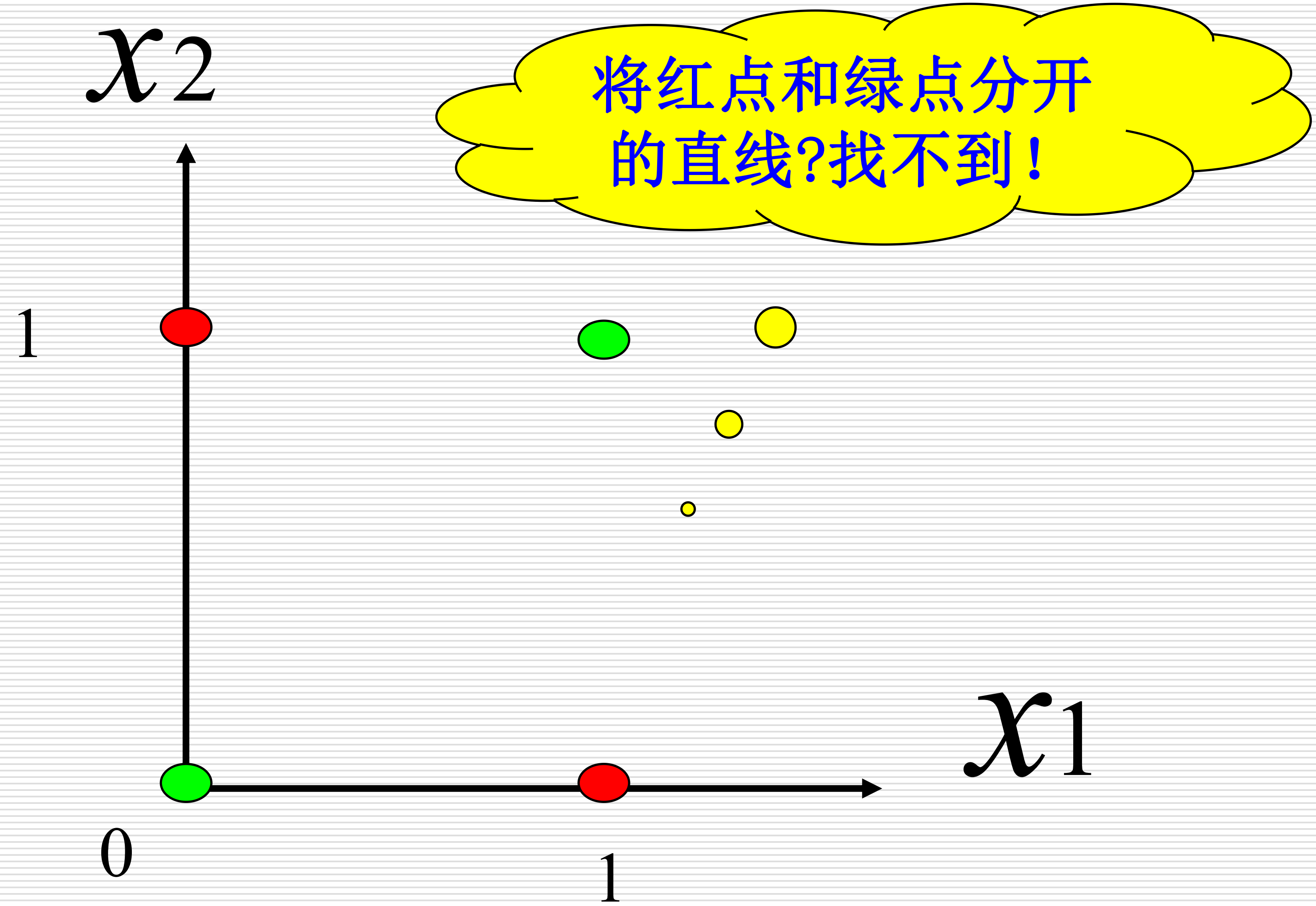
- 1 画出布尔函数的真值表；
- 2 画出x1--x2坐标系的输入点；
- 3 画出对输入点正确分类的直线，标出正方向；
- 4 写出直线方程；
- 5 得到感知器的参数

异或运算

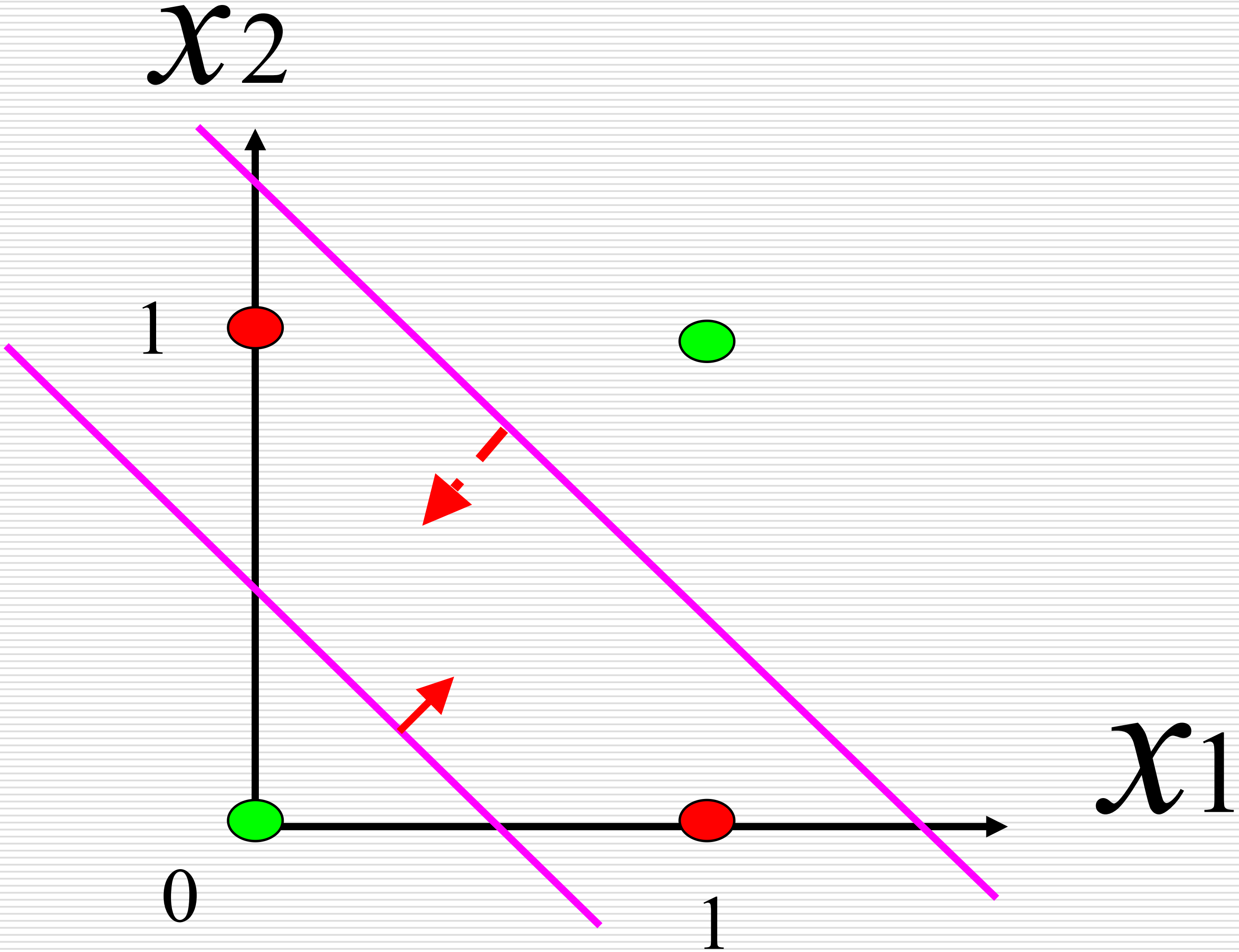
- 异或的真值表

p	q	$p \oplus q$
0	0	0
0	1	1
1	0	1
1	1	0

异或运算



异或运算



异或运算

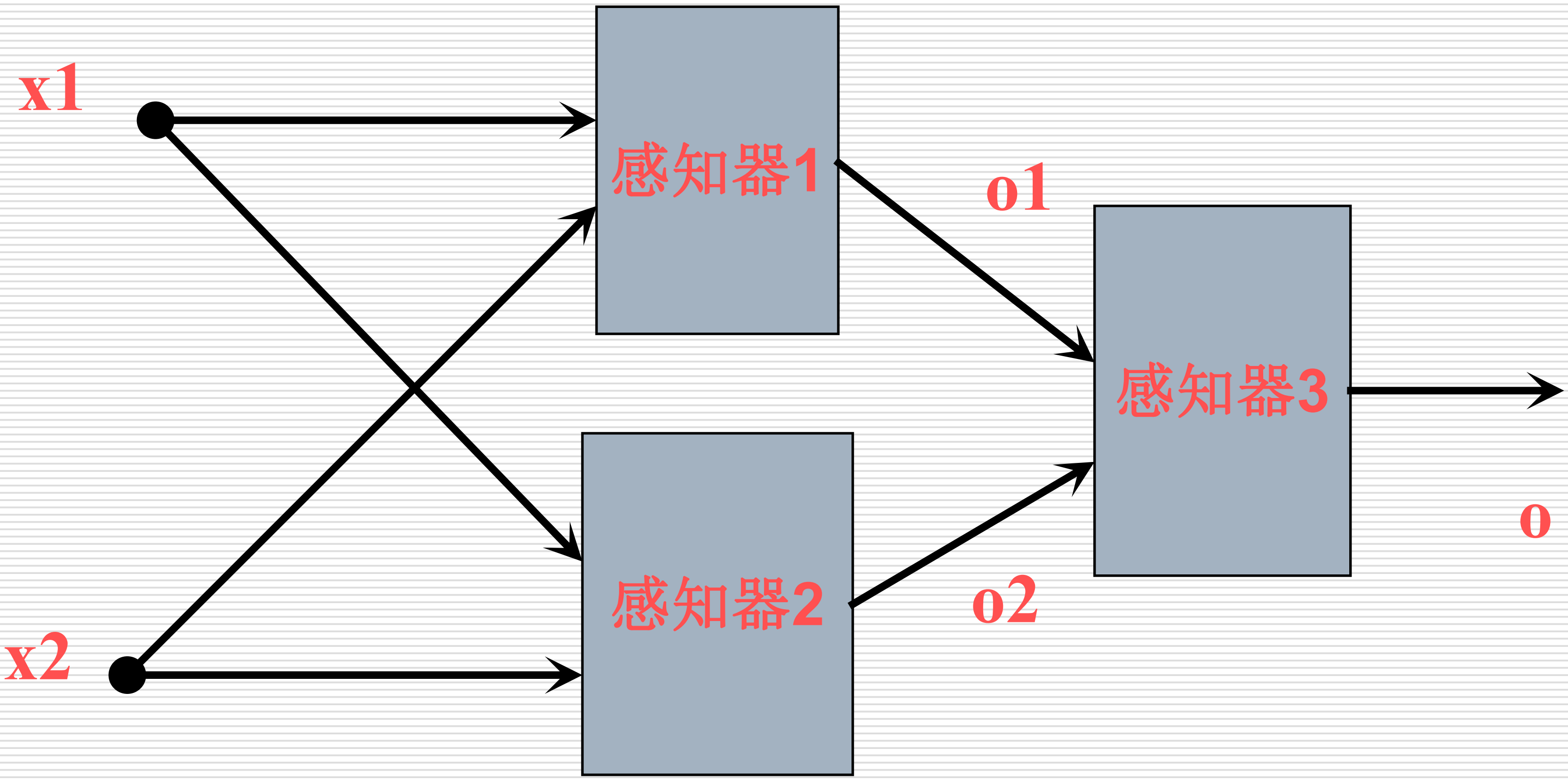
- 异或运算需要使用**3个感知器**来实现，即需要一个简单的感知器网络实现异或运算
- 利用2个感知器，对应2条直线，将红点和绿点分开：

$$x_1 + x_2 - 0.5 = 0$$

$$-x_1 - x_2 + 1.5 = 0$$

- 再利用感知器实现AND运算

异或运算



练习1

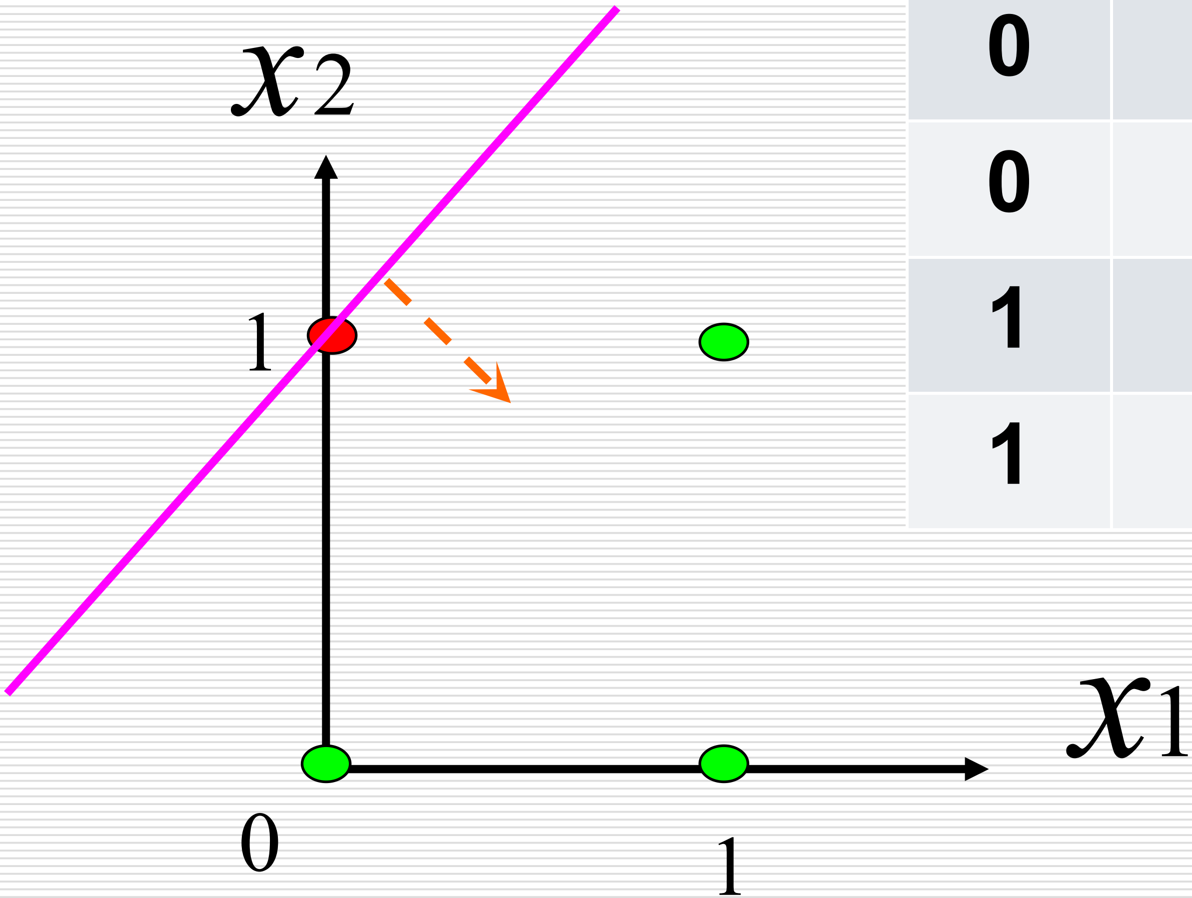
- 利用一个二输入的感知器，可以实现以下哪些布尔函数？如果可以实现，则写出一组权重参数

$$A \vee \neg B, \neg(A \vee \neg B)$$

$$A \vee B \vee \neg A, B \wedge A \vee \neg A$$

练习1(1)

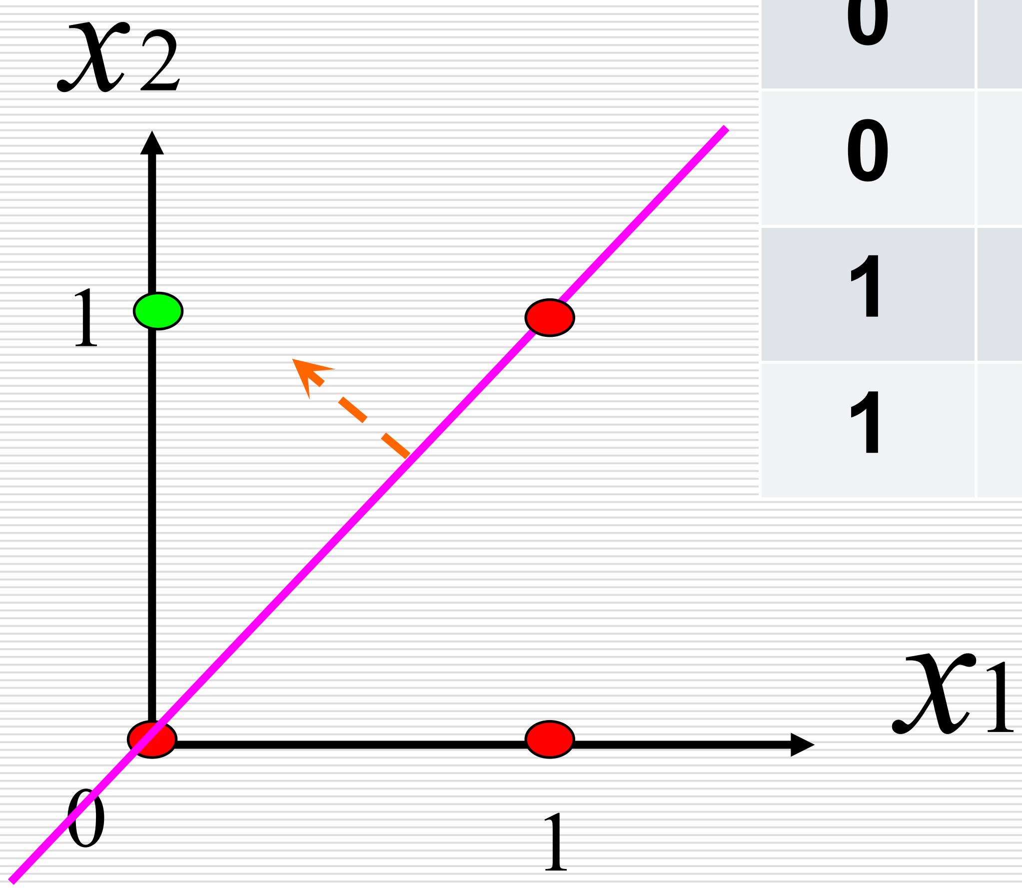
$$eg.x_1 - x_2 + 1 = 0$$



A	B	$A \vee \neg B$
0	0	1
0	1	0
1	0	1
1	1	1

练习1(2)

$$eg. -x_1 + x_2 = 0$$

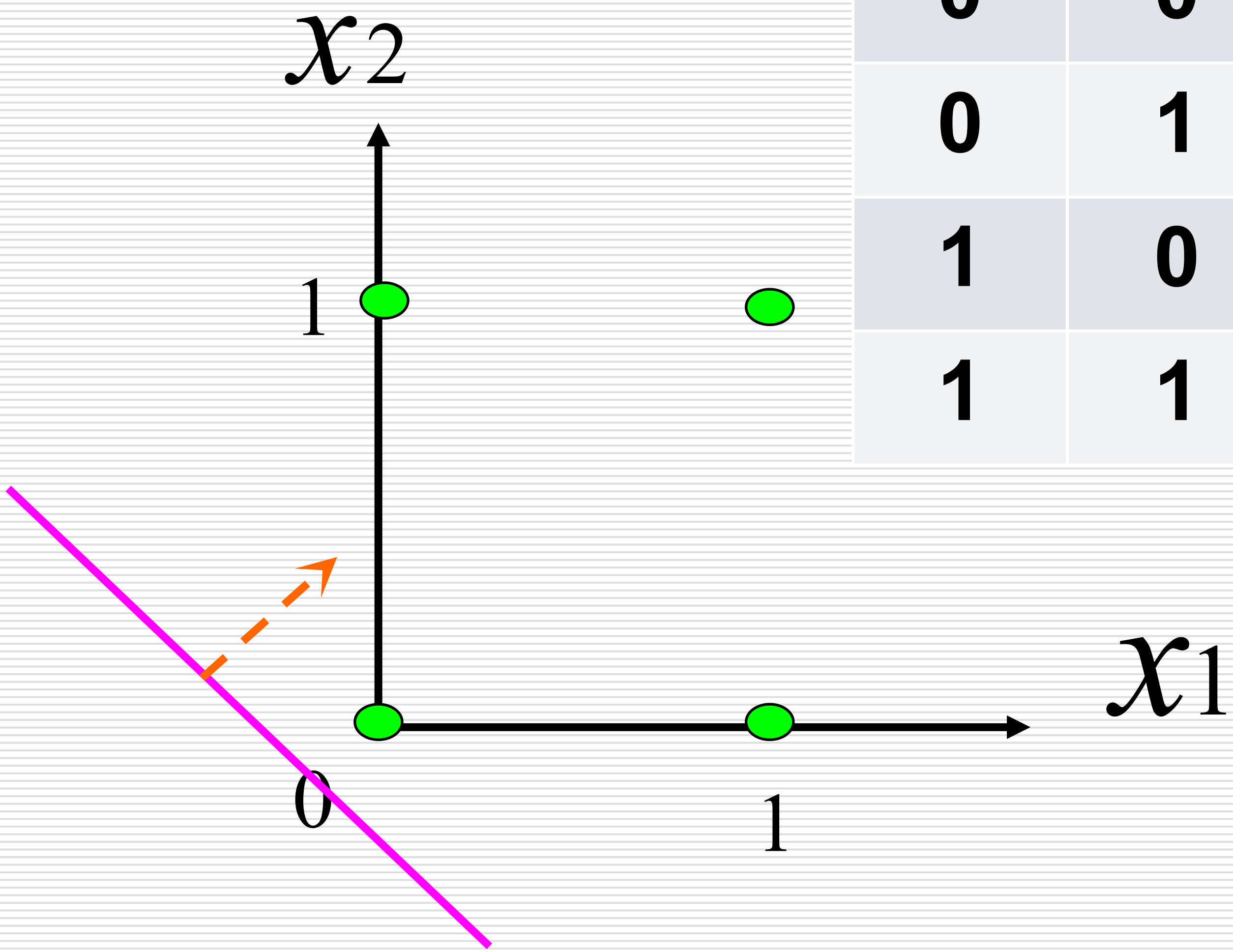


A	B	$\neg(A \vee \neg B)$
0	0	0
0	1	1
1	0	0
1	1	0

练习1(3)

$$eg.x_1 + x_2 + 1 = 0$$

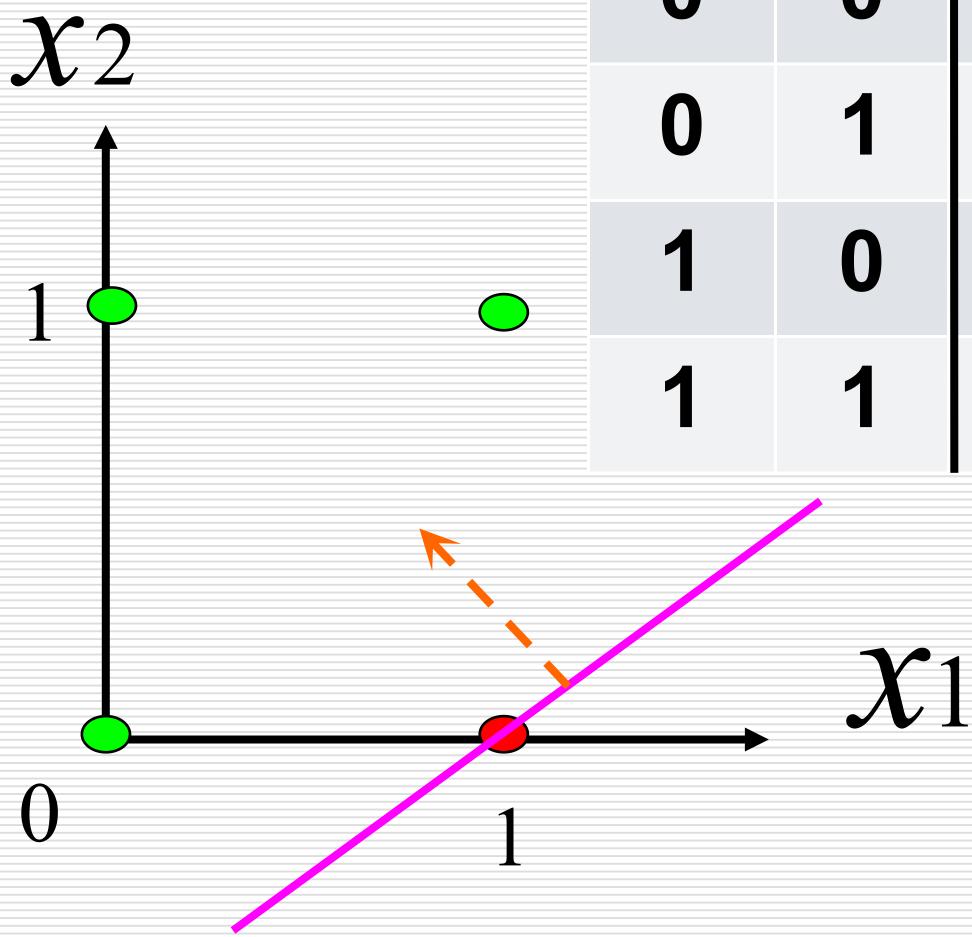
A	B	$A \vee B \vee \neg A$
0	0	1
0	1	1
1	0	1
1	1	1



练习1(4)

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

A	B	$B \wedge A \vee \neg A$
0	0	1
0	1	1
1	0	0
1	1	1



练习2

考虑使用阈值表达式$\omega_1 x_1 + \omega_2 x_2 + b = 0$定义的两个感知器。感知器$A$的权值为

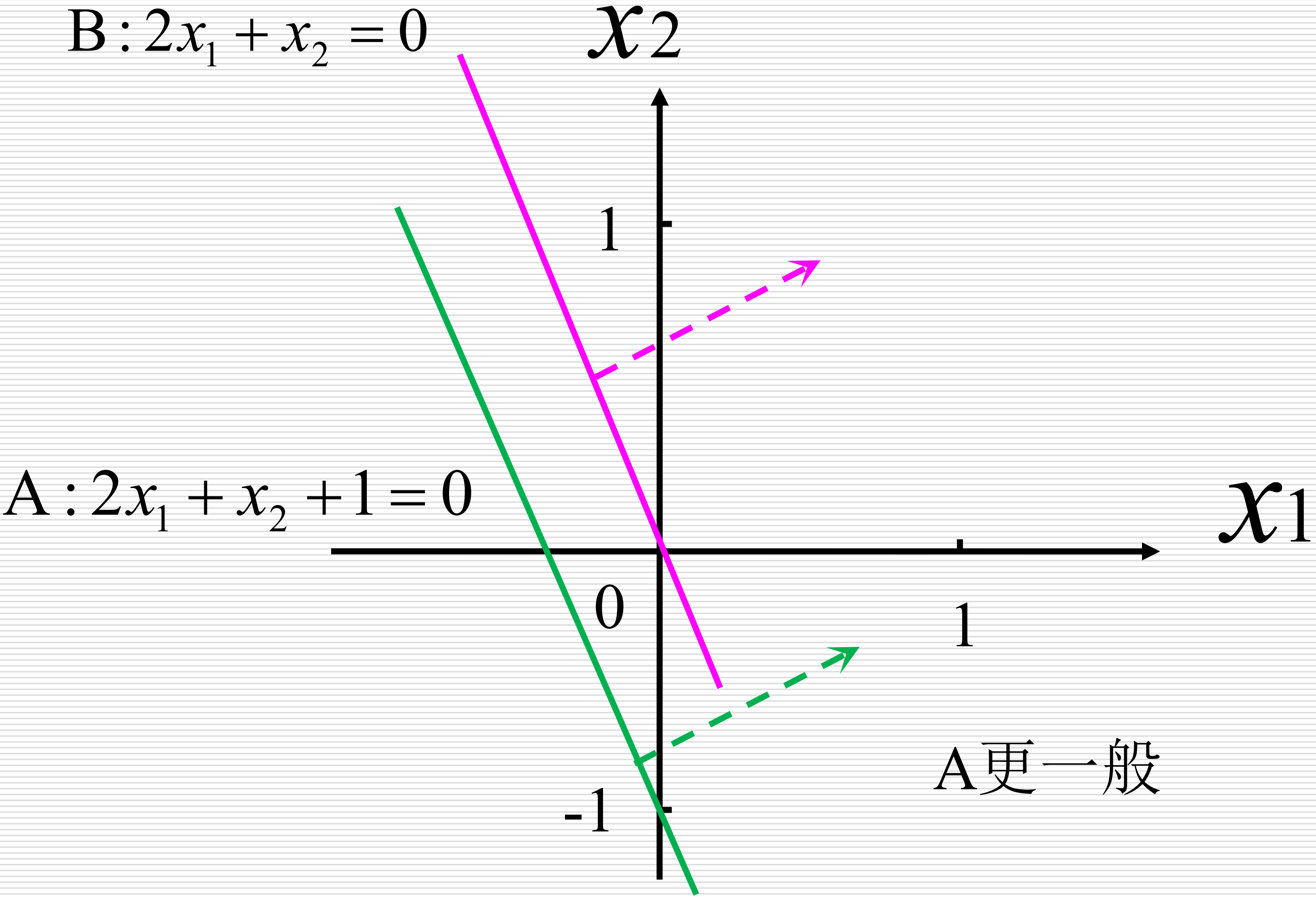
$\omega_1 = 2$，$\omega_2 = 1$，$b = 1$;

感知器B的权值为：

$\omega_1 = 2$，$\omega_2 = 1$，$b = 0$;

试问两个感知器中哪一个更一般？

练习2解答



3. 多输入的感知器

- 单个感知器可以表示布尔函数与、或、与非和或非等
- 三输入感知器的权值表达式为：

$$\omega_1 x_1 + \omega_2 x_2 + \omega_3 x_3 + b = 0$$

它表示三维空间的一个平面；

n个输入的感知器，其权重表达式为

$\sum_{i=1}^{n} w_i x_i + b = 0$表示$n$维空间的一个超平面

3. 多输入的感知器

- 感知器可以看成n维实例空间中的超平面决策面，对于该平面一侧的实例，感知器输出1，对于另一侧的实例输出0；
- 正反例的集合，不一定能被超平面所分割；可以被超平面分割的集合称为线性可分的样例集合。

练习 1

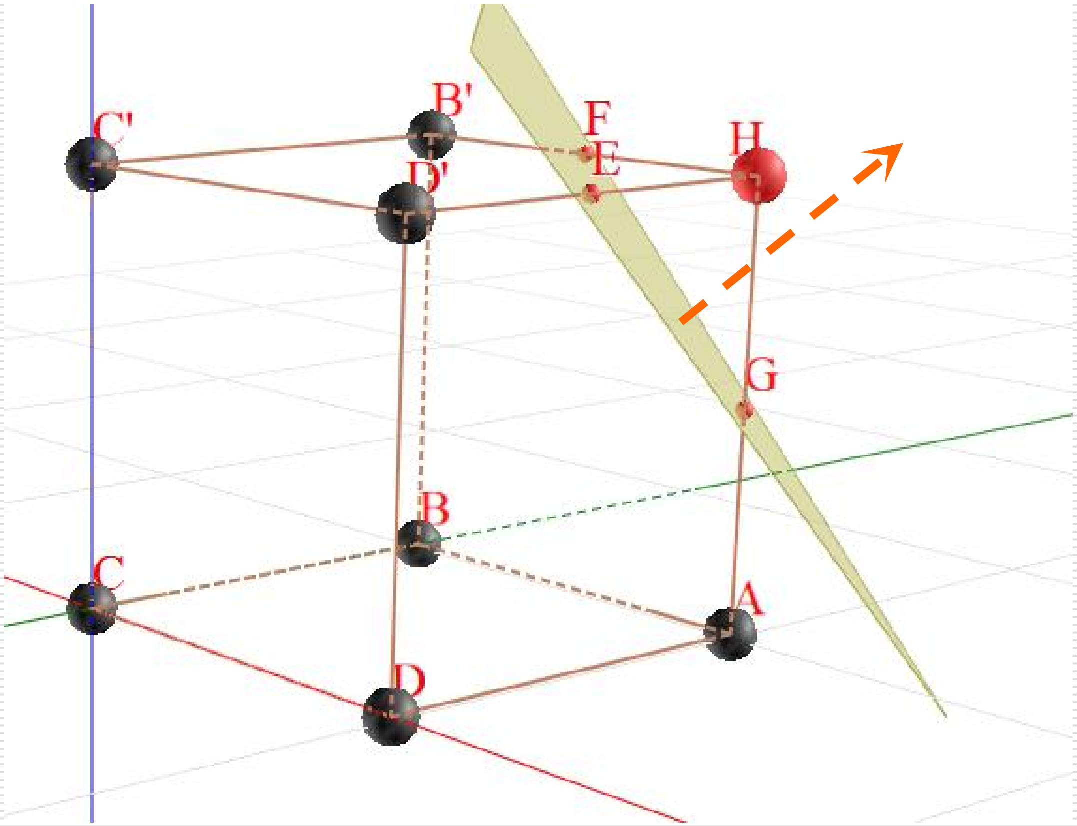
- 利用一个三输入的感知器，可以实现以下哪些布尔函数?

$$A \wedge B \wedge C, A \wedge B \vee C,$$

$$A \wedge B \vee \neg C, \neg A \wedge \neg B \wedge \neg C$$

练习 1 (1)

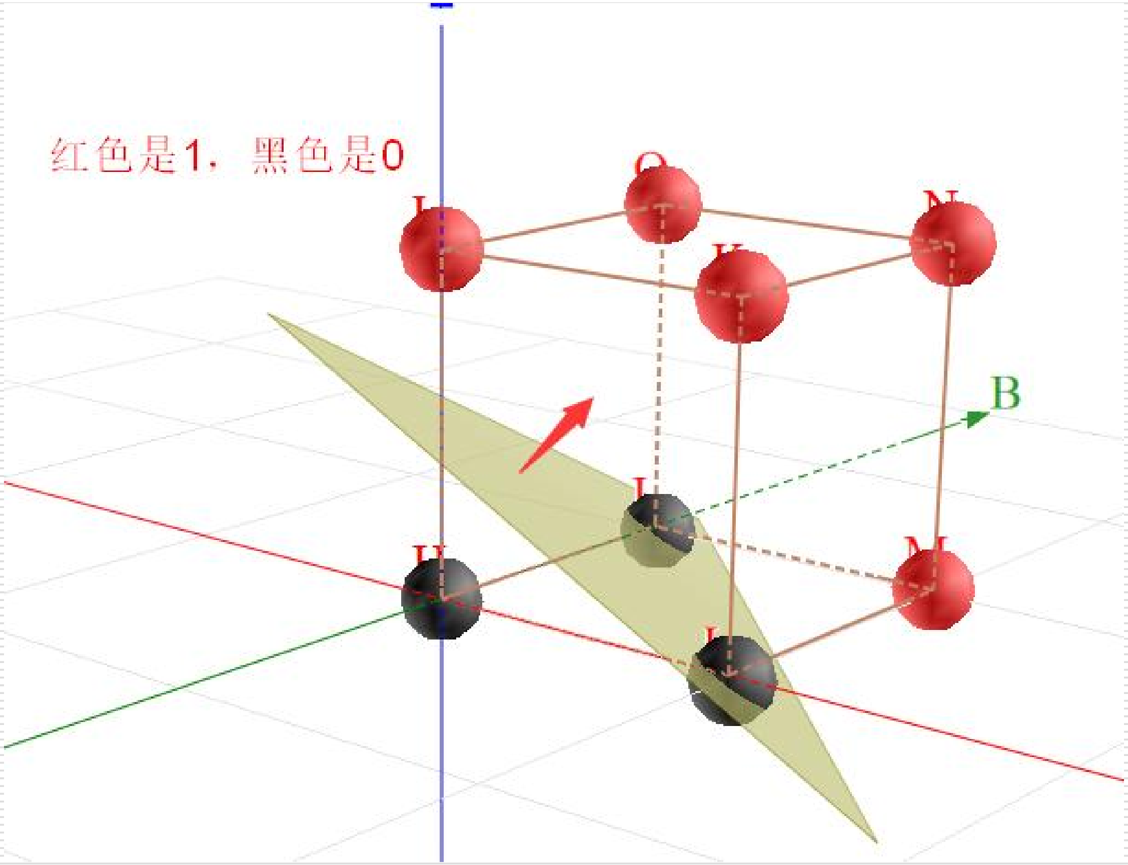
A	B	C	$A \wedge B \wedge C$
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	0
1	0	0	0
1	0	1	0
1	1	0	0
1	1	1	1



可以

练习 1 (2)

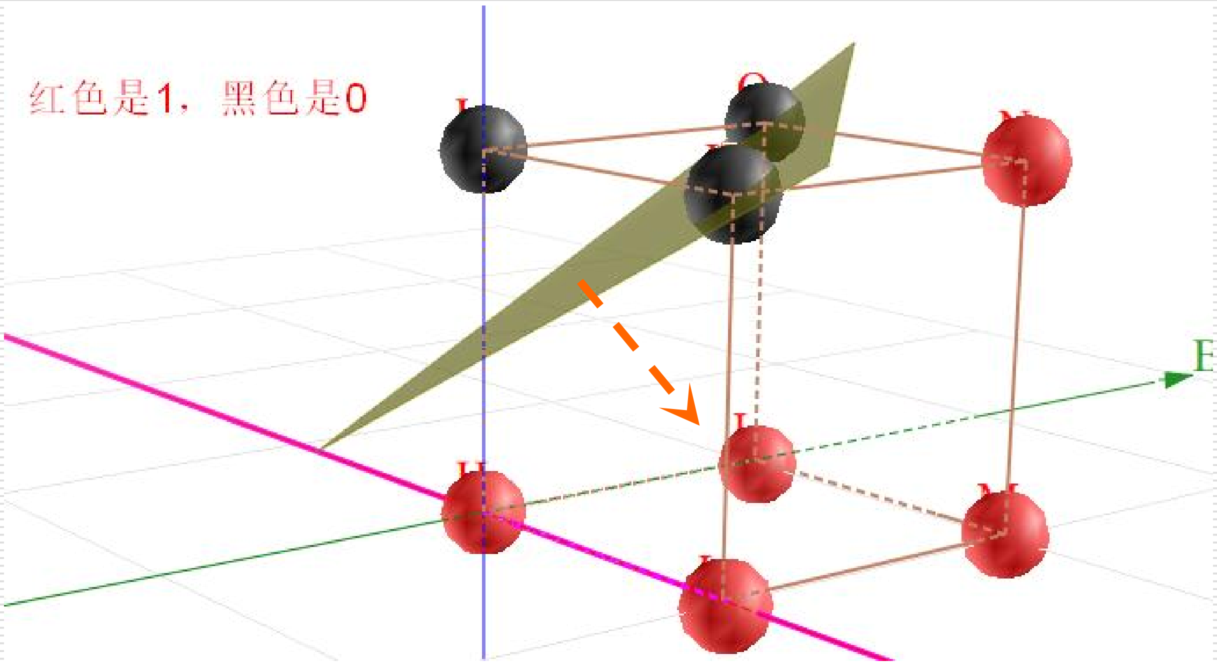
A	B	C	$A \wedge B \vee C$
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1



可以

练习 1 (3)

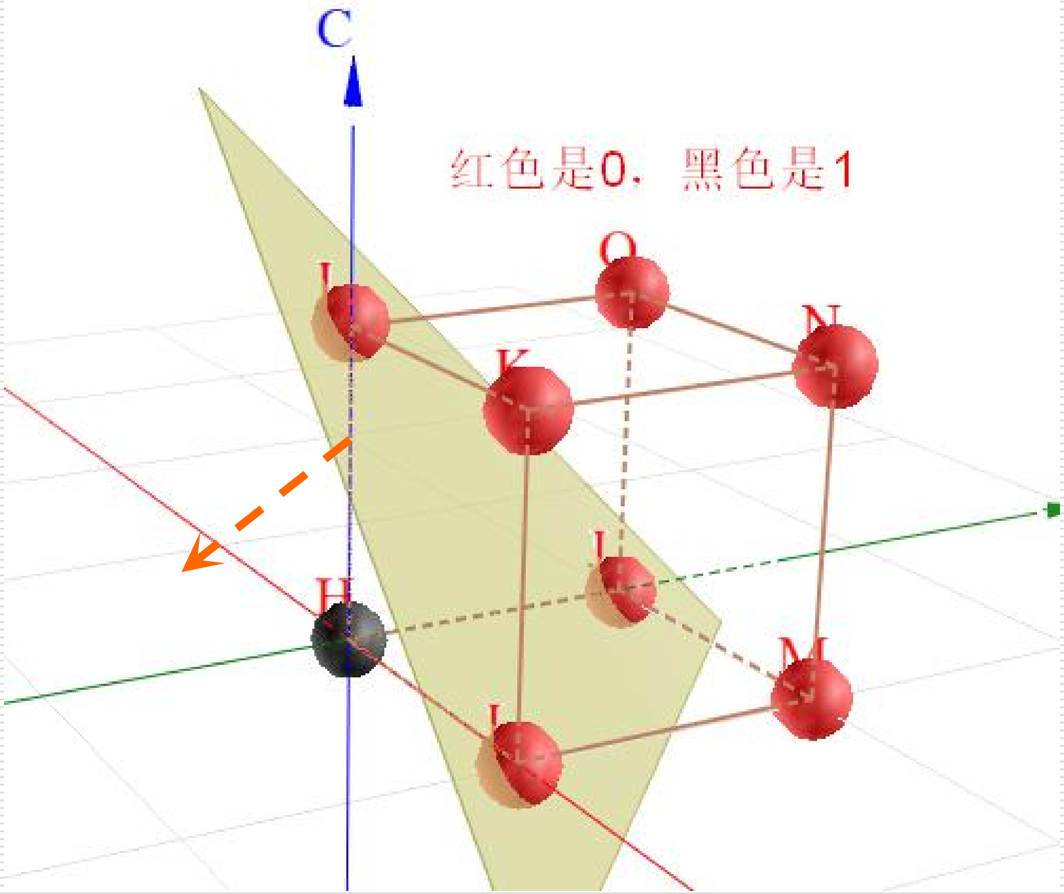
A	B	C	$A\wedge B\vee\neg C$
0	0	0	1
0	0	1	0
0	1	0	1
0	1	1	0
1	0	0	1
1	0	1	0
1	1	0	1
1	1	1	1



可以

练习 1 (4)

A	B	C	$\neg A \wedge \neg B \wedge \neg C$
0	0	0	1
0	0	1	0
0	1	0	0
0	1	1	0
1	0	0	0
1	0	1	0
1	1	0	0
1	1	1	0



可以

练习 2

- 利用直线方程的方法，设计一个三输入的感知器，来实现以下布尔函数：

$$A \vee B \vee C$$

4. 感知器训练法则

- 设计单个感知器时，只要n维实例空间是线性可分的，就可以通过超平面来确定权值；但是超平面不象2维和3维时的直线和平面那么直观
- 对于n>3时，可以采用学习的方法来确定参数

4. 感知器训练法则

设计感知器就是确定一组权重值$\omega_0, \omega_1, \omega_2, \cdots, \omega_n$,的值，使得下式成立：

$$\begin{cases} y = \omega_1 x_1 + \omega_2 x_2 + \cdots + \omega_n x_n + \omega_0 > 0, \text{当样例为正例} \\ y = \omega_1 x_1 + \omega_2 x_2 + \cdots + \omega_n x_n + \omega_0 \leq 0, \text{当样例为反例} \end{cases}$$

这样，确定权值的问题就转换为感知器的学习问题

感知器的学习

从随机的一组权值开始，然后反复地应用这个感知器到每个训练样例，只要它误分类训练样例，那么就修改感知器的权值。重复这一过程，直到感知器正确分类所有的训练样例。

学习过程的关键在于：当感知器对于某个训练样例误判时，如何修改权值？

训练法则

对于每个训练样例，当感知器误判时，则修改与感知器的输入x_i对应的权重ω_i的法则如下：

$\omega_i \leftarrow \omega_i + \Delta\omega_i$,其中$\Delta\omega_i = \eta(t-o)x_i$

η是一个正的常数，称为学习速率,通常较小，可用来缓和每一步调整权重的程度；t是当前训练样例的目标输出，即正例为1，反例为0；o是感知器的输出

训练过程

情形1：假定当前的训练样例为正例，而感知器的输 出为反例，且 $x_i > 0$,那么：

$$\Delta\omega_i = \eta(t - o)x_i = \eta(1 - 0)x_i > 0$$

因 $\omega_i \leftarrow \omega_i + \Delta\omega i, \omega_i$不断增大，

y也相应地不断增大，使 得o

逼近目标输出 t

训练过程

情形2：假定当前的训练样例为反例，而感知器的输 出为正例，且 $x_i > 0$, 那么：

$$\Delta\omega_i = \eta(t - o)x_i = \eta(0 - 1)x_i < 0$$

因 $\omega_i \leftarrow \omega_i + \Delta\omega i, \omega_i$不断缩小，$y$也相应地不断减小，使 得$o$逼近目标输出 t

训练过程

情形3：如果当前的训练样例为正（反）例，而感知器的输出也为正（反）例，那么：

$$\Delta\omega_i = \eta(t - o)x_i = 0$$

ω_i保持不变

训练过程

结论

- 数学上可以证明，只要训练样例集合是线性可分的，并且使用了充分小的学习速率，感知器的训练过程就一定可以在有限步骤内收敛，且收敛到一个能够正确分类所有训练样例的权向量

5. delta法则简介

- 当训练样例集合不是线性可分的，那么感知器的学习过程就不能收敛，那么就需要采用delta法则。
- delta法则可以使得感知器的学习过程收敛到目标函数的最佳近似

5. delta法则简介

- **delta训练法则的思想：**
  - 采用梯度下降方法来搜索可能的权向量的假设空间，来找到最佳拟合训练样例的权向量

5. delta法则简介

*delta*训练法则可以理解为训练一个无阈值的感知器，感知器输出为：

$$y(\mathbf{x}) = \sum_{i=1}^{n} \omega_i x_i = \boldsymbol{\omega} \bullet \mathbf{x}$$

训练误差（能量）为：

$$E(\boldsymbol{\omega}) = \frac{1}{2} \sum_{d \in D} (t_d - y_d)^2$$

5. delta法则简介

分析$E(\boldsymbol{\omega})$:

$$E(\boldsymbol{\omega}) = \frac{1}{2}\sum_{d\in D}(t_d - y_d)^2$$

当权值仅有ω_1和ω_2的时候，

$E(\boldsymbol{\omega})$是在$\omega_1 - \omega_2$坐标系中的抛物面，

该抛物面的开口向上，又称误差曲面。

(注意此处输出采用连续值y_d，而不用离散值。)

梯度下降

- 梯度下降搜索，是指不断地修改权重，使得误差E(x)的值，逐步沿着误差曲面最陡峭的方向变化，最后达到曲面的最低点
- 误差曲面的最低点，表示在该组权重下，感知器的误差值最小

感知器的局限性

- 由于感知器的激活函数采用的是阈值函数，输出矢量只能取0或1，所以只能用它来解决简单的分类问题；
- 感知器仅能够线性地将输入矢量进行分类。

练 习

- 利用感知器学习的方法，设计一个两输入的感知器，来实现以下布尔函数：

$$A \vee B$$

- 列出学习的关键步骤

小 结

- 感知器的模型
- 感知器的参数获取
  - 代数方法: 直线方程、平面方程
  - 学习方法: 样例训练
- 感知器的训练法则
  - 适合线性可分情形
- Delta 法则