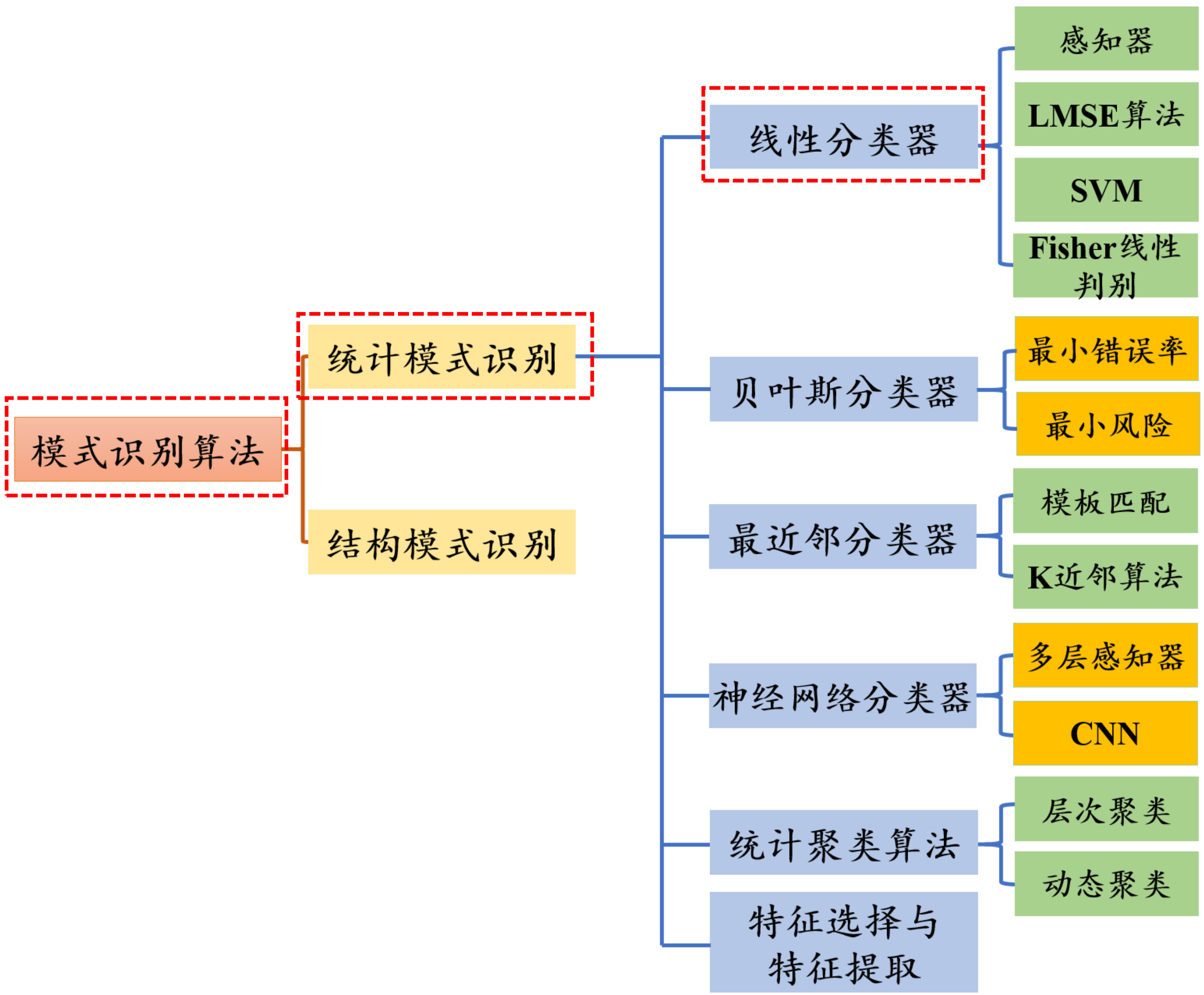
模式识别课程

线性分类器

张俊超

中南大学
航空航天学院

模式识别-线性分类器


模式识别算法
统计模式识别
结构模式识别
线性分类器
贝叶斯分类器
最近邻分类器
神经网络分类器
统计聚类算法
特征选择与特征提取
感知器
LMSE算法
SVM
Fisher线性判别
最小错误率
最小风险
模板匹配
K近邻算法
多层感知器
CNN
层次聚类
动态聚类

本章主要内容：

主要讨论**有监督的分类问题**

- ◆线性判别
- ◆感知器算法
- ◆LMSE算法(HK算法)
- ◆Fisher线性判别分析
- ◆支持向量机(SVM)
- ◆多分类的线性判别

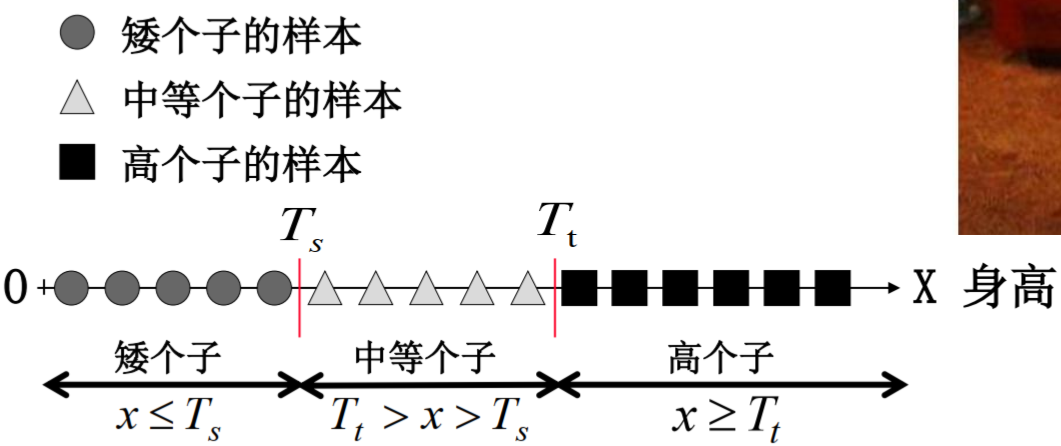
模式识别-线性分类器

学习目标：

- **了解**线性分类器的基本概念
- **理解**感知器算法、LMSE算法、Fisher判别法和SVM
- **应用**线性分类器解决实际分类问题

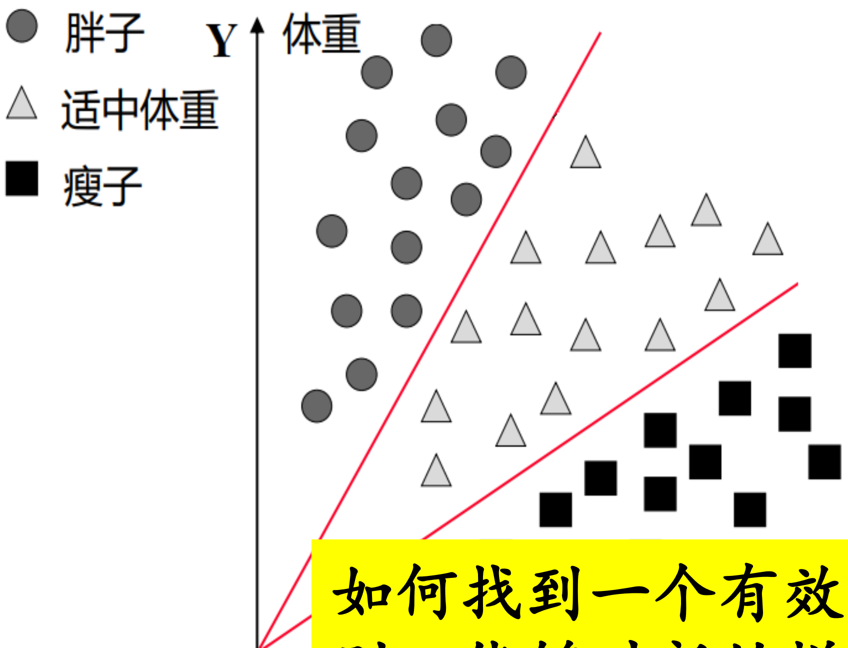
模式识别-线性分类器

- 根据身高特征，判断待识别个体的高矮程度



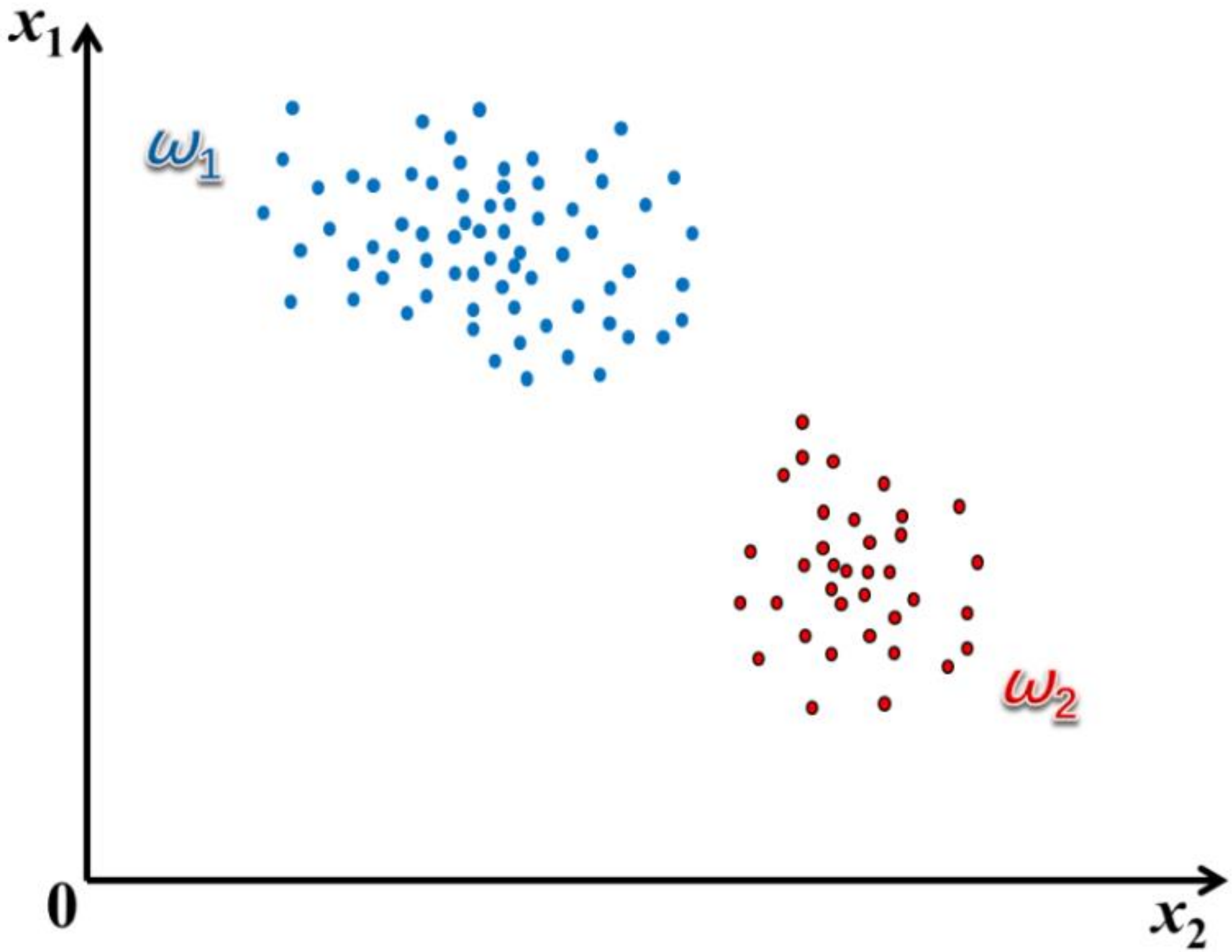
模式识别-线性分类器

- 根据身高和体重特征，判断待识别个体的胖瘦程度

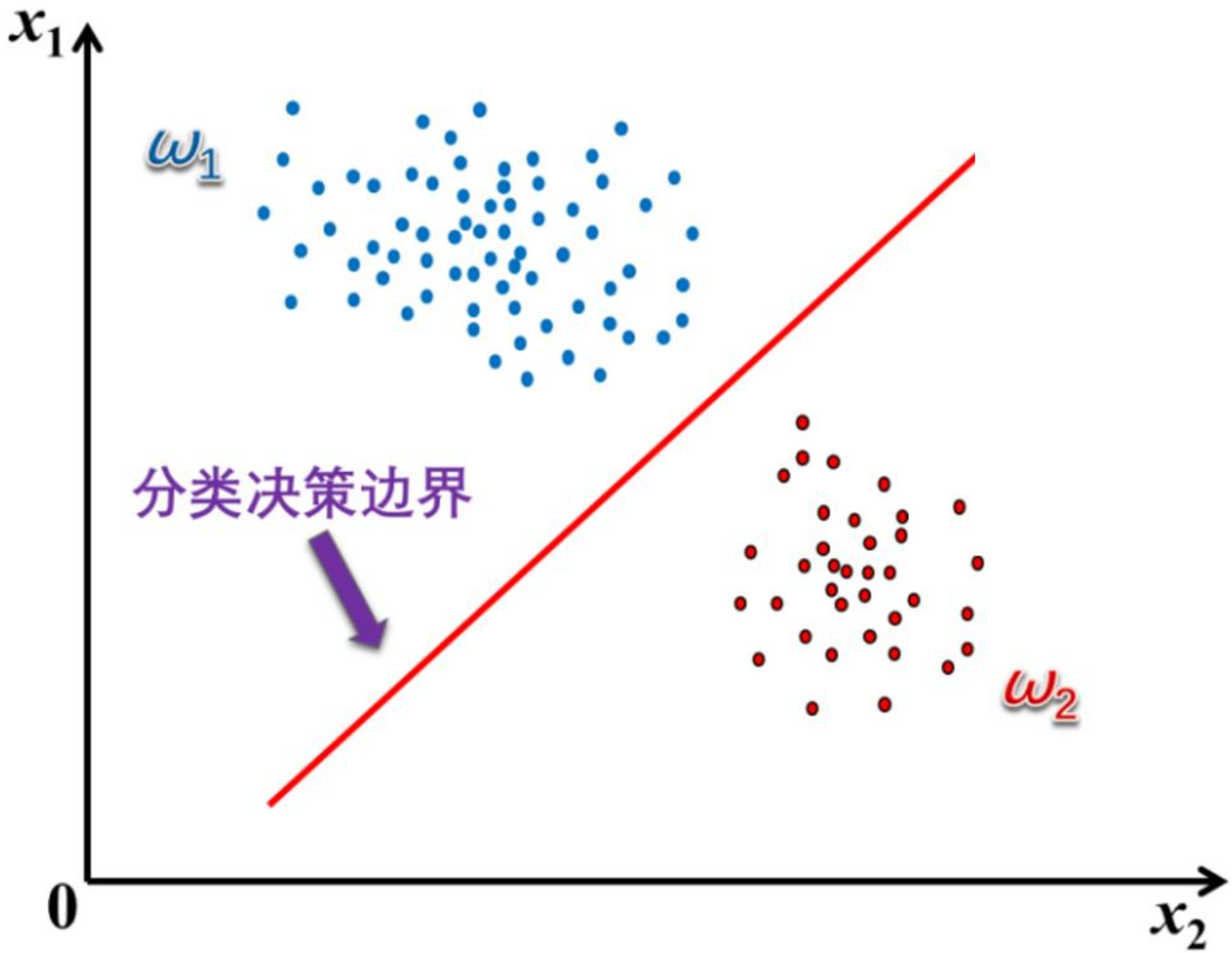


如何找到一个有效的分类决策规则，能够对新的样本正确地分类？

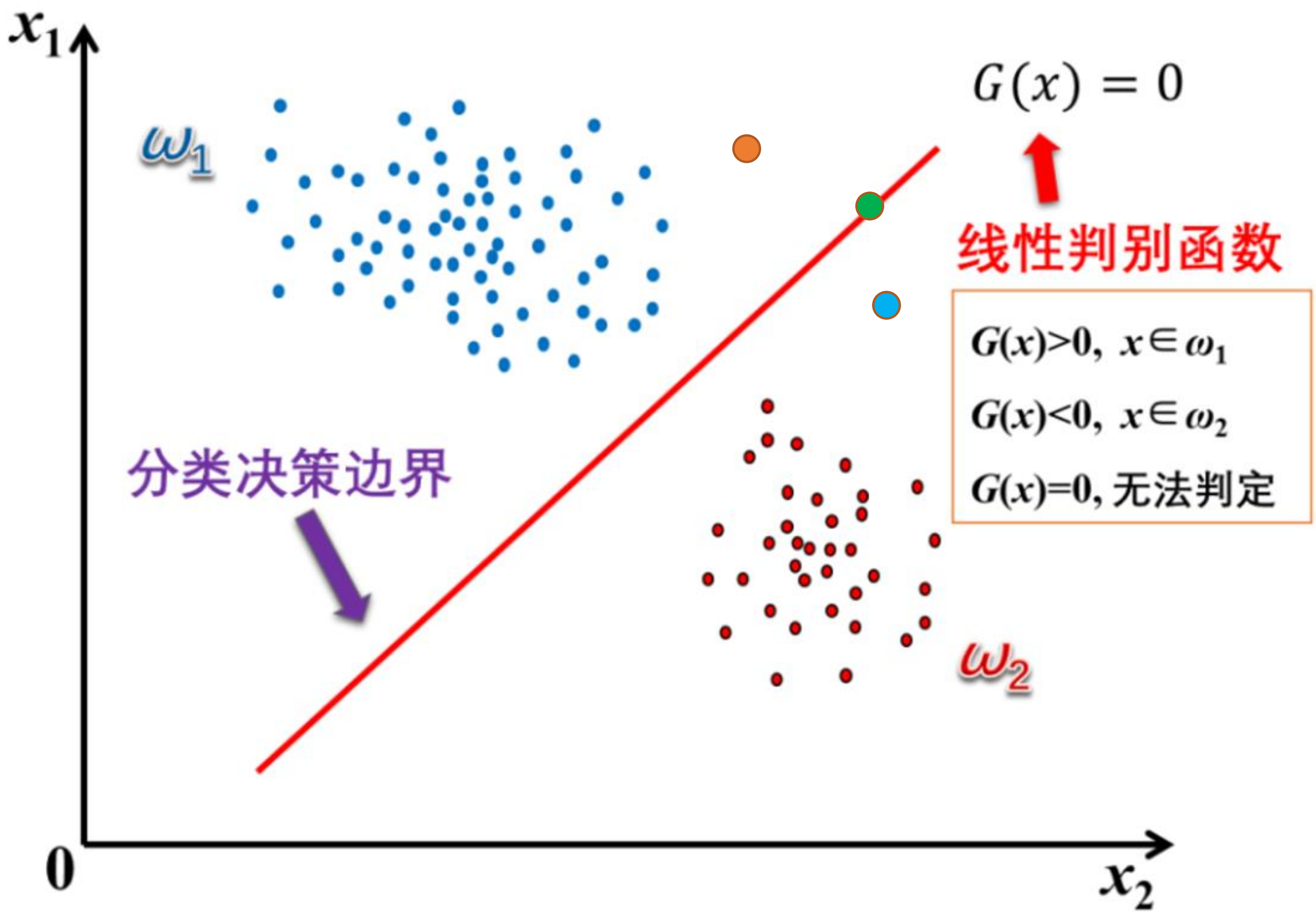
线性分类器－线性判别



线性分类器－线性判别



线性分类器－线性判别



x_1
ω_1
$G(x) = 0$
线性判别函数
$G(x)>0,\ x\in\omega_1$
$G(x)<0,\ x\in\omega_2$
$G(x)=0$, 无法判定
分类决策边界
ω_2
0
x_2

线性分类器－线性判别

线性判别函数： 判别函数是线性的，可以表示为：

$$G(\mathbf{x}) = \mathbf{w}^T\mathbf{x} + w_0$$

其中：特征向量 $\mathbf{x} = [x_1, x_2, \ldots, x_n]^T$

权重 $\mathbf{w} = [w_1, w_2, \ldots, w_n]^T$

偏差 w_0

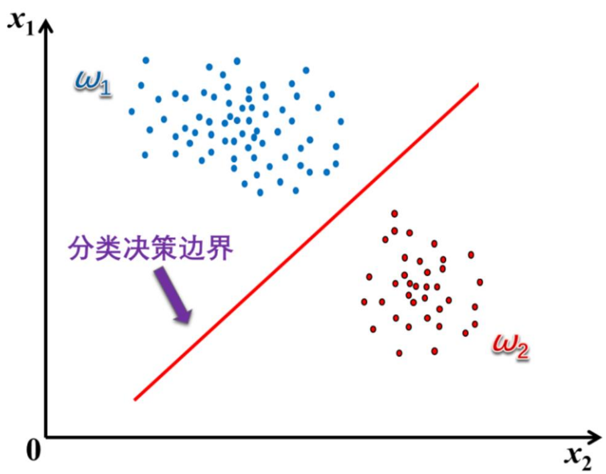
齐次表示为：

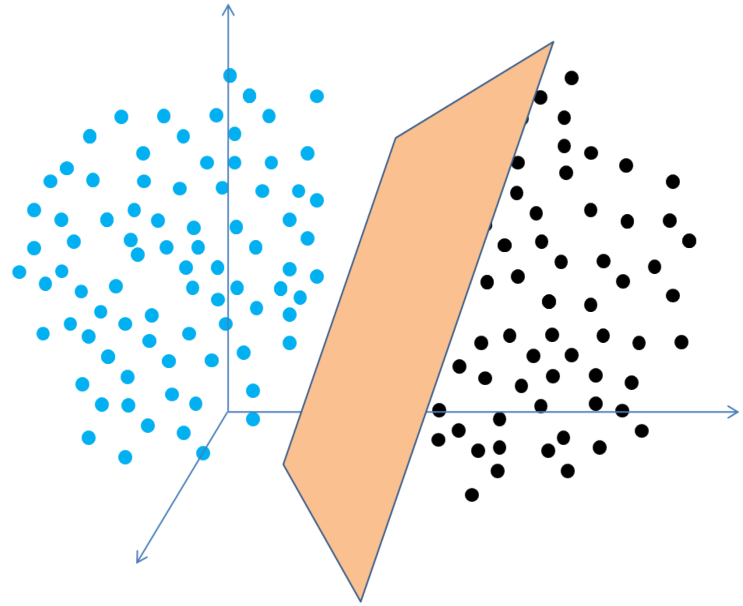
$$G(\mathbf{y}) = \boldsymbol{\alpha}^T\mathbf{y}$$

$$\boldsymbol{\alpha} = [w_1, w_2, \ldots, w_n, w_0]^T$$

$$\mathbf{y} = [x_1, x_2, \ldots, x_n, 1]^T$$

线性分类器－线性判别

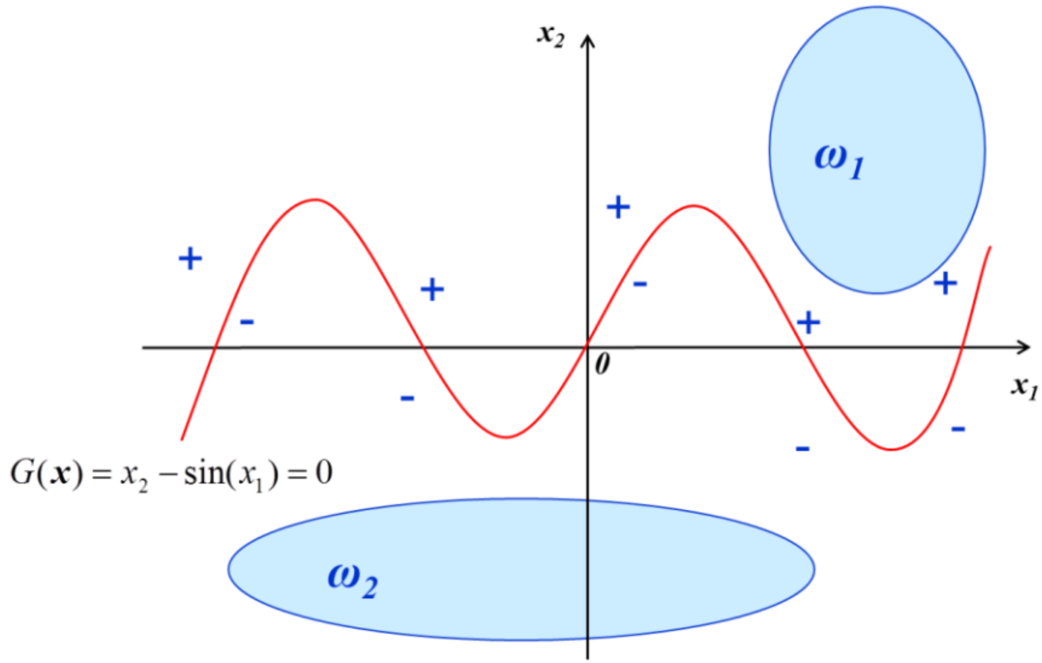




- 如果特征空间是1维的，线性分类器的分类决策边界就是一个点；
- 如果特征空间是2维的，分类决策边界就是一条直线；
- 如果特征空间是3维的，分类决策边界就是一个平面；
- 如果特征空间是更高维的，分类决策边界就是一个超平面。

线性分类器－线性判别

➢Q1:是不是任何一个模式识别问题，都可以找到线性分类决策边界呢？

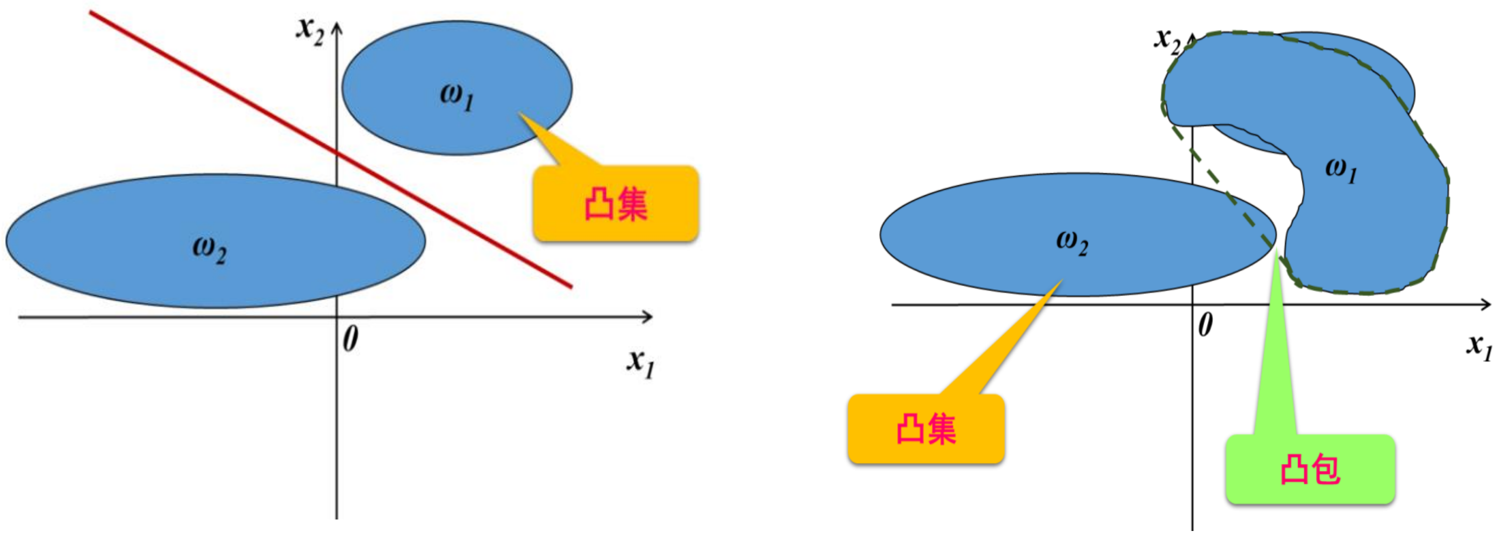


➢Q2:给定一个样本集，它是线性可分的吗？

线性分类器－线性判别

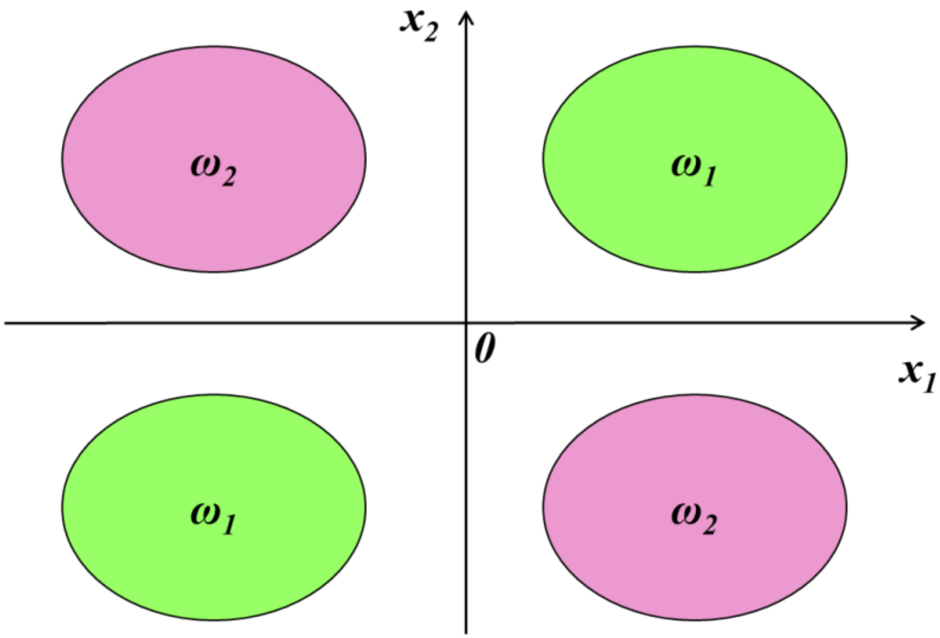
- 样本集的线性可分性：

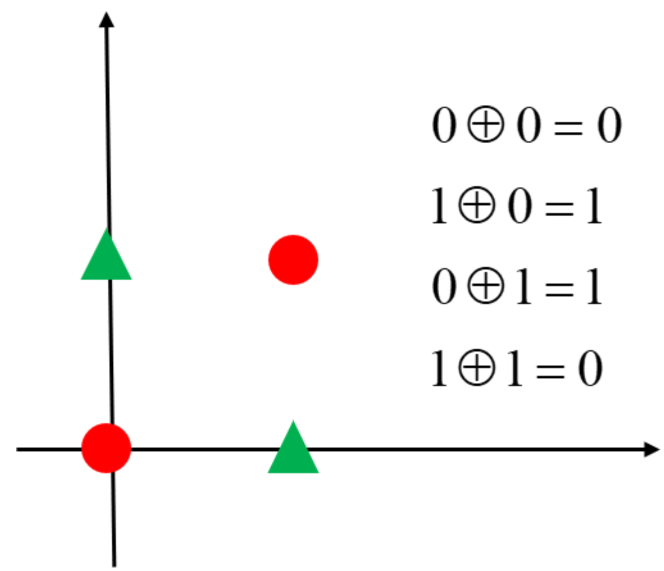
　　如果各个类别样本的分布区域互不相交，并且都是凸集，那么它一定是线性可分的。



线性分类器－线性判别

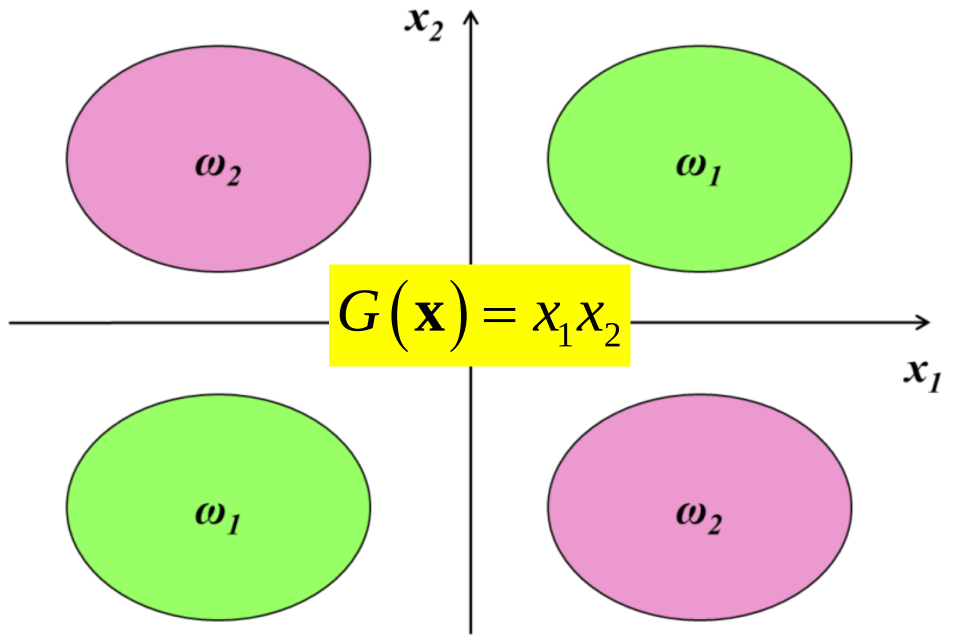
除了凸包相交的情形，如果同一类别样本的分布区域是由不连通的子区域组成的，也会带来线性不可分的问题，这种情形的典型案例就是异或问题 。





线性分类器－线性判别

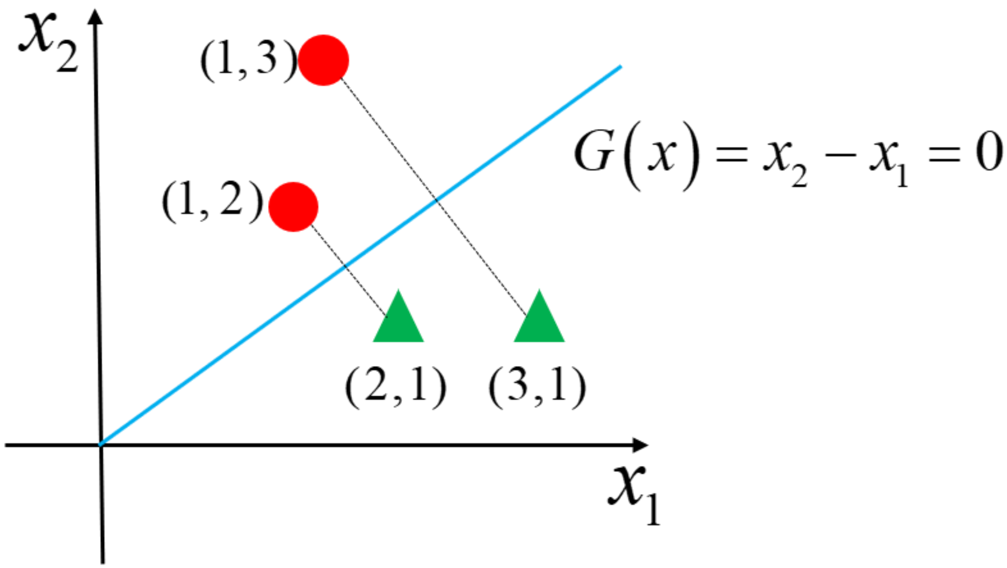
- Q1:有没有什么方法很快就能知道样本集是不是线性可分的呢?



- Q2:有没有什么方法可以将非线性判别问题转化成线性判别问题呢?

线性分类器－线性判别

样本距离决策边界的距离VS判别函数的绝对值



$$G(1,2)=2-1=1$$
$$G(1,3)=3-1=2$$
$$G(2,1)=1-2=-1$$
$$G(3,1)=1-3=-2$$

- 直观地发现，当一个样本距离分类边界越远，判别函数的绝对值也越大。
- 判别函数是样本到决策面距离远近的一种度量。

判别函数的几何意义究竟如何呢？

线性判别函数的几何意义

➢决策边界方程为：

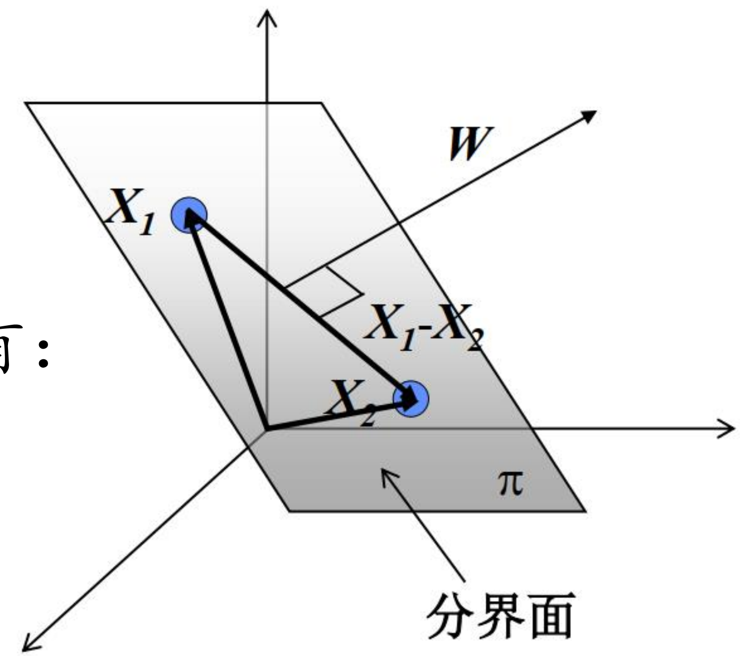
$$G(\mathbf{x}) = \mathbf{w}^T\mathbf{x} + w_0 = 0$$

➢设样本$\mathbf{x}_1, \mathbf{x}_2$ 在决策边界上，则有：

$$\mathbf{w}^T\mathbf{x}_1 + w_0 = \mathbf{w}^T\mathbf{x}_2 + w_0 = 0$$

$$\Downarrow$$

$$\mathbf{w}^T(\mathbf{x}_1 - \mathbf{x}_2) = 0$$



权向量**w**和决策边界上任一向量正交，即其与决策面法向量方向一致，方向指向正样本。

线性分类器 - 线性判别

线性判别函数的几何意义

➢设样本x 距离决策边界的距离为r，其在决策边界上的投影点为 $\mathbf{x}_p$,则：

$$\mathbf{x} = \mathbf{x}_p + r\frac{\mathbf{w}}{\|\mathbf{w}\|}$$

➢代入判别函数中，可得：

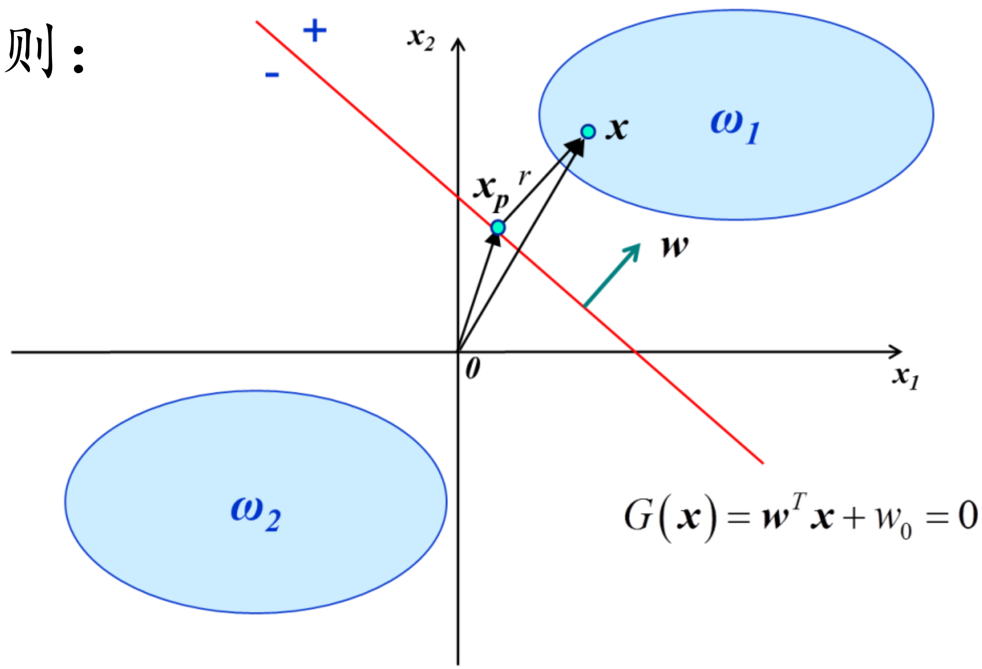
$$G(\mathbf{x}) = \boxed{\mathbf{w}^T\mathbf{x}_p + w_0} + r\frac{\mathbf{w}^T\mathbf{w}}{\|\mathbf{w}\|}$$

⇓

$$G(\mathbf{x}) = r\|\mathbf{w}\|$$

⇓

$$r = \frac{G(\mathbf{x})}{\|\mathbf{w}\|}$$



线性分类器－线性判别

◆线性判别小结：

- ➢决策边界在特征空间可以是点、线、面和超平面
- ➢样本集的线性可分性，仍需深入研究
- ➢非线性问题可以进一步转化为线性问题(后面章节会学到)

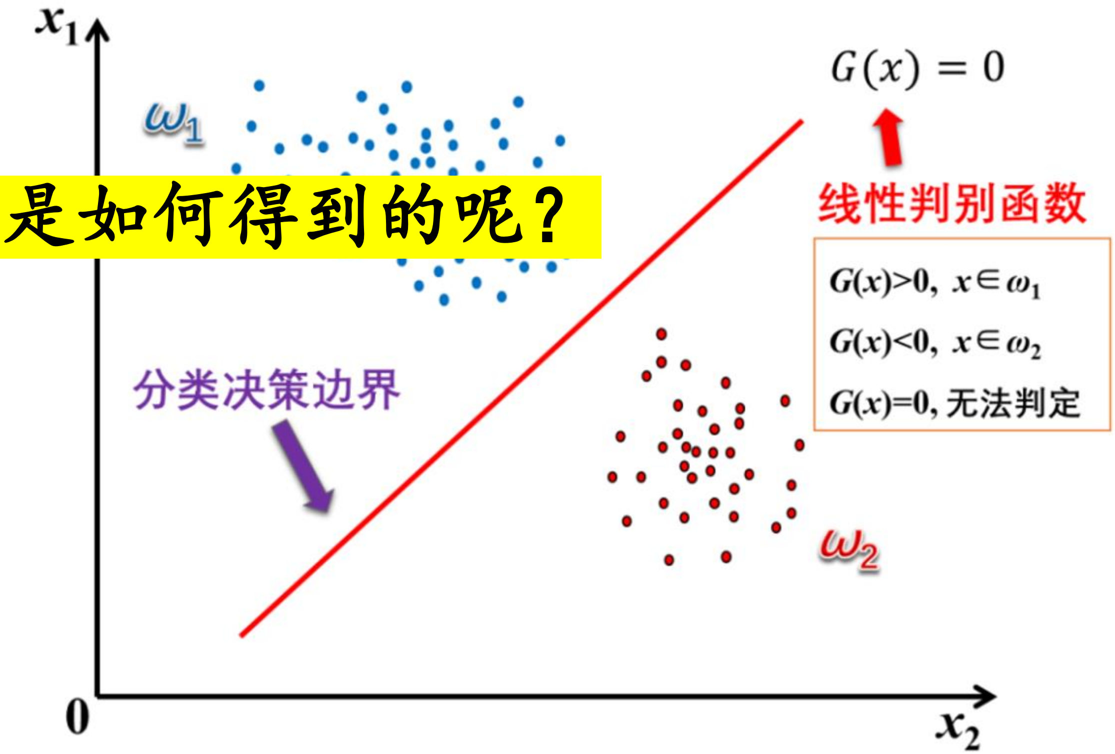
◆线性判别函数的几何意义小结：

- ➢样本到决策边界的距离正比于判别函数的绝对值
- ➢判别函数大于(或小于)0，表示该样本位于决策边界的正侧(或负侧)
- ➢权向量与决策边界的法线方向一致，其长度不会影响决策边界在特征空间中的方向，完全可以取$\|\mathbf{w}\|=1$

线性分类器-算法前言

回顾： 线性分类器由线性判别函数和相应的分类决策规则构成。

Q1: 线性判别函数是如何得到的呢？

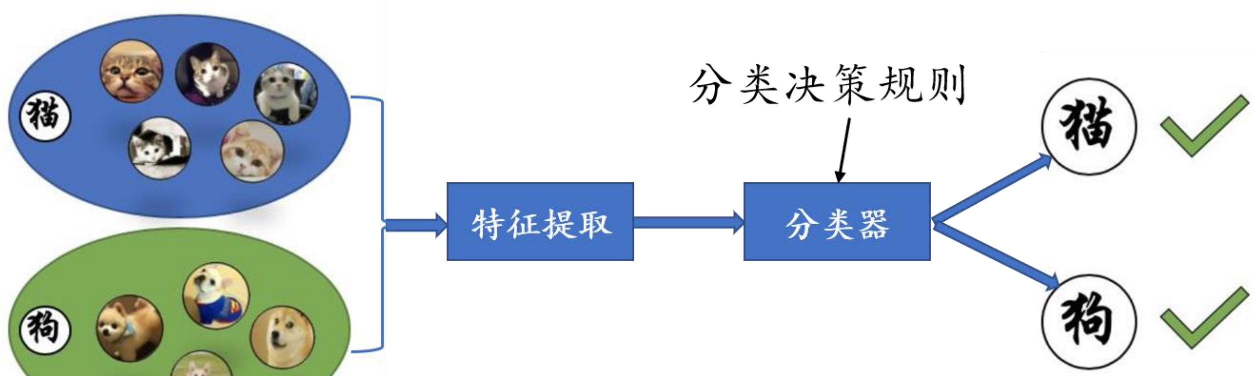


Q2: 一个模式识别问题，我们怎么设计出一个合适的线性分类器，使它能对未知样本进行正确分类呢？

线性分类器－算法前言

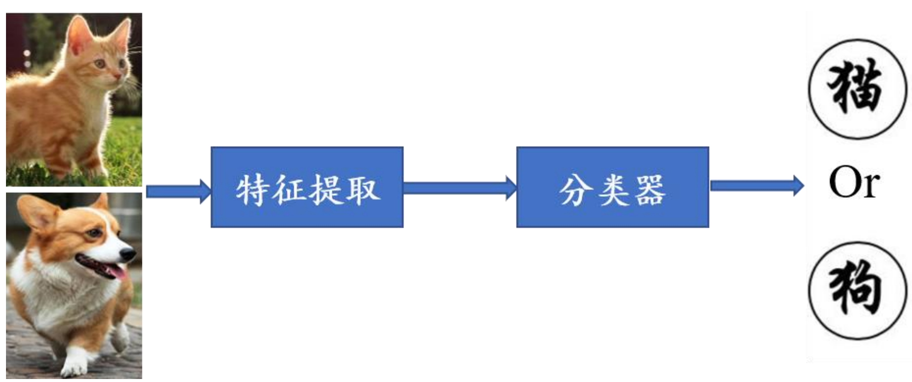
有监督的分类问题一般流程如下：

➢训练(学习)



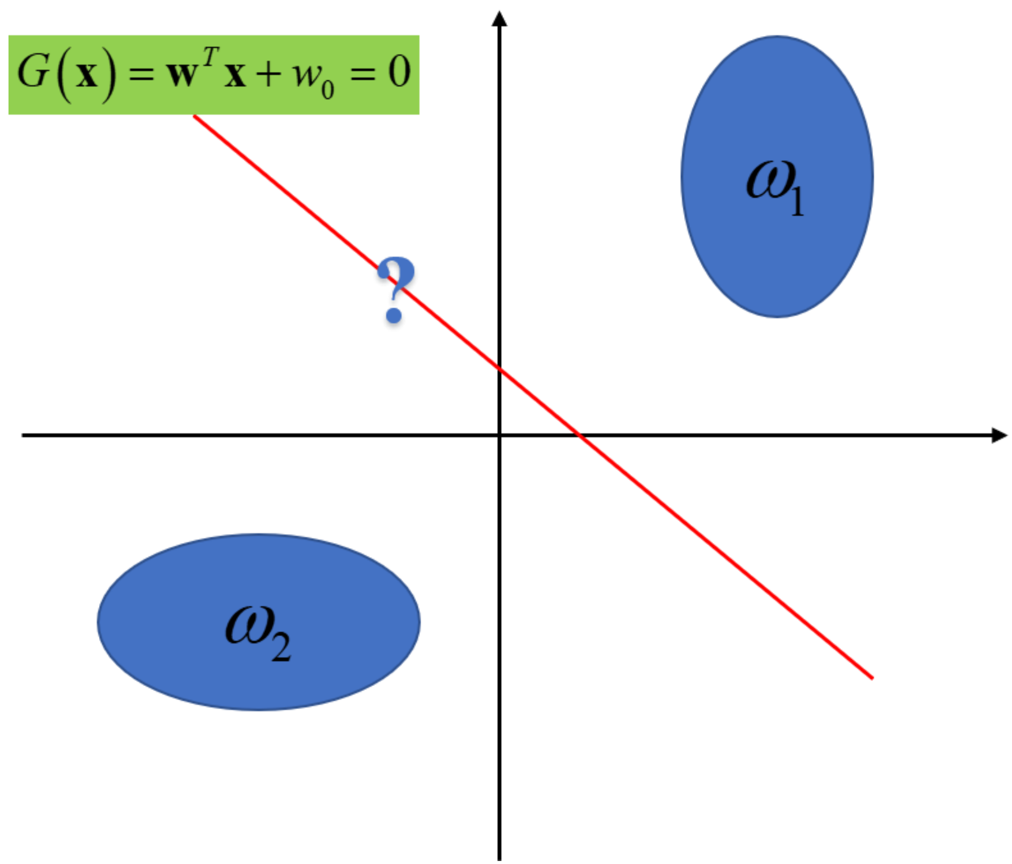
Q1:如何训练一个线性分类器呢？

➢测试

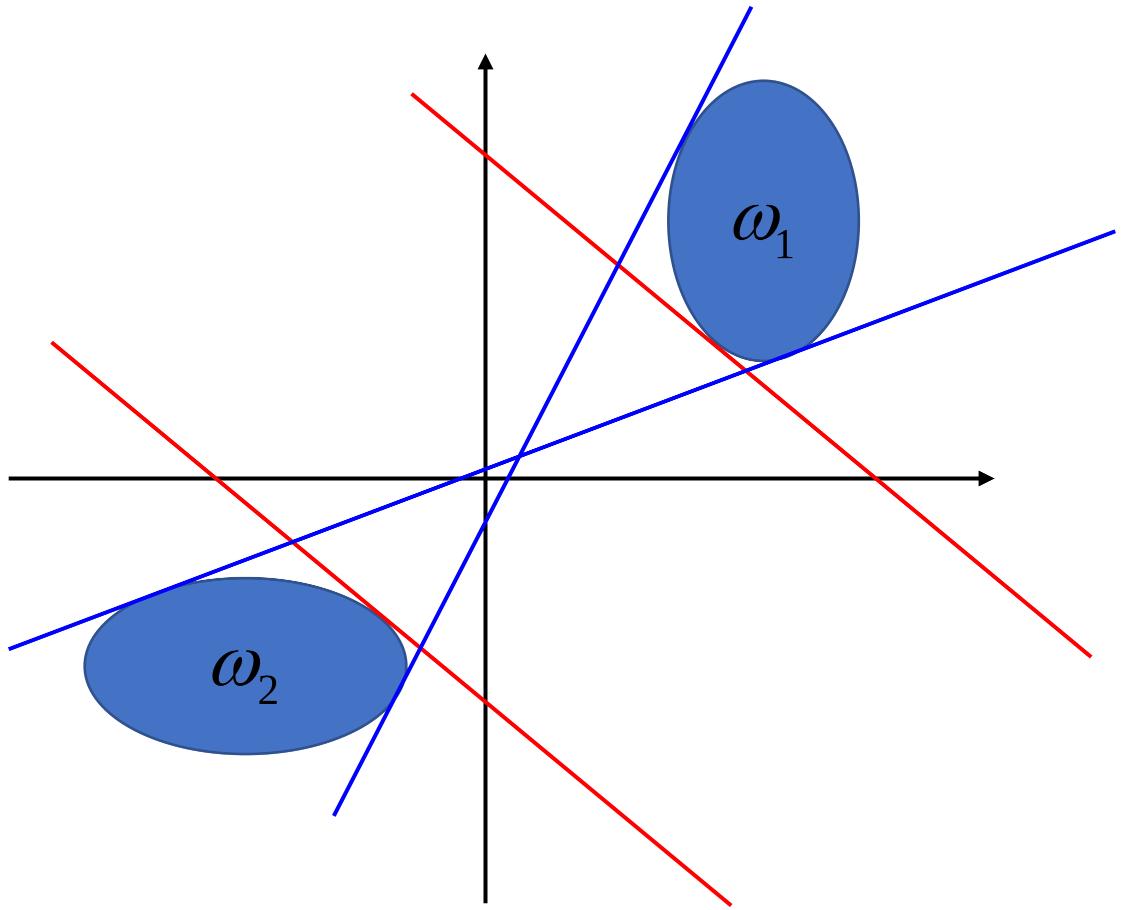


线性分类器-算法前言

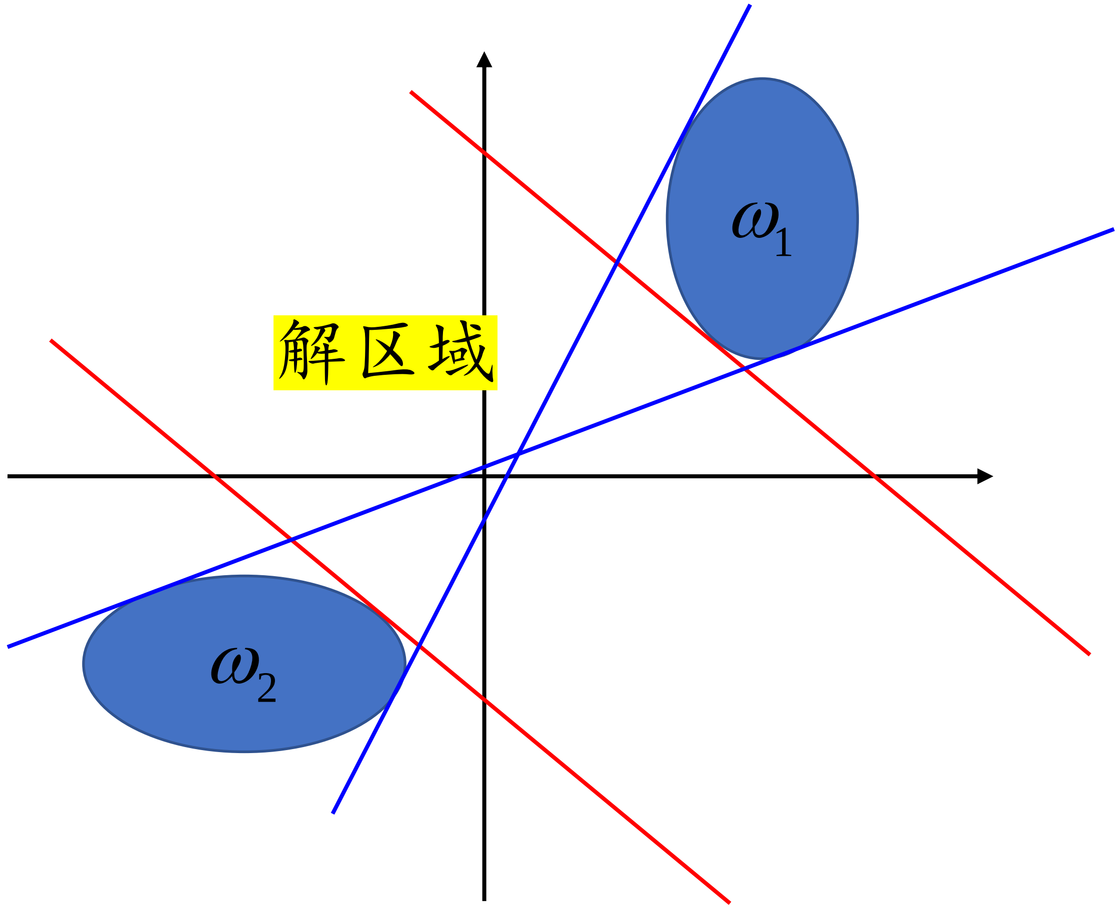
给定有标签的样本，训练获得$\mathbf{w}, w_0$



线性分类器-算法前言



线性分类器-算法前言



线性分类器－算法前言

解区域的确定：

$$G(\mathbf{y})=\boldsymbol{\alpha}^T\mathbf{y}$$

$$\boldsymbol{\alpha}=[w_1,w_2,\ldots,w_n,w_0]^T$$

$$\mathbf{y}=[x_1,x_2,\ldots,x_n,1]^T$$

对于一组线性可分的样本$\{\mathbf{y}_1,\mathbf{y}_2,\mathbf{y}_3,\ldots,\mathbf{y}_N\}$，有：

$$G(\mathbf{y}_i)=\begin{cases}\boldsymbol{\alpha}^T\mathbf{y}_i>0 & \mathbf{y}_i\in\omega_1\\ \boldsymbol{\alpha}^T\mathbf{y}_i<0 & \mathbf{y}_i\in\omega_2\end{cases}$$

令

$$\mathbf{y}_i^{'}=\begin{cases}\mathbf{y}_i & \mathbf{y}_i\in\omega_1\\ -\mathbf{y}_i & \mathbf{y}_i\in\omega_2\end{cases}$$

书中为了方便，把y'仍然记作y

$$\boldsymbol{\alpha}^T\mathbf{y}_i>0,(i=1,2,\ldots,N)$$

规范化增广样本向量

线性分类器－算法前言

对于一组线性可分的样本$\{\mathbf{y}_1, \mathbf{y}_2, \mathbf{y}_3, \ldots, \mathbf{y}_N\}$，如果一个权向量 $\boldsymbol{\alpha}^*$ 满足：

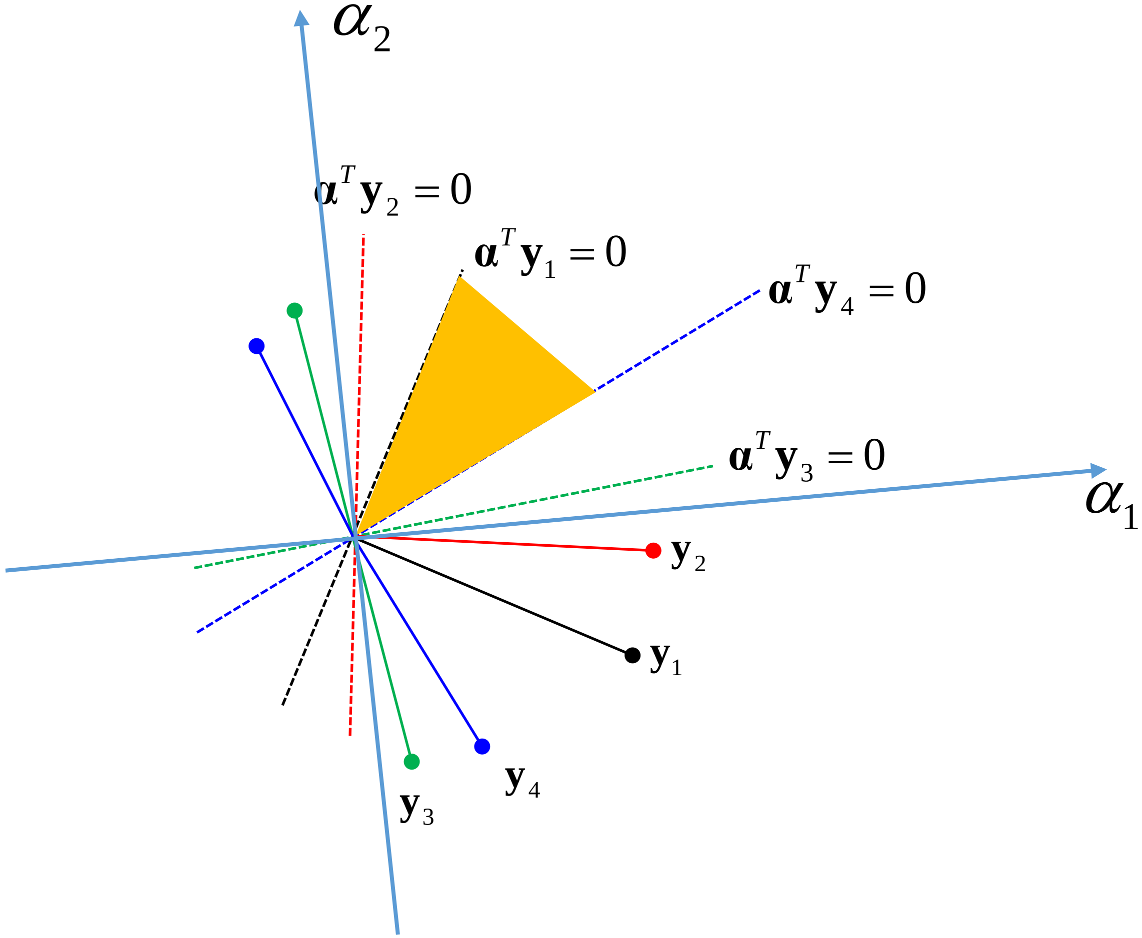
$$\left(\boldsymbol{\alpha}^*\right)^T \mathbf{y}_i > 0, (i = 1, 2, \ldots, N)$$

则称 $\boldsymbol{\alpha}^*$ 为一个解向量。

- 对于一个样本 $\mathbf{y}_i$，$\boldsymbol{\alpha}^T \mathbf{y}_i = 0$ 定义了**权空间(见补充材料)**中一个过原点的超平面。
- 处于超平面正侧的任何一个向量都能使 $\boldsymbol{\alpha}^T \mathbf{y}_i > 0$。
- 所有样本对应超平面的正侧的交集就是解区域。

线性分类器－算法前言

$$\begin{cases} \mathbf{y}_1, \mathbf{y}_2 \in \omega_1 \\ \mathbf{y}_3, \mathbf{y}_4 \in \omega_2 \end{cases}$$



线性分类器－算法前言

线性分类器训练一般思路如下：

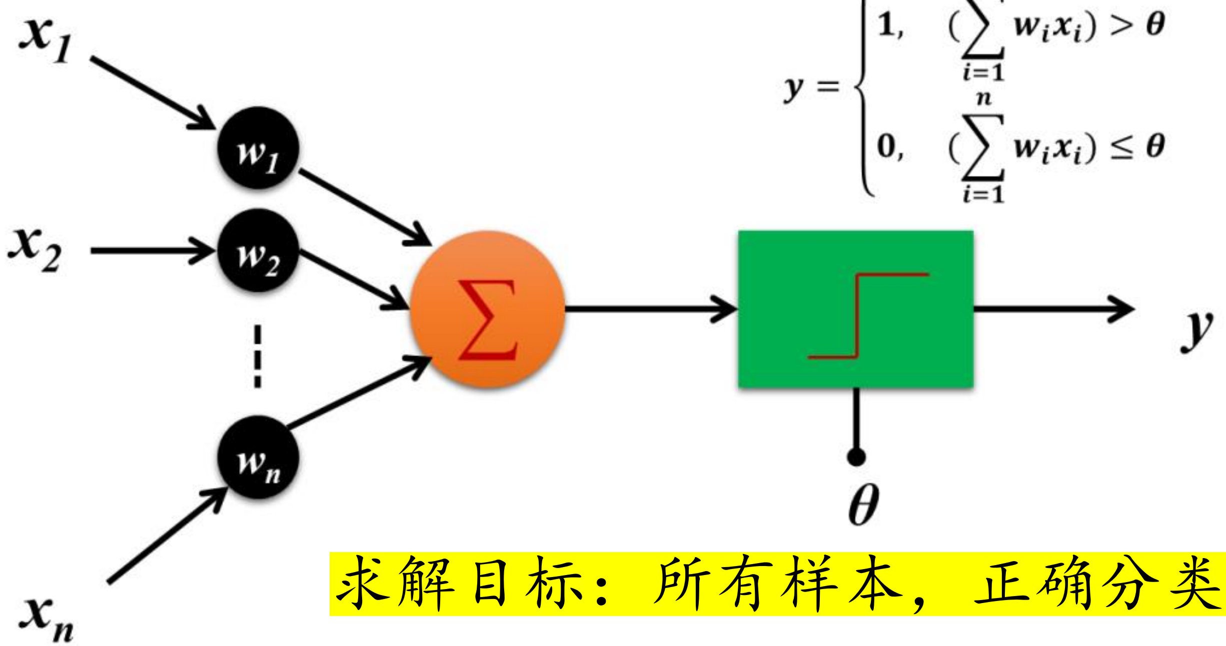
- 首先设定一个标量的**准则函数**，使其值能够代表解的优劣程度，准则函数值越小，说明越符合要求，越好；
- 通过寻找准则函数的极小值，就能找到最优的一个解。

注意：准则函数和判别函数的区别

线性分类器－感知器算法

感知器：

1957年，美国计算机科学家罗森布拉特(F. Rosenblatt)提出的。是一种神经元模型，一个神经元的“树突”搜集的信号，达到一定的激发阈值，就会激活“轴突”的输出。



$$y = \begin{cases} 1, & (\sum_{i=1}^{n} w_i x_i) > \theta \\ 0, & (\sum_{i=1}^{n} w_i x_i) \le \theta \end{cases}$$

求解目标：所有样本，正确分类

线性分类器-感知器算法

准则函数：所有错分样本的判别函数值之和，再乘以(-1)，

【自学】：

1. 什么是梯度下降算法？
2. 为什么梯度下降要选负梯度的方向？
3. 每次移动的步长(学习率)能不能选太大？

学习网址：

https://blog.csdn.net/pengchengliu/article/details/80932232

$$J_p(\boldsymbol{\alpha}) = \sum_{\boldsymbol{\alpha}^T \mathbf{y}_i \le 0} \left(-\boldsymbol{\alpha}^T \mathbf{y}_i\right)$$

$$\boldsymbol{\alpha}^* = \arg\min_{\boldsymbol{\alpha}} J_p(\boldsymbol{\alpha})$$

梯度下降算法

$$\boldsymbol{\alpha}(t+1) = \boldsymbol{\alpha}(t) - \rho_t \nabla J_p(\boldsymbol{\alpha})$$

$$\nabla J_p(\boldsymbol{\alpha}) = -\sum_{\boldsymbol{\alpha}^T \mathbf{y}_i \le 0} \mathbf{y}_i$$

$$\boldsymbol{\alpha}(t+1) = \boldsymbol{\alpha}(t) + \rho_t \sum_{\boldsymbol{\alpha}^T \mathbf{y}_i \le 0} \mathbf{y}_i$$

线性分类器－感知器算法

所有样本

感知器算法具体步骤：

(1) 任意选择初始的权向量$\boldsymbol{\alpha}(0)$，置 $t=0$

(2) 修正权向量：$\boldsymbol{\alpha}(t+1)=\boldsymbol{\alpha}(t)+\rho_t\sum_{\boldsymbol{\alpha}^T\mathbf{y}_i\le 0}\mathbf{y}_i$

(3)$t=t+1$，重复(2)，直至对所有样本都有$\boldsymbol{\alpha}(t)^T\mathbf{y}>0$

单样本

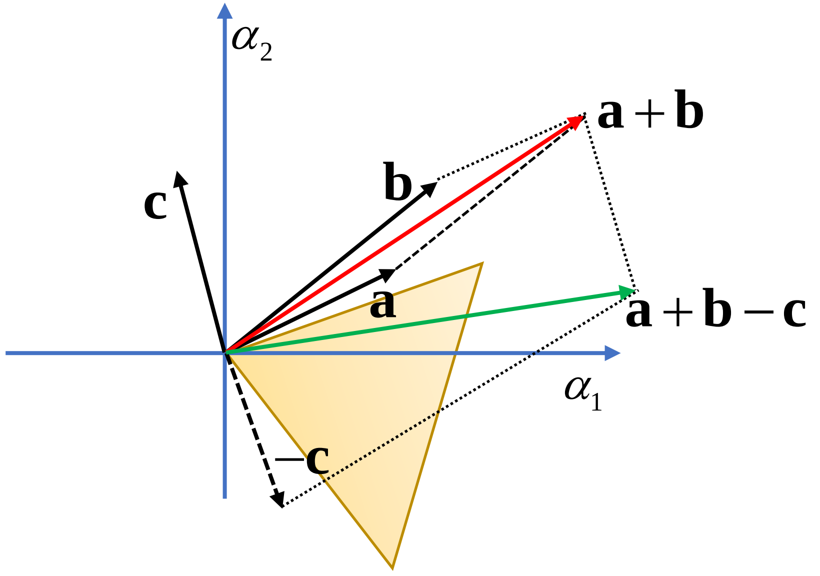
感知器算法具体步骤：

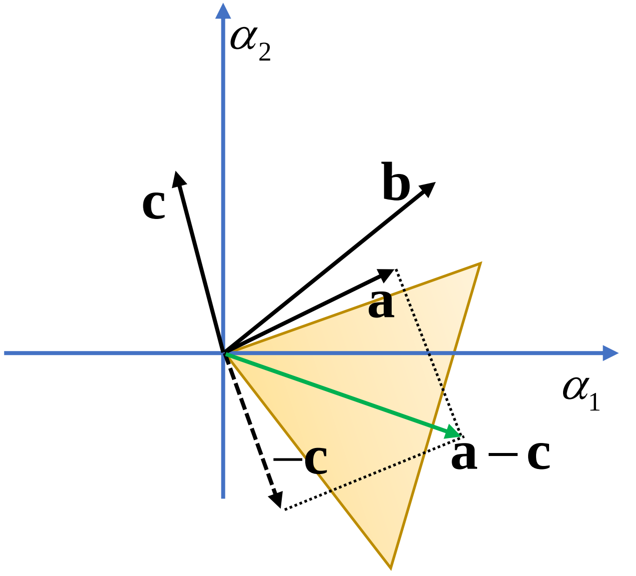
(1) 任意选择初始的权向量$\boldsymbol{\alpha}(0)$，置 $t=0$

(2) 考查样本$\mathbf{y}_i$，若$\boldsymbol{\alpha}(t)^T\mathbf{y}_i\le 0$，则 $\boldsymbol{\alpha}(t+1)=\boldsymbol{\alpha}(t)+\rho_t\mathbf{y}_i$

(3) 考察另一个样本，重复(2)，直至所有样本都正确分类

线性分类器-感知器算法





$$\boldsymbol{\alpha}(0)=(0,0),\ \rho=1$$
$$\boldsymbol{\alpha}(1)=\mathbf{a}+\mathbf{b}-\mathbf{c}$$

$$\boldsymbol{\alpha}(0)=(0,0),\ \rho=1$$
$$\boldsymbol{\alpha}(1)=\mathbf{a}$$
$$\boldsymbol{\alpha}(2)=\mathbf{a}-\mathbf{c}$$

Q1: 如何评判哪一个解更优呢?

线性分类器－感知器算法

Q:对于线性可分的样本集，感知器算法一定能找到最优解吗？

可以证明，对于线性可分的样本集，采用梯度下降的迭代算法，经过有限次修正后一定会收敛到一个解向量。

$$\boldsymbol{\alpha}(0) = \mathbf{0}$$

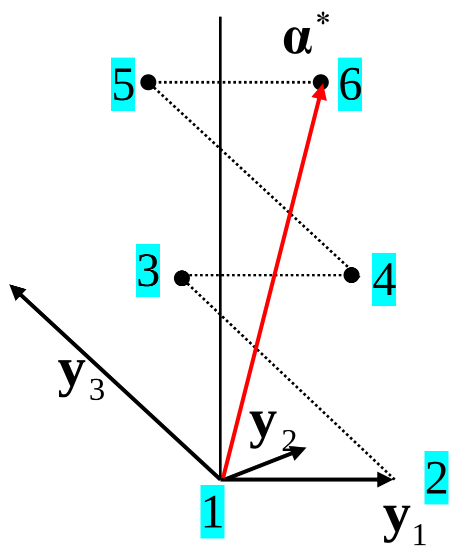
$$\boldsymbol{\alpha}(1) = \mathbf{0} + \mathbf{y}_1$$

$$\boldsymbol{\alpha}(2) = \boldsymbol{\alpha}(1) + \mathbf{y}_3$$

$$\boldsymbol{\alpha}(3) = \boldsymbol{\alpha}(2) + \mathbf{y}_1$$

$$\boldsymbol{\alpha}(4) = \boldsymbol{\alpha}(3) + \mathbf{y}_3$$

$$\boldsymbol{\alpha}(5) = \boldsymbol{\alpha}(4) + \mathbf{y}_1$$



线性分类器－感知器算法

例题：已知两类训练样本，(0,0), (0,1)属于w1, (1,0), (1,1)属于w2, 试用感知器算法求$\boldsymbol{\alpha}^*$

解：

1. 样本增广化和规范化：y1=(0,0,1)', y2=(0,1,1)', y3=(-1,0,-1)', y4=(-1,-1,-1)'

2. 权向量初始化$\boldsymbol{\alpha}(0)=\mathbf{0}$，学习速率$\rho=1$

$$t=1, \boldsymbol{\alpha}(0)^T\mathbf{y}_1=0, \boldsymbol{\alpha}(1)=\mathbf{0}+\mathbf{y}_1=(0,0,1)^T$$

$$t=2, \boldsymbol{\alpha}(1)^T\mathbf{y}_2=1>0, \boldsymbol{\alpha}(2)=\boldsymbol{\alpha}(1)=(0,0,1)^T$$

$$t=3, \boldsymbol{\alpha}(2)^T\mathbf{y}_3=-1<0, \boldsymbol{\alpha}(3)=\boldsymbol{\alpha}(2)+\mathbf{y}_3=(-1,0,0)^T$$

$$t=4, \boldsymbol{\alpha}(3)^T\mathbf{y}_4=1>0, \boldsymbol{\alpha}(4)=\boldsymbol{\alpha}(3)=(-1,0,0)^T$$

$$\vdots$$

线性分类器－感知器算法

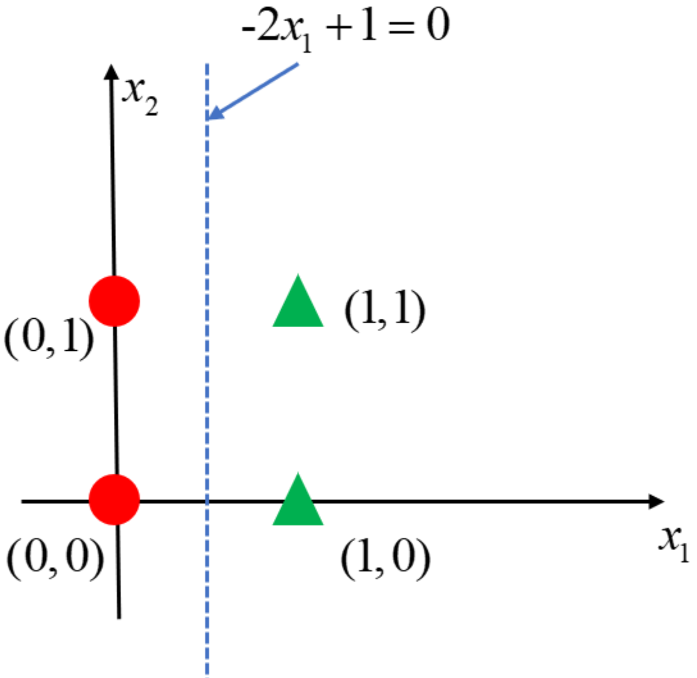
权向量： $\boldsymbol{\alpha}^* = (-2,0,1)$

决策边界方程为：$-2x_1 + 0 \cdot x_2 + 1 = 0$

【作业】

$\boldsymbol{\alpha}(0) = (1,1,1),\ \rho=2$

给出最优权向量和决策边界的方程，并给出图形表示。



线性分类器-

```
%%
Y = [0 0 1;
    0 1 1;
    -1 0 -1;
    -1 -1 -1];
p = 1;% learning rate
a_cur = [0 0 0];
while(1)
    c = 0;
    for i = 1:size(Y,1)
        tmp = a_cur(end,:)*Y(i,:)';
        if tmp>0
            c = c + 1;
        else
            a_t = a_cur(end,:) + p*Y(i,:);
            a_cur = [a_cur;a_t];
        end
    end
    if c == size(Y,1)
        break;
    end
end
```

案例：字写数字的识别

基于Mnist数据集，请用感知器算法对手写数字“8”和“6”进行识别。

线性分类器 - 感知器算法

【说明】

1. 数据集(提供，去学习委员那里copy)
2. 数据集结构：

名称	值
test_images	28x28x10000 double
test_labels1	1x10000 double
train_images	28x28x60000 double
train_labels1	1x60000 double

3. Label: 范围[0,9]，分别对应数字0-9
4. 图像大小是28*28，需要转成向量(提取特征或直接拉成784维的向量)
5. 训练感知器是基于train_images和train_labels1,测试时用测试图像

线性分类器-感知器算法

【要求】

1. 编程语言：Matlab (或Python)
2. 不能使用额外的库函数，自己编写实现感知器算法。
3. 采用基于单样本(或集体样本)的梯度下降算法。
4. 调整学习率，总结算法的求解精度和速度，与学习率之间的关系。
5. 采用不同数量的训练样本(比如5000，3000，1000，500)训练感知器，对比测试结果。
6. 提交实验报告和源代码(**命名规则：感知器_学号_姓名，先发给助教，助教再发到邮箱：zhangjunchao_work@163.com**)
7. 作业迟交n天，本次作业分数乘以0.98^n。

线性分类器－感知器算法

小结：

1.权向量的解区域：

- 对于一个样本$\mathbf{y}_i$，$\boldsymbol{\alpha}^T\mathbf{y}_i=0$定义了**权空间**中一个过原点的超平面。
- 处于超平面正侧的任何一个向量都能使$\boldsymbol{\alpha}^T\mathbf{y}_i>0$。
- 所有样本对应超平面的正侧的交集就是解区域。

2. 感知器算法：

- 只对线性可分的问题有效。
- 不能在解区域中选出最优解。
- 不能对训练样本集是否线性可分作出判断。

线性分类器－感知器算法

线性分类器-感知器算法

权空间：

决策面方程：$\boldsymbol{\alpha}^T \mathbf{y} = 0 \qquad \mathbf{y} \in \mathbb{R}^n$

$$\Downarrow$$

$$\mathbf{y}^T \boldsymbol{\alpha} = 0$$

权向量全体构成一个n维的权向量空间，$\boldsymbol{\alpha}^T \mathbf{y} = 0$ 确定了通过权空间原点的超平面，法向量与 $\mathbf{y}$ 一致。