2023-2024学年第一学期本科生课程

《神经网络与深度学习》

第二节：线性模型

主讲人：戴金晟(副教授，博士生导师)

daijincheng@bupt.edu.cn

神经网络与深度学习课程组

北京郵電大學

Beijing University of Posts and Telecommunications

内容导览

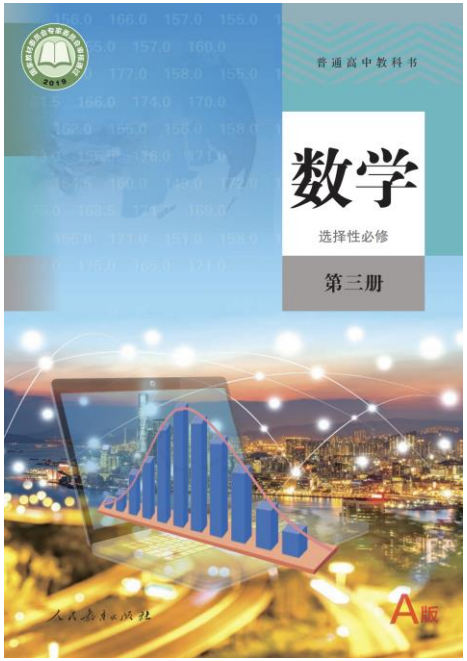
- 线性回归：多项式拟合
- 线性分类：概率生成模型
- Logistic回归：神经元模型
- 多分类问题
- 线性模型的局限与非线性模型

内容导览

- 线性回归：多项式拟合
- 线性分类：概率生成模型
- Logistic回归：神经元模型
- 多分类问题
- 线性模型的局限与非线性模型

线性回归

高中时期的经典问题



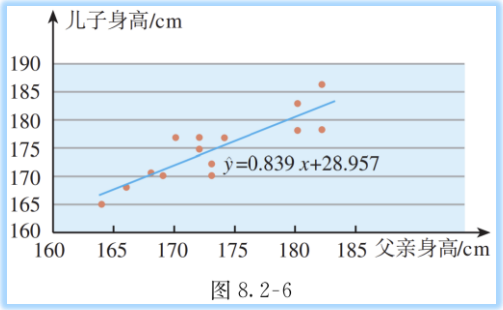


图 8.2-6

8.2 一元线性回归模型及其应用

通过前面的学习我们已经了解到，根据成对样本数据的散点图和样本相关系数，可以推断两个变量是否存在相关关系、是正相关还是负相关，以及线性相关程度的强弱等. 进

8.2.2 一元线性回归模型参数的最小二乘估计

在一元线性回归模型中，表达式 $Y=bx+a+e$ 刻画的是变量 Y 与变量 x 之间的线性相关关系，其中参数 a 和 b 未知，需要根据成对样本数据进行估计. 由模型的建立过程可知，参数 a 和 b 刻画了变量 Y 与变量 x 的线性关系，因此通过成对样本数据估计这两个参数，相当于寻找一条适当的直线，使表示成对样本数据的这些散点在整体上与这条直线最接近.

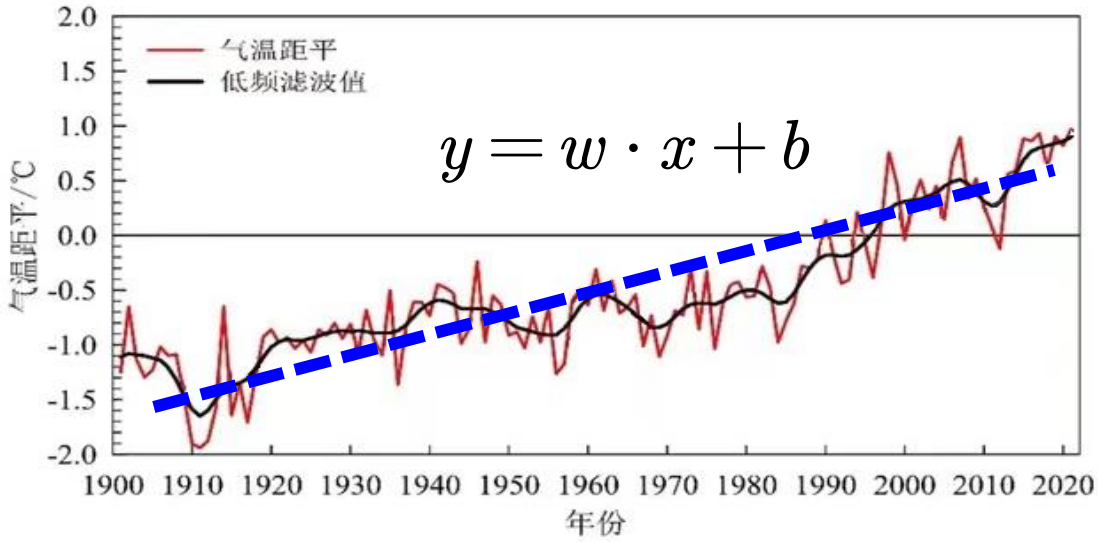
探究

利用散点图 8.2-1 找出一条直线，使各散点在整体上与此直线尽可能接近.

有的同学可能会想，可以采用测量的方法，先画出一条直线，测量出各点与它的距离，然后移动直线，到达一个使距离的和最小的位置. 测量出此时的斜率和截距，就可得到一条直线，如图 8.2-2 所示.

回归问题的解决

步骤1：定义一组函数集合 → 步骤2：定义函数好坏的标准 → 步骤3：选取最优的函数



$$L(w,b) = \frac{1}{2N}\sum_{i=1}^{N}\left(\hat{y}_i - (w \cdot x_i + b)\right)^2$$

高中的解法

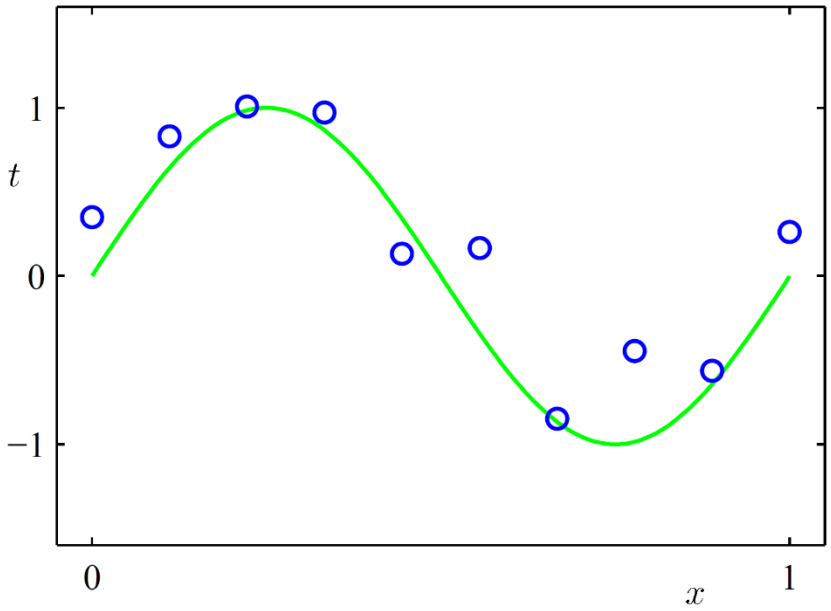
通过最小二乘法确定最优的参数 w 和 b

人工智能方法

利用梯度下降法求解参数

多项式拟合问题

步骤1：定义一组函数集合 → 步骤2：定义函数好坏的标准 → 步骤3：选取最优的函数



模型

$$y(x,\boldsymbol{\omega}) = \omega_0 + \omega_1 x + \omega_2 x^2 + \cdots + \omega_M x^M$$

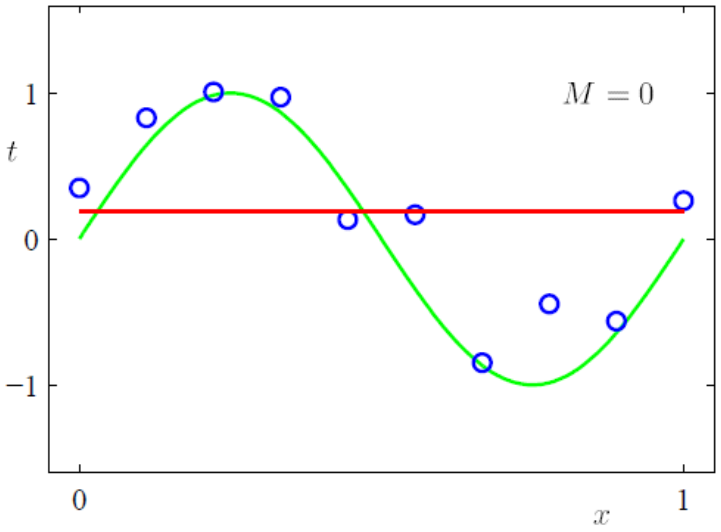
损失函数

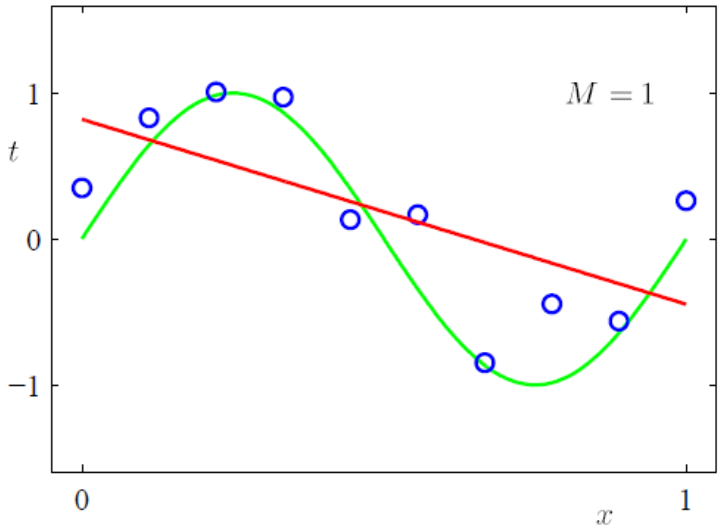
$$L(\boldsymbol{\omega}) = \frac{1}{2}\sum_{n=1}^{N}\{y(x_n,\boldsymbol{\omega}) - t_n\}^2$$

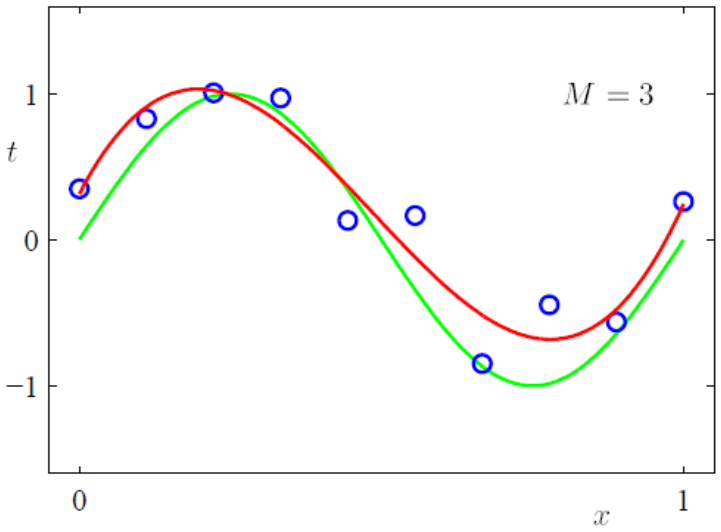
机器学习本质：利用数据样本，估计模型中的参数取值

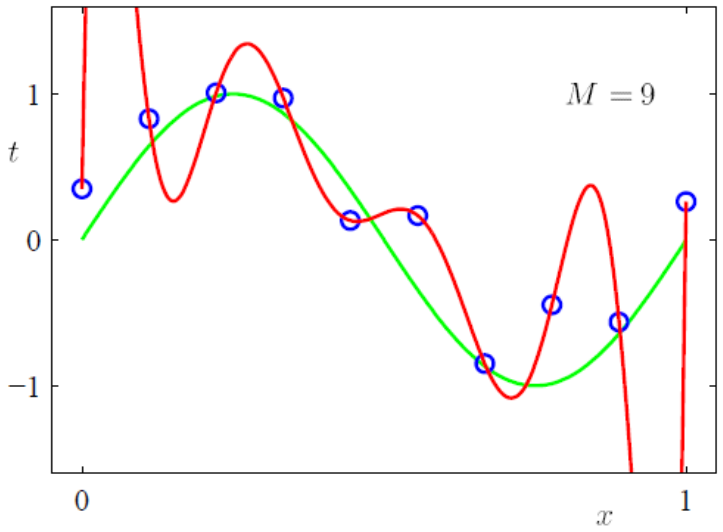
多项式拟合问题

如何在拟合能力和模型复杂度之间取得一个好的平衡





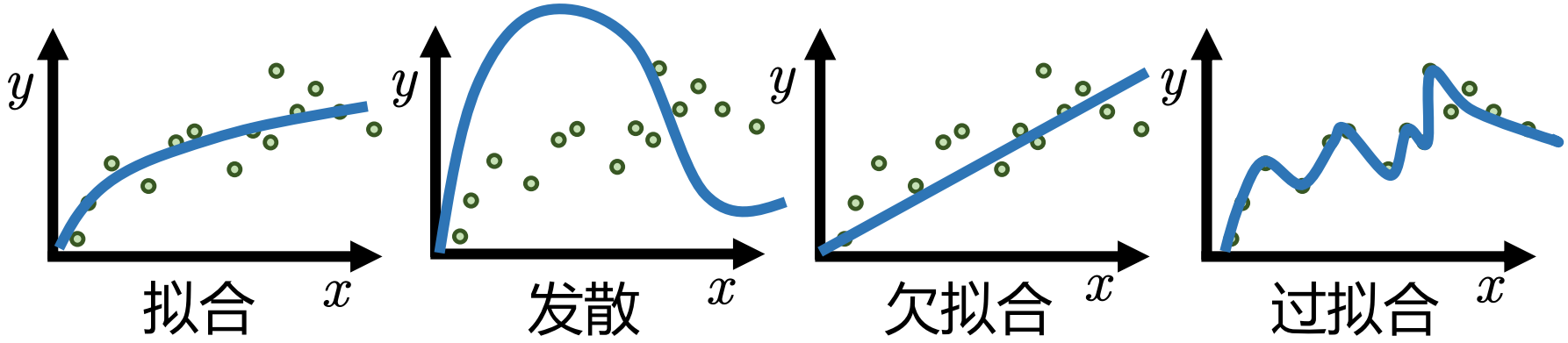




模型性能的评价

口机器学习的本质是找一个函数

利用样本估计模型参数



在测试集上，更复杂的模型并不总是会得到更好的性能

如何在拟合能力和复杂度之间取得一个好的平衡

模型选择

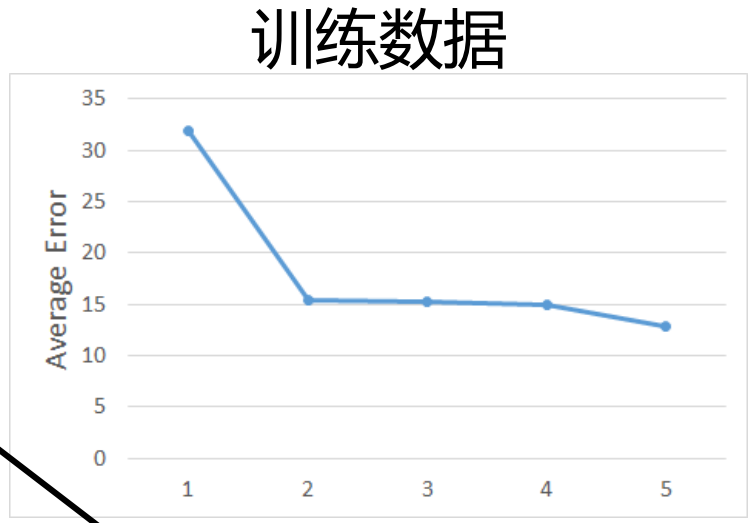
$y = b + w_1 \cdot x$

$y = b + w_1 \cdot x + w_2 \cdot x^2$

$y = b + w_1 \cdot x + w_2 \cdot x^2 + w_3 \cdot x^3$

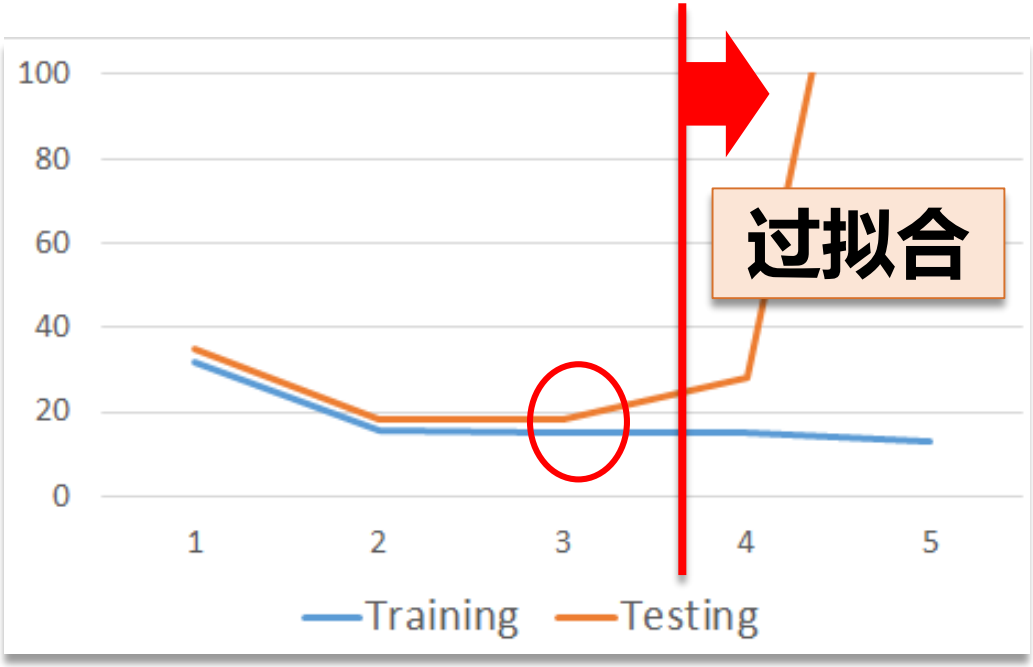
$y = b + w_1 \cdot x + w_2 \cdot x^2 + w_3 \cdot x^3 + w_4 \cdot x^4$

$y = b + w_1 \cdot x + w_2 \cdot x^2 + w_3 \cdot x^3 + w_4 \cdot x^4 + w_5 \cdot x^5$



在存在最优函数的情况下，更复杂的模型会降低训练数据的误差。

模型选择



最高阶数	训练集	测试集
1	31.9	35.0
2	15.4	18.4
3	15.3	18.1
4	14.9	28.2
5	12.8	232.1

更复杂的模型并不总是能够在测试数据上获得更好的性能

如何避免过拟合的情况

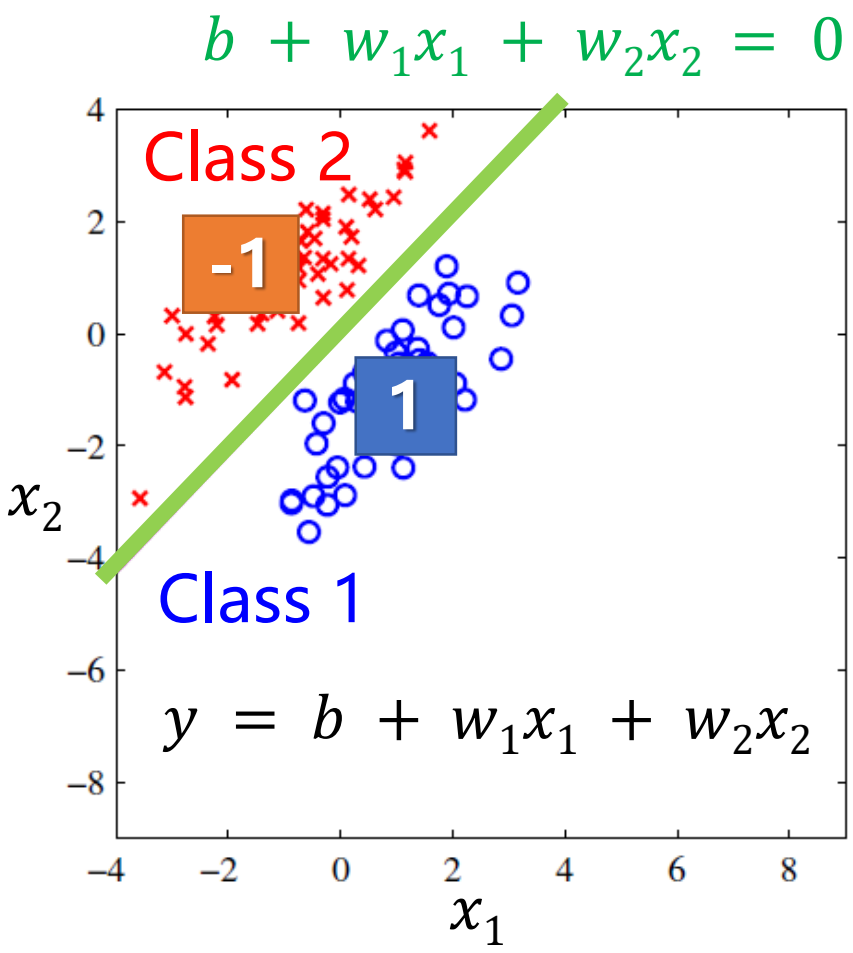
?

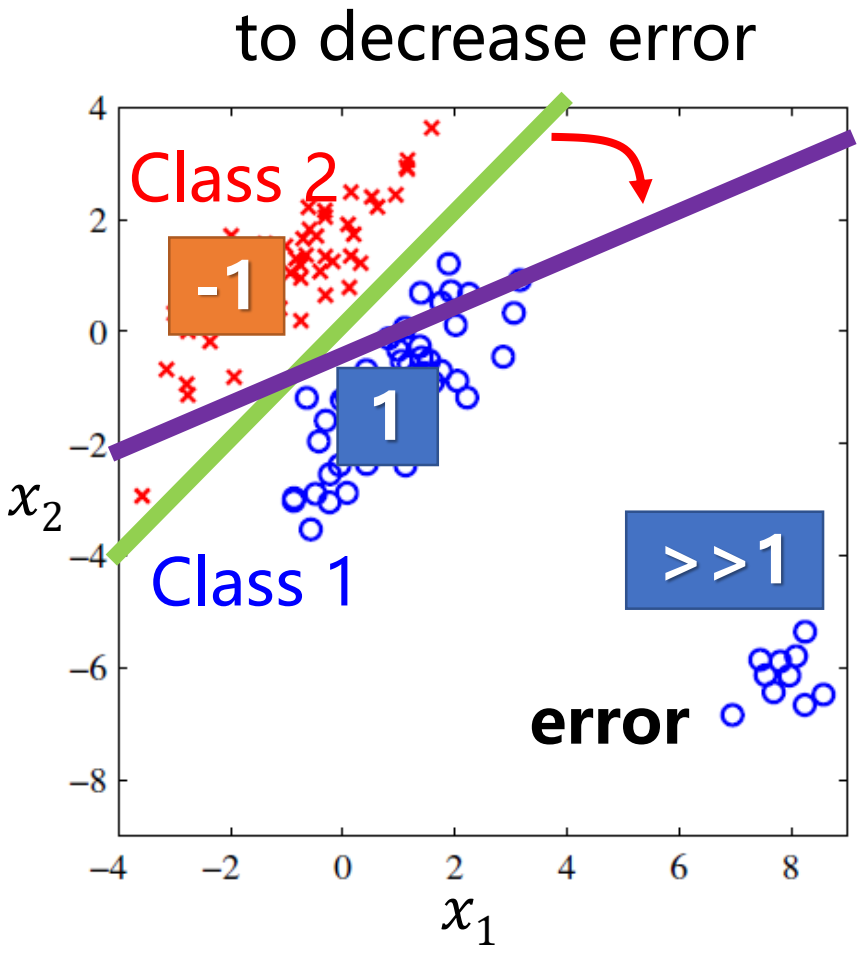
选择合适的模型

内容导览

- 线性回归：多项式拟合
- **线性分类：概率生成模型**
- Logistic回归：神经元模型
- 多分类问题
- 线性模型的局限与非线性模型

能否利用回归思路解决分类问题?



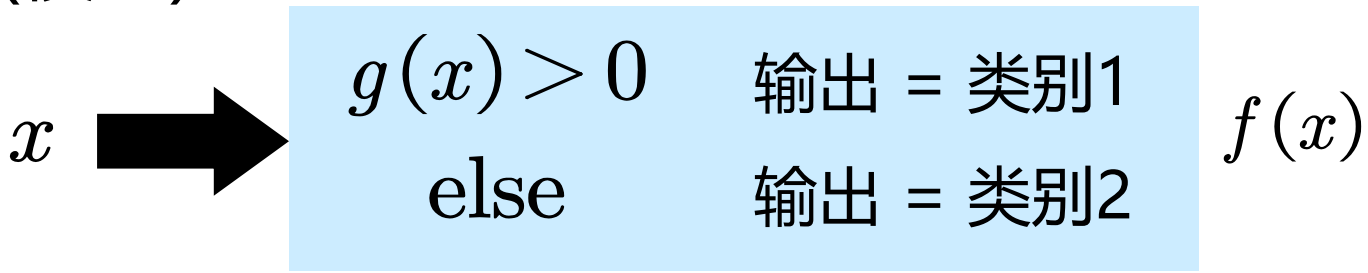


Penalize to the examples that are "too correct" ...

(Bishop, P186)

解决分类问题的三步骤

- 函数(模型)



- 模型度量(损失函数)

$$L(f)=\sum_{n}\delta\left(f(x^n)\neq\hat{y}^n\right)$$

f在训练数据上得到错误结果的次数

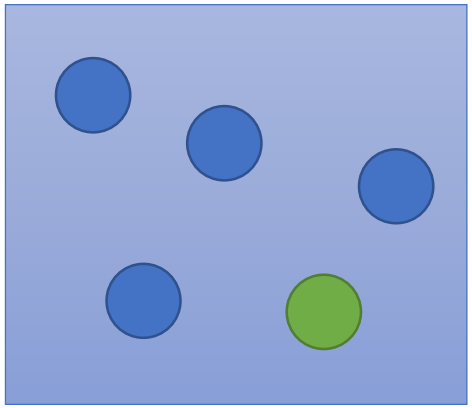
- 寻找最佳函数
  - 示例：感知器，SVM

不是当前使用的主流方法

概率生成问题

Box 1

$P(B_1) = 2/3$

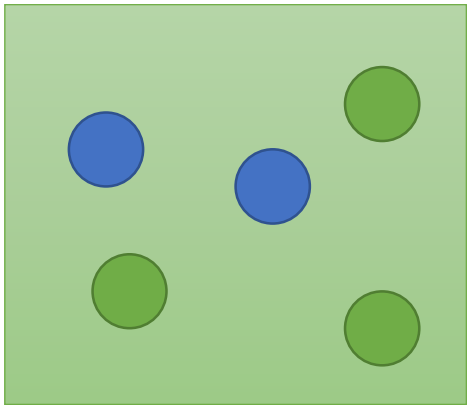


$P(Blue|B_1) = 4/5$

$P(Green|B_1) = 1/5$

Box 2

$P(B_2) = 1/3$



$P(Blue|B_2) = 2/5$

$P(Green|B_2) = 3/5$

- 是从某一个盒子取出的蓝色球

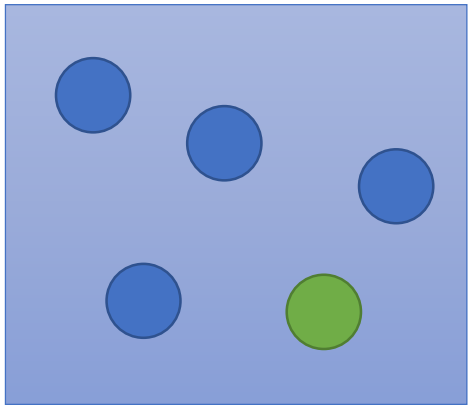
  是从哪个盒子取出?

$$P(B_1|Blue) = \frac{P(Blue|B_1)P(B_1)}{P(Blue|B_1)P(B_1) + P(Blue|B_2)P(B_2)}$$

分类问题实质：概率生成模型

Class 1

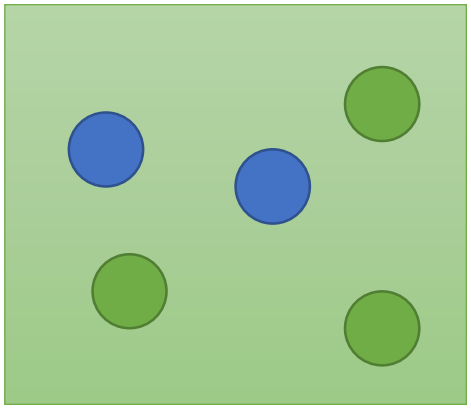
$P(C_1)$



$P(x|C_1)$

Class 2

$P(C_2)$



$P(x|C_2)$

对于给定样本x, 其属于某一类别的概率为

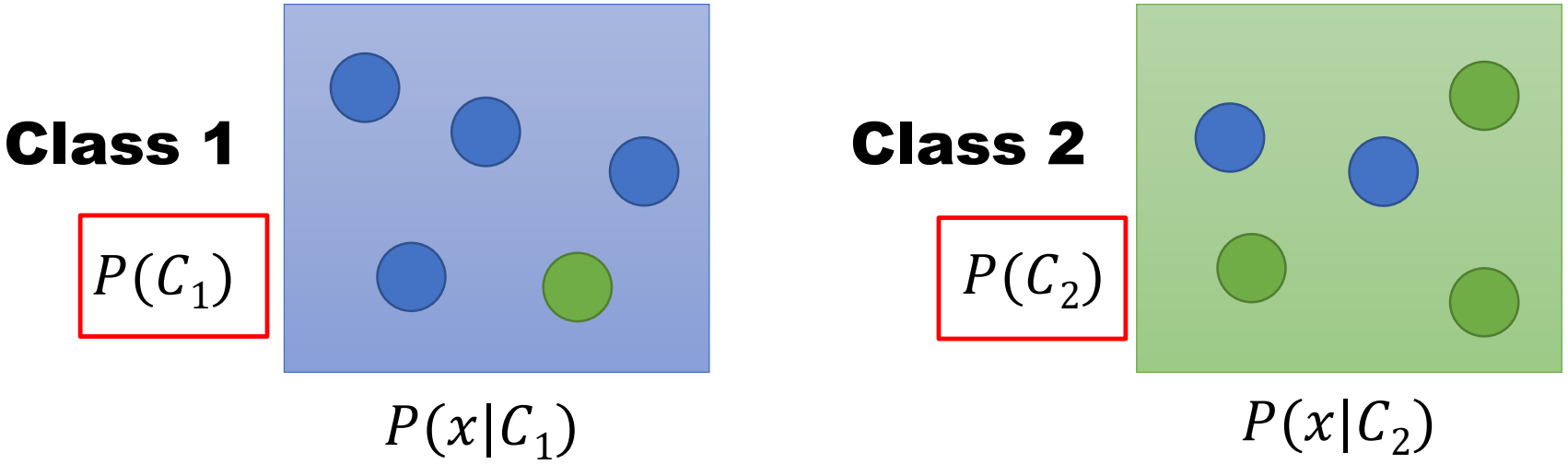
贝叶斯公式

$$P(C_1|x) = \frac{P(x|C_1)P(C_1)}{P(x|C_1)P(C_1) + P(x|C_2)P(C_2)}$$

概率生成模型：$P(x) = P(x|C_1)P(C_1) + P(x|C_2)P(C_2)$

生成模型：获得数据概率分布

概率生成模型：先验概率获取



以Pokémon分类为例，**水系(class 1)**和**常规系(class 2)**，**训练数据**包含79只水系Pokémon，61只常规系Pokémon

先验概率

$$P(C_1) = \frac{79}{79+61} = 0.56$$

$$P(C_2) = \frac{61}{79+61} = 0.44$$

新样本来自某一类的概率

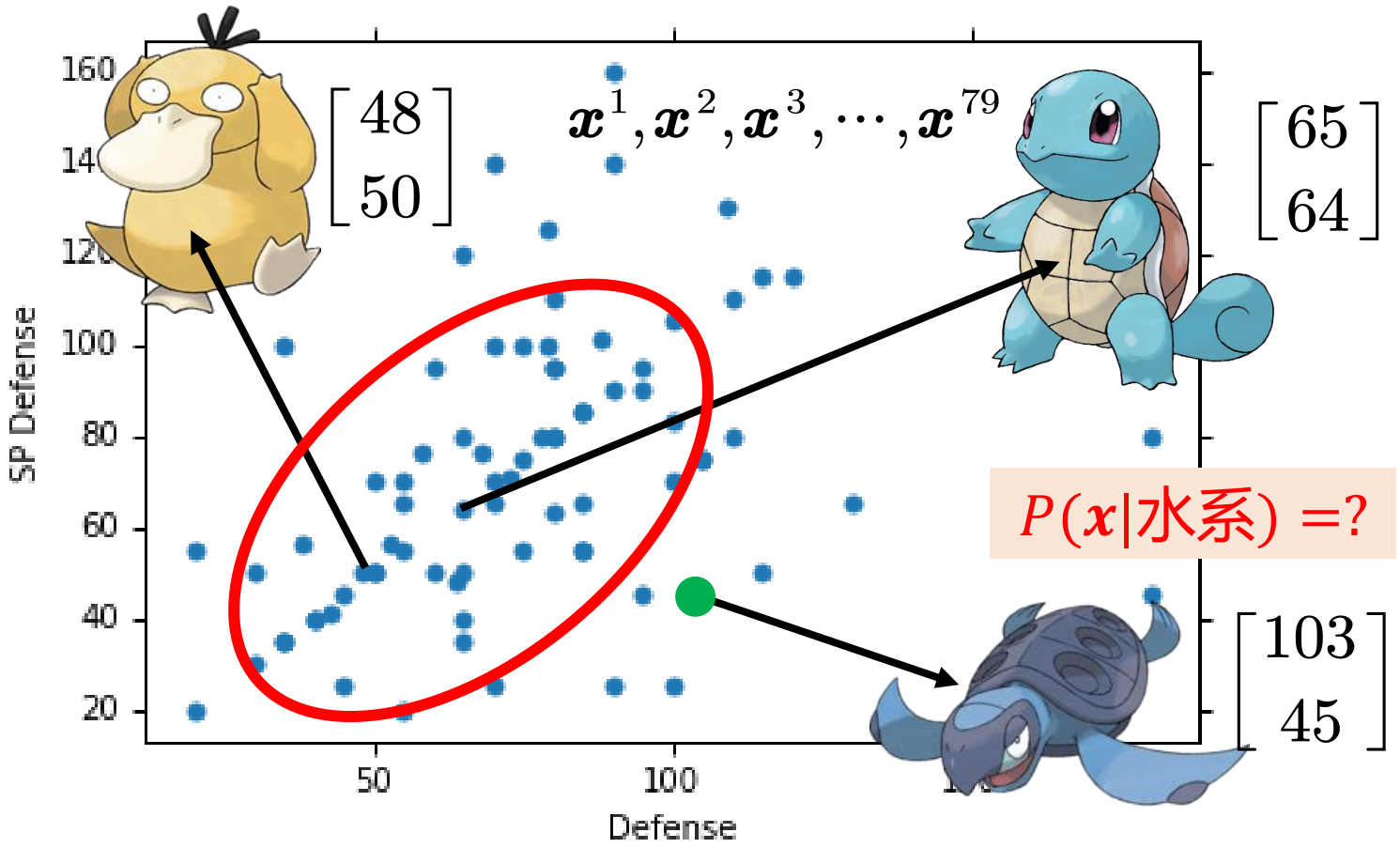
$P(x|C_1)=?$ $P($  $|$ 水系 $)=?$

每个Pokémon都通过其属性表示为**一个向量**

Pokémon属性组成的向量可看作是**特征**

- Pokémon实例源自李宏毅机器学习公开课

新样本来自某一类的概率



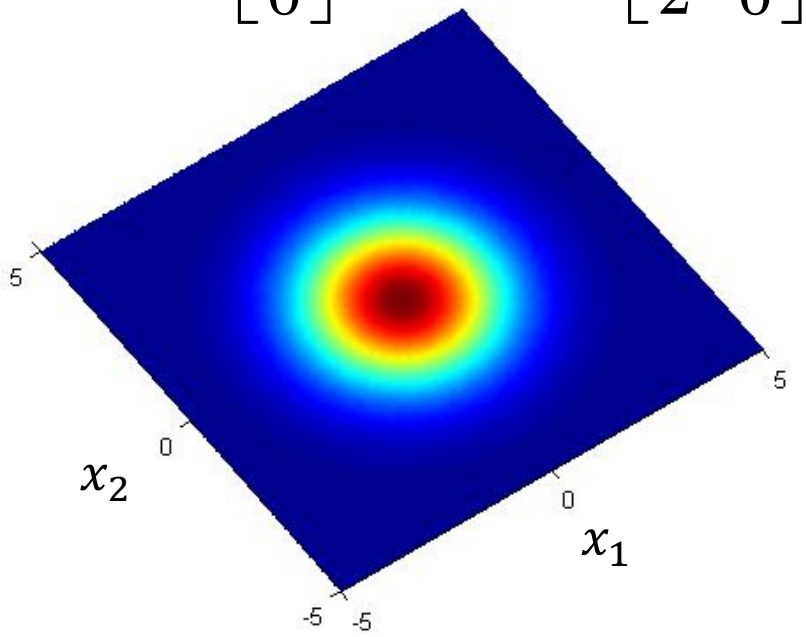
假设所有点采样自某个**高斯分布**

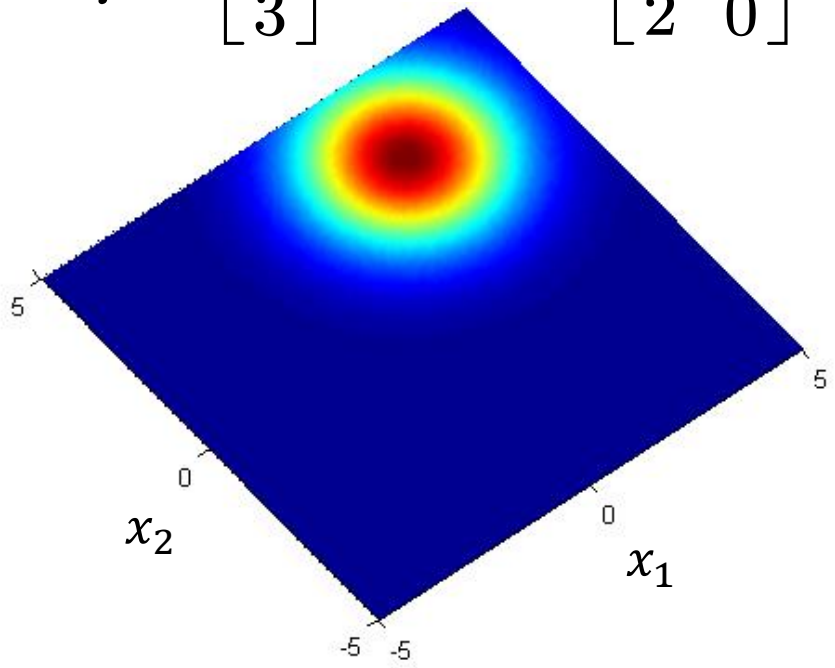
高斯分布

$$f_{\mu,\Sigma}(x)=\frac{1}{(2\pi)^{D/2}}\frac{1}{|\Sigma|^{1/2}}exp\left\{-\frac{1}{2}(x-\mu)^{T}\Sigma^{-1}(x-\mu)\right\}$$

- 输入：向量 x　　输出：抽样得到 x 的概率
- **均值** μ 和 **协方差** Σ 决定了函数的形状

$$\mu=\begin{bmatrix}0\\0\end{bmatrix}\quad\Sigma=\begin{bmatrix}0&2\\2&0\end{bmatrix}\qquad\mu=\begin{bmatrix}2\\3\end{bmatrix}\quad\Sigma=\begin{bmatrix}0&2\\2&0\end{bmatrix}$$



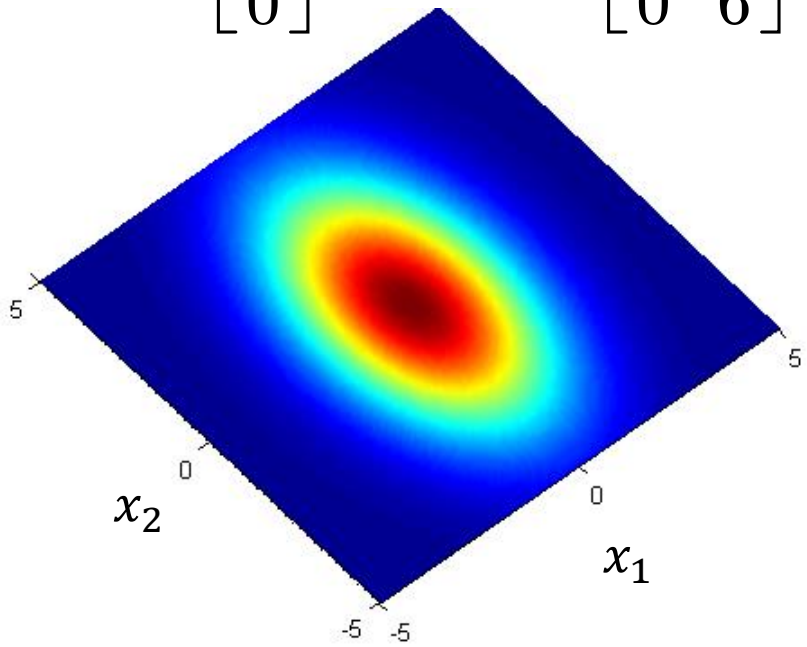


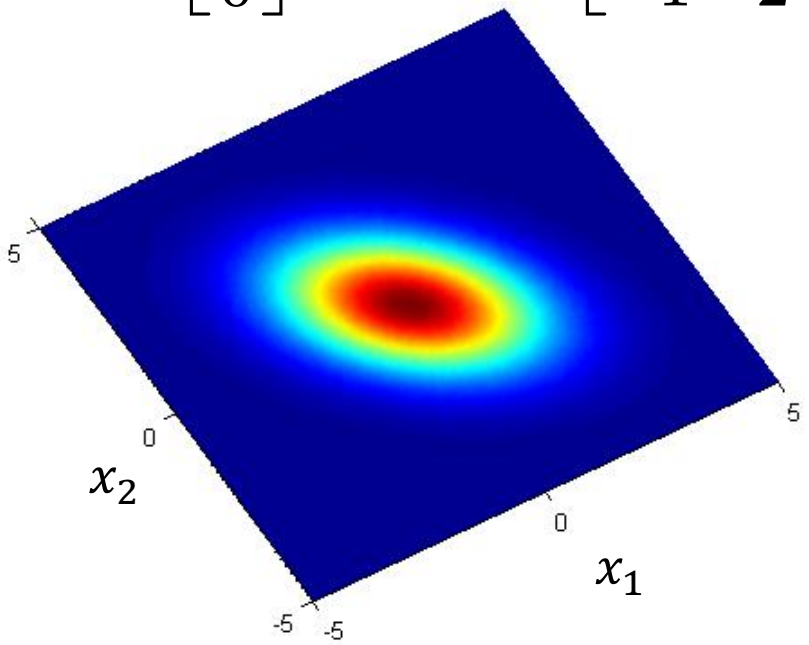
高斯分布

$$f_{\mu,\Sigma}(x)=\frac{1}{(2\pi)^{D/2}}\frac{1}{|\Sigma|^{1/2}}exp\left\{-\frac{1}{2}(x-\mu)^{T}\Sigma^{-1}(x-\mu)\right\}$$

- 输入：向量 x　　输出：抽样得到 x 的概率
- **均值** μ 和 **协方差** Σ 决定了函数的形状

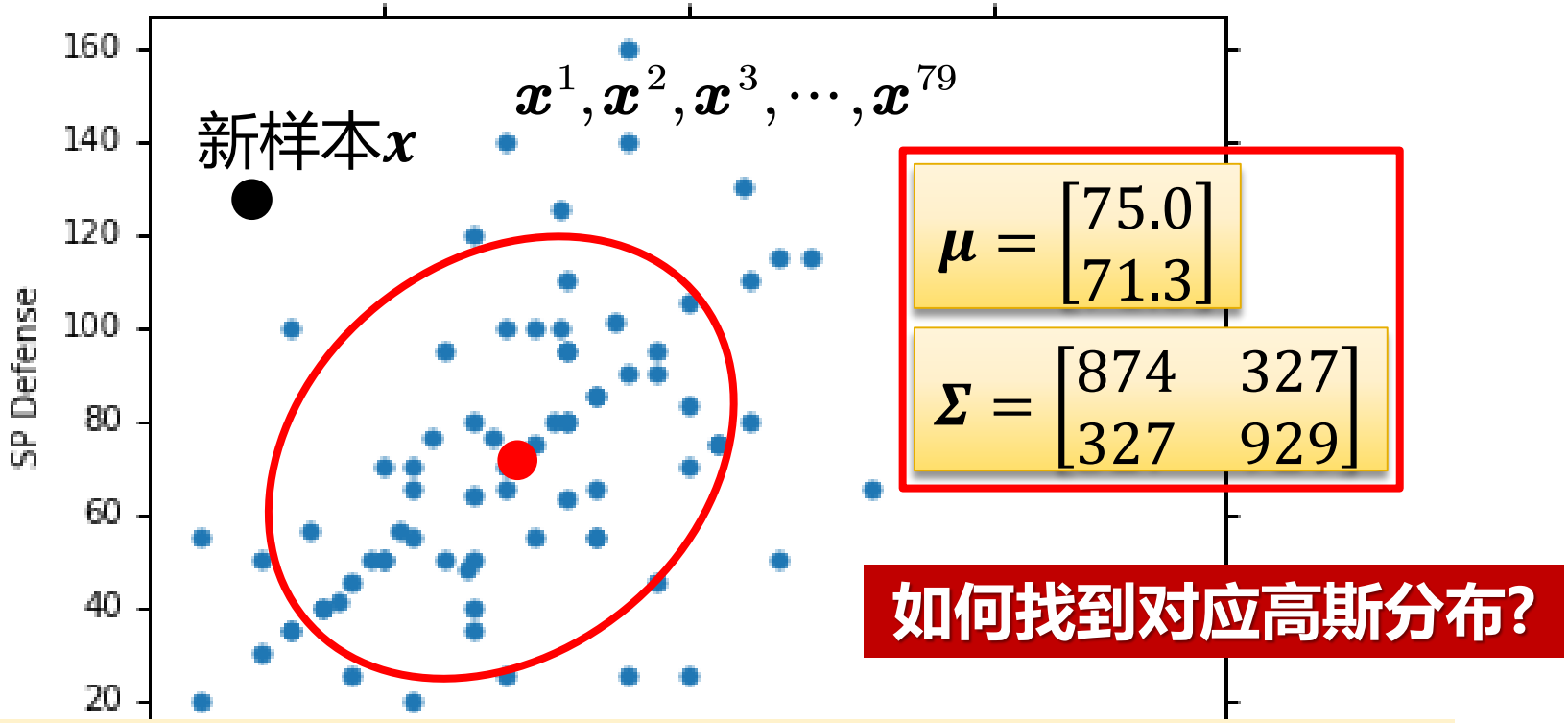
$$\mu=\begin{bmatrix}0\\0\end{bmatrix}\quad \Sigma=\begin{bmatrix}2&0\\0&6\end{bmatrix}\qquad \mu=\begin{bmatrix}0\\0\end{bmatrix}\quad \Sigma=\begin{bmatrix}2&-1\\-1&2\end{bmatrix}$$





新样本来自某一类的概率

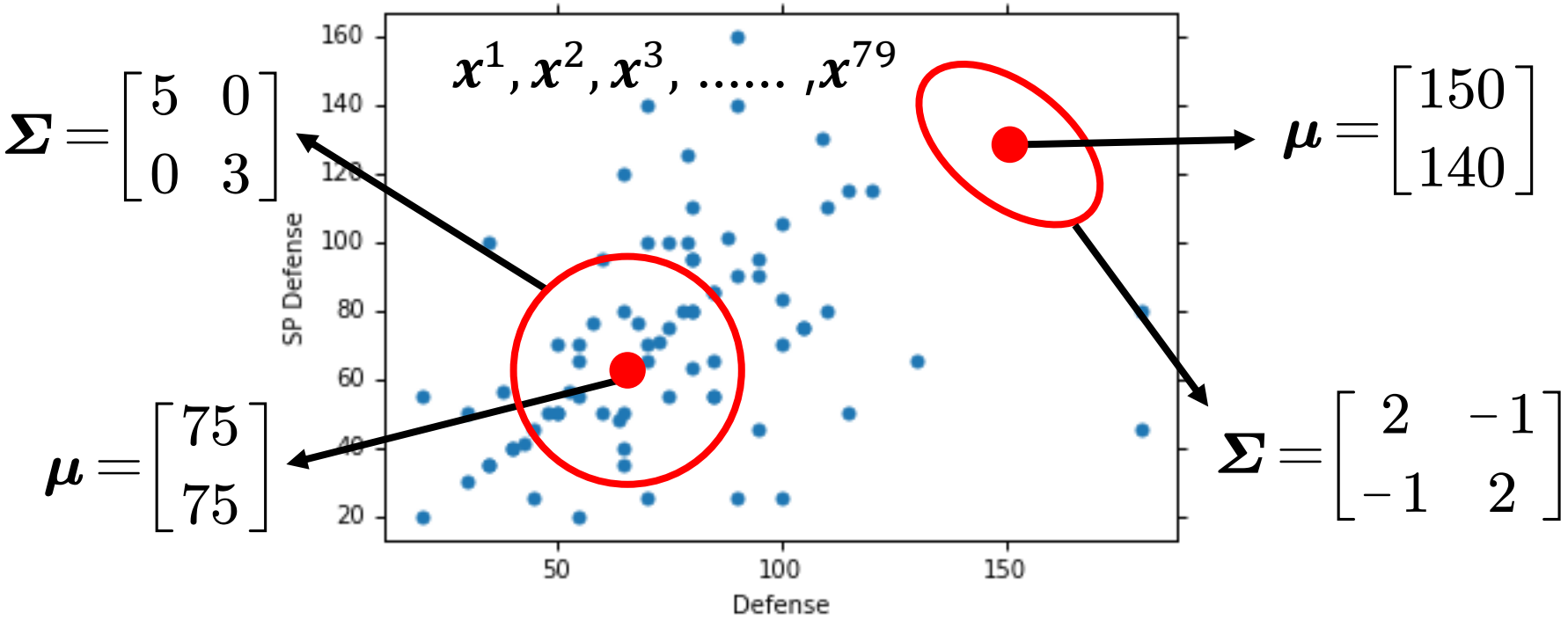
- 假设所有点均采样于某一高斯分布
  - 找出它们背后的高斯分布 ➡ **得到生成新样本的概率**



$$f_{\mu,\Sigma}(x) = \frac{1}{(2\pi)^{D/2}} \frac{1}{|\Sigma|^{1/2}} exp\left\{-\frac{1}{2}(x-\mu)^T \Sigma^{-1}(x-\mu)\right\}$$

最大似然估计

$$f_{\mu,\Sigma}(x)=\frac{1}{(2\pi)^{D/2}}\frac{1}{|\Sigma|^{1/2}}\exp\left\{-\frac{1}{2}(x-\mu)^T\Sigma^{-1}(x-\mu)\right\}$$



- 任意均值 μ 和任意协方差矩阵 Σ 所对应的高斯分布都能生成这些点，**只是生成的概率(似然函数)不同**
- 具有均值 μ 和协方差矩阵 Σ 的某个高斯分布的似然函数=生成高斯样本 $x^1, x^2, x^3, \ldots\ldots, x^{79}$ 的概率

$$L(\mu,\Sigma)=f_{\mu,\Sigma}(x^1)f_{\mu,\Sigma}(x^2)f_{\mu,\Sigma}(x^3)\cdots f_{\mu,\Sigma}(x^{79})$$

最大似然估计

- 训练样本：水系宝可梦$x^1, x^2, x^3, \ldots\ldots, x^{79}$
- 假设 $x^1, x^2, x^3, \ldots\ldots, x^{79}$ 由**高斯分布似然函数** (μ, Σ)采样生成

$$L(\mu, \Sigma) = f_{\mu,\Sigma}(x^1) f_{\mu,\Sigma}(x^2) \cdots f_{\mu,\Sigma}(x^{79})$$

$$f_{\mu,\Sigma}(x) = \frac{1}{(2\pi)^{D/2}} \frac{1}{|\Sigma|^{1/2}} \exp\left\{-\frac{1}{2}(x-\mu)^T (\Sigma)^{-1} (x-\mu)\right\}$$

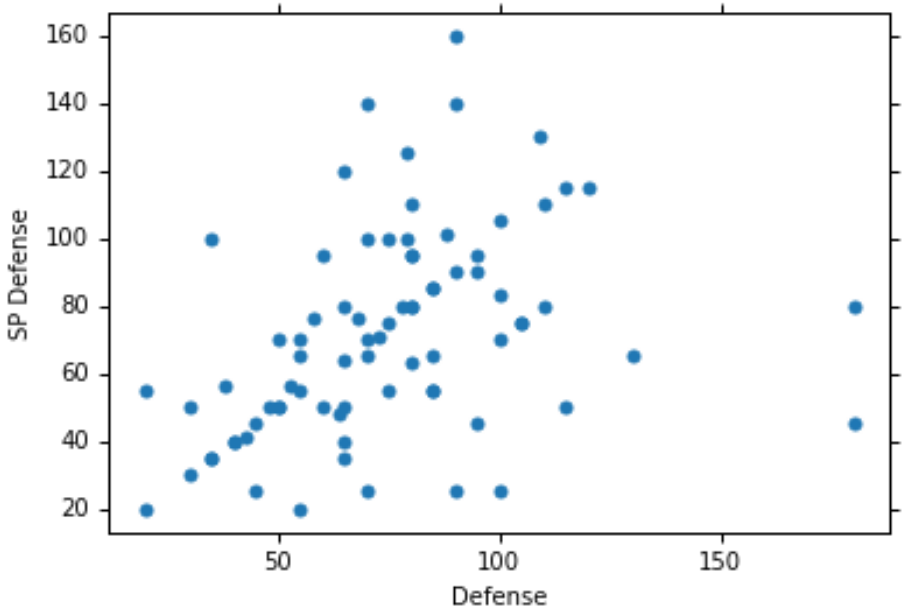
$$\mu^*, \Sigma^* = arg \max_{\mu,\Sigma} L(\mu, \Sigma)$$

$$\mu^* = \frac{1}{79}\sum_{n=1}^{79} x^n \qquad \Sigma^* = \frac{1}{79}\sum_{n=1}^{79} (x^n - \mu^*)(x^n - \mu^*)^T$$

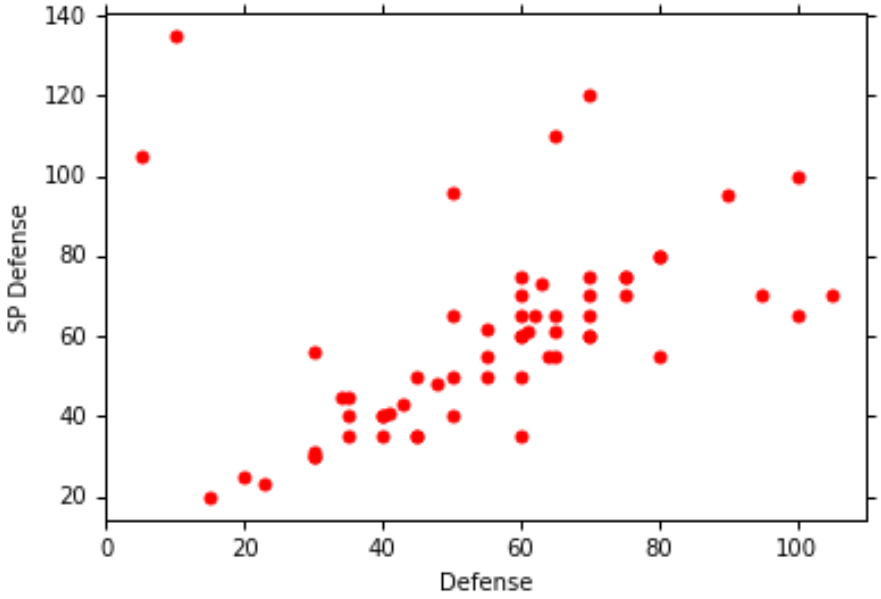
均值　　协方差矩阵

最大似然估计

Class 1



Class 2



$$\boldsymbol{\mu}^1 = \begin{bmatrix} 75.0 \\ 71.3 \end{bmatrix} \quad \boldsymbol{\Sigma}^1 = \begin{bmatrix} 874 & 327 \\ 327 & 929 \end{bmatrix}$$

$$\boldsymbol{\mu}^2 = \begin{bmatrix} 55.6 \\ 59.8 \end{bmatrix} \quad \boldsymbol{\Sigma}^2 = \begin{bmatrix} 874 & 422 \\ 422 & 685 \end{bmatrix}$$

依后验概率进行分类

$$f_{\mu^1,\Sigma^1}(\boldsymbol{x}) = \frac{1}{(2\pi)^{D/2}} \frac{1}{|\boldsymbol{\Sigma}^1|^{1/2}} \exp\left\{-\frac{1}{2}(\boldsymbol{x}-\boldsymbol{\mu}^1)^T(\boldsymbol{\Sigma}^1)^{-1}(\boldsymbol{x}-\boldsymbol{\mu}^1)\right\}$$

$$\boldsymbol{\mu}^1 = \begin{bmatrix} 75.0 \\ 71.3 \end{bmatrix} \quad \boldsymbol{\Sigma}^1 = \begin{bmatrix} 874 & 327 \\ 327 & 929 \end{bmatrix} \qquad P(C_1) = 79/(79+61) = 0.56$$

$$P(C_1|\boldsymbol{x}) = \frac{P(\boldsymbol{x}|C_1)P(C_1)}{P(\boldsymbol{x}|C_1)P(C_1) + P(\boldsymbol{x}|C_2)P(C_2)}$$

$$P(C_2) = 61/(79+61) = 0.44$$

$$f_{\mu^2,\Sigma^2}(\boldsymbol{x}) = \frac{1}{(2\pi)^{D/2}} \frac{1}{|\boldsymbol{\Sigma}^2|^{1/2}} \exp\left\{-\frac{1}{2}(\boldsymbol{x}-\boldsymbol{\mu}^2)^T(\boldsymbol{\Sigma}^2)^{-1}(\boldsymbol{x}-\boldsymbol{\mu}^2)\right\}$$

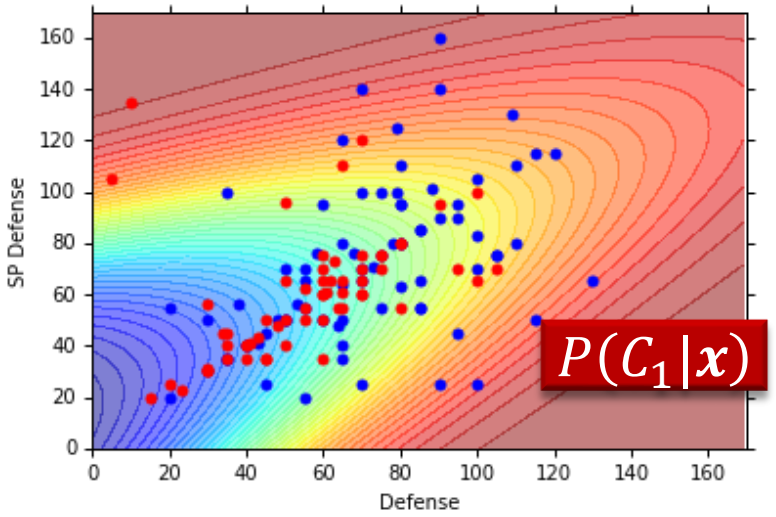
$$\boldsymbol{\mu}^2 = \begin{bmatrix} 55.6 \\ 59.8 \end{bmatrix} \quad \boldsymbol{\Sigma}^2 = \begin{bmatrix} 874 & 422 \\ 422 & 685 \end{bmatrix}$$

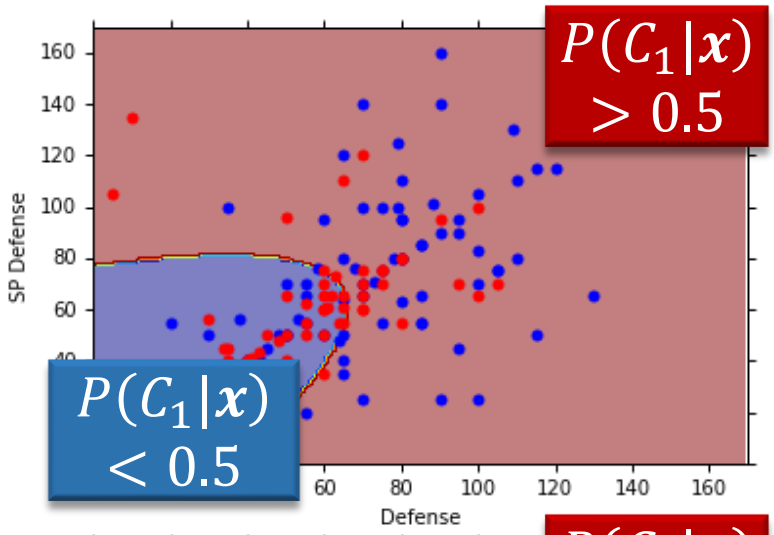
If $P(C_1|\boldsymbol{x}) > 0.5$

样本 $\boldsymbol{x}$ 属于 Class 1

结果分析

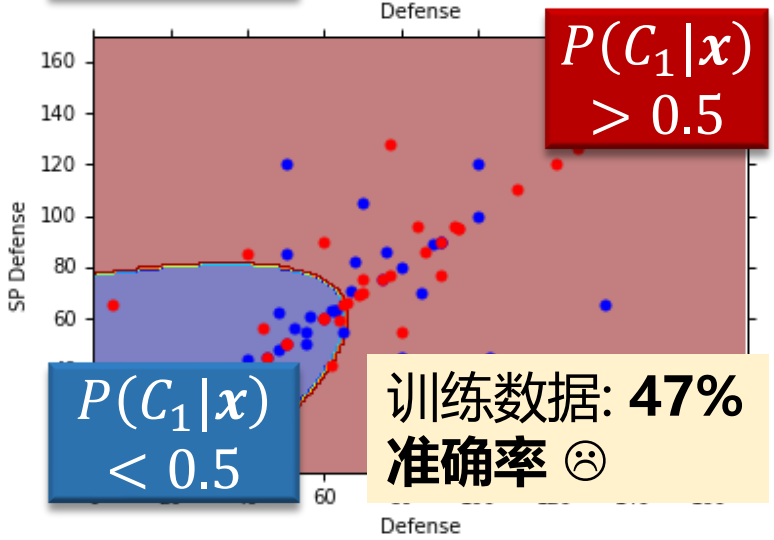
- **蓝点**: C_1 (水系), **红点**: C_2 (一般系)





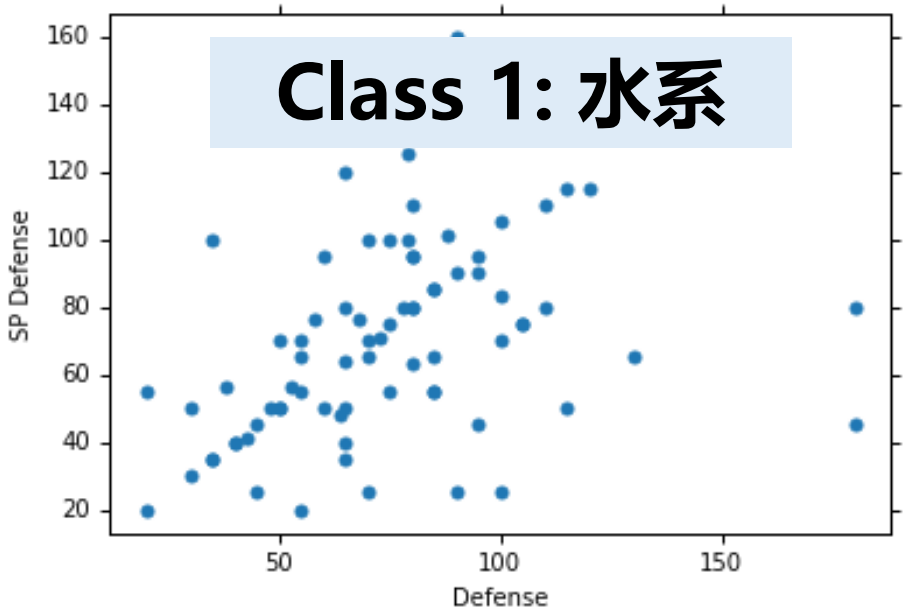
- 若考虑总共6个特征
  - μ^1, μ^2: 6维向量
  - Σ^1, Σ^2: 6 x 6 矩阵

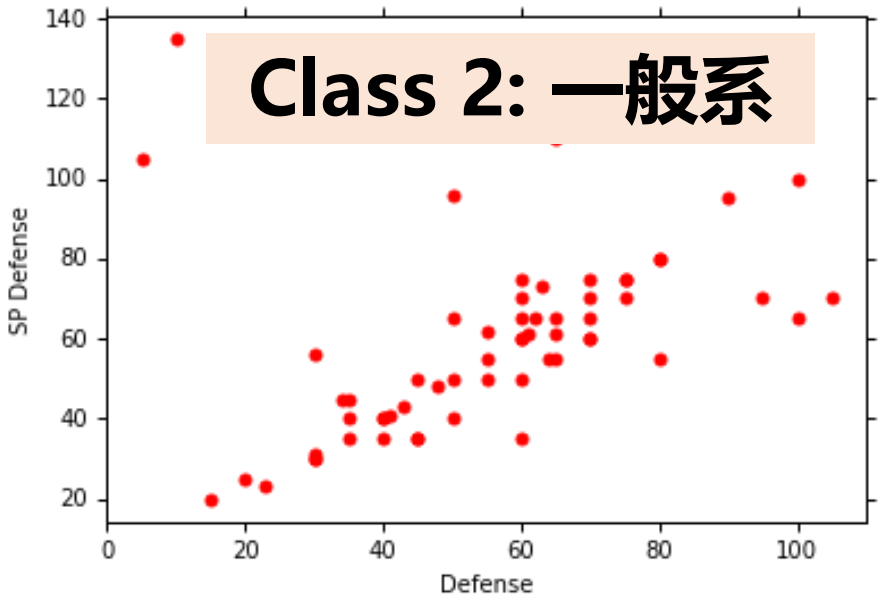
新测试数据:
64% 准确率 ☺



训练数据: **47%**
准确率 ☹

改进模型





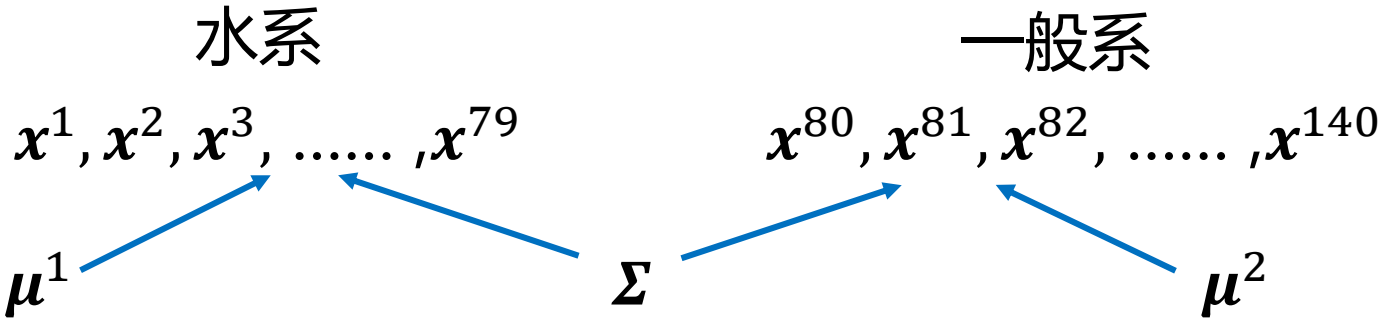
$$\mu^1 = \begin{bmatrix} 75.0 \\ 71.3 \end{bmatrix} \quad \Sigma^1 = \begin{bmatrix} 874 & 327 \\ 327 & 929 \end{bmatrix}$$

$$\mu^2 = \begin{bmatrix} 55.6 \\ 59.8 \end{bmatrix} \quad \Sigma^2 = \begin{bmatrix} 874 & 422 \\ 422 & 685 \end{bmatrix}$$

改进策略 如果两个分布共用同一个Σ，可以节省参数量

改进模型

口最大似然估计

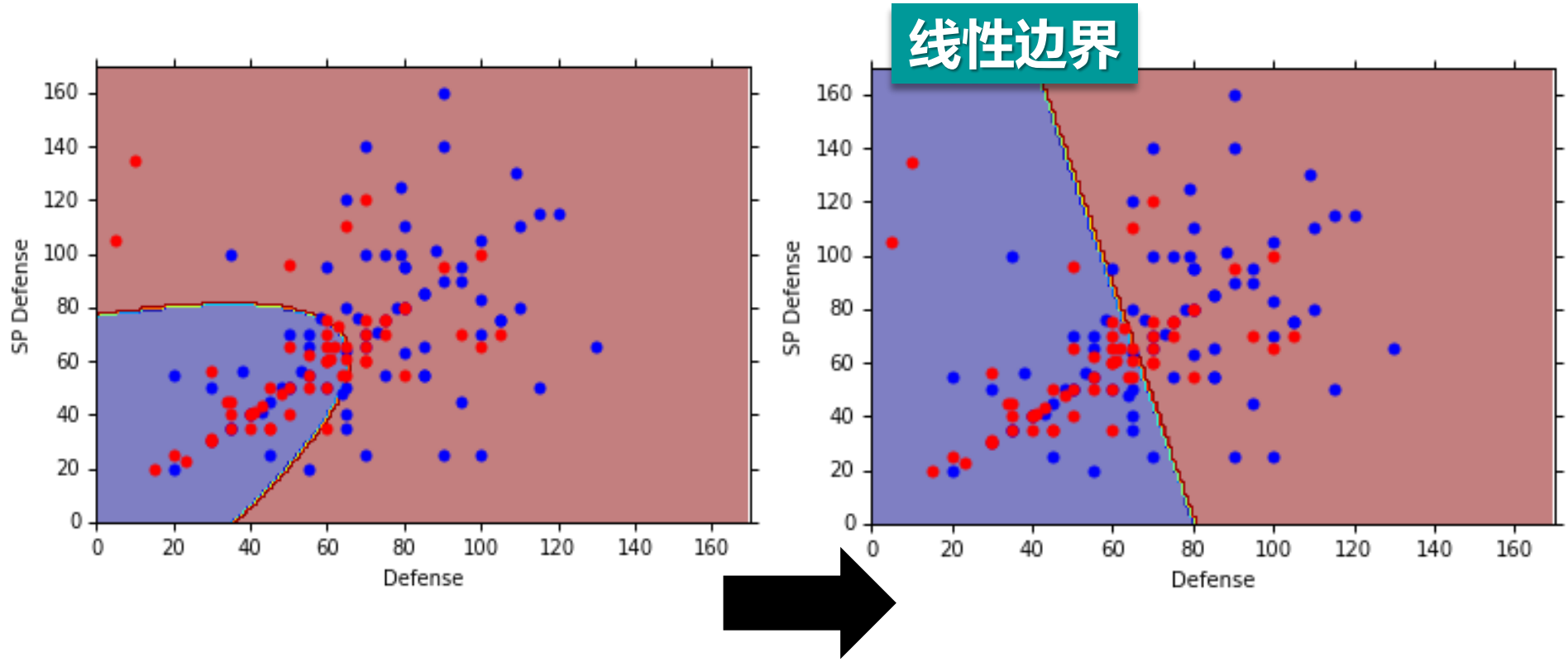


- 找到使似然函数$L(\mu^1, \mu^2, \Sigma)$最大的 μ^1, μ^2, Σ

$$L(\mu^1, \mu^2, \Sigma) = f_{\mu^1,\Sigma}(x^1) \cdot f_{\mu^1,\Sigma}(x^2) \cdots f_{\mu^1,\Sigma}(x^{79}) \times f_{\mu^2,\Sigma}(x^{80}) \cdot f_{\mu^2,\Sigma}(x^{81}) \cdots f_{\mu^2,\Sigma}(x^{140})$$

- μ^1 和 μ^2 计算方法相同 $\Sigma = \frac{79}{140}\Sigma^1 + \frac{61}{140}\Sigma^2$

改进模型



共享同一个协方差矩阵

考虑全部六个特征

64%准确率 → 73%准确率

内容导览

- 线性回归：多项式拟合
- 线性分类：概率生成模型
- Logistic回归：神经元模型
- 多分类问题
- 线性模型的局限与非线性模型

从深度学习角度看待分类问题

口将分类问题看作条件概率估计问题

- 引入非线性函数g来预测类别标签的条件概率$p(y = c|\boldsymbol{x})$
- 以二分类为例，

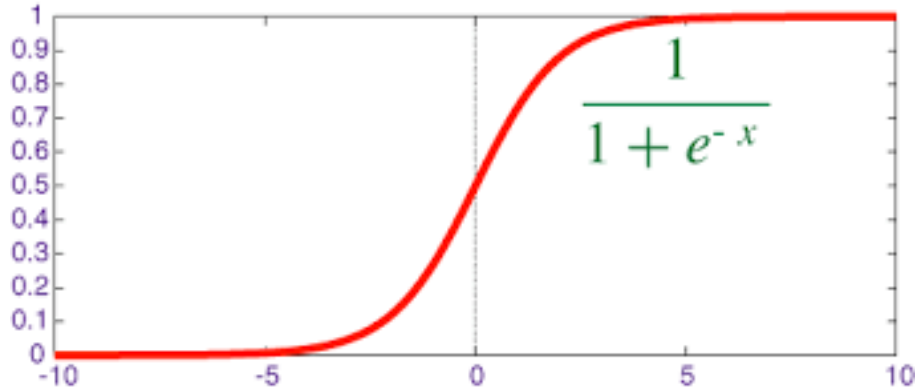
$$p(y = 1|\mathbf{x}) = g(f(\mathbf{x}; \mathbf{w}))$$

- 函数f：线性函数
- 函数g：把线性函数的值域从实数区间**“挤压”**到了$(0,1)$之间，可以用来表示概率。

预备知识：Logistic函数

- Logistic函数

$$\sigma(x) = \frac{1}{1 + \exp(-x)}$$



- Logistic回归

$$p(y = 1|\boldsymbol{x}) = \sigma(\boldsymbol{w}^{\mathsf{T}}\boldsymbol{x})$$

$$\triangleq \frac{1}{1 + \exp(-\boldsymbol{w}^{\mathsf{T}}\boldsymbol{x})}$$

预备知识：学习准则

- 模型预测条件概率$p_\theta(y|\boldsymbol{x})$

$$p_\theta(y=1|\boldsymbol{x}) = \sigma(\boldsymbol{w}^T\boldsymbol{x})$$

- 真实条件概率$p_r(y|\boldsymbol{x})$
  - 对于一个样本(x, y^*),其真实条件概率为

$$p_r(y=1|\boldsymbol{x}) = y^*$$

$$p_r(y=0|\boldsymbol{x}) = 1-y^*$$

如何衡量两个条件分布的差异?

预备知识：信息熵

- 在信息论中，熵用来衡量一个随机事件的不确定性
  - 自信息(Self Information) $I(x) = -\log p(x)$
  - 熵 $H(X) = \mathbb{E}_X[I(x)]$

$$= \mathbb{E}_X[-\log p(x)]$$

$$= -\sum_{x \in \mathcal{X}} p(x) \log p(x)$$

  - 熵越高，则随机变量的信息越多；
  - 熵越低，则随机变量的信息越少
  - 在对分布 $q(y)$ 的符号进行编码时，熵 $H(q)$ 也是**理论上最优的平均编码长度**，这种编码方式称为**熵编码**(Entropy Coding)，广泛应用于**数据压缩的理论性能分析**

预备知识：交叉熵(Cross Entropy)

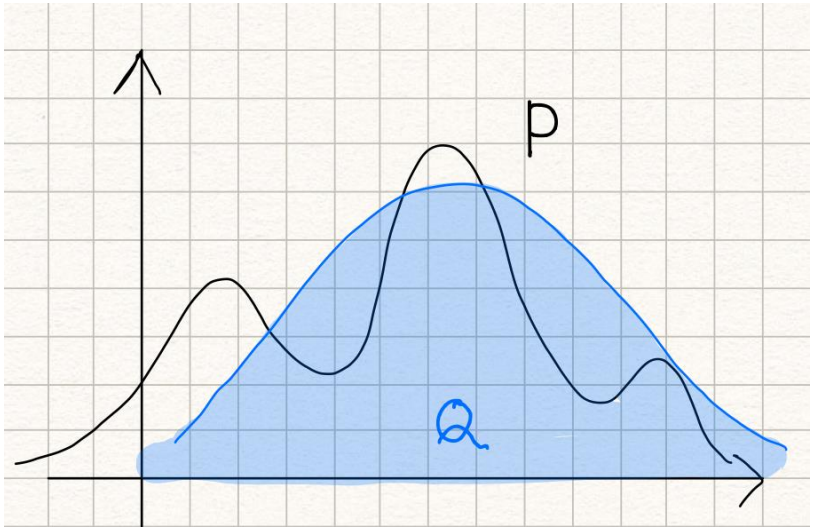
- **交叉熵**是按照**估计分布**为q的最优编码对**真实分布**为p的信息进行编码的长度

$$H(p,q) = \mathbb{E}_p[-\log q(x)]$$
$$= -\sum_x p(x)\log q(x)$$

  - 在给定 q 的情况下，如果 p 和 q 越接近，交叉熵越小
  - 如果 p 和 q 越远，交叉熵就越大

预备知识：KL散度(Kullback-Leibler Divergence)

- KL散度是用**估计分布**q来近似**真实分布**p时所造成的信息损失量
  - KL散度的物理含义是：按照估计分布为q的最优编码对真实分布为p的信息进行编码，其**平均编码长度(即交叉熵)**$H(p,q)$和p的**最优平均编码长度(即熵)**$H(p)$之间的**差异**



$$KL(p,q) = H(p,q) - H(p)$$

$$= \sum_x p(x) \log \frac{p(x)}{q(x)}$$

$$\geqslant 0$$

分类问题的基本步骤

口建立模型

后验概率为分类准则

x ➡

$$P(C_1|\boldsymbol{x}) = \frac{P(\boldsymbol{x}|C_1)P(C_1)}{P(\boldsymbol{x}|C_1)P(C_1) + P(\boldsymbol{x}|C_2)P(C_2)}$$

If $P(C_1|\boldsymbol{x}) > 0.5$, 输出: class 1

否则, 输出: class 2

口确定评价函数

- 使得似然函数（生成数据的概率）最大的均值 $\boldsymbol{\mu}$ 和协方差 $\boldsymbol{\Sigma}$

口选择最佳函数

依后验概率分类

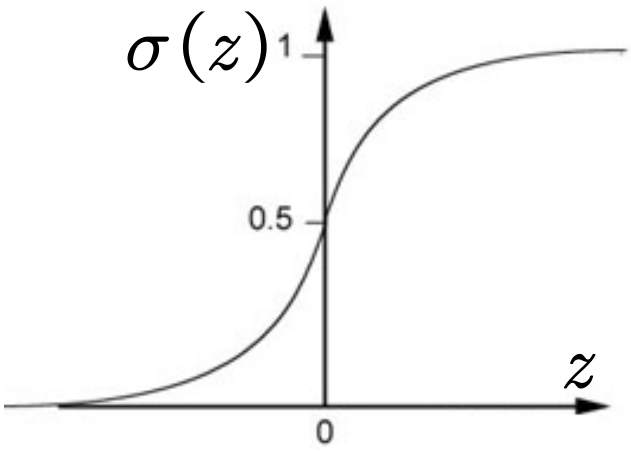
$$P(C_1|\boldsymbol{x}) = \frac{P(\boldsymbol{x}|C_1)P(C_1)}{P(\boldsymbol{x}|C_1)P(C_1)+P(\boldsymbol{x}|C_2)P(C_2)}$$

$$= \frac{1}{1+\frac{P(\boldsymbol{x}|C_2)P(C_2)}{P(\boldsymbol{x}|C_1)P(C_1)}} \quad \text{记 } z = \ln\frac{P(\boldsymbol{x}|C_1)P(C_1)}{P(\boldsymbol{x}|C_2)P(C_2)}$$

$$= \frac{1}{1+\exp(-z)}$$

$$= \sigma(z)$$

Sigmoid 函数



互动 还记得上学期《人工智能数学基础》提到的Sigmoid函数的优缺点有哪些吗?

依后验概率分类

$$P(C_1|\boldsymbol{x}) = \sigma(z)$$ Sigmoid 函数 $$z = \ln \frac{P(\boldsymbol{x}|C_1)P(C_1)}{P(\boldsymbol{x}|C_2)P(C_2)}$$

$$z = \ln \frac{P(\boldsymbol{x}|C_1)}{P(\boldsymbol{x}|C_2)} + \ln \frac{P(C_1)}{P(C_2)} \Rightarrow \frac{\frac{N_1}{N_1+N_2}}{\frac{N_2}{N_1+N_2}} = \frac{N_1}{N_2}$$

$$P(\boldsymbol{x}|C_1) = \frac{1}{(2\pi)^{D/2}} \frac{1}{|\boldsymbol{\Sigma}^1|^{1/2}} \exp\left\{-\frac{1}{2}(\boldsymbol{x}-\boldsymbol{\mu}^1)^T(\boldsymbol{\Sigma}^1)^{-1}(\boldsymbol{x}-\boldsymbol{\mu}^1)\right\}$$

$$P(\boldsymbol{x}|C_2) = \frac{1}{(2\pi)^{D/2}} \frac{1}{|\boldsymbol{\Sigma}^2|^{1/2}} \exp\left\{-\frac{1}{2}(\boldsymbol{x}-\boldsymbol{\mu}^2)^T(\boldsymbol{\Sigma}^2)^{-1}(\boldsymbol{x}-\boldsymbol{\mu}^2)\right\}$$

依后验概率分类

$$z = \ln \frac{P(\boldsymbol{x}|C_1)}{P(\boldsymbol{x}|C_2)} + \ln \frac{P(C_1)}{P(C_2)}$$

$$\ln \frac{P(\boldsymbol{x}|C_1)}{P(\boldsymbol{x}|C_2)}$$

$$= \ln \frac{\cancel{\frac{1}{(2\pi)^{D/2}}} \frac{1}{|\boldsymbol{\Sigma}^1|^{1/2}} \exp\left\{-\frac{1}{2}(\boldsymbol{x}-\boldsymbol{\mu}^1)^T(\boldsymbol{\Sigma}^1)^{-1}(\boldsymbol{x}-\boldsymbol{\mu}^1)\right\}}{\cancel{\frac{1}{(2\pi)^{D/2}}} \frac{1}{|\boldsymbol{\Sigma}^2|^{1/2}} \exp\left\{-\frac{1}{2}(\boldsymbol{x}-\boldsymbol{\mu}^2)^T(\boldsymbol{\Sigma}^2)^{-1}(\boldsymbol{x}-\boldsymbol{\mu}^2)\right\}}$$

$$= \frac{1}{2}\ln\frac{|\boldsymbol{\Sigma}^2|}{|\boldsymbol{\Sigma}^1|} - \frac{1}{2}(\boldsymbol{x}-\boldsymbol{\mu}^1)^T(\boldsymbol{\Sigma}^1)^{-1}(\boldsymbol{x}-\boldsymbol{\mu}^1) + \frac{1}{2}(\boldsymbol{x}-\boldsymbol{\mu}^2)^T(\boldsymbol{\Sigma}^2)^{-1}(\boldsymbol{x}-\boldsymbol{\mu}^2)$$

依后验概率分类

$$(\boldsymbol{x}-\boldsymbol{\mu}^1)^T(\boldsymbol{\Sigma}^1)^{-1}(\boldsymbol{x}-\boldsymbol{\mu}^1)$$

$$=\boldsymbol{x}^T(\boldsymbol{\Sigma}^1)^{-1}\boldsymbol{x}-\boldsymbol{x}^T(\boldsymbol{\Sigma}^1)^{-1}\boldsymbol{\mu}^1-(\boldsymbol{\mu}^1)^T(\boldsymbol{\Sigma}^1)^{-1}\boldsymbol{x}+(\boldsymbol{\mu}^1)^T(\boldsymbol{\Sigma}^1)^{-1}\boldsymbol{\mu}^1$$

$$=\boldsymbol{x}^T(\boldsymbol{\Sigma}^1)^{-1}\boldsymbol{x}-2(\boldsymbol{\mu}^1)^T(\boldsymbol{\Sigma}^1)^{-1}\boldsymbol{x}+(\boldsymbol{\mu}^1)^T(\boldsymbol{\Sigma}^1)^{-1}\boldsymbol{\mu}^1$$

同理 $(\boldsymbol{x}-\boldsymbol{\mu}^2)^T(\boldsymbol{\Sigma}^2)^{-1}(\boldsymbol{x}-\boldsymbol{\mu}^2)$

$$=\boldsymbol{x}^T(\boldsymbol{\Sigma}^2)^{-1}\boldsymbol{x}-2(\boldsymbol{\mu}^2)^T(\boldsymbol{\Sigma}^2)^{-1}\boldsymbol{x}+(\boldsymbol{\mu}^2)^T(\boldsymbol{\Sigma}^2)^{-1}\boldsymbol{\mu}^2$$

综上 $z=\frac{1}{2}\ln\frac{|\boldsymbol{\Sigma}^2|}{|\boldsymbol{\Sigma}^1|}-\frac{1}{2}\boldsymbol{x}^T(\boldsymbol{\Sigma}^1)^{-1}\boldsymbol{x}+(\boldsymbol{\mu}^1)^T(\boldsymbol{\Sigma}^1)^{-1}\boldsymbol{x}$

$$-\frac{1}{2}(\boldsymbol{\mu}^1)^T(\boldsymbol{\Sigma}^1)^{-1}\boldsymbol{\mu}^1+\frac{1}{2}\boldsymbol{x}^T(\boldsymbol{\Sigma}^2)^{-1}\boldsymbol{x}-(\boldsymbol{\mu}^2)^T(\boldsymbol{\Sigma}^2)^{-1}\boldsymbol{x}$$

$$+\frac{1}{2}(\boldsymbol{\mu}^2)^T(\boldsymbol{\Sigma}^2)^{-1}\boldsymbol{\mu}^2+\ln\frac{N_1}{N_2}$$

依后验概率分类

$$z=\frac{1}{2}\ln\frac{|\Sigma^2|}{|\Sigma^1|}-\frac{1}{2}x^T(\Sigma^1)^{-1}x+(\mu^1)^T(\Sigma^1)^{-1}x-\frac{1}{2}(\mu^1)^T(\Sigma^1)^{-1}\mu^1$$

$$+\frac{1}{2}x^T(\Sigma^2)^{-1}x-(\mu^2)^T(\Sigma^2)^{-1}x+\frac{1}{2}(\mu^2)^T(\Sigma^2)^{-1}\mu^2+\ln\frac{N_1}{N_2}$$

$$\Sigma^1=\Sigma^2=\Sigma$$

$$z=(\mu^1-\mu^2)^T(\Sigma)^{-1}x-\frac{1}{2}(\mu^1)^T(\Sigma)^{-1}\mu^1+\frac{1}{2}(\mu^2)^T(\Sigma)^{-1}\mu^2+\ln\frac{N_1}{N_2}$$

w^T b

综上 $P(C_1|x)=\sigma(w^T\cdot x+b)$

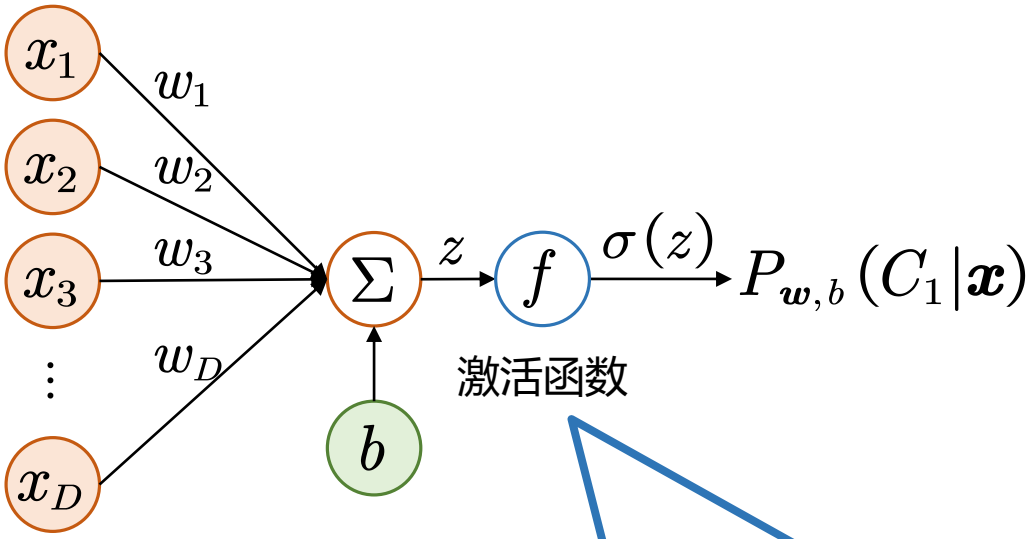
能够直接求解 w^T 和 b 吗

前述概率生成模型做法

先估计 N_1, N_2, μ^1, μ^2, Σ, 最大似然估计计算得到 w^T、b

神经元解决分类问题：①模型建立

$$P_{\boldsymbol{w},\boldsymbol{b}}(C_1|\boldsymbol{x})=\sigma(z) \quad z=\boldsymbol{w}^T\cdot\boldsymbol{x}+b=\sum_i w_i x_i+b$$

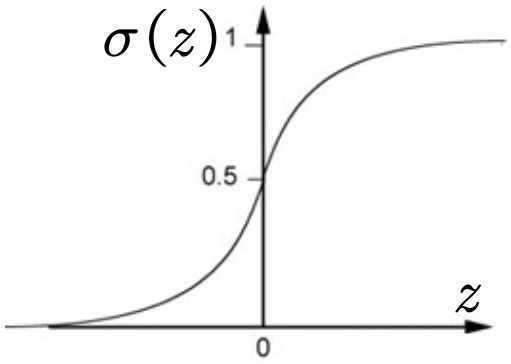


$$\begin{cases} P_{\boldsymbol{w},b}(C_1|\boldsymbol{x})\geqslant 0.5 & \text{class 1} \\ P_{\boldsymbol{w},b}(C_1|\boldsymbol{x})<0.5 & \text{class 2} \end{cases}$$

$$\begin{cases} z\geqslant 0 & \text{class 1} \\ z<0 & \text{class 2} \end{cases}$$

Sigmoid 函数

$$\sigma(z)=\frac{1}{1+e^{-z}}$$



神经元解决分类问题：②损失函数

- 神经元模型 $f_{w,b}(x) = P_{w,b}(C_1|x)$

x^1	x^2	x^3	$\cdots$	x^n
C_1	C_1	C_2	$\cdots$	C_1

x^1	x^2	x^3	$\cdots$	x^n
$\hat{y}^1=1$	$\hat{y}^2=1$	$\hat{y}^3=0$	$\cdots$	$\hat{y}^n=1$

- 给定一组 w 和 b，神经元模型估计对应类别的概率是

$$L(w,b) = f_{w,b}(x^1) f_{w,b}(x^2) (1 - f_{w,b}(x^3)) \cdots f_{w,b}(x^n)$$

- 当 $L(w,b)$ 取最大值时，能得到最有可能的 w^* 和 b^*

$$w^*, b^* = arg \max_{w,b} L(w,b)$$

$$w^*, b^* = arg \min_{w,b} (-\ln L(w,b))$$

$$-\ln L(w,b) =$$

$$-\ln f_{w,b}(x^1) \rightarrow -\left(\mathbf{1} \ln f(x^1) + \mathbf{0} \ln(1 - f(x^1))\right)$$

$$-\ln f_{w,b}(x^2) \rightarrow -\left(\mathbf{1} \ln f(x^2) + \mathbf{0} \ln(1 - f(x^2))\right)$$

$$-\ln(1 - f_{w,b}(x^3)) \rightarrow -\left(\mathbf{0} \ln f(x^3) + \mathbf{1} \ln(1 - f(x^3))\right)$$

神经元解决分类问题：②损失函数

$$L(\boldsymbol{w},b)=f_{\boldsymbol{w},b}(\boldsymbol{x}^1)\,f_{\boldsymbol{w},b}(\boldsymbol{x}^2)\,(1-f_{\boldsymbol{w},b}(\boldsymbol{x}^3))\cdots f_{\boldsymbol{w},b}(\boldsymbol{x}^n)$$

$$-\ln L(\boldsymbol{w},b)=-\ln f_{\boldsymbol{w},b}(\boldsymbol{x}^1)-\ln f_{\boldsymbol{w},b}(\boldsymbol{x}^2)-\ln(1-f_{\boldsymbol{w},b}(\boldsymbol{x}^3))\cdots$$

$\hat{y}^n$: **1** 代表 class 1, **0** 代表 class 2

$$-\ln L(\boldsymbol{w},b)=-\sum_i\left(\hat{y}^i\ln f_{\boldsymbol{w},b}(\boldsymbol{x}^i)+(1-\hat{y}^i)\ln(1-f_{\boldsymbol{w},b}(\boldsymbol{x}^i))\right)$$

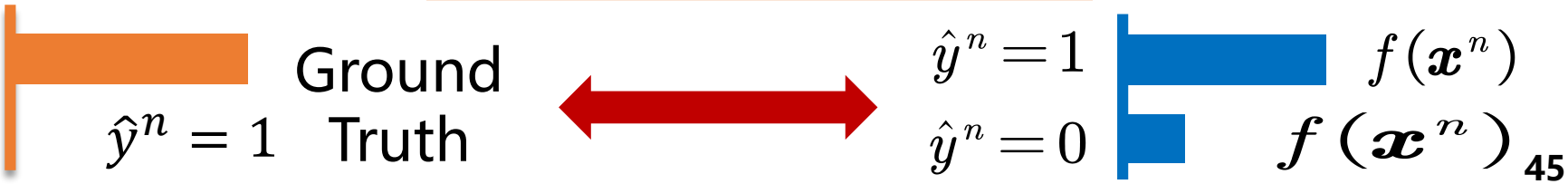
两个伯努利分布的交叉熵

真实分布 p:
$p(x=1)=\hat{y}^n$
$p(x=0)=1-\hat{y}^n$

⟷ **交叉熵** ⟷

估计分布 q:
$q(x=1)=f(x^n)$
$q(x=0)=1-f(x^n)$

$$H(p,q)=-\sum_{\boldsymbol{x}}p(\boldsymbol{x})\ln q(\boldsymbol{x})$$



神经元解决分类问题：③参数优化

$$-\ln L(\boldsymbol{w},b)=-\sum_n\left(\hat{y}^n\ln f_{\boldsymbol{w},\boldsymbol{b}}(\boldsymbol{x}^n)+(1-\hat{y}^n)\ln\left(1-f_{\boldsymbol{w},\boldsymbol{b}}(\boldsymbol{x}^n)\right)\right)$$

w_i w_i w_i

$$f_{\boldsymbol{w},b}(\boldsymbol{x})=\sigma(z)$$

$$z=\boldsymbol{w}^T\cdot\boldsymbol{x}+b=\sum_i w_i x_i+b$$

$$\frac{\partial\ln f_{\boldsymbol{w},b}(\boldsymbol{x}^n)}{\partial w_i}=\frac{\partial\ln f_{\boldsymbol{w},b}(\boldsymbol{x}^n)}{\partial z}\frac{\partial z}{\partial w_i}$$

$$\frac{\partial z}{\partial w_i}=x_i^n$$

$$\frac{\partial\ln\sigma(z)}{\partial z}=\frac{1}{\sigma(z)}\sigma(z)\left(1-\sigma(z)\right)=1-\sigma(z)$$

$$\frac{\partial\ln\left(1-f_{\boldsymbol{w},b}(\boldsymbol{x}^n)\right)}{\partial w_i}=\frac{\partial\ln\left(1-f_{\boldsymbol{w},b}(\boldsymbol{x}^n)\right)}{\partial z}\frac{\partial z}{\partial w_i}$$

$$\frac{\partial\ln\left(1-\sigma(z)\right)}{\partial z}=-\frac{1}{1-\sigma(z)}\sigma(z)\left(1-\sigma(z)\right)=-\sigma(z)$$

神经元解决分类问题：③参数优化

$$-\frac{\partial \ln L(\boldsymbol{w},b)}{\partial w_i}=-\sum_n\left(\hat{y}^n\left(1-f_{\boldsymbol{w},b}(\boldsymbol{x}^n)\right)-\left(1-\hat{y}^n\right)f_{\boldsymbol{w},b}(\boldsymbol{x}^n)\right)x_i^n$$

$$=-\sum_n\left(\hat{y}^n-\hat{y}^n f_{\boldsymbol{w},b}(\boldsymbol{x}^n)-f_{\boldsymbol{w},b}(\boldsymbol{x}^n)+\hat{y}^n f_{\boldsymbol{w},b}(\boldsymbol{x}^n)\right)x_i^n$$

$$=-\sum_n\left(\hat{y}^n-f_{\boldsymbol{w},b}(\boldsymbol{x}^n)\right)x_i^n$$

拓展到向量形式，即有

$$-\frac{\partial \ln L(\boldsymbol{w},b)}{\partial \boldsymbol{w}}=-\sum_n\left(\hat{y}^n-f_{\boldsymbol{w},b}(\boldsymbol{x}^n)\right)\boldsymbol{x}^n$$

参数 $\boldsymbol{w}$ 的梯度下降更新式为 $\boldsymbol{w}\leftarrow\boldsymbol{w}+\eta\sum_n\left(\hat{y}^n-f_{\boldsymbol{w},b}(\boldsymbol{x}^n)\right)\boldsymbol{x}^n$

差异越大，参数更新幅度越大

Logistic回归与线性回归对比

模型	$f_{\boldsymbol{w},b}(\boldsymbol{x}) = \sigma(\sum_i w_i x_i + b)$	$f_{\boldsymbol{w},b}(\boldsymbol{x}) = \sum_i w_i x_i + b$
	输出：介于 0~1 之间	输出：任意值

Logistic回归与线性回归对比

	Logistic回归	线性回归
模型	$f_{w,b}(\boldsymbol{x}) = \sigma(\sum_i w_i x_i + b)$ 输出：介于 0~1 之间	$f_{w,b}(\boldsymbol{x}) = \sum_i w_i x_i + b$ 输出：任意值
目标	训练数据: $(\boldsymbol{x}^n, \hat{y}^n)$ $\hat{y}^n$: **1**、**0** 代表 class 1、2 $L(f) = \sum_n l(f(\boldsymbol{x}^n), \hat{y}^n)$	训练数据: $(\boldsymbol{x}^n, \hat{y}^n)$ $\hat{y}^n$: **真实的数值** $L(f) = \frac{1}{2}\sum_n (f(\boldsymbol{x}^n) - \hat{y}^n)^2$

交叉熵：

$$l(f(\boldsymbol{x}^n), \hat{y}^n) = -\left[\hat{y}^n ln f(\boldsymbol{x}^n) + (1-\hat{y}^n) ln\left(1 - f(\boldsymbol{x}^n)\right)\right]$$

Logistic回归与线性回归对比

	Logistic回归	线性回归
模型	$f_{w,b}(x) = \sigma(\sum_i w_i x_i + b)$ 输出：介于 0~1 之间	$f_{w,b}(x) = \sum_i w_i x_i + b$ 输出：任意值
目标	训练数据: $(x^n, \hat{y}^n)$ $\hat{y}^n$: **1**、**0** 代表 class 1、2 $L(f) = \sum_n l(f(x^n), \hat{y}^n)$	训练数据: $(x^n, \hat{y}^n)$ $\hat{y}^n$: **真实的数值** $L(f) = \frac{1}{2}\sum_n (f(x^n) - \hat{y}^n)^2$
求解	Logistic回归: $w_i \leftarrow w_i - \eta \sum_n -\left(\hat{y}^n - f_{w,b}(x^n)\right) x_i^n$	
	线性回归: $w_i \leftarrow w_i - \eta \sum_n -\left(\hat{y}^n - f_{w,b}(x^n)\right) x_i^n$	

均方误差MSE准则下的Logistic回归

口建立模型

$$f_{\boldsymbol{w},b}(\boldsymbol{x})=\sigma\left(\sum_i w_i x_i+b\right)$$

口确定评价函数

- 训练数据: $(x^n,\hat{y}^n)$, $\hat{y}^n$: **1** for class 1, **0** for class 2

$$L(f)=\frac{1}{2}\sum_n\left(f_{\boldsymbol{w},b}(\boldsymbol{x}^n)-\hat{y}\right)^2$$

口找到最佳的函数：梯度下降

$$\frac{\partial\left(f_{\boldsymbol{w},b}(\boldsymbol{x}^n)-\hat{y}\right)^2}{\partial w_i}=2\left(f_{\boldsymbol{w},b}(\boldsymbol{x}^n)-\hat{y}\right)\frac{\partial f_{\boldsymbol{w},b}(\boldsymbol{x}^n)}{\partial z}\frac{\partial z}{\partial w_i}$$

$$=2\left(f_{\boldsymbol{w},b}(\boldsymbol{x}^n)-\hat{y}\right)f_{\boldsymbol{w},b}(\boldsymbol{x}^n)\left(1-f_{\boldsymbol{w},b}(\boldsymbol{x}^n)\right)x_i^n$$

均方误差MSE准则下的Logistic回归

- 当 $\hat{y}^n = 1$ 时

  If $f_{\boldsymbol{w},b}(\boldsymbol{x}^n) = 1$ (close to target) ➡ $\partial L/\partial w_i = 0$

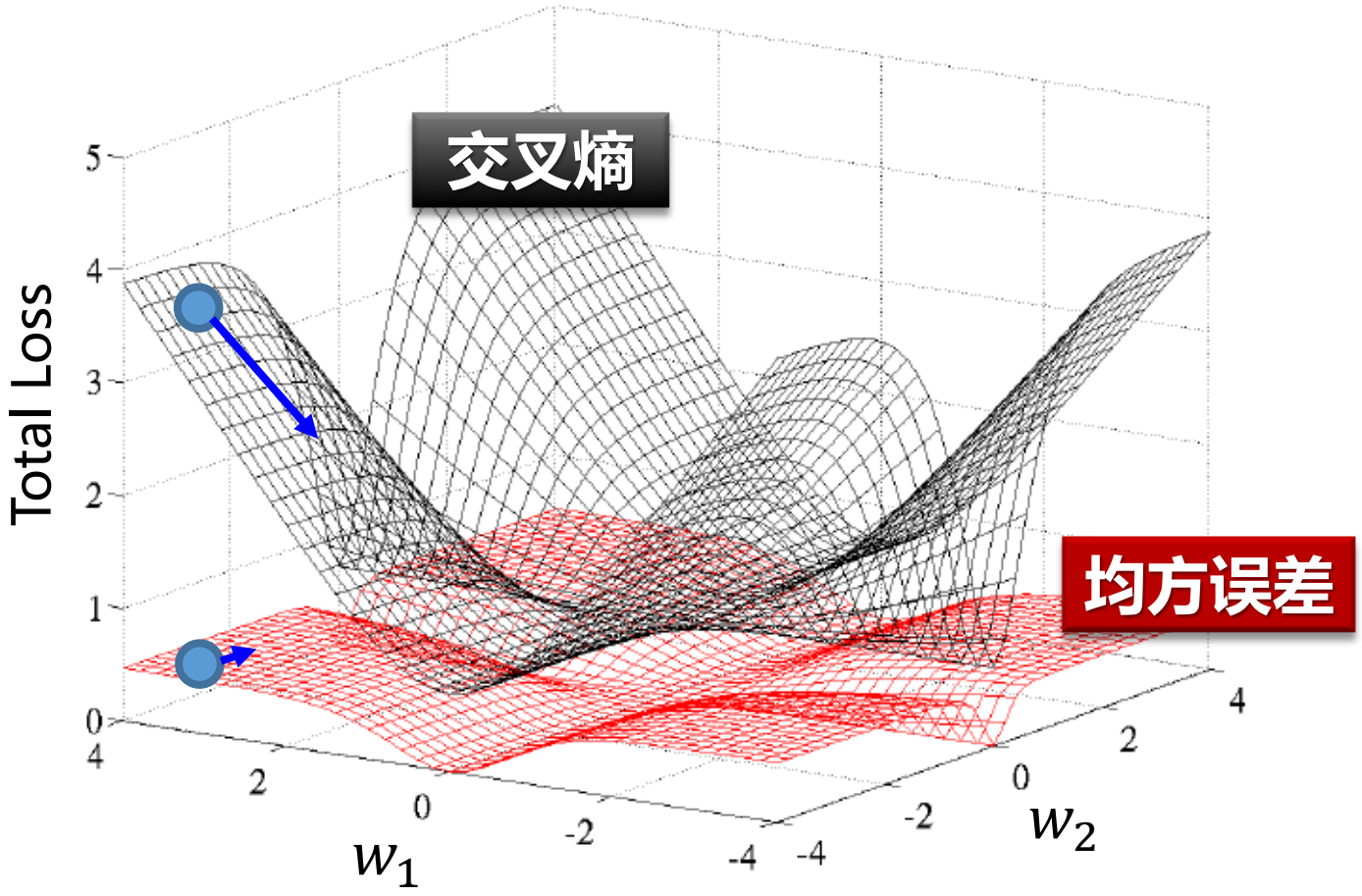
  If $f_{\boldsymbol{w},b}(\boldsymbol{x}^n) = 0$ (far from target) ➡ $\partial L/\partial w_i = 0$

- 当 $\hat{y}^n = 0$ 时

  If $f_{\boldsymbol{w},b}(\boldsymbol{x}^n) = 1$ (far from target) ➡ $\partial L/\partial w_i = 0$

  If $f_{\boldsymbol{w},b}(\boldsymbol{x}^n) = 0$ (close to target) ➡ $\partial L/\partial w_i = 0$

交叉熵与均方误差对比



http://jmlr.org/proceedings/papers/v9/glorot10a/glorot10a.pdf

概率生成模型与神经元模型对比

$$P(C_1|\boldsymbol{x}) = \sigma(\boldsymbol{w}^T \cdot \boldsymbol{x} + b)$$

- **概率生成模型**：先找到 $\boldsymbol{\mu}^1, \boldsymbol{\mu}^2, \boldsymbol{\Sigma}^{-1}$ ，再进行计算

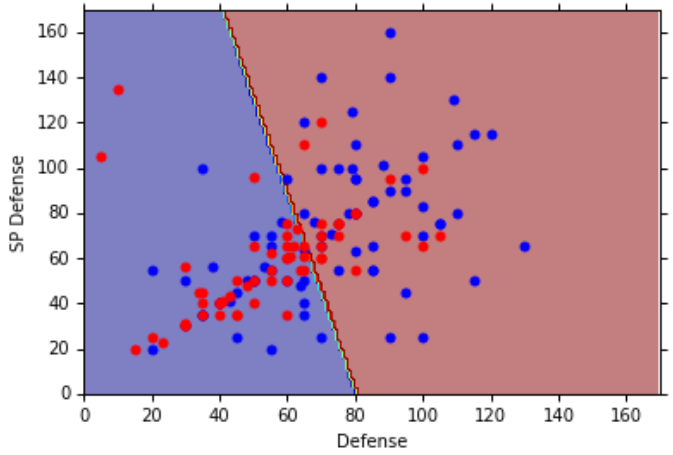
$$\boldsymbol{w}^T = (\boldsymbol{\mu}^1 - \boldsymbol{\mu}^2)^T (\boldsymbol{\Sigma})^{-1}$$

$$b = -\frac{1}{2}(\boldsymbol{\mu}^1)^T (\boldsymbol{\Sigma})^{-1} \boldsymbol{\mu}^1 + \frac{1}{2}(\boldsymbol{\mu}^2)^T (\boldsymbol{\Sigma})^{-1} \boldsymbol{\mu}^2 + \ln \frac{N_1}{N_2}$$

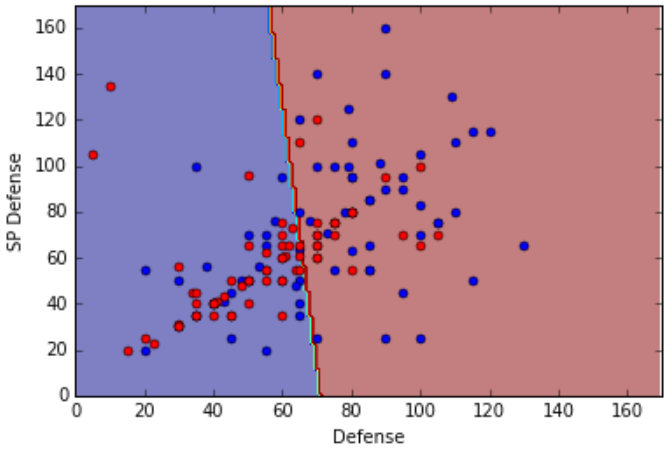
两种方式能解得同一组 w^T 和 b 吗

- **神经元模型**：直接找到 $\boldsymbol{w}$ 与 b

同样的模型，同样的数据也可能优化得到不同函数



概率生成模型 (73%准确度)



神经元模型 (79%准确度)

为什么神经元模型结果更好?

口朴素贝叶斯分类器

- 假设所有维度都是独立的

$$P(\boldsymbol{x}|C_1) = P(x_1|C_1)\,P(x_2|C_1)\cdots P(x_k|C_1)\cdots$$

$$\begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_k \\ \vdots \\ x_N \end{bmatrix}$$

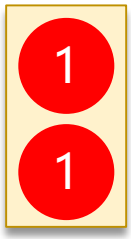
$P(x_k|C_1)$服从一维高斯分布

小贴士

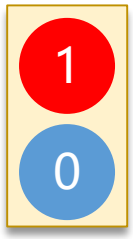
对于二进制特征，可以假设它们来自伯努利分布

从朴素贝叶斯分类角度来理解

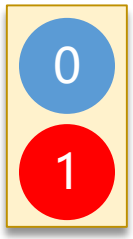
训练数据



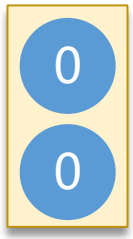
Class 1



× 4 Class 2

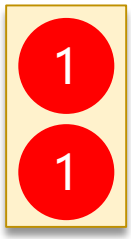


× 4 Class 2



× 4 Class 2

测试数据



Class 1?
Class 2?

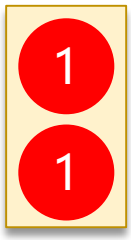
朴素贝叶斯的结果是?

$$P(x|C_i) = P(x_1|C_i)P(x_2|C_i)$$

$$P(C_1) = \frac{1}{13} \qquad P(x_1 = 1|C_1) = 1 \qquad P(x_2 = 1|C_1) = 1$$

$$P(C_2) = \frac{12}{13} \qquad P(x_1 = 1|C_2) = \frac{1}{3} \qquad P(x_2 = 1|C_2) = \frac{1}{3}$$

朴素贝叶斯分类

测试数据 

$$P(C_1|x) = \frac{P(x|C_1)P(C_1)}{P(x|C_1)P(C_1) + P(x|C_2)P(C_2)} < 0.5$$

1×1 $\frac{1}{13}$ 1×1 $\frac{1}{13}$ $\frac{1}{3} \times \frac{1}{3}$ $\frac{12}{13}$

$$P(C_1) = \frac{1}{13} \quad P(x_1 = 1|C_1) = 1 \quad P(x_2 = 1|C_1) = 1$$

$$P(C_2) = \frac{12}{13} \quad P(x_1 = 1|C_2) = \frac{1}{3} \quad P(x_2 = 1|C_2) = \frac{1}{3}$$

朴素贝叶斯假设各特征相互独立，即 $P(x|C_2) = \frac{1}{3} \times \frac{1}{3}$，而实际上 $P(x|C_2) = 0$

概率生成模型与神经元模型对比

$$P(C_1|\boldsymbol{x}) = \sigma(\boldsymbol{w}^T \cdot \boldsymbol{x} + b)$$

- **概率生成模型**：先找到 $\boldsymbol{\mu}^1, \boldsymbol{\mu}^2, \boldsymbol{\Sigma}^{-1}$ ，再进行计算

$$\boldsymbol{w}^T = (\boldsymbol{\mu}^1 - \boldsymbol{\mu}^2)^T (\boldsymbol{\Sigma})^{-1}$$

$$b = -\frac{1}{2}(\boldsymbol{\mu}^1)^T (\boldsymbol{\Sigma})^{-1} \boldsymbol{\mu}^1 + \frac{1}{2}(\boldsymbol{\mu}^2)^T (\boldsymbol{\Sigma})^{-1} \boldsymbol{\mu}^2 + \ln \frac{N_1}{N_2}$$

- **神经元模型**：直接找到 $\boldsymbol{w}$ 与 b
- **性能对比**
  - 通常**神经元模型效果更好**
  - **概率生成模型**的优势
    - 在概率分布的假设下**只需要更少的数据、面对噪声更鲁棒**
    - 先验和类相关概率可以从不同的来源估计

内容导览

- 线性回归：多项式拟合
- 线性分类：概率生成模型
- Logistic回归：神经元模型
- **多分类问题**
- 线性模型的局限与非线性模型

多分类问题：以三分类为例

C_1: w^1, b_1 $\quad z_1 = w^1 \cdot \boldsymbol{x} + b_1$

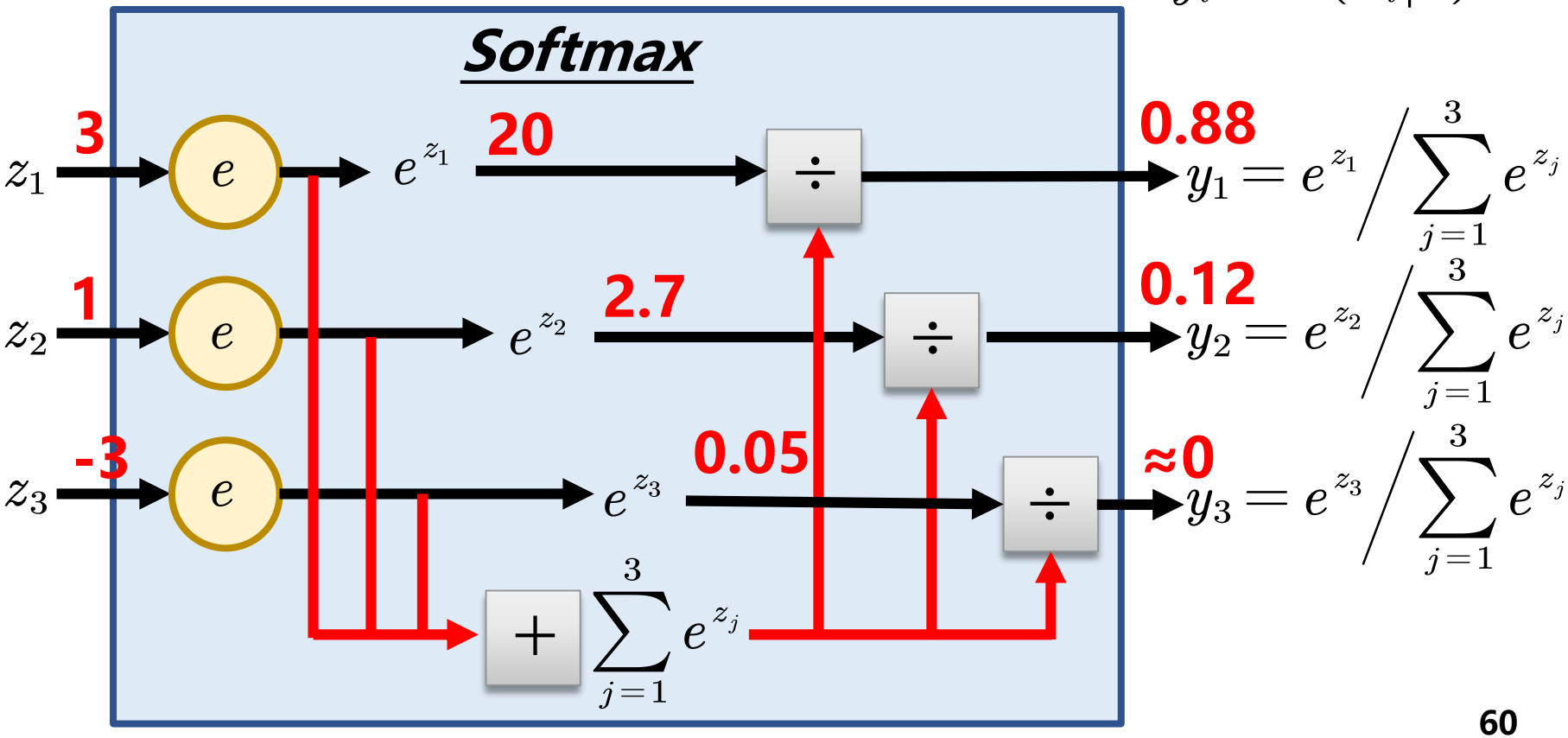
C_2: w^2, b_2 $\quad z_2 = w^2 \cdot \boldsymbol{x} + b_2$

C_3: w^3, b_3 $\quad z_3 = w^3 \cdot \boldsymbol{x} + b_3$

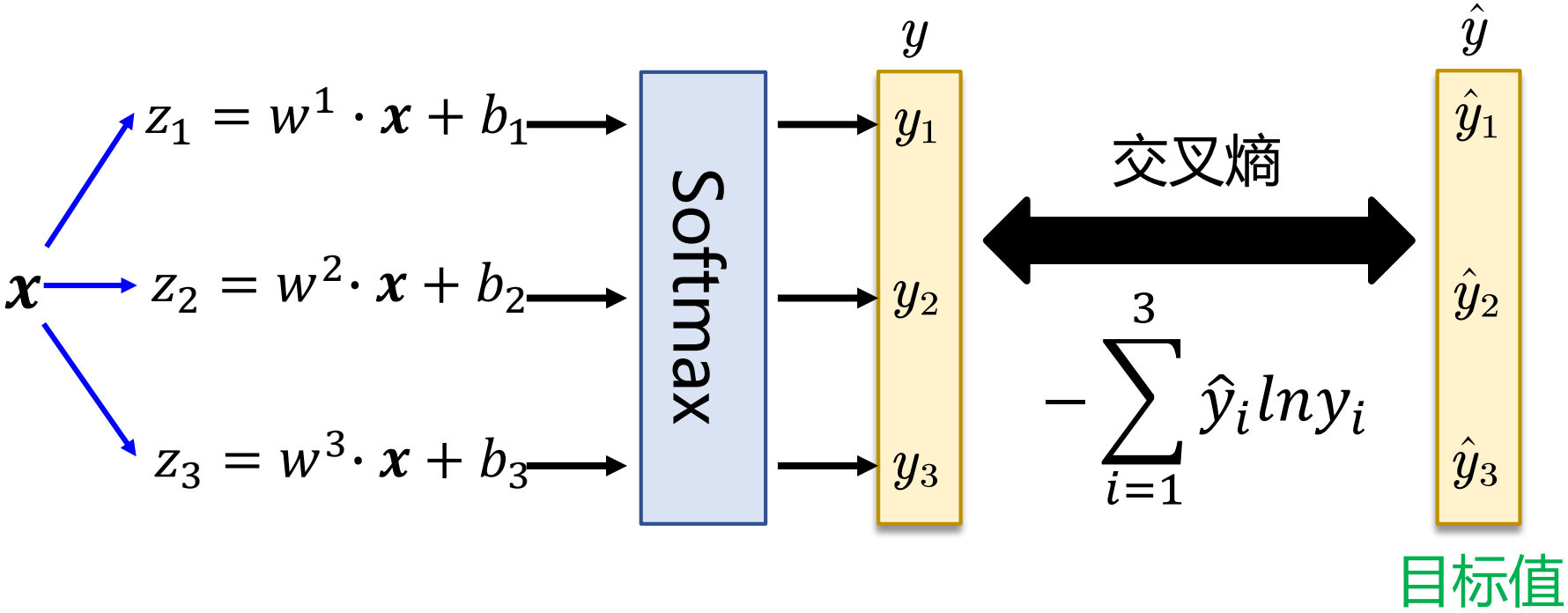
概率

- $1 > y_i > 0$
- $\sum_i y_i = 1$

$y_i = P(C_i|\boldsymbol{x})$



多分类问题：以三分类为例



If $x \in class\ 1$

$$\hat{y} = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}$$

$$-ln y_1$$

If $x \in class\ 2$

$$\hat{y} = \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}$$

$$-ln y_2$$

If $x \in class\ 3$

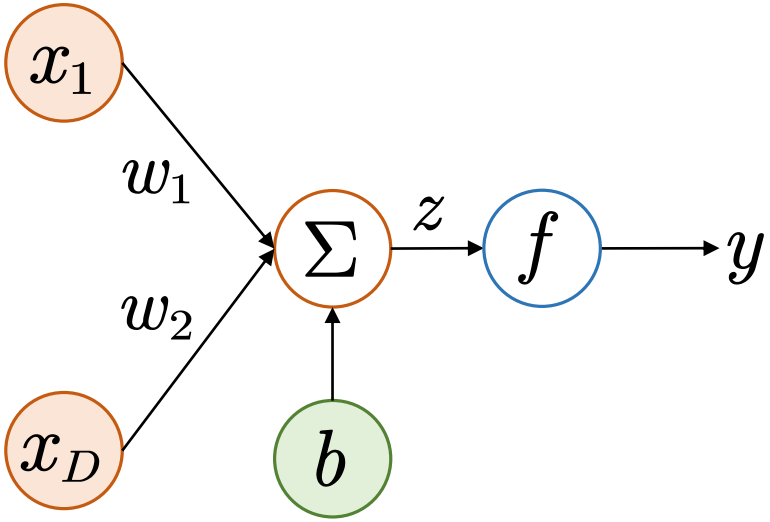
$$\hat{y} = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}$$

$$-ln y_3$$

内容导览

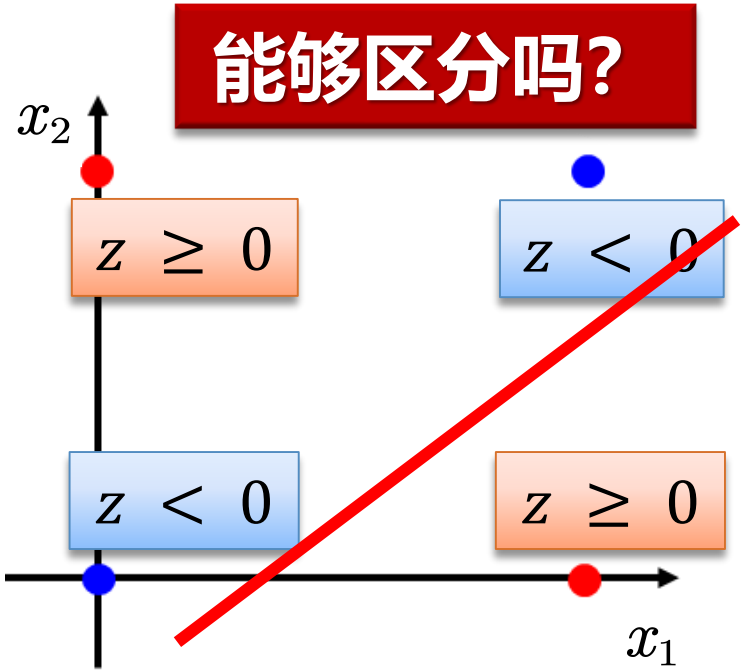
- 线性回归：多项式拟合
- 线性分类：概率生成模型
- Logistic回归：神经元模型
- 多分类问题
- **线性模型的局限与非线性模型**

Logistic回归的局限



$$\begin{cases} Class\ 1 & y \geqslant 0.5 \quad (z \geq 0) \\ Class\ 2 & y < 0.5 \quad (z < 0) \end{cases}$$

输入特征		标签
x_1	x_2	
0	0	Class 2
0	1	Class 1
1	0	Class 1
1	1	Class 2



Logistic回归的局限

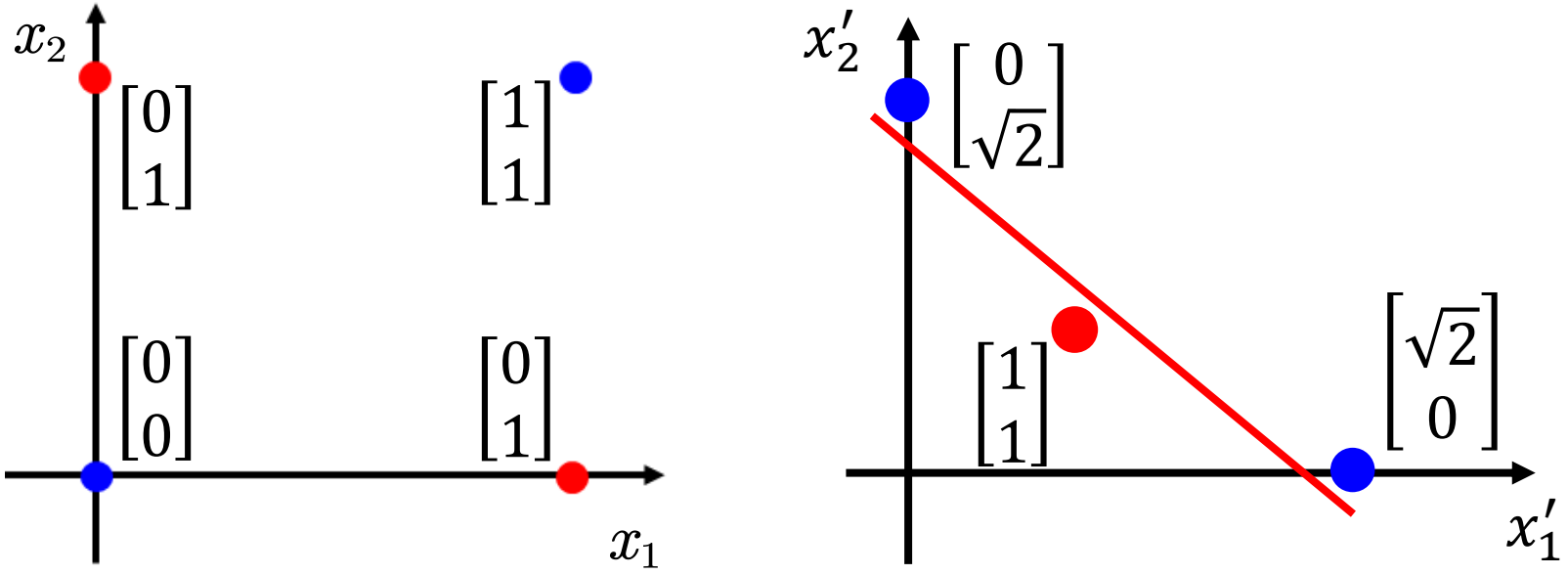
口特征变换

x_1': 到 $\begin{bmatrix}0\\0\end{bmatrix}$ 的距离

x_2': 到 $\begin{bmatrix}1\\1\end{bmatrix}$ 的距离

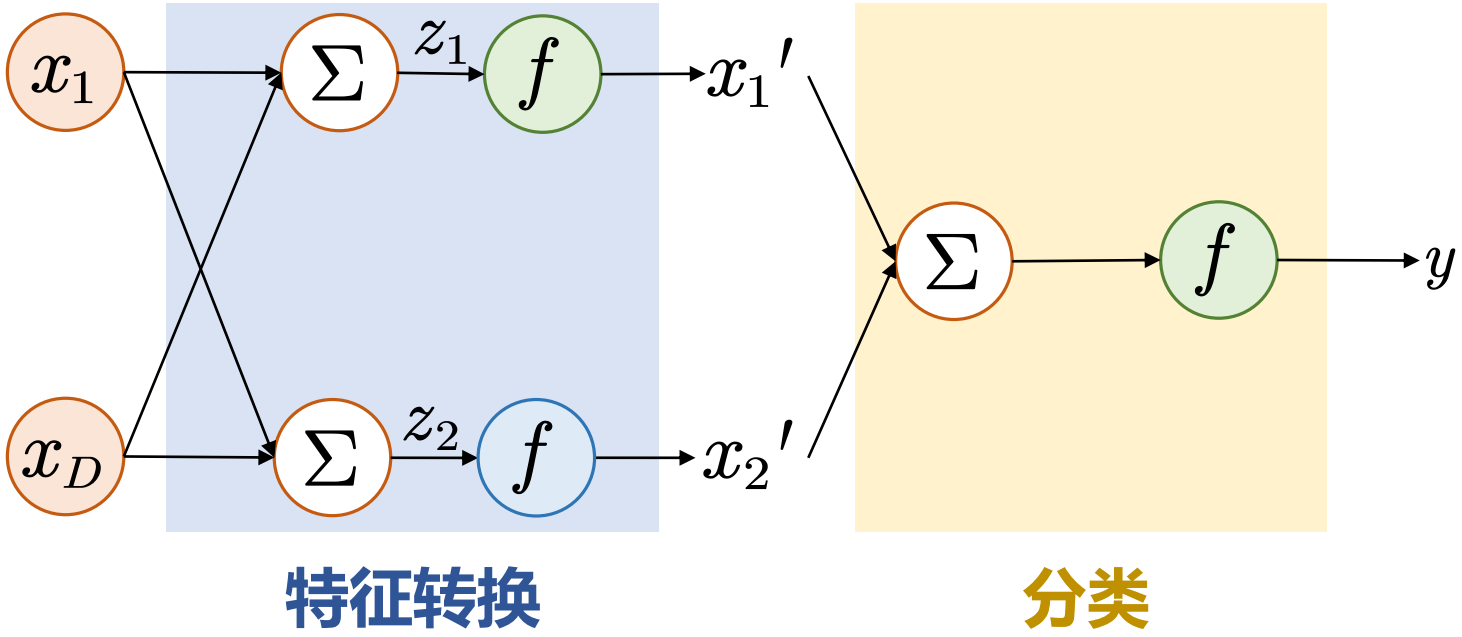
$\begin{bmatrix}x_1\\x_2\end{bmatrix}$ → $\begin{bmatrix}x_1'\\x_2'\end{bmatrix}$

并不总是有效

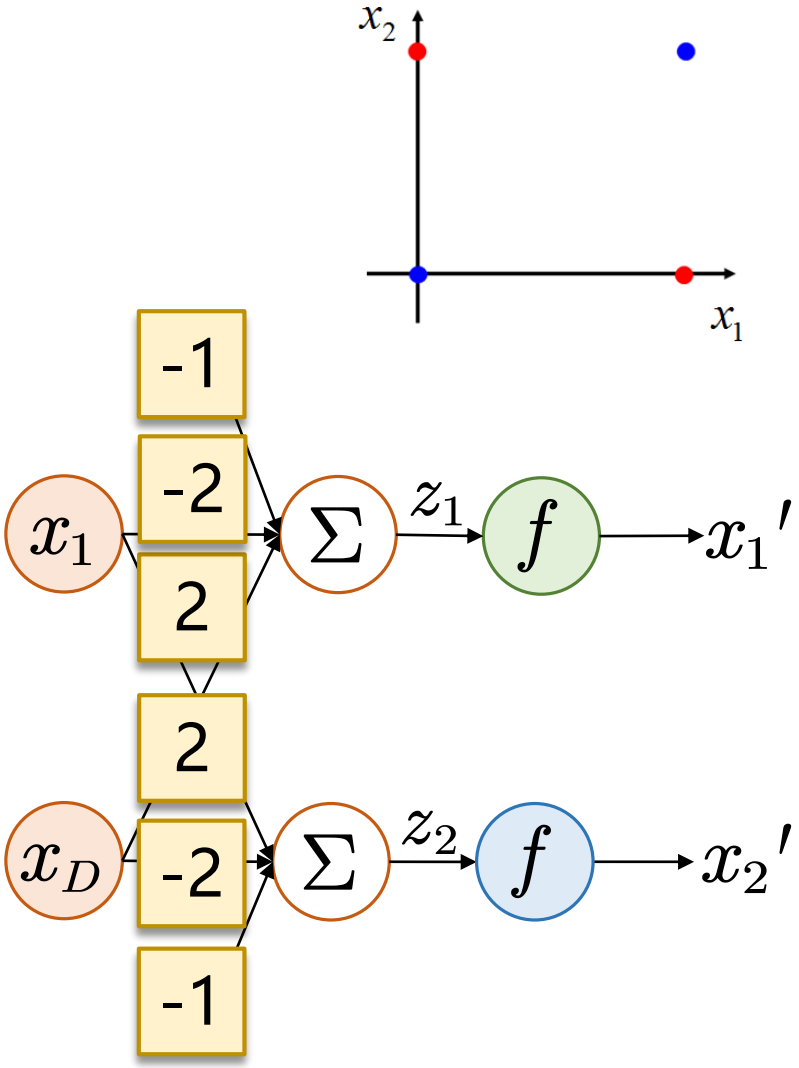


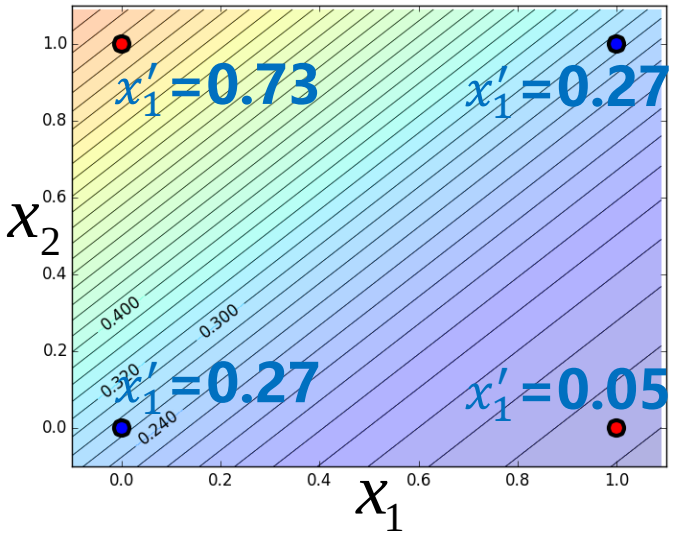
Logistic回归的局限

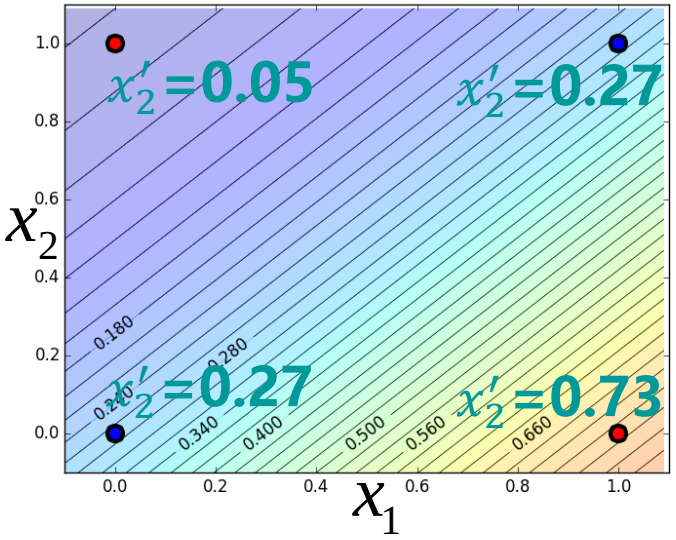
- 连接多个Logistic回归模型（图中忽略bias项）



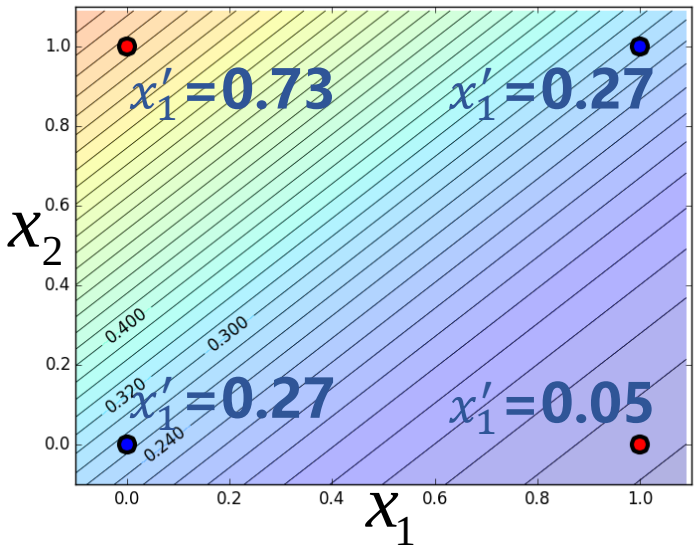
连接多个Logistic回归模型

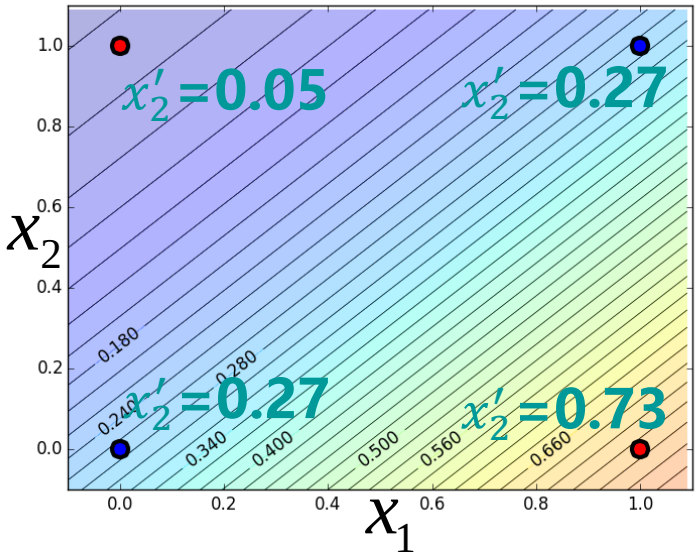


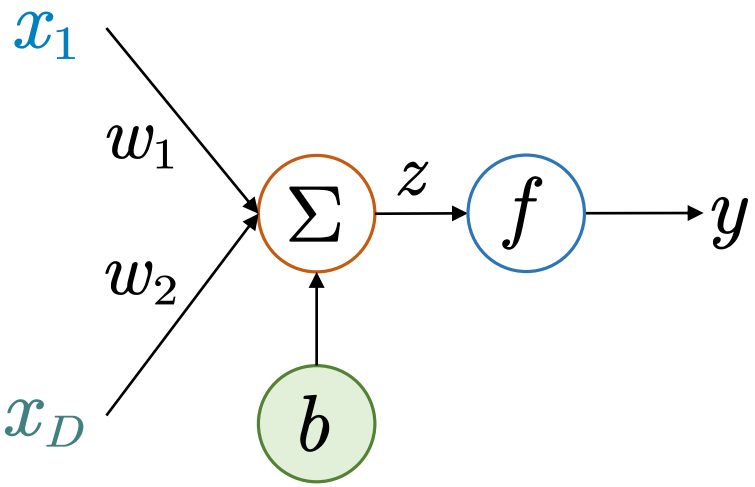


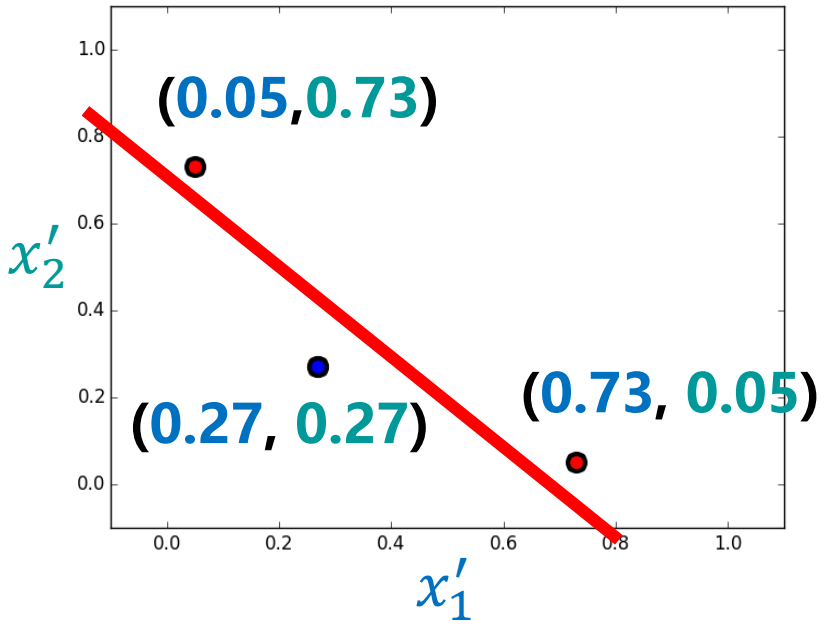


连接多个Logistic回归模型



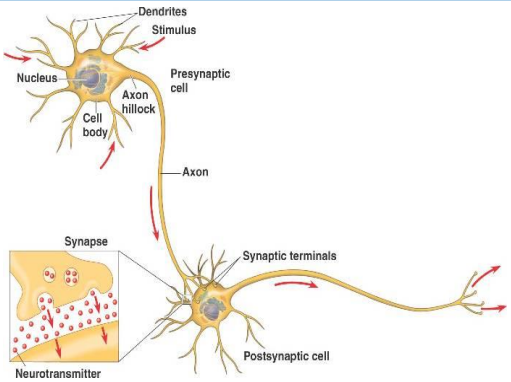


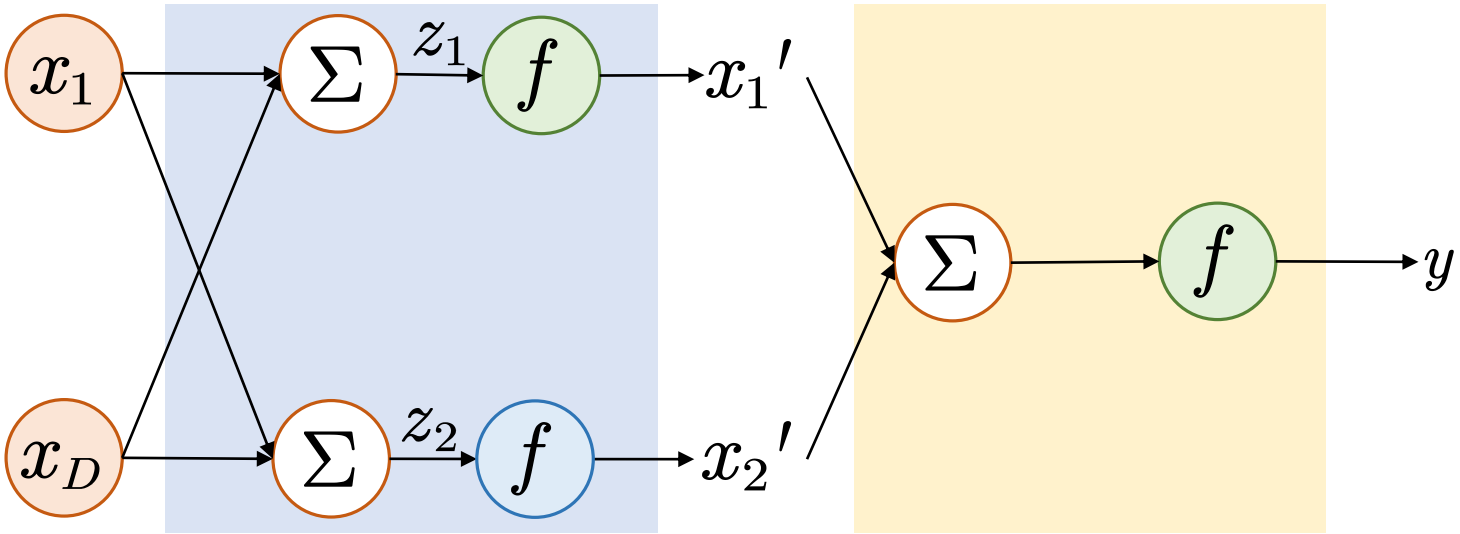




Logistic回归：深度学习方案

所有Logistic回归模型的参数一起联合学习





特征转换　　分类

神经网络