北京航空航天大學
COLLEGE OF SOFTWARE BEIHANG UNIVERSITY 软件学院

人工智能

第5讲：机器学习-有监督学习I

张晶

2023年春季

- 参考教材： 吴飞，《人工智能导论：模型与算法》，高等教育出版社
- 在线课程： https://www.icourse163.org/course/ZJU-1003377027?from=searchPage

北京航空航天大學 软件学院

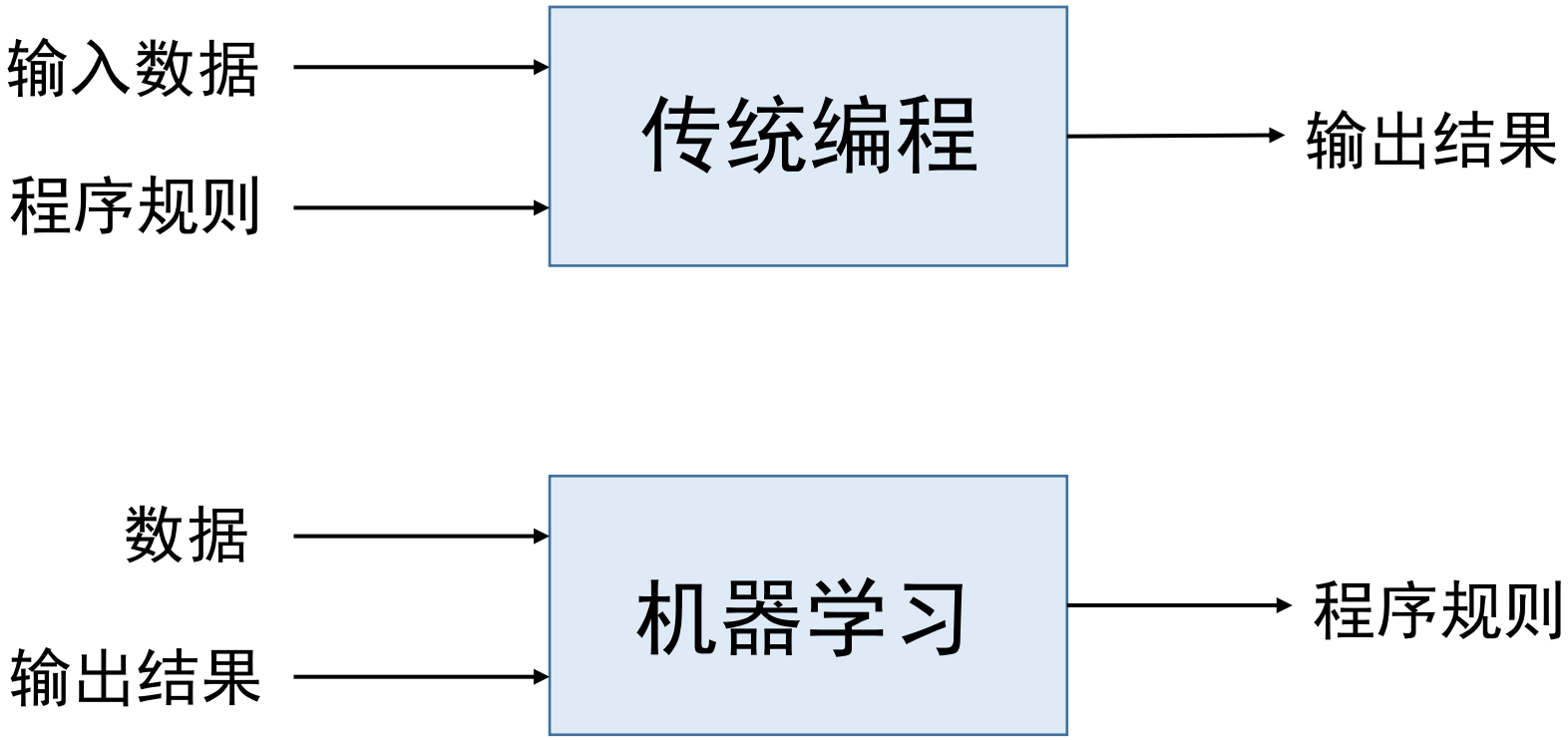
提纲

北京航空航天大學
COLLEGE OF SOFTWARE BEIHANG UNIVERSITY 软件学院

提纲

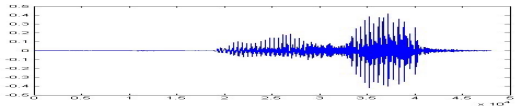
北京航空航天大學 软件学院

机器学习 v.s. 传统编程



机器学习：从数据中学习知识

- 语音识别

$f($  $)=$ “你好”

- 图像分类

$f($  $)=$ “猫”

- 围棋游戏

$f($  $)=$ “5–5”

（下一步落子位置）

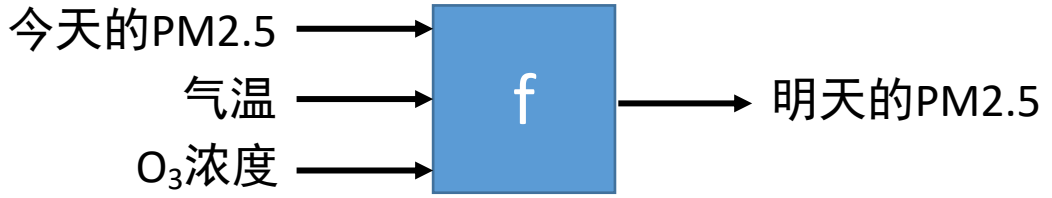
让机器去帮我们找个函数

- 从原始数据中提取特征
- 学习映射函数f（模型）
- 通过映射函数f将原始数据映射到任务空间，即寻找数据和任务目标之间的关系

机器学习的分类（按问题分类）

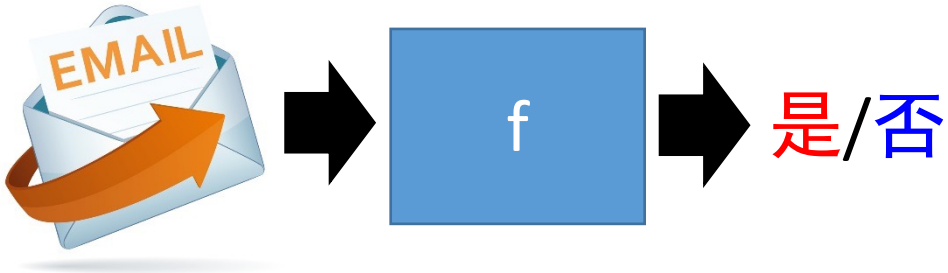
- 回归问题：函数输出标量（连续）

预测PM2.5



- 分类问题：给定选项（类别），函数输出其中一个正确选项（离散）

垃圾邮件过滤



北京航空航天大學 软件学院

机器学习的分类（按数据标注情况分类）

监督学习(supervised learning)
数据有标签、一般为回归或分类等任务

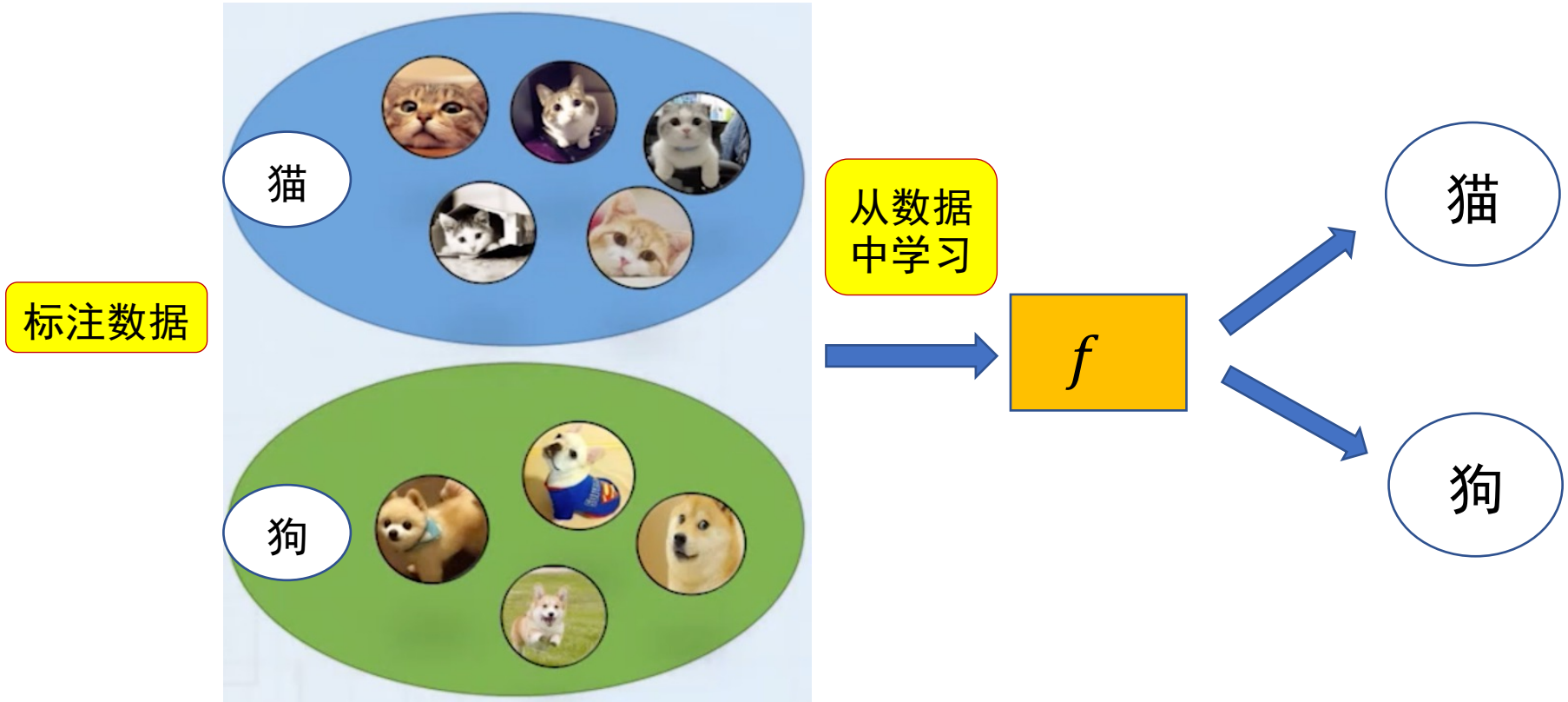
半监督学习(semi-supervised learning)
一部分数据有标签，一部分数据没有标签

无监督学习(un-supervised learning)
数据无标签、一般为聚类或若干降维任务

强化学习(reinforcement learning)
序列数据决策学习，一般为与从环境交互中学习

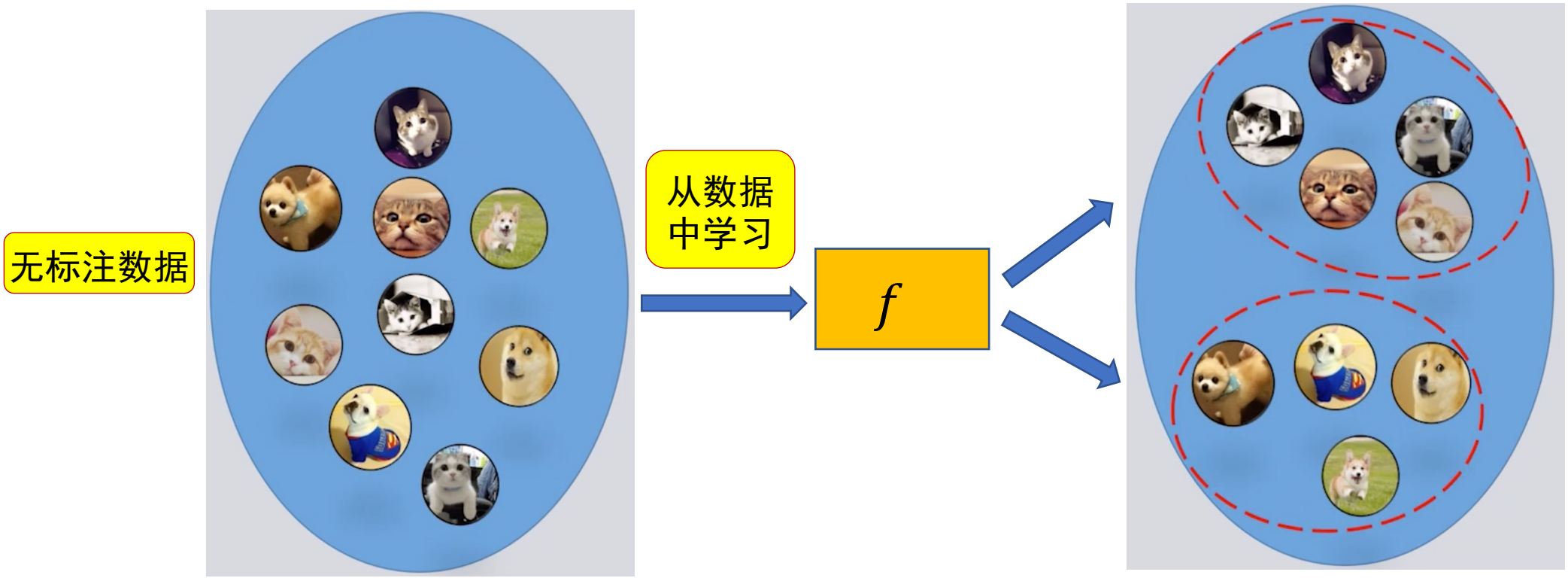
机器学习的分类（按数据标注情况分类）

- 有监督学习（supervised learning）



机器学习的分类（按数据标注情况分类）

- 无监督学习（unsupervised learning）



北京航空航天大學 软件学院

监督学习的重要元素

标注数据

学习模型

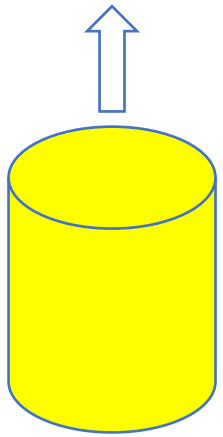
损失函数

- 标识了类别信息的数据
  学什么
- 如何学习得到映射模型
  如何学
- 如何对学习结果进行度量
  学到否

监督学习：主要要素

训练映射函数f

使得$f(x_i)$预测结果尽量等于y_i



训练数据集

$(x_i, y_i), i = 1, \ldots, n$

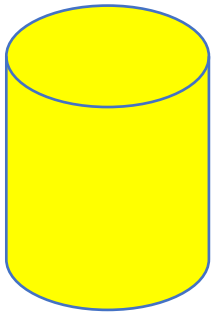
- 训练集中一共有n个标注数据，第i个标注数据记为(x_i, y_i)，
  - x_i：第i个样本数据
  - y_i：x_i的标注信息（真值，groundtruth）。
- 定义一个映射函数f（也被称为模型），f对x_i的预测结果记为$f(x_i)$。
- 损失函数$Loss$就是用来计算x_i真值y_i与预测值$f(x_i)$之间差值的函数。
- 在训练过程中希望优化映射函数，使得在训练数据集上得到“损失”之和最小，即$\min \sum_{i=1}^{n} Loss(f(x_i), y_i)$。

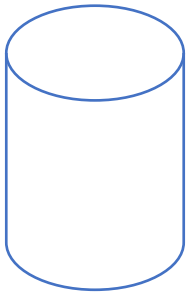
监督学习：训练数据与测试数据

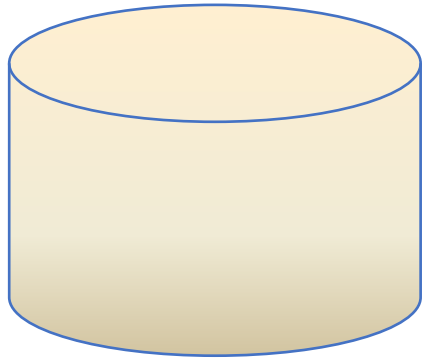
从训练数据集学习得到
映射函数f的未知参数

在测试数据集
测试映射函数f

未知数据集
上测试映射函数f







训练数据集
$(x_i, y_i), i = 1, \dots, n$

测试数据集
$(x_i', y_i'), i = 1, \dots, m$

监督学习：映射函数f

- 映射函数f（模型）：比如线性模型

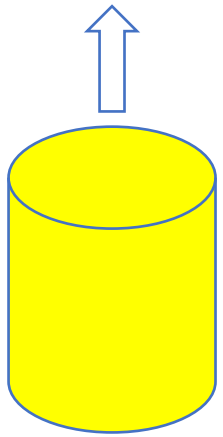
$$\hat{y}_i = f(x_i) = wx_i + b\ (1 \le i \le n)$$

  - x_i：输入样本数据
  - $\hat{y}_i$：模型输出结果
  - w 和 b 为未知参数(需要从数据中学习)
  - 基于领域知识定义模型

监督学习：损失函数

训练映射函数f

使得$f(x_i)$预测结果尽量等于y_i



训练数据集
$(x_i, y_i), i = 1, \ldots, n$

损失函数名称	损失函数定义
0-1损失函数	$Loss(y_i, f(x_i) = \begin{cases} 1, f(x_i) \neq y_i \\ 0, f(x_i) = y_i \end{cases}$
平方损失函数	$Loss(y_i, f(x_i)) = (y_i - f(x_i))^2$
绝对损失函数	$Loss(y_i, f(x_i)) = \lvert y_i - f(x_i) \rvert$
对数损失函数/对数似然损失函数	$Loss(y_i, P(y_i \lvert x_i)) = -logP((y_i \lvert x_i)$

典型的损失函数

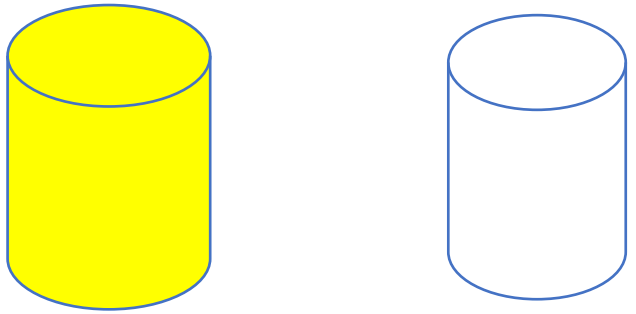
监督学习：经验风险与期望风险

从训练数据集学习得到映射函数f

在测试数据集测试映射函数f

经验风险(empirical risk)

- 训练集中数据产生的损失。经验风险越小说明学习模型对训练数据拟合程度越好。



期望风险(expected risk):

- 当测试集中存在无穷多数据时产生的损失。期望风险越小，学习所得模型越好。

训练数据集
$(x_i, y_i), i = 1, \ldots, n$

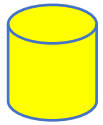
测试数据集
$(x_i', y_i'), i = 1, \ldots, m$

监督学习：经验风险与期望风险

映射函数训练目标：经验风险最小化
(empirical risk minimization，ERM)

$$\min_{f\in\Phi}\frac{1}{n}\sum_{i=1}^{n}Loss(y_i,f(x_i))$$

选取一个使得训练集所有数据损失平均值最小的映射函数。这样的考虑是否够？



训练数据集
$(x_i,y_i),i=1,\dots,n$

映射函数训练目标：期望风险最小化
(expected risk minimization)

$$\min_{f\in\Phi}\int_{x\times y}Loss(y,f(x))P(x,y)dxdy$$

测试数据集数据无穷多
$(x_i',y_i'),i=1,\dots,\infty$

- 期望风险是模型关于联合分布期望损失，经验风险是模型关于训练样本集平均损失。
- 根据大数定律，当样本容量趋于无穷时，经验风险趋于期望风险。所以在实践中很自然用经验风险来估计期望风险。
- 由于现实中训练样本数目有限，用经验风险估计期望风险并不理想，要对经验风险进行一定的约束。

监督学习：“过学习(over-fitting)”与“欠学习(under-fitting)”

- 经验风险最小化

$$\min_{f\in\Phi}\frac{1}{n}\sum_{i=1}^{n}Loss(y_i,f(x_i))$$

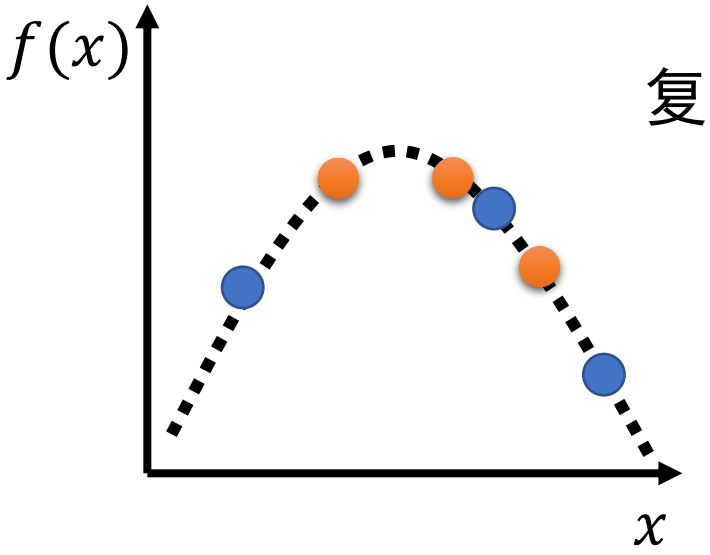
- 期望风险最小化

$$\min_{f\in\Phi}\int_{x\times y}Loss\big(y,f(x)\big)P(x,y)dxdy$$

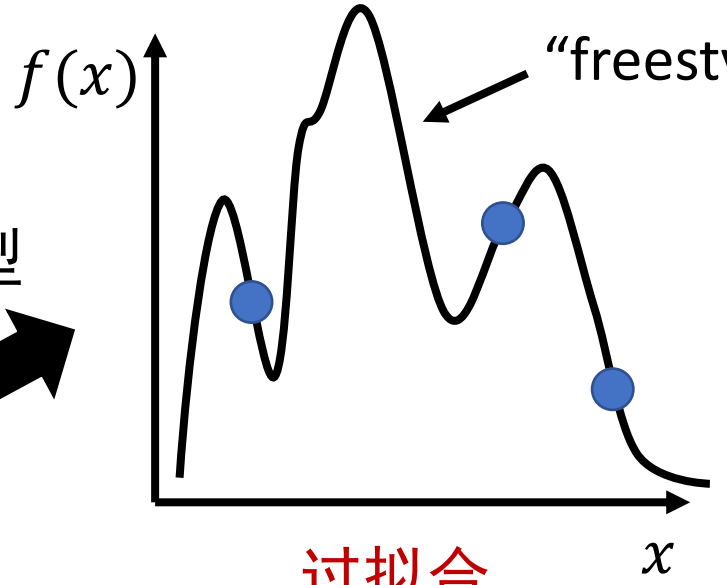
经验风险小 （训练集上表现好）	期望风险小 （测试集上表现好）	泛化能力强
经验风险小 （训练集上表现好）	期望风险大 （测试集上表现不好）	过学习 （模型过于复杂）
经验风险大 （训练集上表现不好）	期望风险大 （测试集上表现不好）	欠学习
经验风险大 （训练集上表现不好）	期望风险小 （测试集上表现好）	“神仙算法”或 “黄粱美梦”

表4.3 模型泛化能力与经验风险、期望风险的关系

监督学习："过学习(over-fitting)"与"欠学习(under-fitting)"

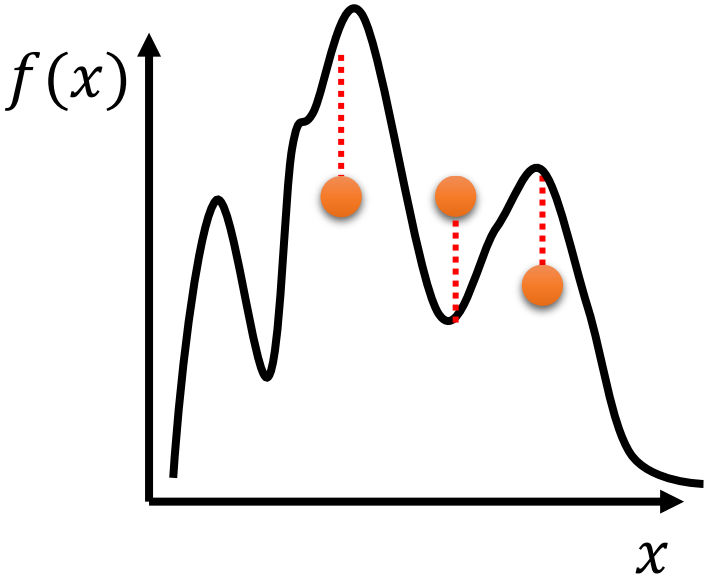


复杂模型



经验风险小

过拟合



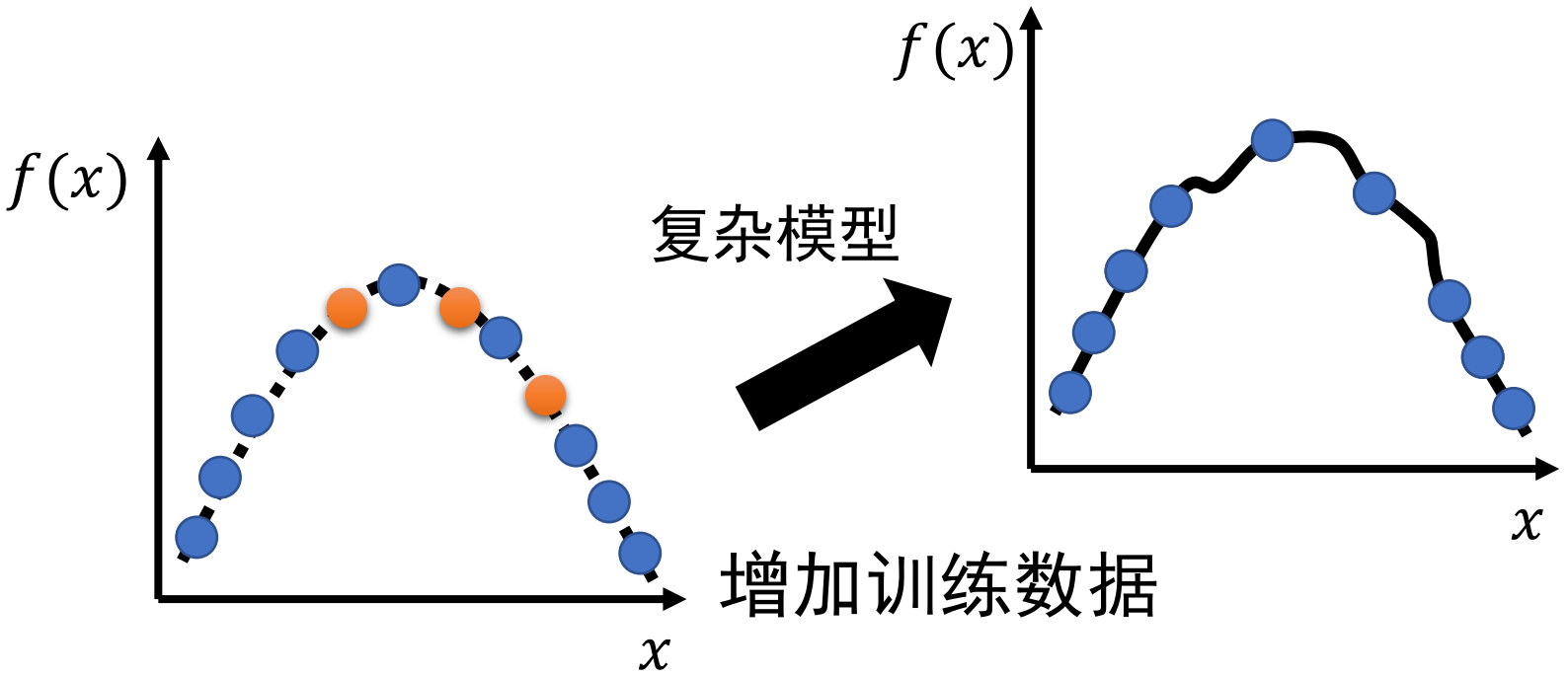
期望风险大

真实数据分布
(不可观测)

训练数据

测试数据

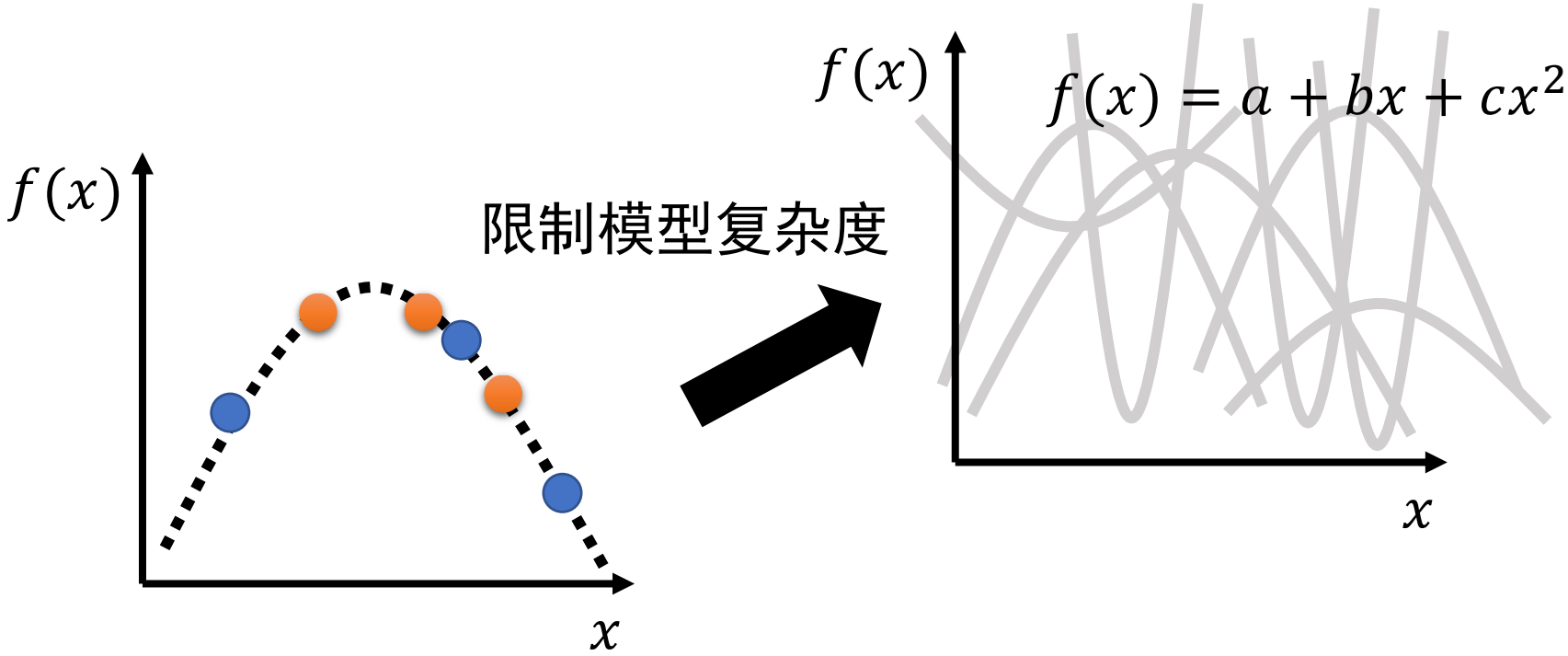
监督学习:“过学习(over-fitting)”与“欠学习(under-fitting)”



经验风险小

期望风险小

监督学习:“过学习(over-fitting)”与“欠学习(under-fitting)”

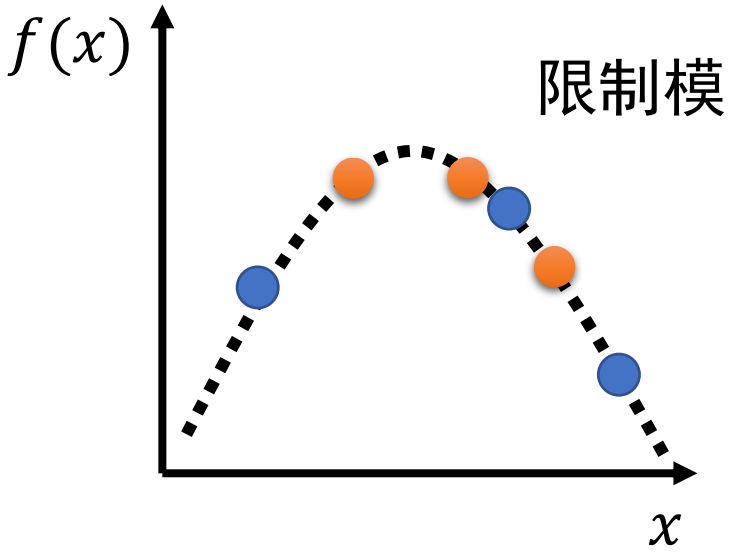


真实数据分布
(不可观测)

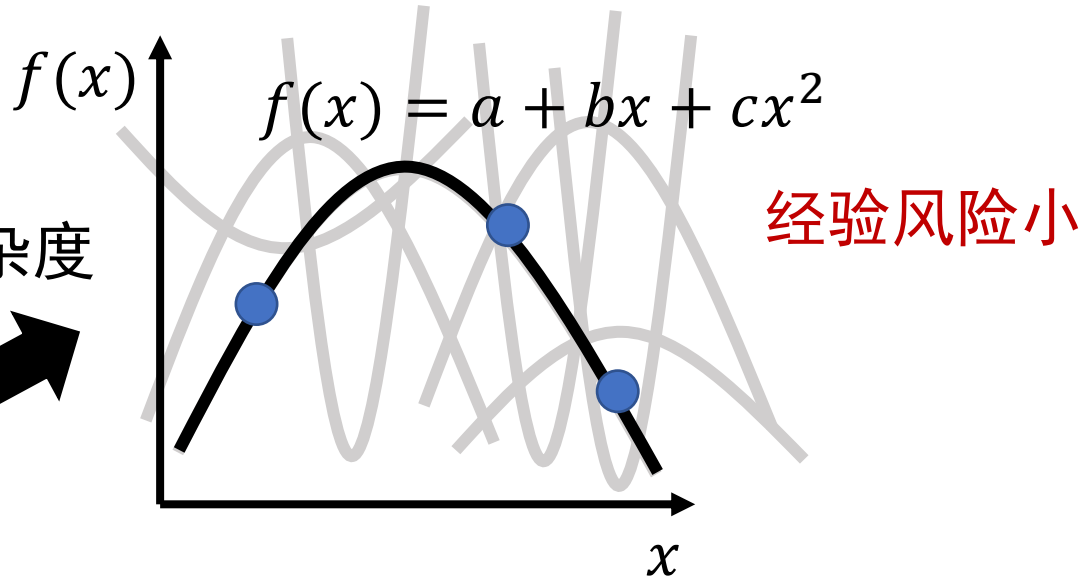
训练数据

测试数据

监督学习：“过学习(over-fitting)”与“欠学习(under-fitting)”



限制模型复杂度

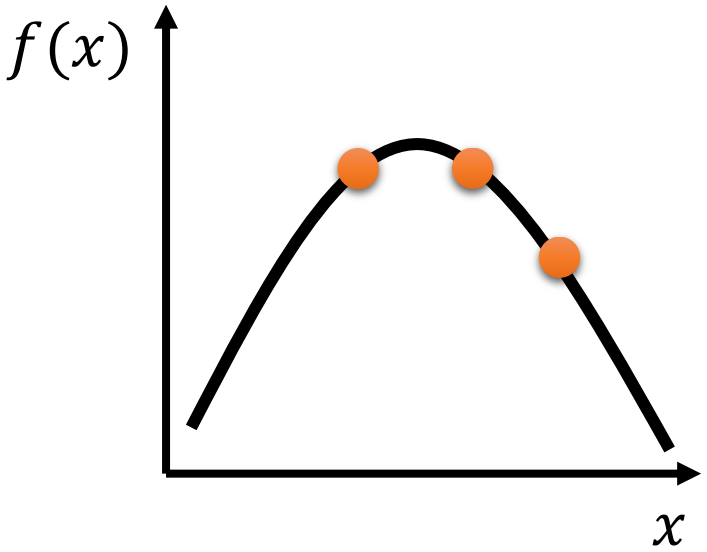


经验风险小

真实数据分布
(不可观测)

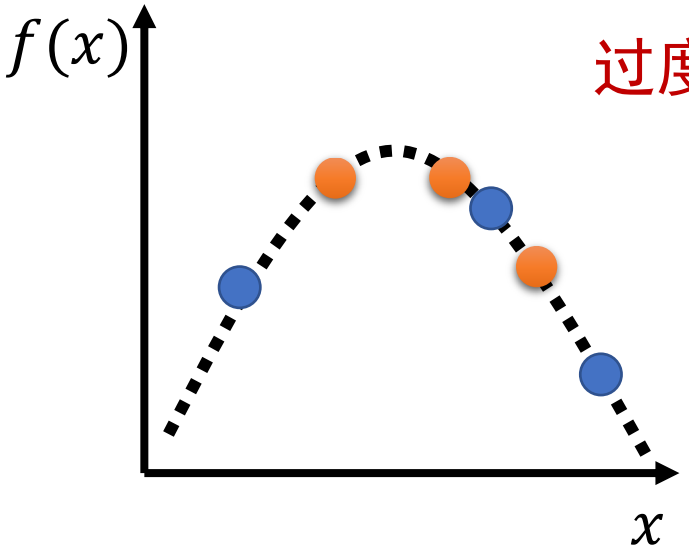
训练数据

测试数据

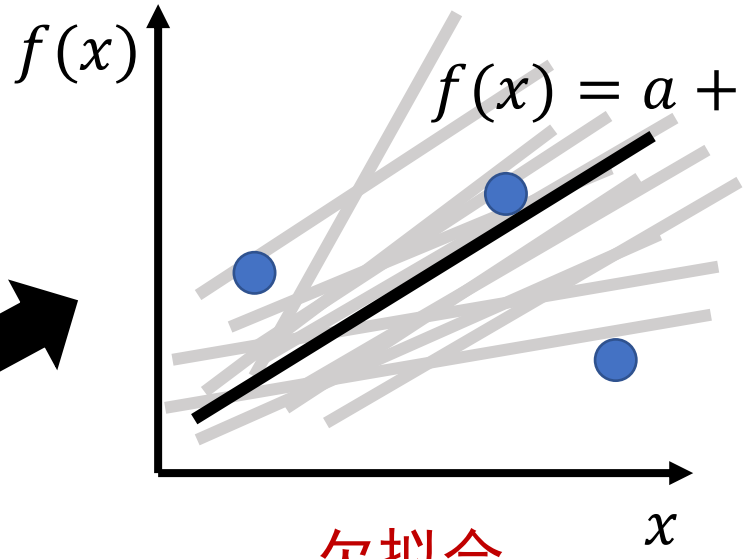


期望风险小

监督学习：“过学习(over-fitting)”与“欠学习(under-fitting)”

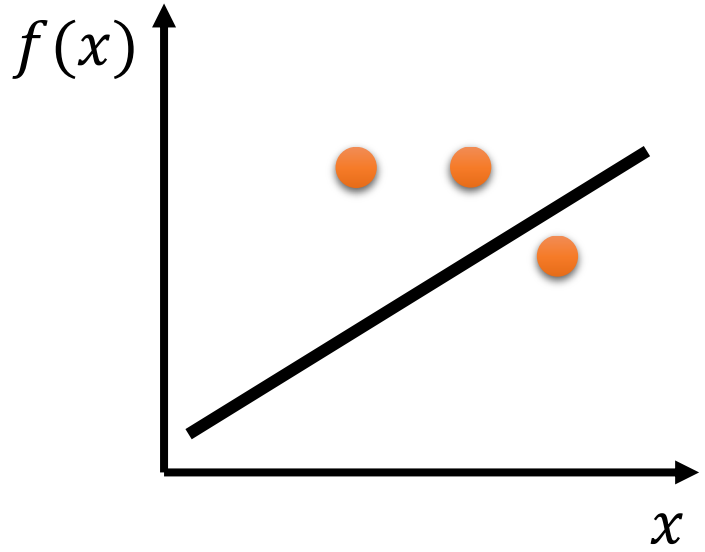


过度约束



经验风险大

欠拟合 …



期望风险大

真实数据分布
(不可观测)

训练数据

测试数据

监督学习: 结构风险最小

- 经验风险最小化：仅反映了局部数据

$$\min_{f\in\Phi}\frac{1}{n}\sum_{i=1}^{n}Loss(y_i,f(x_i))$$

- 期望风险最小化：无法得到全量数据

$$\min_{f\in\Phi}\int_{x\times y}Loss\big(y,f(x)\big)P(x,y)dxdy$$

- 结构风险最小化(structural risk minimization)：为了防止过拟合，在经验风险上加上表示模型复杂度的正则化项(regulatizer)或惩罚项(penalty term)：

$$\min_{f\in\Phi}\frac{1}{n}\sum_{i=1}^{n}Loss(y_i,f(x_i)) + \lambda J(f)$$

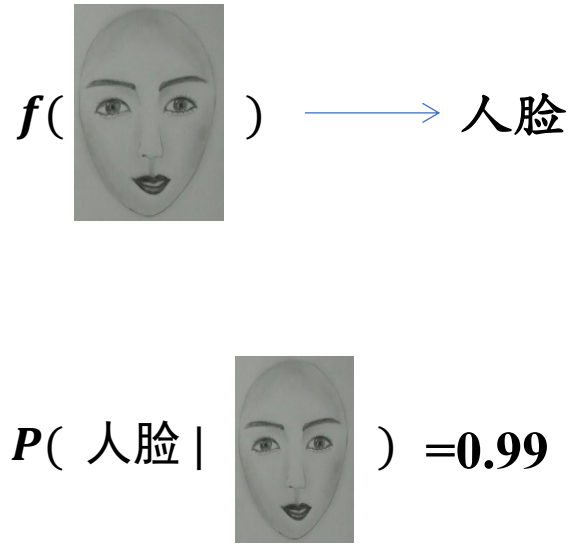
经验风险　　模型复杂度

在最小化经验风险与降低模型复杂度之间寻找平衡

监督学习两种方法：判别模型与生成模型

监督学习方法又可以分为生成方法(generative approach)和判别方法(discriminative approach)。所学到的模型分别称为生成模型(generative model)和判别模型(discriminative model).

- 判别方法直接学习判别函数$f(X)$或者条件概率分布$P(Y|X)$作为预测的模型，即判别模型。
- 判别模型关心在给定输入数据下，预测该数据的输出是什么。
- 典型判别模型包括回归模型、神经网络、支持向量机和Ada boosting等。



监督学习两种方法：判别模型与生成模型

- 生成模型从数据中学习联合概率分布$P(X,Y)$（通过似然概率$P(X|Y)$ 和类概率$P(Y)$ 的乘积来求取）

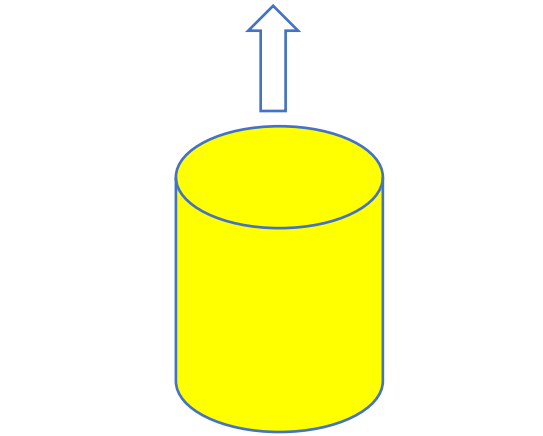
  $P(Y|X)=\frac{P(X,Y)}{P(X)}$ 或者 $P(Y|X)=\frac{P(X|Y)\times P(Y)}{P(X)}$
- 典型方法为贝叶斯方法、隐马尔可夫链
- 授之于鱼、不如授之于“渔”
- 联合分布概率$P(X,Y)$或似然概率$P(X|Y)$求取很困难

似然概率：计算导致样本X出现的模型参数值

$$P(Y|X)=\frac{P(X|Y)\times P(Y)}{P(X)}$$

监督学习

训练映射函数f



训练数据集

$(x_i, y_i), i = 1, \ldots, n$

- 回归
- 分类
- 识别
- 推荐
- 生成
- …

提纲

线性回归 (linear regression)

- 在现实生活中，往往需要分析若干变量之间的关系，如碳排放量与气候变暖之间的关系、某一商品广告投入量与该商品销售量之间的关系等，这种分析不同变量之间存在关系的研究叫回归分析，刻画不同变量之间关系的模型被称为回归模型。如果这个模型是线性的，则称为线性回归模型。

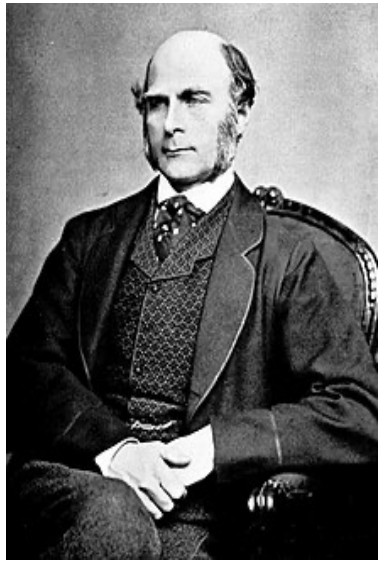
- 一旦确定了回归模型，就可以进行预测等分析工作，如从碳排放量预测气候变化程度、从广告投入量预测商品销售量等。

线性回归 (linear regression)

$$y = 33.73(\text{英寸}) + 0.516x$$

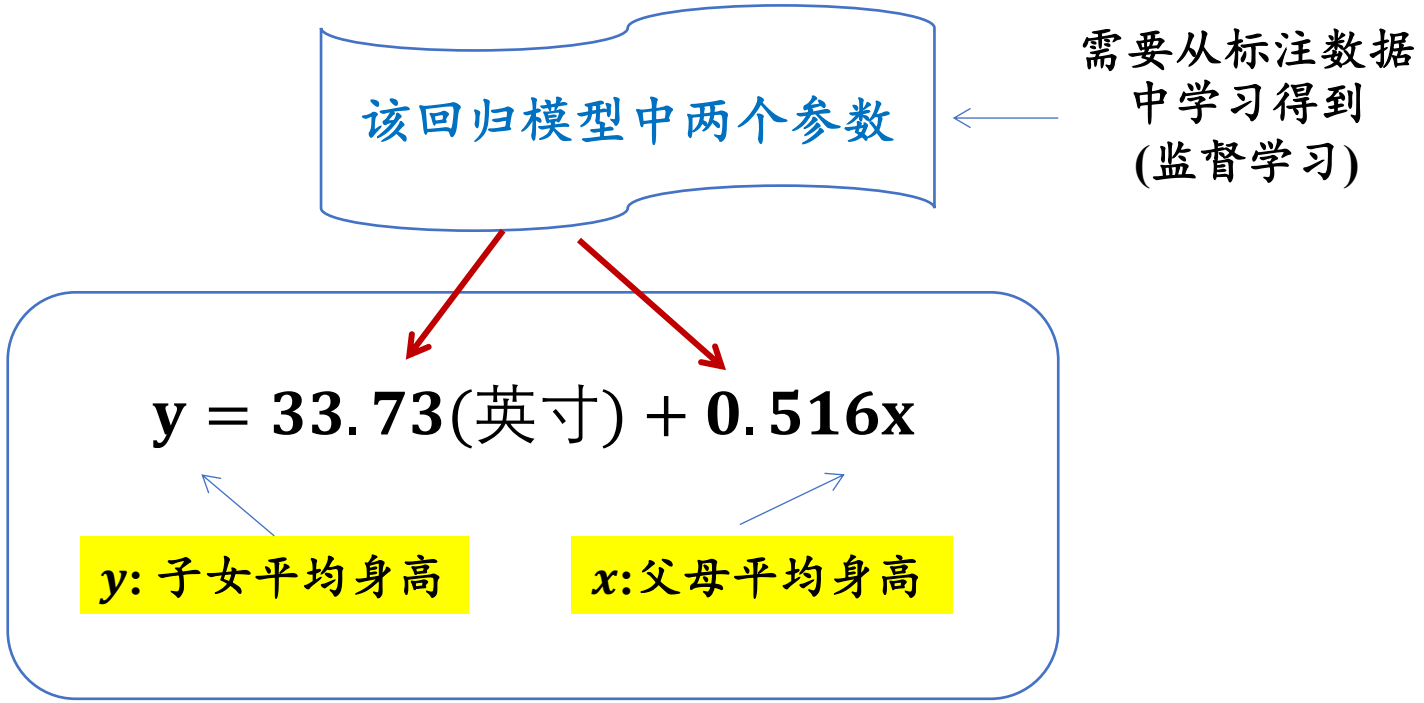
y: 子女平均身高　　x:父母平均身高

- 父母平均身高每增加一个单位，其成年子女平均身高增加0.516个单位。另外，他还发现一种“衰退(regression)”效应（“回归”到正常人平均身高）。
- 虽然x和y之间并不总是具有“衰退”（回归）关系，但是“线性回归”这一名称就保留了下来了。



英国著名生物学家兼统计学家高尔顿
Sir Francis Galton (1822-1911)

线性回归 (linear regression)



- 给出任意一对父母平均身高，则可根据上述方程，计算得到其子女平均身高
- 从父母平均身高来预测其子女平均身高
- 如何求取上述线性方程（预测方程）的参数？

线性回归：一元线性回归

一元线性回归模型例子

下表给出了芒提兹尼欧（Montesinho）地区发生森林火灾的部分历史数据，表中列举了每次发生森林火灾时的气温温度取值x和受到火灾影响的森林面积y。

气温温度x	5.1	8.2	11.5	13.9	15.1	16.2	19.6	23.3
火灾影响面积y	2.14	4.62	8.24	11.24	13.99	16.33	19.23	28.74

可否对气温温度与火灾所影响的森林面积之间关系进行建模呢？初步观察之后，可以使用简单的线性模型构建两者之间关系，即气温温度x与火灾所影响的森林面积y之间存在$y = ax + b$形式的关系。

线性回归：一元线性回归

一元线性回归模型例子

气温温度x	5.1	8.2	11.5	13.9	15.1	16.2	19.6	23.3
火灾影响面积y	2.14	4.62	8.24	11.24	13.99	16.33	19.23	28.74

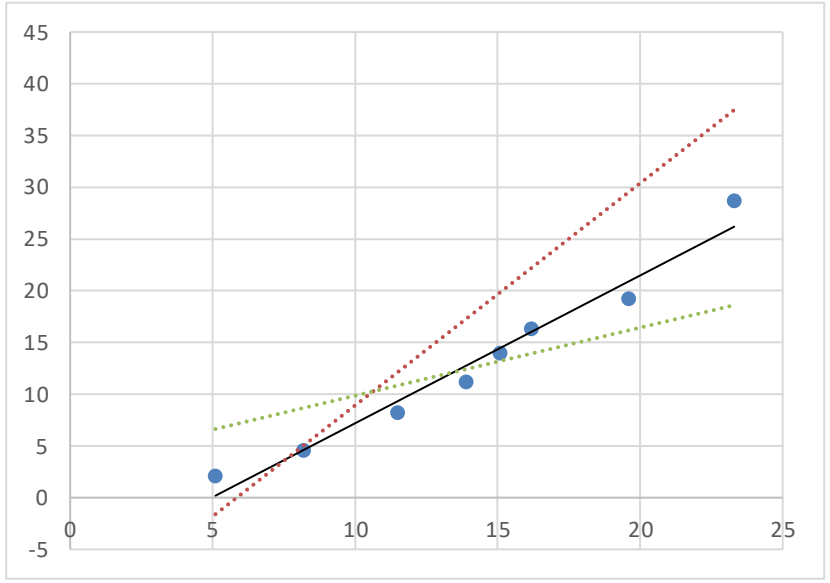


图4. 2 气温温度取值和受到火灾影响森林面积之间的一元线性回归模型（实线为最佳回归模型）

线性回归模型：$f(x) = wx + b$

- 求取：最佳回归模型是最小化残差平方和的均值
- 即要求8组(x, y)数据得到的残差平均值$\frac{1}{N}\sum(y - f(x))^2$最小。
- 残差平均值最小只与参数w和b有关，最优解即是使得残差最小所对应的w和b的值。

线性回归：一元线性回归

线性回归模型参数求取：$f(x_i) = wx_i + b\ (1 \le i \le n)$

- 记在当前参数下第i个训练样本x_i的预测值为$\hat{y}_i = f(x_i)$
- x_i的标注值（实际值）y_i与预测值$\hat{y}_i$之差记为$(y_i - \hat{y}_i)^2$
- 训练集中n个样本所产生误差总和为：$L(w,b) = \sum_{i=1}^{n}(y_i - wx_i - b\ )^2$

目标：寻找一组w和b，使得误差总和$L(w,b)$值最小。在线性回归中，解决如此目标的方法叫最小二乘法。

一般而言，要使函数具有最小值，可对$L(w,b)$ 参数w和b分别求导，令其导数值为零，再求取参数w和b的取值。

线性回归：一元线性回归

回归模型参数求取：$f(x_i) = wx_i + b\ (1 \le i \le n)$　　$\min\limits_{a,b} L(w,b) = \sum\limits_{i=1}^{n}(y_i - wx_i - b)^2$

对b求偏导　　$\dfrac{\partial L(w,b)}{\partial b} = \sum\limits_{i=1}^{n} 2(y_i - wx_i - b)(-1) = 0$

$$\rightarrow \sum_{i=1}^{n}(y_i - wx_i - b) = 0$$

$$\rightarrow \sum_{i=1}^{n}(y_i) - w\sum_{i=1}^{n}x_i - \sum_{i=1}^{n}b = 0$$

$\rightarrow n\bar{y} - wn\bar{x} - nb = 0$ ⇨ $b = \bar{y} - w\bar{x}$

线性回归：一元线性回归

回归模型参数求取：$f(x_i) = wx_i + b\ (1 \le i \le n)$

$$\min_{a,b} L(w,b) = \sum_{i=1}^{n} (y_i - wx_i - b)^2$$

对w求偏导 $\dfrac{\partial L(w,b)}{\partial w} = \sum_{i=1}^{n} 2(y_i - wx_i - b)(-x_i) = 0$

将$b = \bar{y} - w\bar{x}$ $\left(\bar{y} = \dfrac{\sum_{i=1}^{n} y_i}{n},\ \bar{x} = \dfrac{\sum_{i=1}^{n} x_i}{n}\right)$

代入上式

$$\rightarrow \sum_{i=1}^{n} (y_i - wx_i - \bar{y} + w\bar{x})(x_i) = 0$$

$$\rightarrow \sum_{i=1}^{n} (y_i x_i - wx_i x_i - \bar{y}x_i + w\bar{x}x_i) = 0$$

$$\rightarrow \sum_{i=1}^{n} (y_i x_i - \bar{y}x_i) - w\sum_{i=1}^{n} (x_i x_i - \bar{x}x_i) = 0$$

$$\rightarrow (\sum_{i=1}^{n} x_i y_i - n\bar{x}\bar{y}) - w(\sum_{i=1}^{n} x_i x_i - n\bar{x}^2) = 0$$

$$w = \frac{\sum_{i=1}^{n} x_i y_i - n\bar{x}\,\bar{y}}{\sum_{i=1}^{n} x_i x_i - n\bar{x}^2}$$

线性回归：一元线性回归

气温温度x	5.1	8.2	11.5	13.9	15.1	16.2	19.6	23.3
火灾影响面积y	2.14	4.62	8.24	11.24	13.99	16.33	19.23	28.74

回归模型参数求取：$f(x_i) = wx_i + b\ (1 \le i \le n)$

$$\min_{a,b} L(w,b) = \sum_{i=1}^{n} (y_i - wx_i - b)^2$$

可以看出：只要给出了训练样本$(x_i, y_i)(i = 1, \dots, n)$，我们就可以从训练样本出发，建立一个线性回归方程，使得对训练样本数据而言，该线性回归方程预测的结果与样本标注结果之间的差值和最小。

$$b = \bar{y} - w\bar{x}$$

$$w = \frac{\sum_{i=0}^{n} x_i y_i - n\bar{x}\,\bar{y}}{\sum_{i=1}^{n} x_i x_i - n\bar{x}^2}$$

$$w = \frac{x_1 y_1 + x_2 y_2 + \cdots + x_8 y_8 - 8\bar{x}\bar{y}}{x_1^2 + x_2^2 + \cdots + x_8^2 - 8\bar{x}^2} = 1.428$$

$$b = \bar{y} - w\bar{x} = -7.09$$

即预测芒提兹尼欧地区火灾所影响森林面积与气温温度之间的一元线性回归模型为“火灾所影响的森林面积 $= 1.428\times$气温温度 $- 7.09$”，即$y = 1.428x - 7.09$

线性回归：多元线性回归

多元线性回归模型例子

接下来扩展到数据特征的维度是多维的情况，在上述数据中增加一个影响火灾影响面积的潜在因素—风力。

气温$x_{i,1}$	5.1	8.2	11.5	13.9	15.1	16.2	19.6	23.3
风力$x_{i,2}$	4.5	5.8	4	6.3	4	7.2	6.3	8.5
火灾影响面积y	2.14	4.62	8.24	11.24	13.99	16.33	19.23	28.74

多维数据特征中线性回归的问题定义如下：假设总共有m个训练数据$\{(x_i, y_i)\}_{i=1}^{m}$，其中$x_i = [x_{i,1}, x_{i,2}, \dots, x_{i,D}] \in \mathbb{R}^D$，$D$为数据特征的维度，线性回归就是要找到一组参数$\boldsymbol{w} = [w_0, w_1, \dots, w_D]$，使得线性函数：

$$f(\boldsymbol{x_i}) = w_0 + \sum_{j=1}^{D} w_j \, x_{i,j} = w_0 + \boldsymbol{w^T x_i}$$

线性回归：多元线性回归

最小化均方误差函数：

$$J_m = \frac{1}{m}\sum_{i=1}^{m}\left(y_i - f(\boldsymbol{x_i})\right)^2$$

为了方便，使用矩阵来表示所有的训练数据和数据标签。

$$X = [\boldsymbol{x_1}, \dots, \boldsymbol{x_m}], \qquad \boldsymbol{y} = [y_1, \dots, y_m]$$

其中每一个数据x_i会扩展一个维度，其值为1，对应参数a_0。均方误差函数可以表示为：

$$J_m(\boldsymbol{w}) = (\boldsymbol{y} - X^T\boldsymbol{w})^T(\boldsymbol{y} - X^T\boldsymbol{w})$$

均方误差函数$J_n(\boldsymbol{w})$对所有参数$\boldsymbol{w}$求导可得：

$$\nabla J(\boldsymbol{w}) = -2X(\boldsymbol{y} - X^T\boldsymbol{w})$$

因为均方误差函数$J_n(\boldsymbol{w})$是一个二次的凸函数，所以函数只存在一个极小值点，也同样是最小值点，所以令$\nabla J(\boldsymbol{w}) = 0$可得

$$XX^T\boldsymbol{w} = X\boldsymbol{y}$$

$$\boldsymbol{w} = (XX^T)^{-1}X\boldsymbol{y}$$

线性回归：多元线性回归

对于上面的例子，转化为矩阵的表示形式为：

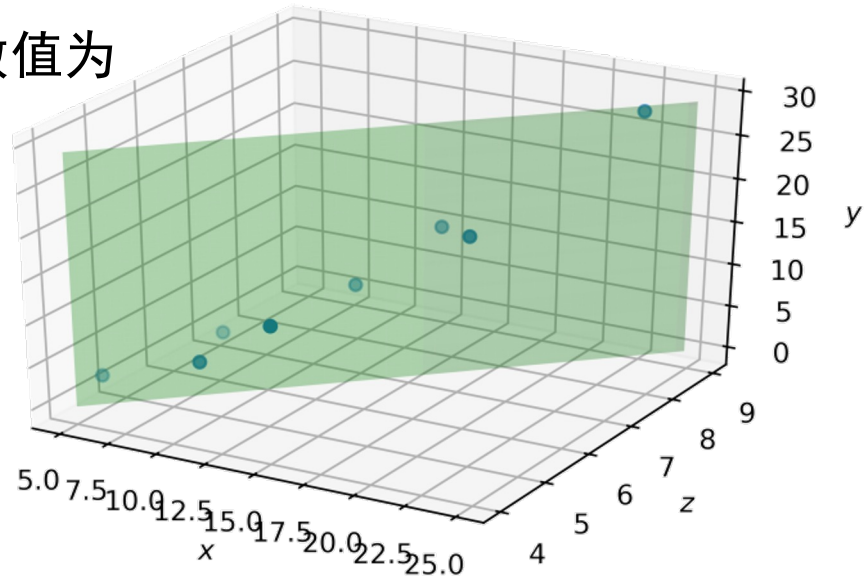
$$X = \begin{matrix} \boldsymbol{x_1} & \boldsymbol{x_2} & \cdots \\ \end{matrix} \begin{bmatrix} 5.1 & 8.2 & 11.5 & 13.9 & 15.1 & 16.2 & 19.6 & 23.3 \\ 4.5 & 5.8 & 4. & 6.3 & 4. & 7.2 & 6.3 & 8.5 \\ 1. & 1. & 1. & 1. & 1. & 1. & 1. & 1. \end{bmatrix} \begin{matrix} x_{i,1} \\ x_{i,2} \\ \\ \end{matrix}$$

$$\boldsymbol{y} = [2.14 \quad 4.62 \quad 8.24 \quad 11.24 \quad 13.99 \quad 16.33 \quad 19.23 \quad 28.74]^{\mathrm{T}}$$

其中矩阵X多出一行全1，是因为常数项a_0，可以看作是数值为全1的特征的对应系数。计算可得

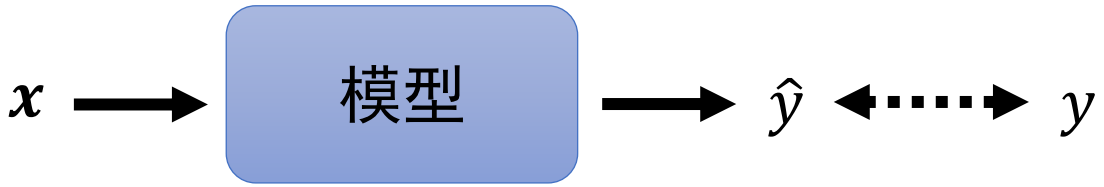
$$\boldsymbol{w} = [1.312 \quad 0.626 \quad -9.103]$$

$$f(x_i) = -9.103 + 1.312x_{i,1} + 0.626x_{i,2}$$

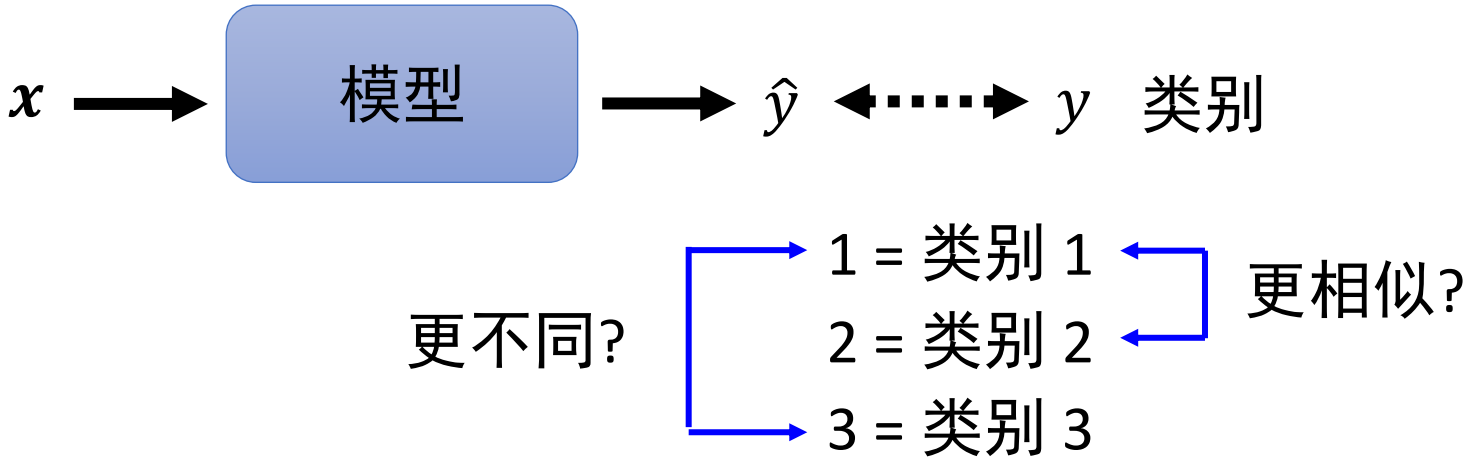


线性分类：用线性回归做分类？

- 线性回归

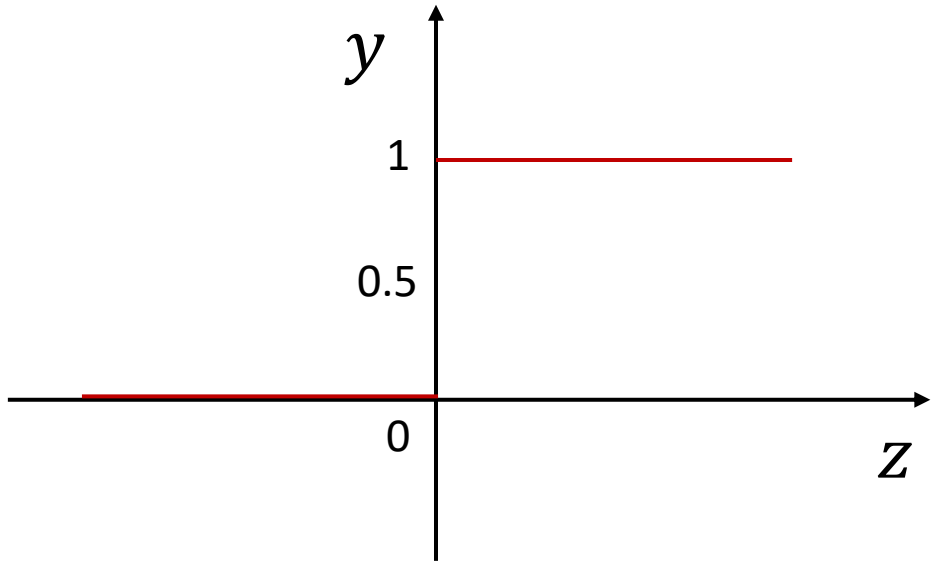


- 用线性回归做分类?



线性分类

- 线性归回产生的实值输出：$z_i = wx_i + b$
- 线性分类期望的输出： $y_i \in \{0,1\}$
- $y_i = \begin{cases} 1 & z_i > 0 \\ 0.5 & z_i = 0 \\ 0 & z_i < 0 \end{cases}$



北京航空航天大學 软件学院

线性分类：广义线性模型

- 线性模型：通过特征的线性组合来进行预测的函数

$$\hat{y} = f(x) = \boldsymbol{w}^{\boldsymbol{T}}\boldsymbol{x} + b$$

- 广义线性模型：

$$\hat{y} = g^{-1}(\boldsymbol{w}^{\boldsymbol{T}}\boldsymbol{x} + b)$$

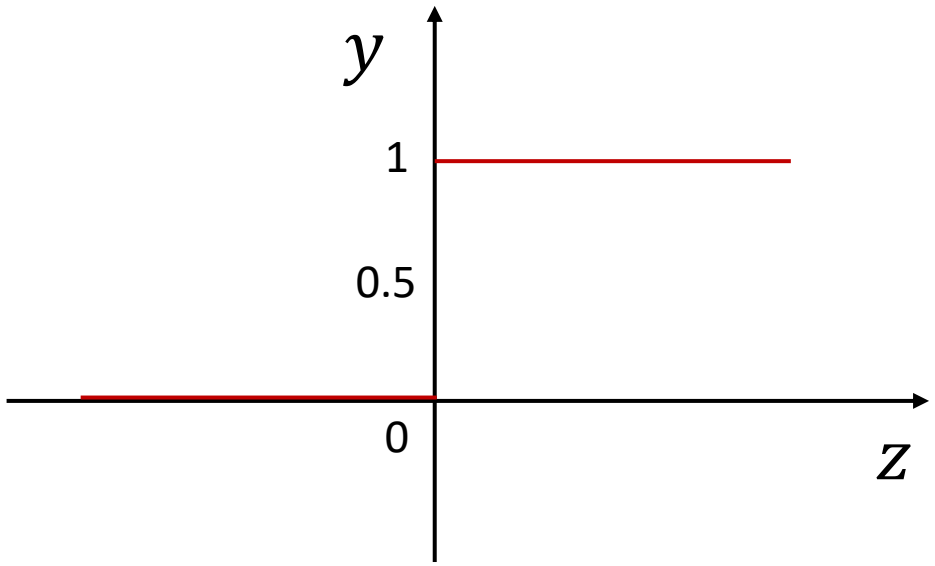
- 线性模型用于分类：
  - 找一个单调可微函数将分类任务的真实标记y与线性回归模型的预测值联系起来。
  - 比如：

$$\ln\left(\frac{\hat{y}}{1-\hat{y}}\right) = \boldsymbol{w}^{\boldsymbol{T}}\boldsymbol{x} + b$$

即

$$\hat{y} = \frac{1}{1 + e^{-(\boldsymbol{w}^{T}\boldsymbol{x}+b)}}$$

线性分类：对数几率回归 (Logistic Regression)



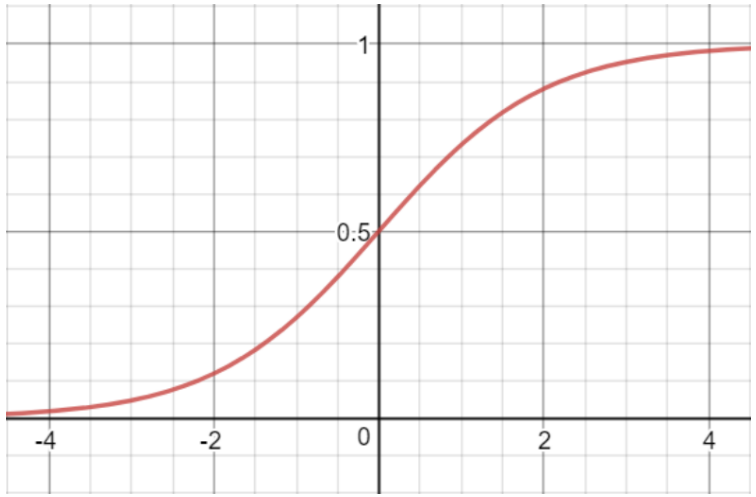


图4.6 Sigmoid函数

- 对数几率回归(logistic regression)就是在回归模型中引入 sigmoid函数的一种广义线性模型。
- Logistic回归模型可如下表示：

$$\hat{y} = \frac{1}{1+e^{-z}} = \frac{1}{1+e^{-(\boldsymbol{w}^T\boldsymbol{x}+b)}} \quad ，其中\hat{y} \in (0,1), z = \boldsymbol{w}^T\boldsymbol{x} + \mathrm{b}$$

- 这里$\frac{1}{1+e^{-z}}$是sigmoid函数、$\boldsymbol{x} \in \mathbb{R}^d$是输入数据、$\boldsymbol{w} \in \mathbb{R}^d$和$b \in \mathbb{R}$是回归函数的参数。

线性分类：对数几率回归 (Logistic Regression)

Sigmoid函数的特点 $\hat{y} = \frac{1}{1+e^{-z}} = \frac{1}{1+e^{-(w^T x+b)}}$

- 概率形式输出：sigmoid函数是单调递增的，其值域为(0，1)，因此使sigmoid函数输出可作为概率值。在前面介绍的线性回归中，回归函数的值域一般为$(-\infty, +\infty)$
- 数据特征加权累加：对输入z取值范围没有限制，但当z大于一定数值后，函数输出无限趋近于1，而小于一定数值后，函数输出无限趋近于0。特别地，当$z = 0$时，函数输出为0.5。这里z是输入数据x和回归函数的参数w内积结果（可视为$\boldsymbol{x}$各维度进行加权叠加）
- 非线性变化：$\boldsymbol{x}$各维度加权叠加之和结果取值在0附近时，函数输出值的变化幅度比较大（函数值变化陡峭），且是非线性变化。但是，各维度加权叠加之和结果取值很大或很小时，函数输出值几乎不变化。

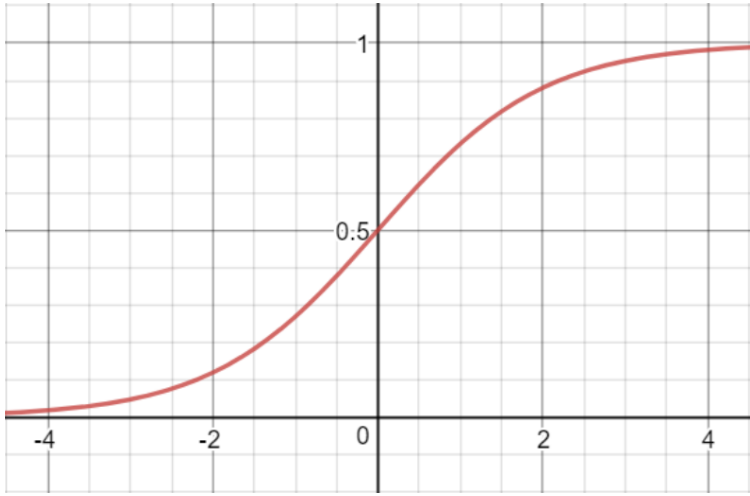


图4.6 Sigmoid函数

线性分类：对数几率回归 (Logistic Regression)

概率输出：从回归到分类

- 假设模型输出$\hat{y}$表示输入数据x属于正例的后验概率

$$\hat{y} = p(y = 1|\boldsymbol{x}) = f_\theta(x) = \frac{1}{1 + e^{-(\boldsymbol{w}^T\boldsymbol{x}+b)}}$$

和x属于负例的后验概率

$$p(y = 0|\boldsymbol{x}) = 1 - f_\theta(x) = \frac{e^{-(\boldsymbol{w}^T\boldsymbol{x}+\mathrm{b})}}{1 + e^{-(\boldsymbol{w}^T\boldsymbol{x}+\mathrm{b})}}$$

θ表示模型参数（$\theta = \{\boldsymbol{w}, b\}$）。于是有：

$$\mathrm{logit}\big(p(y = 1|\boldsymbol{x})\big) = \log\left(\frac{p(y = 1|\boldsymbol{x})}{p(y = 0|\boldsymbol{x})}\right) = \log\left(\frac{p}{1-p}\right) = \boldsymbol{w}^T\boldsymbol{x} + \mathrm{b}$$

几率(odds)：x属于正例的相对可能性

对数几率(log odds)或logit

线性回归方程

线性分类：对数几率回归 (Logistic Regression)

概率输出：从回归到分类

$$\text{logit}\big(p(y=1|\boldsymbol{x})\big) = \log\left(\frac{p(y=1|\boldsymbol{x})}{p(y=0|\boldsymbol{x})}\right) = \log\left(\frac{p}{1-p}\right) = \boldsymbol{w}^T\boldsymbol{x} + \mathrm{b}$$

- 如果输入数据$\boldsymbol{x}$属于正例的概率大于其属于负例的概率，即$p(y=1|\boldsymbol{x}) > 0.5$，则输入数据$\boldsymbol{x}$可被判断属于正例。

- 这一结果等价于$\frac{p(y=1|\boldsymbol{x})}{p(y=0|\boldsymbol{x})} > 1$，即$\log\left(\frac{p(y=1|\boldsymbol{x})}{p(y=0|\boldsymbol{x})}\right) > \log 1 = 0$，也就是$\boldsymbol{w}^T\boldsymbol{x} + \mathrm{b} > 0$成立。

- 从这里可以看出，logistic回归是一个线性模型。在预测时，可以计算线性函数$\boldsymbol{w}^T\boldsymbol{x} + \mathrm{b}$取值是否大于0来判断输入数据$\boldsymbol{x}$的类别归属。

线性分类：对数几率回归 (Logistic Regression)

概率输出：从回归到分类

- 为了估计后验概率$\hat{y} = p(y = 1|\boldsymbol{x})$，可以用极大似然估计！
- 模型参数的似然函数被定义为$\mathcal{L}(\theta|\mathcal{D}) = \mathrm{p}(\mathcal{D}|\theta)$，其中$\mathcal{D} = \{(x_i, y_i\}(1 \leq \mathrm{i} \leq \mathrm{n})$表示所有观测数据（或训练数据），$\theta$表示模型参数（$\theta = \{\boldsymbol{w}, b\}$）。
- 在最大化对数似然函数过程中，一般假设观测所得每一个样本数据是独立同分布（independent and identically distributed, i.i.d），于是可得：

$$\mathcal{L}(\theta|\mathcal{D}) = \mathrm{p}(\mathcal{D}|\theta) = \prod_{i=1}^{n} \mathrm{p}(y_i|\mathrm{x}, \theta) = \prod_{i=1}^{n} \big(f_\theta(x_i)\big)^{y_i} \big(1 - f_\theta(x_i)\big)^{1-y_i}$$

- 对上述公式取对数：

$$J(\theta) = \log\big(L(\theta|\mathcal{D})\big) = \sum_{i=1}^{n} y_i \log\big(f_\theta(x_i)\big) + (1 - y_i)\log(1 - f_\theta(x_i))$$

线性分类：对数几率回归 (Logistic Regression)

概率输出：从回归到分类

- 最大似然估计目的是计算似然函数的最大值，而分类过程是需要损失函数最小化。因此，在上式前加一个负号得到损失函数（交叉熵）：

$$J(\theta) = -\log\big(L(\theta|\mathcal{D})\big) = -(\sum_{i=1}^{n} y_i \log\big(f_\theta(x_i)\big) + (1-y_i)\log(1-f_\theta(x_i)))$$

真实值　　预测值

- $J(\theta)$等价于：$J(\theta) = \begin{cases} -\log\big(f_\theta(x_i)\big) & if\ y=1 \\ -\log\big(1-f_\theta(x_i)\big) & if\ y=0 \end{cases}$

线性分类：对数几率回归 (Logistic Regression)

概率输出：从回归到分类

- 需要最小化损失函数来求解参数。损失函数对参数θ的偏导如下（其中，$f_\theta'(x) = f_\theta(x)\big(1 - f_\theta(x)\big), \log' x = \frac{1}{x}$ ）

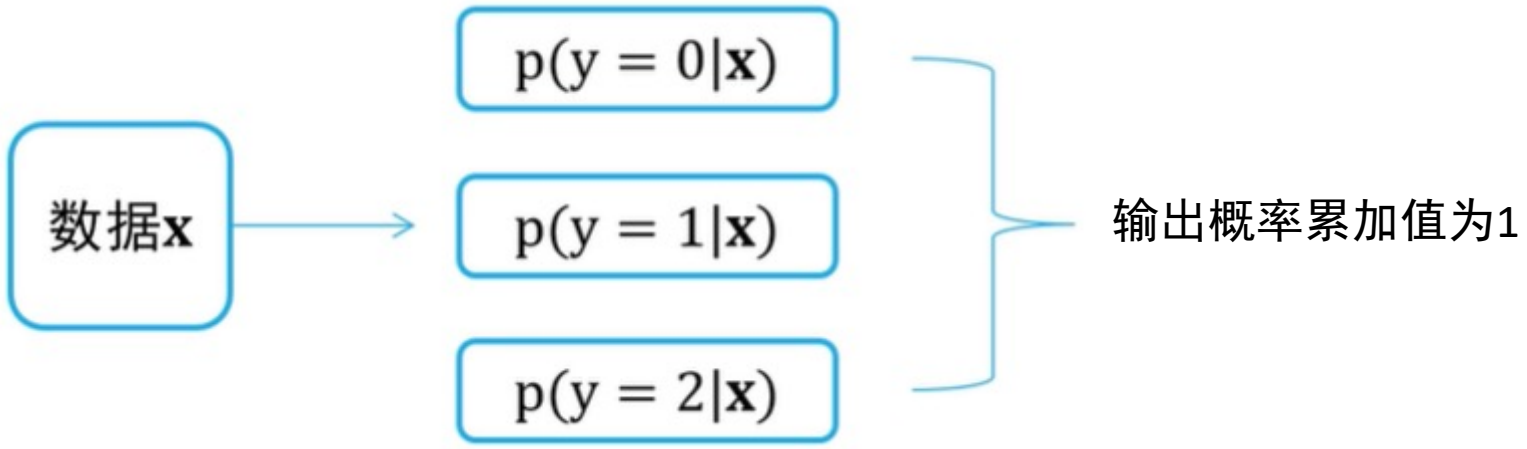
$$\begin{aligned}
\frac{\partial J(\theta)}{\partial \theta_j} &= -\sum_{i=1}^{n}\Big(y_i \frac{1}{f_\theta(x_i)}\frac{\partial f_\theta(x_i)}{\partial \theta_j} + (1-y_i)\frac{1}{1-f_\theta(x_i)}\frac{\partial(1-f_\theta(x_i))}{\partial \theta_j}\Big) \\
&= -\sum_{i=1}^{n}\frac{1}{f_\theta(x_i)}\Big(\frac{y_i}{f_\theta(x_i)} - \frac{1-y_i}{1-f_\theta(x_i)}\Big) \\
&= -\sum_{i=1}^{n} x_i f_\theta(x_i)\big(1-f_\theta(x_i)\big)\Big(\frac{y_i}{f_\theta(x_i)} - \frac{1-y_i}{1-f_\theta(x_i)}\Big) \\
&= -\sum_{i=1}^{n} x_i\big(y_i\big(1-f_\theta(x_i)\big) - (1-y_i)f_\theta(x_i)\big) \\
&= \sum_{i=1}^{n}\big(y_i - f_\theta(x_i)\big)x_i
\end{aligned}$$

- 将求导结果代入梯度下降迭代公式得：$\theta_j = \theta_j - \eta\sum_{i=1}^{n}\big(y_i - f_\theta(x_i)\big)x_i$

线性分类：对数几率回归 (Logistic Regression)

从回归到分类：从两类分类到多类分类

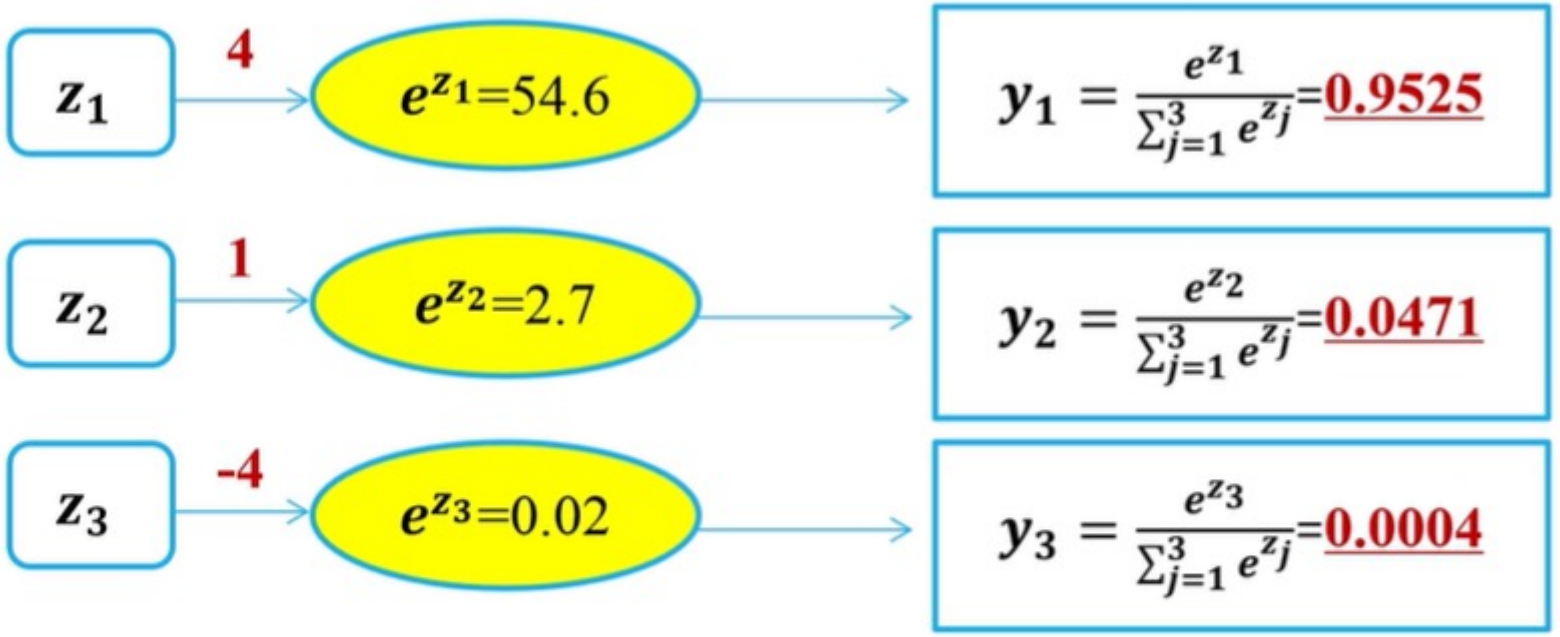
Logistic回归只能用于解决二分类问题，将它进行推广为多项逻辑斯蒂回归模型（multi-nominal logistic model，也即softmax函数），用于处理多类分类问题，可以得到处理多类分类问题的softmax回归



三个类别互斥

线性分类：对数几率回归 (Logistic Regression)

从回归到分类（ softmax 分类）：从两类分类到多类分类



$$0 < y_i < 1, \sum_i y_i = 1$$

指数级扩大最后一层输出，每个输出值都会增大，然而值最大的输出相比其他值都大很多，然后再将所有结果归一化到（0，1）概率空间。