重庆邮电大学

计算机科学与技术学院

人工智能原理

监督学习

机器学习：从数据中学习知识

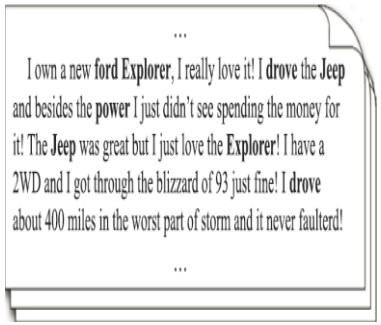
f $\begin{Bmatrix} 81 & 116 & \cdots & 133 \\ 104 & 130 & \cdots & 159 \\ \vdots & \vdots & \ddots & \vdots \\ 155 & 189 & \cdots & 218 \\ 197 & 221 & \cdots & 216 \end{Bmatrix}$

- Person
- Dog
- …

图像数据

类别分类

1. 原始数据中提取特征
2. 学习映射函数f
3. 通过映射函数f将原始数据映射到语义空间，即寻找数据和任务目标之间的关系



f {car, money, drive,…}

- 喜悦
- 愤怒
- …

文本数据

情感分类

机器学习的分类

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

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

半监督学习
(semi-supervised learning)

机器学习：分类问题

人员	数学好	身体好	会编程	嗓门大
程序员A	Yes	No	Yes	Yes
作家A	No	No	Yes	No
程序员B	Yes	Yes	No	No
…	…	…	…	…
医生A	Yes	Yes	Yes	Yes
程序员C	Yes	Yes	Yes	Yes
程序员D	Yes	Yes	Yes	No

标签数据

从数据中学习

映射函数

模式

f (数学好 = Yes, 会编程 = Yes, 身体好 =?，嗓门大 =?)

→ 程序员

类别

监督学习的重要元素

标注数据

- 标识了类别信息的数据

学习模型

- 如何学习得到映射模型

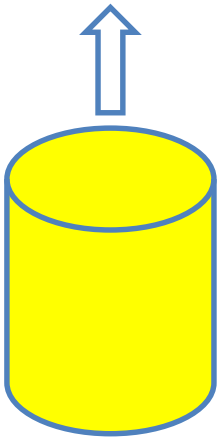
损失函数

- 如何对学习结果进行度量

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

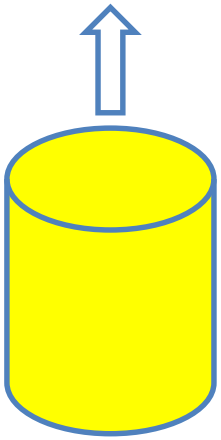
$(x_i, y_i), i = 1, \ldots, n$

- 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),其中第i个样本数据为x_i，y_i是x_i的标注信息。
- 从训练数据中学习得到的映射函数记为f， f对x_i的预测结果记为$f(x_i)$。损失函数就是用来计算x_i真实值y_i与预测值$f(x_i)$之间差值的函数。
- 很显然，在训练过程中希望映射函数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：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	$Loss(y_i, P(y_i \lvert x_i)) = -logP((y_i \lvert x_i)$

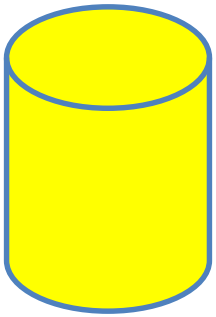
典型的损失函数

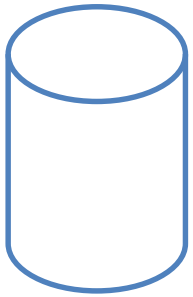
监督学习：训练数据与测试数据

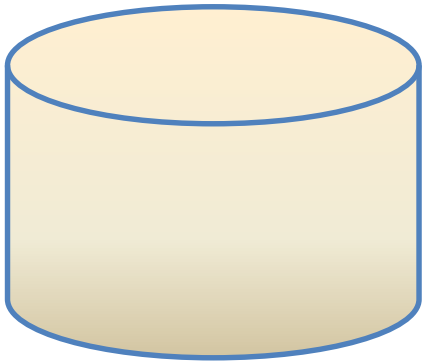
从训练数据集学习
得到映射函数f

在测试数据集
测试映射函数f

未知数据集
上测试映射函数f







训练数据集
$(x_i, y_i), i = 1, \ldots, n$

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

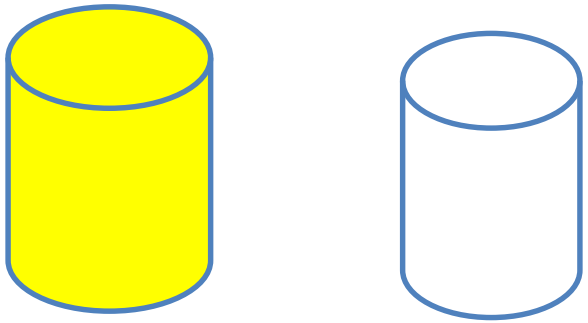
监督学习：经验风险与期望风险

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

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”

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。

$f($  $)$ → 人脸

$P($ 人脸 $|$  $)$ =0.99

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

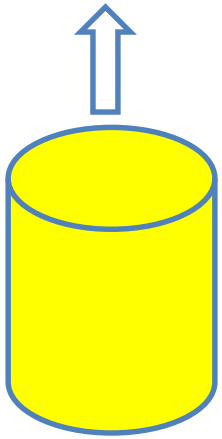
- 典型方法为贝叶斯方法、隐马尔可夫链
- 授之于鱼、不如授之于“渔”
- 联合分布概率$P(X,Y)$或似然概率$P(X|Y)$求取很困难

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

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

- 分类
- 识别
- 推荐
- …

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。

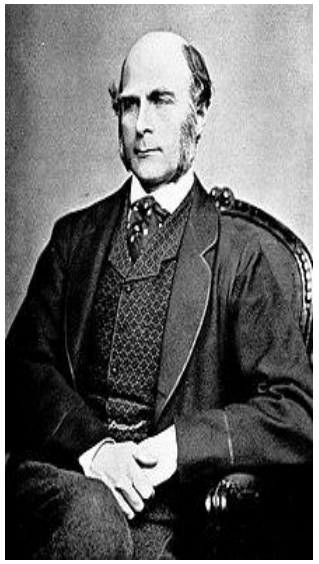
- 一旦确定了回归模型，就可以进行预测等分析工作，如从碳排放量预测气候变化程度、从广告投入量预测商品销售量等。

线性回归 (linear regression)

$$y = 33.73(英寸) + 0.516x$$

y: 子女平均身高

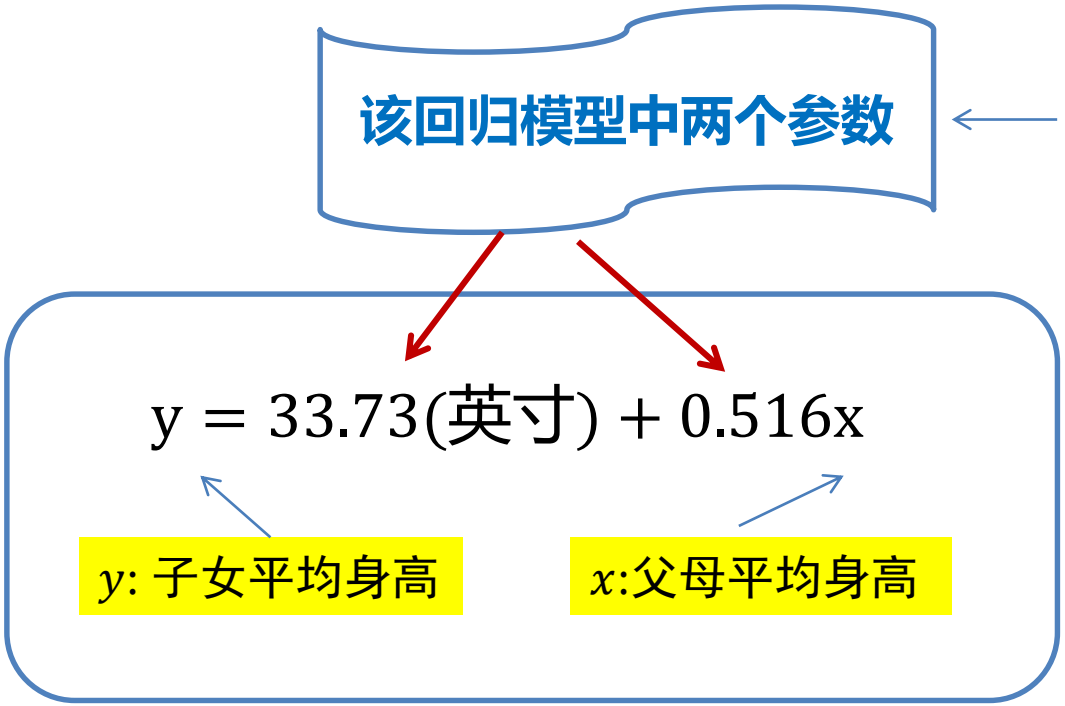
x:父母平均身高



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

- 父母平均身高每增加一个单位，其成年子女平均身高只增加0.516个单位，它反映了这种“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。

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：参数学习

线性回归模型例子

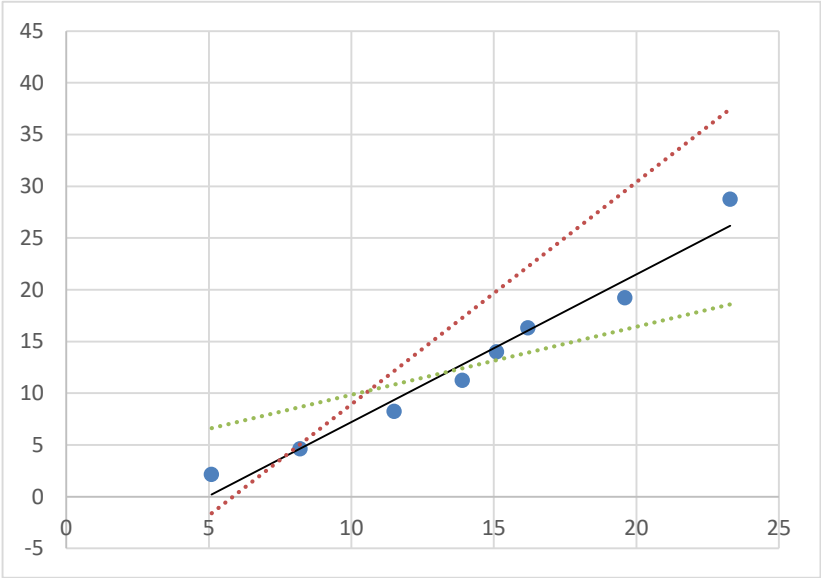
下表给出了芒提兹尼欧（Montesinho）地区发生森林火灾的部分历史数据，表中列举了每次发生森林火灾时的气温温度取值x和受到火灾影响的森林面积y。

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：参数学习

线性回归模型例子



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）

回归模型：$y = ax + b$

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$\frac{1}{N}\sum(y - \tilde{y})^2$最小。残差平均值最小只与参数$a$和$b$有关，最优解即是使得残差最小所对应的$a$和$b$的值。

线性回归：参数学习

回归模型参数求取：$y_i = ax_i + b\ (1 \le i \le n)$

- 记在当前参数下第i个训练样本x_i的预测值为$\hat{y}_i$
- 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：$L(a,b) = \sum_{i=1}^{n}(y_i - a \times x_i - b\ )^2$

目标：寻找一组a和b，使得误差总和$L(a,b)$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$L(a,b)$ 参数a和b分别求导，令其导数值为零，再求取参数a和b的取值。

线性回归：参数学习

回归模型参数求取：$y_i = ax_i + b\ (1 \le i \le n)$

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

$$\frac{\partial L(a,b)}{\partial a} = \sum_{i=1}^{n} 2(y_i - ax_i - b)(-x_i) = 0$$

将$b = \bar{y} - a\bar{x}$ $(\bar{y} = \frac{\sum_{i=1}^{n} y_i}{n},\ \bar{x} = \frac{\sum_{i=1}^{n} x_i}{n})$ 代入上式

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

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

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

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

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

线性回归：参数学习

回归模型参数求取：$y_i = ax_i + b\ (1 \le i \le n)$

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

$$\frac{\partial L(a,b)}{\partial b} = \sum_{i=1}^{n} 2(y_i - ax_i - b)(-1) = 0$$

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

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

$$\rightarrow n\bar{y} - an\bar{x} - nb = 0$$

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

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

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。

线性回归：参数学习

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

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

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

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

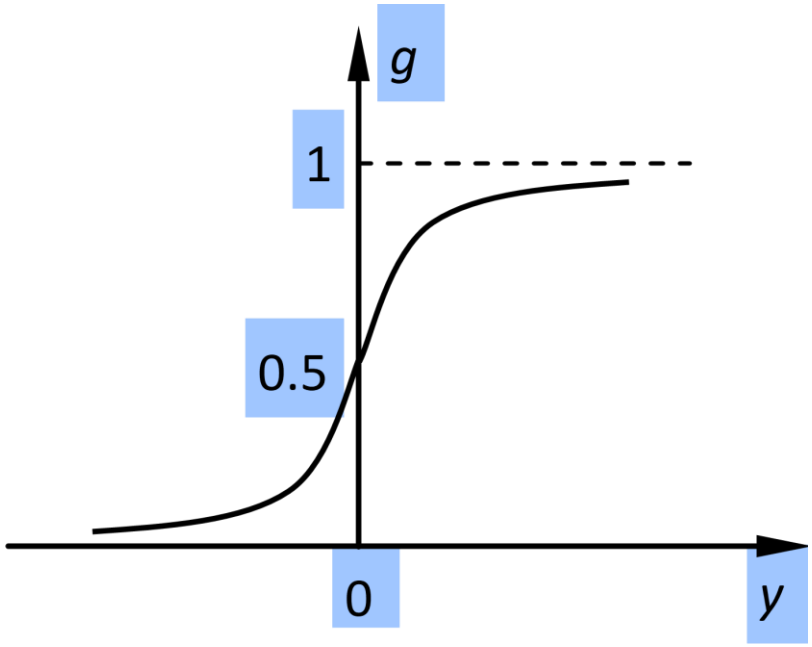
$$b = \bar{y} - a\bar{x} = -7.09$$

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$y = 1.428x - 7.09$

Logistic回归模型

Logistic函数（也称Sigmoid函数）

$$g(y)=\frac{1}{1+e^{-y}}$$



取变换

$$y= w_0+w_1x_1+w_2x_2= (w_0, w_1, w_2)^{\mathrm{T}}\bullet(1, x_1, x_2)=W^{\mathrm{T}}\boldsymbol{x}$$

得

$$g(y)=g(W^{\mathrm{T}}\boldsymbol{x})=\frac{1}{1+e^{-W^T X}}$$

将函数Logistic(x)作为分类问题的一种假设概率模型而表示为：

$$P(Y=1 \mid \boldsymbol{x})= \frac{1}{1+e^{-W^T X}}=\frac{e^{W^T X}}{1+e^{W^T X}}=\frac{\exp(W^T x)}{1+\exp(W^T x)}$$

$$P(Y=0 \mid \boldsymbol{x})=1-P(Y=1|\boldsymbol{x})=\frac{1}{1+e^{W^T X}}=\frac{1}{1+\exp(W^T x)}$$

这两个等式称为二项Logistic回归模型的条件概率分布。

可以看出，当$W^{\mathrm{T}}x$的值越接近正无穷，概率值$P(Y=1 \mid x)$就越接近1；当$W^{\mathrm{T}}x$的值越接近负无穷，概率值$P(Y=1 \mid x)$就越接近0.

由上述公式可得

$$P(Y=1 \mid x)= \frac{1}{1+e^{-W^{T}X}} = g(W^{\mathrm{T}}x)$$

现在，考虑如何确定式中参数$W^{\mathrm{T}}=(w_0,w_1,w_2)$的值？

将对数据x的一次分类决策的损失定义为（用极大似然估计可以规避非凸优化的难题）：

$$l(W, \boldsymbol{x})=\begin{cases} -\ln(g(\boldsymbol{x}; W)), & \text{当 } y=1 \\ -\ln(1-g(\boldsymbol{x}; W)), & \text{当 } y=0 \end{cases}$$

这一函数称为负对数似然函数。如果将这里的y值0、1当作数值来用（它们本来是符号值），则上面的两个表达式也可合并为：

$$l(W, \boldsymbol{x})= -y\ln(g(\boldsymbol{x}; W)) - (1-y)\ln(1-g(\boldsymbol{x}; W))$$

将全部n个样本在参数W下的损失相加，得

$$L(W)= -\sum_{i=1}^{n}[y_i \ln(g(x_i;W)) + (1-y_i)\ln(1-g(x_i;W))]$$

这就是我们给出的准则函数，可称为损失函数（或误差函数、代价函数等），也是一种交叉熵（cross-entropy））。

分类判决规则：

对于任一$\boldsymbol{x}\in U\times V$，

如果$P(Y=0\mid \boldsymbol{x})\geq 0.5$，则$\boldsymbol{x}\in C_1$；

否则，则$\boldsymbol{x}\in C_0$；

也可以一次定义关于全部样本$(\boldsymbol{x}_1, y_1), (\boldsymbol{x}_2, y_2), \ldots, (\boldsymbol{x}_n, y_n)$的参数$W$的似然函数，如

$$l(W)=\prod_{i=1}^{n}(P(Y=1\mid x_i)^{y_i}(1-P(Y=1\mid x_i))^{1-y_i}$$

而相应的对数似然函数则为

$$L(W)=\ln(l(W))$$

$$=\sum_{i=1}^{n}[y_i \ln P(Y=1\mid x_i)+(1-y_i)\ln(1-P(Y=1\mid x_i)]$$

负对数似然函数则为

$$-L(W)=-\sum_{i=1}^{n}[y_i \ln P(Y=1\mid x_i)+(1-y_i)\ln(1-P(Y=1\mid x_i)]$$

由于$P(Y=1\mid \boldsymbol{x})= g(W^{\mathrm{T}}\boldsymbol{x})$，所以，这个负对数似然函数$-L(W)$也就是上面的损失函数$L(W)$。

对于负对数似然函数$-L(W)$可用梯度下降法求解最佳参数W^*，但对于对数似然函数$L(W)$，用梯度上升（gradient rise）法同样也可获得最佳参数W^*。

梯度下降法

基本思想

梯度下降法是用梯度来建立迭代关系式的迭代法。对于无约束优化问题$\underset{W}{\operatorname{argmin}} f(\boldsymbol{x})$，其梯度下降法求解的迭代关系式为：

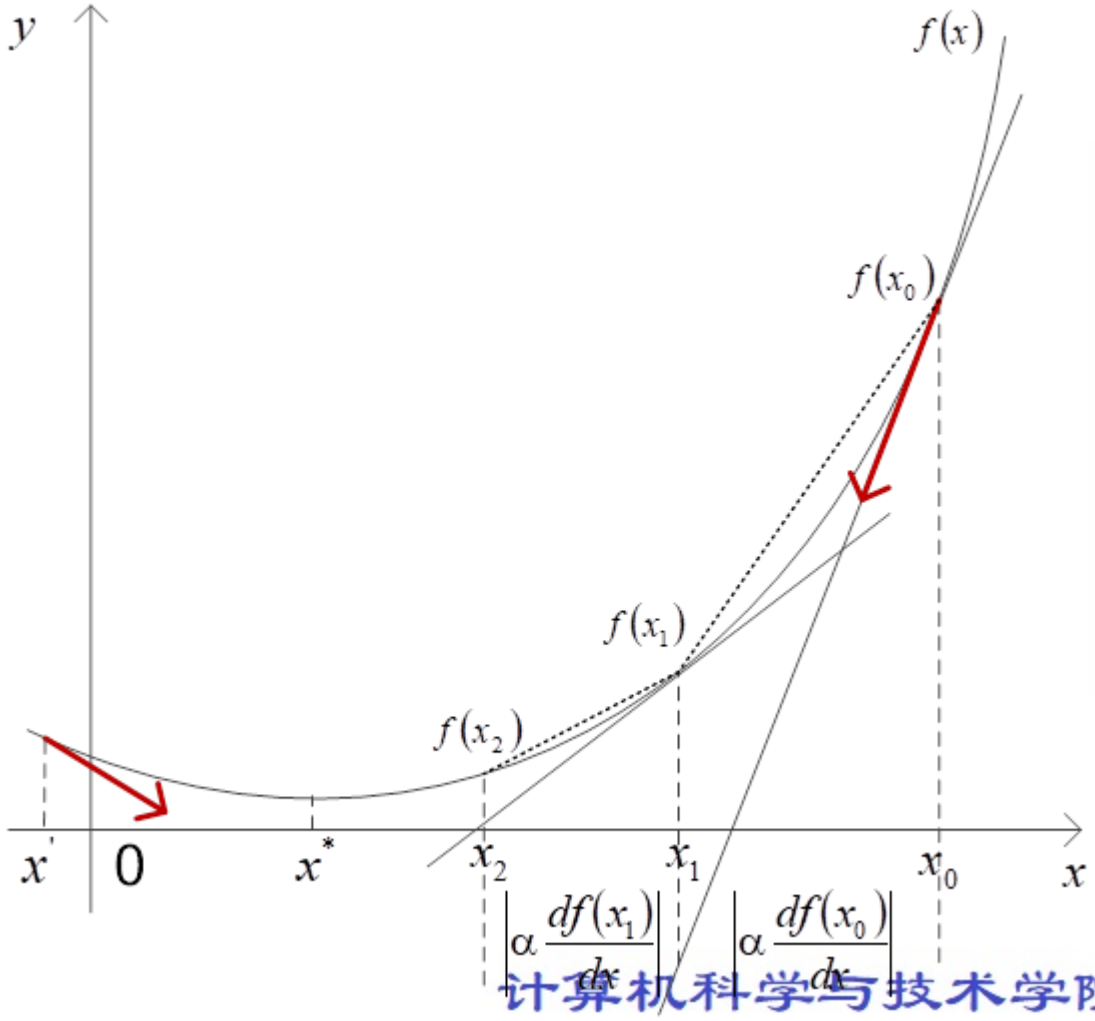
$$\boldsymbol{x}_{i+1} = \boldsymbol{x}_i + \alpha \cdot \left(-\frac{df(\boldsymbol{x})}{d\boldsymbol{x}}\right)\Bigg|_{\boldsymbol{x}=\boldsymbol{x}_i} = \boldsymbol{x}_i - \alpha \cdot \frac{df(\boldsymbol{x})}{d\boldsymbol{x}}\Bigg|_{\boldsymbol{x}=\boldsymbol{x}_i}$$

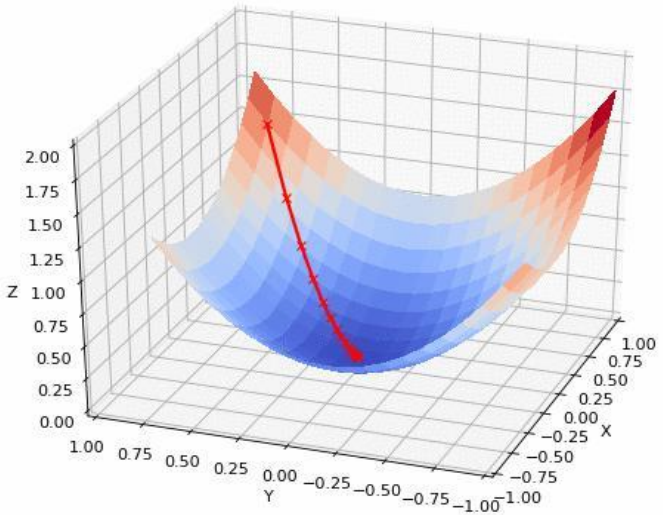
式中，$\boldsymbol{x}$为多维向量，记为$\boldsymbol{x} = \left(x^{(1)}, x^{(2)}, \ldots, x^{(n)}\right)$；$\alpha$为正实数，称为步长；$\frac{df(\boldsymbol{x})}{d\boldsymbol{x}} = \left(\frac{\partial f(\boldsymbol{x})}{\partial x^{(1)}} \quad \frac{\partial f(\boldsymbol{x})}{\partial x^{(2)}} \quad \cdots \quad \frac{\partial f(\boldsymbol{x})}{\partial x^{(n)}}\right)$是$f(\boldsymbol{x})$的梯度函数。

梯度下降法

示意

将向量x的函数简化为一元函数及二元函数时的示意。



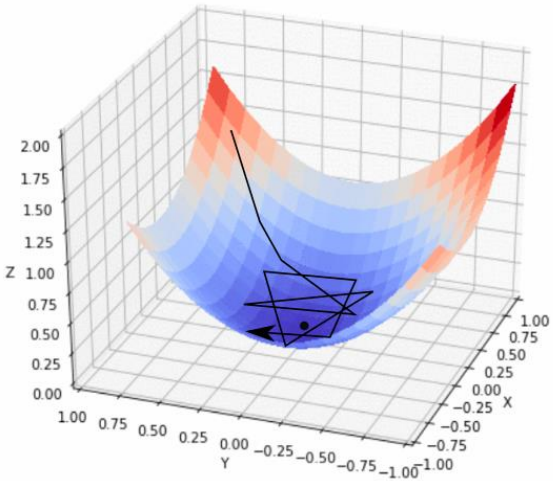


梯度下降法

几个问题

1）梯度下降法的结束条件，一般采用：①迭代次数达到了最大设定；②损失函数降低幅度低于设定的阈值。

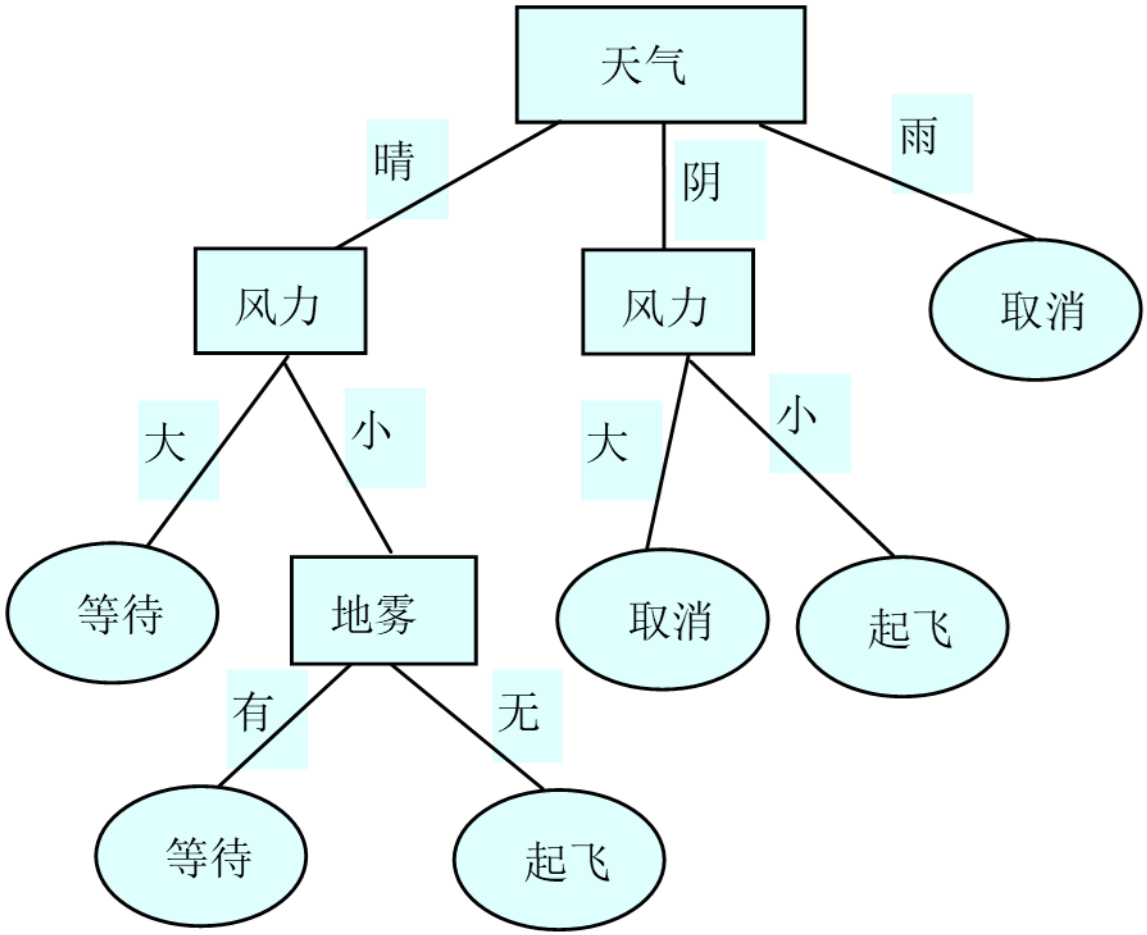
2）关于步长α，过大时，初期下降的速度很快，但有可能越过最低点，如果“洼地”够大，会再折回并反复振荡，如果“洼地”不够大，则会冲过“洼地”。如果步长过小，则收敛的速度会很慢。因此，可以采取先大后小的策略调整步长，具体大小的调节可根据$f(x)$降低的幅度或者x前进的幅度进行。



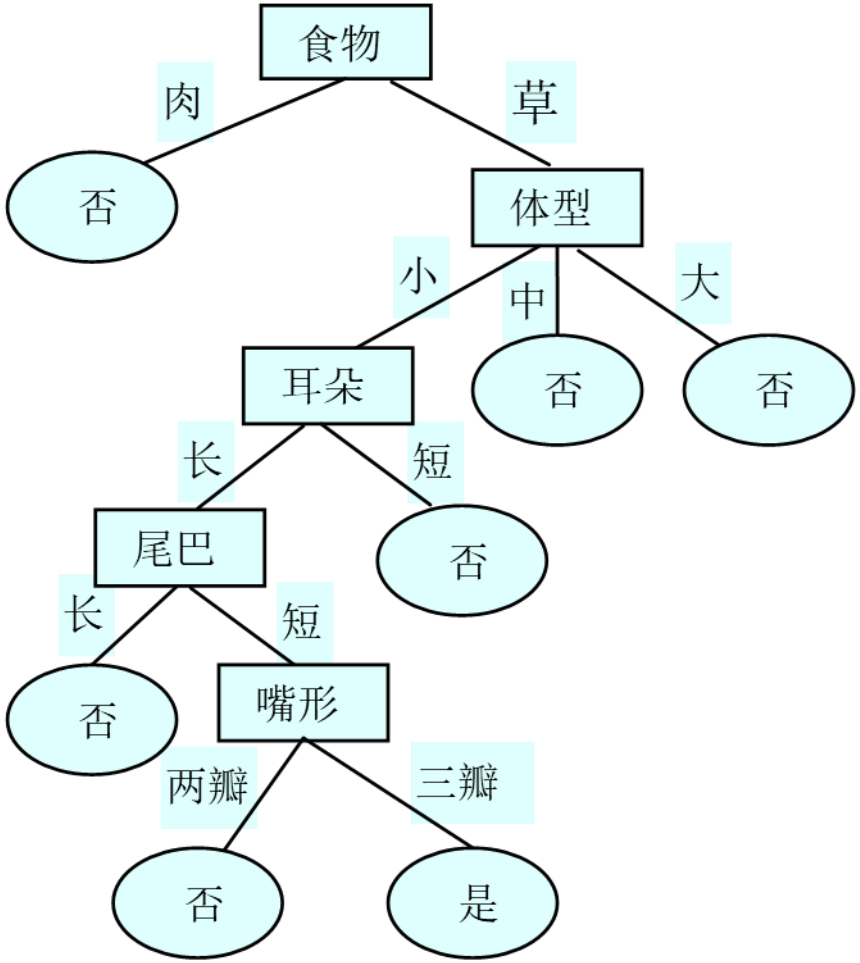
决策树分类算法

决策树模型是一种对测试样本进行分类的树形结构，该结构由结点（node）和有向边（directed edge）组成，结点分为内部节点（internal node）和叶节点（leaf node）两类。内部节点表示对样本的一个特征进行测试，内部节点下面的分支表示该特征测试的输出。如果只对特征的1个具体值进行测试，那么将只有正（大于等于）或负（小于）2个输出，生成的将是二叉树。如果对特征的多个具体值进行测试，那么将产生多个输出，生成的将是多叉树。叶节点表示样本的一个分类，如果样本只有两个分类类别，那么该模型是二分类模型，否则是多分类模型。

下图所示是机场指挥台关于飞机起飞的简单决策树。



下图是一个描述“兔子”概念的决策树。



决策树分类算法

决策树学习的基本方法和步骤：

首先，选取一个属性，按这个属性的不同取值对实例集进行分类；并以该属性作为根节点，以这个属性的诸取值作为根节点的分枝，进行画树。

然后，考察所得的每一个子类，看其中的实例的结论是否完全相同。如果完全相同，则以这个相同的结论作为相应分枝路径末端的叶子节点；否则，选取一个非父节点的属性，按这个属性的不同取值对该子集进行分类，并以该属性作为节点，以这个属性的诸取值作为节点的分枝，继续进行画树。

如此继续，直到所分的子集全都满足：实例结论完全相同，而得到所有的叶子节点为止。

决策树分类算法

建立二叉决策树流程

步数	操作
1	对输入的训练集，如果集合为空，算法结束
2	如果不能选择到一个合适的特征及其决策值，则建立叶子节点，算法结束
3	根据选择到的特征及其决策值，建立内部节点
4	依据选择到的特征及其决策值将输入的训练集划分为左、右两个子集，对每个子集应用本算法

第2步中，选择哪一个特征及其决策值来划分样本集对生成的树结构影响很大，对决策树的研究基本上集中于该问题，该问题习惯上称为样本集分裂，依其解决方法可将决策树算法分为ID3、C4.5、CART等算法。

信息量

信息的概念：信息就是对不确定性的消除。如一条天气预报消息“明天气温下降8度”可以消除人们对明天天气变化的不确定性。

消除的不确定性越大，那么信息量就应该越大。不确定性的消除是根据人们的先验知识来比较的。再比如，“中国足球队打败巴西足球队”比“中国乒乓球队打败巴西乒乓球队”所消除的不确定性就大的多。因此，预言以往发生小概率的事件的消息所带来的信息量就要大。以往发生的概率叫做先验概率，用p表示。香农基于先验概率来定义信息量公式：

$$I(x) = \log\left(\frac{1}{p}\right) = -\log p$$

信息量

假设中国足球队和巴西足球队曾经有过8次比赛，其中中国队胜1次。以U表示未来的中巴比赛中国队胜的事件，那么U的先验概率就是$\frac{1}{8}$，因此其信息量就是：

$$I(U) = -\log_2 \frac{1}{8} = 3$$

如果以$\overline{U}$表示巴西队胜，那么$\overline{U}$的先验概率是$\frac{7}{8}$，其信息量就是：

$$\mathrm{I}\left(\overline{U}\right) = -\log_2 \frac{7}{8} = 0.19$$

信息熵

信息量描述的是信源发出的单个事件消除的不确定性，还不能刻画信源消除的平均不确定性。如果把信源发出的所有事件的信息量求均值，就可以刻画信源消除的平均不确定性，定义为信息熵：

$$\mathrm{H}(X) = E[I(x_i)] = -\sum_{i=1}^{n} p_i \log_2 p_i$$

样本集合的信息熵越大，说明各样本相对均衡。

a) P=(1/4, 1/4, 1/4, 1/4)

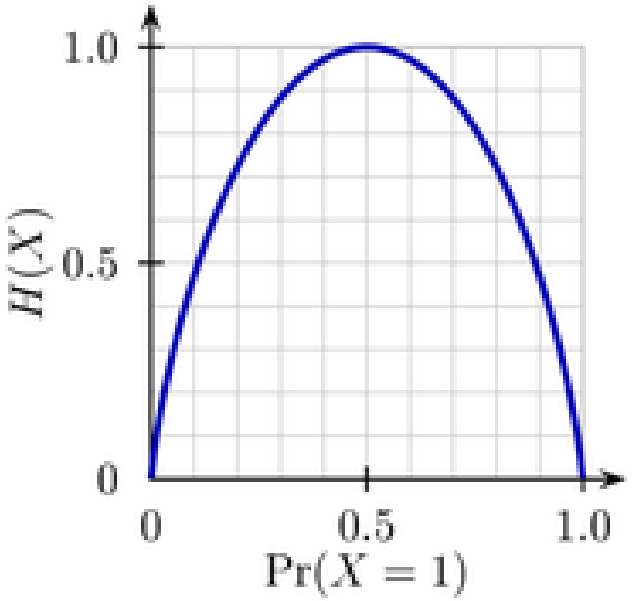
H(P)=log4

b) P=(1/2, 0, 0, 1/2)

H(P)=log2

c) P=(0, 0, 0, 1)

H(P)=log1= 0



信息增益

当把样本集A按照第j个特征$F^{(j)}$的某决策值f划分成两个独立的子集A_1和A_2时，此时A的信息熵为两个子集A_1和A_2的信息熵按样本数量的比例作加权的和：

$$\mathrm{H}\left(A, F^{(j)} = f\right) = \frac{|A_1|}{|A|} H(A_1) + \frac{|A_2|}{|A|} H(A_2)$$

其中，$|A|$，$|A_1|$，$|A_2|$表示A，A_1，A_2三个集合的样本个数。这个公式也被称为条件熵。

信息增益

划分前后信息熵的减少量称为信息增益，即：

$$\mathrm{Gain}\left(A, F^{(j)} = f\right) = \mathrm{H}(A) - \mathrm{H}\left(A, F^{(j)} = f\right)$$

$$= \mathrm{H}(A) - \left(\frac{|A_1|}{|A|}H(A_1) + \frac{|A_2|}{|A|}H(A_2)\right)$$

ID3决策树算法采用信息增益作为划分样本集的指标。在生成决策树时，选择使$\mathrm{Gain}\left(A, F^{(j)} = f\right)$**最大的那个特征**$F^{(j)}$**及其决策值**$f$**作为分裂点。**

ID3决策树算法

序号	天气	学生放假	促销	收入
1	坏	是	否	低
2	好	否	是	高
3	好	否	否	高
4	坏	否	否	低
5	坏	否	是	高
6	坏	是	是	低
7	好	是	是	低
8	好	否	否	高
9	坏	是	否	低
10	好	否	是	高

ID3决策树算法

步骤1 总信息熵:

$$I(5,5) = -\frac{5}{10}log_2\frac{5}{10} - \frac{5}{10}log_2\frac{5}{10} = 1$$

ID3决策树算法

步骤2 天气属性:

天气好的情况下，收入为“高”的记录为4条，收入为“低”的记录为1条，可以表示为(4,1)；天气不好的情况下，收入为“高”的记录为1条，收入为“低”的记录为4条,可以表示为(1,4).

则天气属性的信息熵的计算过程如下:

$$I(4,1) = -\frac{4}{5}log_2\frac{4}{5} - \frac{1}{5}log_2\frac{1}{5} = 0.721924$$

$$I(1,4) = -\frac{1}{5}log_2\frac{1}{5} - \frac{4}{5}log_2\frac{4}{5} = 0.721924$$

$$E(天气) = \frac{5}{10}I(4,1) + \frac{5}{10}I(1,4) = 0.721924$$

ID3决策树算法

步骤3 学生放假属性:

学生放假时，收入为“高”有0条，收入为“低”有4条。记为（0,4）；

学生不放假时，收入为“高”有5条，收入为“低”有1条。记为（5,1）；

学生放假属性的计算过程为:

$$I(0,4) = 0 - \frac{4}{4} log_2 \frac{4}{4} = 0$$

$$I(5,1) = -\frac{5}{6} log_2 \frac{5}{6} - \frac{1}{6} log_2 \frac{1}{6} = 0.650292$$

$$E(学生放假) = \frac{4}{10} I(0,4) + \frac{6}{10} I(5,1) = 0.390175$$

ID3决策树算法

步骤4 促销属性:

当小贩进行促销时，收入为“高”有3条，收入为“低”有2条。记为（3,2）；

当小贩不进行促销时，收入为“高”有2条，收入为“低”有3条。记为（2,3）；

促销属性的计算过程为:

$$I(3,2) = -\frac{2}{5}log_2\frac{2}{5} - \frac{3}{5}log_2\frac{3}{5} = 0.970951$$

$$I(2,3) = -\frac{2}{5}log_2\frac{2}{5} - \frac{3}{5}log_2\frac{3}{5} = 0.970951$$

$$E(促销) = \frac{5}{10}I(3,2) + \frac{5}{10}I(2,3) = 0.970951$$

ID3决策树算法

步骤5 计算信息增益值

$$Gain(天气) = I(5,5) - E(天气) = 0.278076$$

$$Gain(学生放假) = I(5,5) - E(学生放假) = 0.609825$$

$$Gain(促销) = I(5,5) - E(促销) = 0.029049$$

由步骤4可以知道“学生放假”的信息增益值最大，所以以“学生放假“为节点进行划分，划分为两个分支分别为“是”、“否”（指学生是否放假），然后再循环进行步骤1 ~步骤5的过程（排除已经进行划分的节点步骤），对剩下的2个节点分支进行划分，再进行信息增益的计算。

随堂练习：ID3算法

设表所示的是某保险公司的汽车驾驶保险类别划分的部分事例。我们将这张表作为一个实例集，用决策树学习来归纳该保险公司的汽车驾驶保险类别划分规则。

序　号	实　　例			
	性别	年龄段	婚状	保险类别
1	女	＜21	未	C
2	女	＜21	已	C
3	男	＜21	未	C
4	男	＜21	已	B
5	女	≥21 且≤25	未	A
6	女	≥21 且≤25	已	A
7	男	≥21 且≤25	未	C
8	男	≥21 且≤25	已	B
9	女	＞25	未	A
10	女	＞25	已	A
11	男	＞25	未	B
12	男	＞25	已	B

按性别划分，实例集S被分为两个子类：

$S_{男}$ ={(3,C), (4,B), (7,C), (8,B), (11,B), (12,B)}

$S_{女}$ ={(1,C), (2,C), (5,A), (6,A), (9,A), (10,A)}

从而，对子集$S_{男}$而言，

$$P(A)=\frac{0}{6}=0, P(B)=\frac{4}{6}, P(C)=\frac{2}{6}$$

对子集$S_{女}$而言，

$$P(A)=\frac{4}{6}, P(B)=\frac{0}{6}=0, P(C)=\frac{2}{6}$$

于是

$$
\begin{aligned}
H(S_{男}) &= -(P(A)\mathrm{lb}P(A)+(P(B)\mathrm{lb}P(B)+P(C)\mathrm{lb}P(C)) \\
&= -\left(\frac{0}{6}\times\mathrm{lb}\left(\frac{0}{6}\right)+\frac{4}{6}\times\mathrm{lb}\left(\frac{4}{6}\right)+\frac{2}{6}\times\mathrm{lb}\left(\frac{2}{6}\right)\right) \\
&= -\left(0+\frac{4}{6}\times(-0.58850)+\frac{2}{6}\times(-1.5850)\right) \\
&= -(-0.39-0.5283) \\
&= 0.9183
\end{aligned}
$$

$$H(S_{女}) = -(P(A)\text{lb}P(A) + (P(B)\text{lb}P(B) + P(C)\text{lb}P(C))$$

$$= -\left(\frac{4}{6}\times\text{lb}\left(\frac{4}{6}\right) + \frac{4}{6}\times\text{lb}\left(\frac{0}{6}\right) + \frac{2}{6}\times\text{lb}\left(\frac{2}{6}\right)\right)$$

$$= -\left(\frac{4}{6}\times(-0.5850) + 0 + \frac{2}{6}\times(-1.5850)\right)$$

$$= -(-0.5283 - 0.39)$$

$$= 0.9183$$

又

$$\frac{|S_{男}|}{|S|} = \frac{|S_{女}|}{|S|} = \frac{6}{12}$$

$$H(S|\text{性别}) = \frac{6}{12} \times H(S_{\text{男}}) + \frac{6}{12} \times H(S_{\text{女}})$$
$$= \frac{6}{12} \times 0.9183 + \frac{6}{12} \times 0.9183 = 0.9183$$

用同样的方法可求得:

$$H(S|\text{年龄}) = \frac{4}{12} \times H(S_{\text{大}}) + \frac{4}{12} \times H(S_{\text{中}}) + \frac{4}{12} \times H(S_{\text{小}}) = 1.1035$$

$$H(S|\text{婚状}) = \frac{6}{12} \times H(S_{\text{未}}) + \frac{6}{12} \times H(S_{\text{已}}) = 1.5062$$

可见，条件熵$H(S|\text{性别})$为最小，即对应的信息增益最大。所以，应取“性别”这一属性对实例集进行分类。

C4.5决策树分类算法

增益率

使用信息增益来选择特征时，算法会偏向于取值多的特征，也就是说取值越多可能会使得信息增益越大，但有时候并没有实际意义，C4.5决策树算法对此做了修正，它采用增益率作为选择特征的依据。增益率定义如下：$\mathrm{GainRatio}\left(A,F^{(j)}\right)=\frac{\mathrm{Gain}\left(A,F^{(j)}\right)}{SplitInfo\left(F^{(j)}\right)}$

其中，划分信息$SplitInfo\left(F^{(j)}\right)$定义如下：$SplitInfo\left(F^{(j)}\right)=-\sum\frac{|A_i|}{|A|}\log_2\frac{|A_i|}{|A|}$。其中，$A_i$是依据特征$F^{(j)}$的取值划分的样本子集。显然，在样本子集数增加的时，$SplitInfo\left(F^{(j)}\right)$也有增加的趋势，因此，信息增益增加的趋势得到了一定的修正。

CART决策树分类算法

基尼指数

CART决策树算法采用基尼指数（Gini Index）来选择划分特征。对于样本集A，假设有K个分类，则样本属于第k类的概率为p_k，则此概率分布的基尼指数为：$\mathrm{Gini}(p)=\sum_{k=1}^{K}p_k(1-p_k)=1-\sum_{k=1}^{K}{p_k}^2$

对于样本集A，其基尼指数为：$\mathrm{Gini}(A)=1-\sum_{k=1}^{K}\left(\frac{|A_k|}{|A|}\right)^2=1-\frac{\sum_{k=1}^{K}|A_k|^2}{|A|^2}$

CART决策树分类算法

基尼指数

基尼指数也是一种不等性度量的指标，取值介于0-1之间，分类越不平衡，基尼指数就越小。

如果样本集A划分成独立的两个子集A_1和A_2，其基尼指数为：

$$\text{Gini}(\{A_1, A_2\}) = \frac{|A_1|}{|A|} Gini(A_1) + \frac{|A_2|}{|A|} Gini(A_2)$$

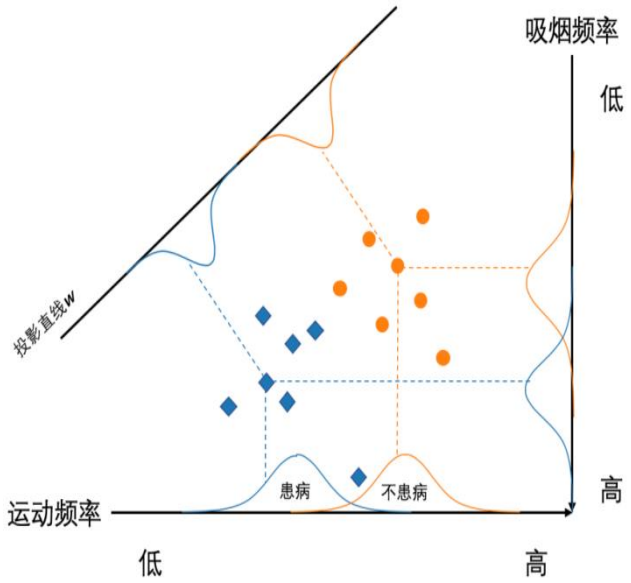
在样本集分裂时，要选择使分开后两个集合基尼指数最小的那个特征及其决策值作为分裂点，即与分裂前基尼指数相比，选择使之减少最多的那个特征及其决策值。

线性判别分析

线性区别分析(linear discriminant analysis， LDA)是一种基于监督学习的降维方法，也称为Fisher线性区别分析（Fisher's Discriminant analysis，FDA)。

对于一组具有标签信息的高维数据样本，LDA利用其类别信息，将其线性投影到一个低维空间上，在低维空间中**同一类别样本尽可能靠近，不同类别样本尽可能彼此远离**。

线性判别分析



君子和而不同，小人同而不和

虽然同一类别中每个数据各有特色，但是它们均具有相同内在模式，因而同一类别数据可被投影映射到一起。

- 左图给出了用矩形和圆圈来表示的患有某一疾病或者不患有某一疾病的两类人群。这两类人群分别用吸烟频率高低和运动频率高低来描述。
- 为了对这两类人群进行区分，需要将其投影到一个低维空间中。从图中可见，将其向x轴方向和y轴方向投影后，总会存在重叠部分（即若干人群在投影后的空间中不可区分）。
- 将数据向直线**w**所位于方向投影，则可见两类数据已经被完全区分开来。
- 在所得投影空间中，同一类别人群数据聚集在一起、不同类别人群数据具有较大间隔，体现了“类内方差小、类间间隔大”的原则。

线性判别分析

假设样本集为D $= \{(\boldsymbol{x}_1, y_1), (\boldsymbol{x}_2, y_2), (\boldsymbol{x}_3, y_3), \ldots, (\boldsymbol{x}_N, y_N)\}$，样本$\boldsymbol{x}_i \in R^n$的类别标签为$y_i$。其中，$y_i$的取值范围是$\{\mathcal{C}_1, \mathcal{C}_2, \ldots, \mathcal{C}_K\}$，即一共有K类样本。

定义N_i为第i类样本的个数、X_i为第i类样本的集合、$\boldsymbol{m}$为所有样本的均值向量、$\boldsymbol{m}_i$为第i类样本的均值向量。Σ_i为第i类样本的协方差矩阵，其定义为:

$$\Sigma_i = \sum_{\boldsymbol{x} \in \boldsymbol{X_i}} (\mathbf{x} - \boldsymbol{m}_i)(\mathbf{x} - \boldsymbol{m}_i)^T$$

线性判别分析

先来看$K = 2$的情况，即二分类问题。在二分类问题中，训练样本归属于$\mathcal{C}_1$或$\mathcal{C}_2$两个类别，并通过如下的线性函数投影到一维空间上：

$y(\mathbf{x}) = \mathbf{w}^{\mathrm{T}}\mathbf{x}\ (\mathbf{w} \in R^n)$

投影之后类别$\mathcal{C}_1$的协方差矩阵s_1为：

$$s_1 = \sum_{x \in \mathcal{C}_1} \left(\mathbf{w}^{\mathrm{T}}\mathbf{x} - \mathbf{w}^{\mathrm{T}}\mathbf{m}_1\right)^2 = \mathbf{w}^{\mathrm{T}} \sum_{x \in \mathcal{C}_1} [(\mathbf{x} - \mathbf{m}_1)\,(\mathbf{x} - \mathbf{m}_1)^{\mathrm{T}}]\mathbf{w}$$

同理可得到投影之后类别$\mathcal{C}_2$的协方差矩阵s_2。

线性判别分析

协方差矩阵s_1和s_2可用来衡量同一类别数据样本之间“分散程度”。为了使得归属于同一类别的样本数据在投影后的空间中尽可能靠近，需要最小化s_1+s_2取值。

$$\textbf{minimize}\ (s_1+s_2)$$

在投影之后的空间中，归属于两个类别的数据样本中心可如下分别计算：

$$m_1 = \mathbf{w}^{\mathrm{T}}\mathbf{m}_1,\quad m_2 = \mathbf{w}^{\mathrm{T}}\mathbf{m}_2$$

可通过$\|m_2 - m_1\|_2^2$来衡量不同类别之间的距离。为了使得归属于不同类别的样本数据在投影后空间中尽可能彼此远离，需要最大化$\|m_2 - m_1\|_2^2$取值。

$$maxmine\|m_2 - m_1\|_2^2$$

$$\mathrm{J}(\mathbf{w}) = \frac{\|m_2 - m_1\|_2^2}{s_1 + s_2}$$

投影方向优化目标

线性判别分析

$$\mathrm{J}(\mathbf{w}) = \frac{\|m_2 - m_1\|_2^2}{s_1 + s_2}$$

可以把上述式子改写成与**w**相关的式子：

$$\mathrm{J}(\mathbf{w}) = \frac{\left\|\mathbf{w}^{\mathrm{T}}(\mathbf{m}_2 - \mathbf{m}_1)\right\|_2^2}{\mathbf{w}^{\mathrm{T}}\Sigma_1\mathbf{w} + \mathbf{w}^{\mathrm{T}}\Sigma_2\mathbf{w}} = \frac{\mathbf{w}^{\mathrm{T}}(\mathbf{m}_2 - \mathbf{m}_1)(\mathbf{m}_2 - \mathbf{m}_1)^{\mathrm{T}}\mathbf{w}}{\mathbf{w}^{\mathrm{T}}(\Sigma_1 + \Sigma_2)\mathbf{w}} = \frac{\mathbf{w}^{\mathrm{T}}\mathbf{S}_b\mathbf{w}}{\mathbf{w}^{\mathrm{T}}\mathbf{S}_w\mathbf{w}}$$

其中，$\mathbf{S}_b$称为类间散度矩阵(between-class scatter matrix)，其定义如下:

$$\mathbf{S}_b = (\mathbf{m}_2 - \mathbf{m}_1)(\mathbf{m}_2 - \mathbf{m}_1)^{\mathrm{T}}$$

$\mathbf{S}_w$则称为类内散度矩阵(within-class scatter matrix)，其定义如下：

$$\mathbf{S}_w = \Sigma_1 + \Sigma_2$$

线性判别分析

$$J(\mathbf{w}) = \frac{\mathbf{w}^{\mathrm{T}}\mathbf{S}_b\mathbf{w}}{\mathbf{w}^{\mathrm{T}}\mathbf{S}_w\mathbf{w}}$$

由于J(**w**)的分子和分母都是关于**w**的二项式，因此最后的解只与**w**的方向有关，与**w**的长度无关

将分母$\mathbf{w}^{\mathrm{T}}\mathbf{S}_{\mathrm{W}}\mathbf{w} = 1$作为约束条件，将上述优化问题转变为拉格朗日函数：

$$L(\boldsymbol{w}) = \mathbf{w}^{\mathrm{T}}\mathbf{S}_b\mathbf{w} - \lambda(\mathbf{w}^{\mathrm{T}}\mathbf{S}_w\mathbf{w} - 1)$$

带约束条件（即$\mathbf{w}^{\mathrm{T}}\mathbf{S}_{\mathrm{W}}\mathbf{w} - 1 = 0$）的函数极大值（即$\mathbf{w}^{\mathrm{T}}\mathbf{S}_b\mathbf{w}$取值最大）优化问题所对应拉格朗日函数

对**w**求偏导并使其求导结果为零，可得：

$$\mathbf{S}_w{}^{-1}\mathbf{S}_b\mathbf{w} = \lambda\mathbf{w}$$

注：拉格朗日乘子法就是求在某个/某些约束条件下函数极值方法，其主要思想是将约束条件函数与原函数联立，从而求出使原函数取得极值时各个变量的解。

拉格朗日乘子法

设给定二元函数z=f(x,y)和附加条件φ(x,y)=0，为寻找z=f(x,y)在附加条件下的极值点，先做拉格朗日函数 $F(x,y,\lambda)=f(x,y)+\lambda\varphi(x,y)$ ，其中λ为参数。

令F(x,y,λ)对x和y和λ的一阶偏导数等于零，即

$F'_x=f'_x(x,y)+\lambda\varphi'_x(x,y)=0$ [1]

$F'_y=f'_y(x,y)+\lambda\varphi'_y(x,y)=0$

$F'_\lambda=\varphi(x,y)=0$

由上述方程组解出x,y及λ，如此求得的(x,y)，就是函数z=f(x,y)在附加条件φ(x,y)=0下的可能极值点。

若这样的点只有一个，由实际问题可直接确定此即所求的点。

线性判别分析

$$J(\mathbf{w}) = \frac{\mathbf{w}^{\mathrm{T}}\mathbf{S}_b\mathbf{w}}{\mathbf{w}^{\mathrm{T}}\mathbf{S}_w\mathbf{w}}$$

拉格朗日乘子法

$$\mathbf{S}_w{}^{-1}\mathbf{S}_b\mathbf{w} = \lambda\mathbf{w}$$

Fisher线性判别
(Fisher linear discrimination)

$\mathbf{w}$和λ是$\mathbf{S}_w{}^{-1}\mathbf{S}_b$的特征向量和特征根

因为 $\mathbf{S}_b = (\mathbf{m}_2 - \mathbf{m}_1)(\mathbf{m}_2 - \mathbf{m}_1)^{\mathrm{T}}$，那么 $\mathbf{S}_b\mathbf{w} = (\mathbf{m}_2 - \mathbf{m}_1)(\mathbf{m}_2 - \mathbf{m}_1)^{\mathrm{T}}\boldsymbol{w} = (\mathbf{m}_2 - \mathbf{m}_1) \times \lambda_w$，将其代入Fisher线性判别，可得：

$$\mathbf{S}_w{}^{-1}\mathbf{S}_b\mathbf{w} = \mathbf{S}_w{}^{-1}(\mathbf{m}_2 - \mathbf{m}_1) \times \lambda_w = \lambda\mathbf{w}$$

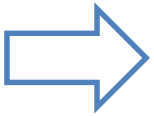
由于对$\mathbf{w}$的扩大和缩小操作不影响结果，因此可约去上式中的未知数λ和λ_w，得到：

$$w = S_w{}^{-1}(m_2 - m_1)$$

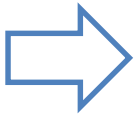
为了获得“类内汇聚、类间间隔”的最佳投影结果，只需要分别求出待投影数据的均值和方差，就可以设计得到最佳投影方向$\mathbf{w}$。 LDA模型可从两类问题被拓展到多类问题

线性判别分析

$$J(\mathbf{w}) = \frac{\mathbf{w}^T \mathbf{S}_b \mathbf{w}}{\mathbf{w}^T \mathbf{S}_w \mathbf{w}}$$



$$\mathbf{w}^T \mathbf{x}$$



$$p(class|\mathbf{w}^T \mathbf{x})$$

寻找“类内汇聚、类间间隔”的最佳投影结果

降维：从高维到低维的映射

分类：将降维后结果判断为某个类别

主成分分析（PCA）是一种无监督学习的降维方法（无需样本类别标签），线性区别分析（LDA）是一种监督学习的降维方法（需要样本类别标签。PCA和LDA均是优化寻找一定特征向量**w**来实现降维，其中PCA寻找投影后数据之间方差最大的投影方向、LDA寻找“类内方差小、类间距离大”投影方向。

PCA对高维数据降维后的维数是与原始数据特征维度相关（与数据类别标签无关）。假设原始数据维度为 d ，那么PCA所得数据的降维维度可以为小于d的任意维度。LDA降维后所得到维度是与数据样本的类别个数K有关（与数据本身维度无关）。假设原始数据一共有K个类别，那么LDA所得数据的降维维度小于或等于K – 1。

集成学习

集成学习基本思想

集成学习的基本思想是集体决策，对多个模型的预测结果进行表决来提高准确性。一般地来说，集成学习可以获得比单一模型效果显著提高的预测性能。集成学习对“弱学习器”效果尤其明显。弱学习器是指预测效果略高于随机猜测的模型，而预测效果明显的学习器，称为“强学习器”。个体模型可以是相同类型的，如都是同一种决策树，称为同质的，也可以是不同类型的，称为异质的。同质集成的个体学习器又称为基学习器（base learner），在分类模型中也称为基分类器（base classifier）。

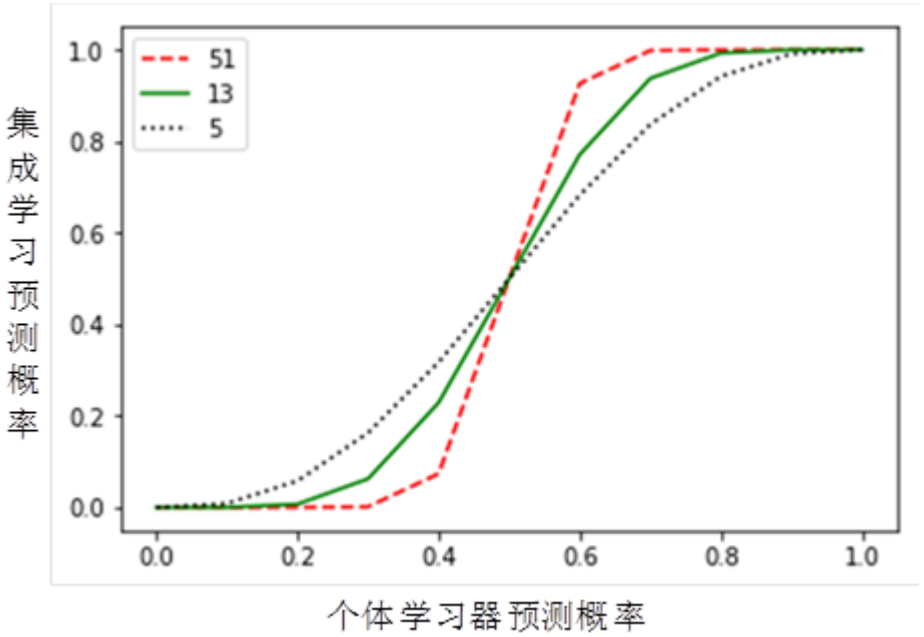
集成系统的预测概率

考虑在集成学习模型中，按个体学习器结果中的最多数给出最终结果。假设每个个体学习器的预测概率相同，对样本正确预测的概率为R，同时个体学习器之间相对独立。因为随机预测成功的概率为0.5，所以R要大于0.5。当R稍大于0.5时，为弱学习器，当R接近于1.0时，为强学习器。假如有n个个体学习器，对于某样本，认为只要有$k(1<k<n)$个个体学习器认为为正样本，那么输出最终结果为正样本。最终输出为正样本的概率为：

$$R_S=\sum_{i=k}^{n}\binom{n}{i}R^i(1-R)^{n-i}=\sum_{i=k}^{n}\frac{n!}{i!\,(n-i)!}R^i(1-R)^{n-i}\quad k\le n$$

集成系统的预测概率

当用5，13，51个个体学习器来集成，并投票决定输出，超过半数个体学习器输出为正类时，集成系统输出正类的概率变化



横坐标是R的取值，即每个个体学习器的正确预测概率，纵坐标是系统的输出正样本的概率。三条折线分别代表在5（点线）、13（实线）、51（虚线）个个体学习器的系统中的表现。由图中可以看出，当个体学习器为弱学习器时（R稍大于0.5），通过集成，可以显著提高预测概率，即变成强学习器，而且个体学习器越多，提高的越快。

Ada Boosting：思路描述

- Ada Boosting算法中两个核心问题：
  - 在每个弱分类器学习过程中，如何改变训练数据的权重：提高在上一轮中分类错误样本的权重。
  - 如何将一系列弱分类器组合成强分类器：通过加权多数表决方法来提高分类误差小的弱分类器的权重，让其在最终分类中起到更大作用。同时减少分类误差大的弱分类器的权重，让其在最终分类中仅起到较小作用。

Ada Boosting：算法描述---数据样本权重初始化

- 给定包含N个标注数据的训练集合Γ，$\Gamma = \{(x_1, y_1), \ldots, (x_N, y_N)\}$。

  $x_i (1 \le i \le N) \in X \subseteq R^n, y_i \in Y = \{-1, 1\}$

- Ada Boosting算法将从这些标注数据出发，训练得到一系列弱分类器，并将这些弱分类器线性组合得到一个强分类器。

  1. 初始化每个训练样本的权重

  $D_1 = (w_{11}, \ldots, w_{1i}, \ldots, w_{1N})$，其中 $w_{1i} = \frac{1}{N} (1 \le i \le N)$

Ada Boosting：算法描述---第m个弱分类器训练

2. 对$m = 1,2,\ldots,M$

a) 使用具有分布权重D_m的训练数据来学习得到第m个基分类器（弱分类器）G_m：

$$G_m(x): X \to \{-1,1\}$$

b) 计算$G_m(x)$在训练数据集上的分类误差

$err_m = \sum_{i=1}^{N} w_{mi} I(G_m(x_i) \neq y_i)$ 这里：$I(\cdot) = 1$，如果$G_m(x_i) \neq y_i$；否则为0

c) 计算弱分类器$G_m(x)$的权重： $\alpha_m = \frac{1}{2}\ln\frac{1-err_m}{err_m}$

d) 更新训练样本数据的分布权重： $D_{m+1} = w_{m+1,i} = \frac{w_{m,i}}{Z_m} e^{-\alpha_m y_i G_m(x_i)}$，其中$Z_m$是归一化因子以使得$D_{m+1}$为概率分布， $Z_m = \sum_{i=1}^{N} w_{m,i}\, e^{-\alpha_m y_i G_m(x_i)}$

Ada Boosting：算法描述---弱分类器组合成强分类器

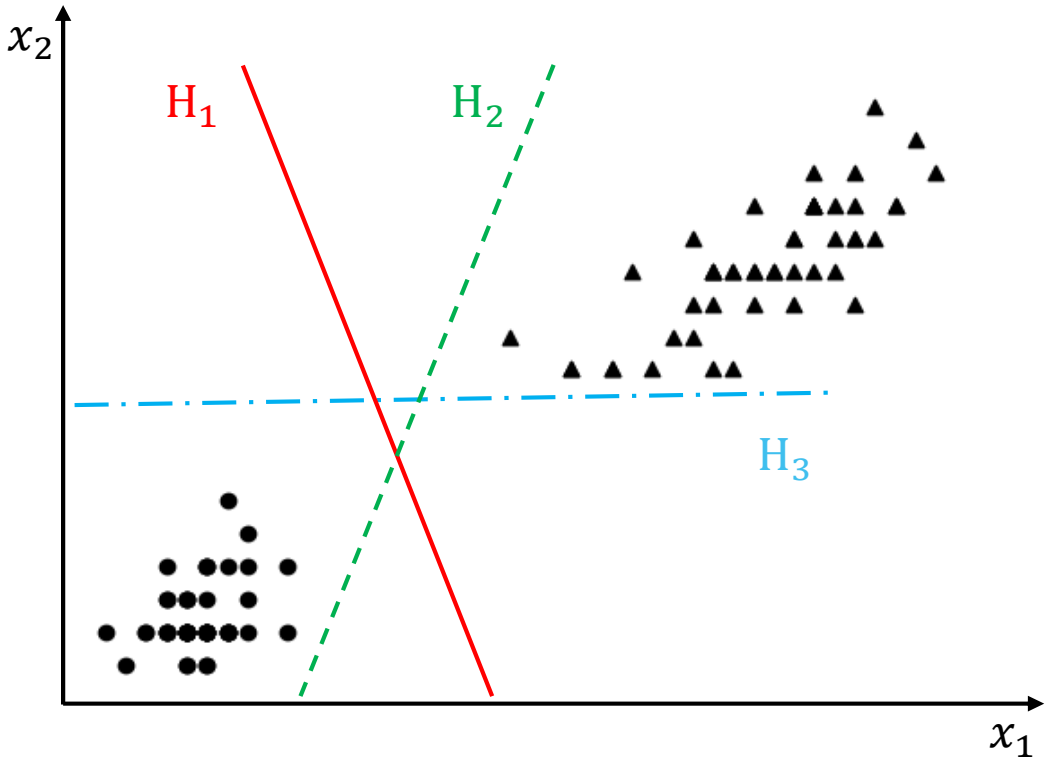
3. 以线性加权形式来组合弱分类器$f(x)$

$$f(x) = \sum_{i=1}^{M} \alpha_m G_m(x)$$

得到强分类器$G(x)$

$$G(x) = sign(f(x)) = sign(\sum_{i=1}^{M} \alpha_m G_m(x))$$

支持向量机（SVM）



支持向量机（SVM）

- 基本分类模型

给定一个面向两类分类问题的线性可分训练集，其中包含N个样本 $(\mathbf{x}_1, y_1), (\mathbf{x}_2, y_2), \cdots, (\mathbf{x}_N, y_N)$**。通常，可令标签**$y \in \{+1, -1\}$**。**

需要学习到一个分类超平面，设对应的参数化表示为

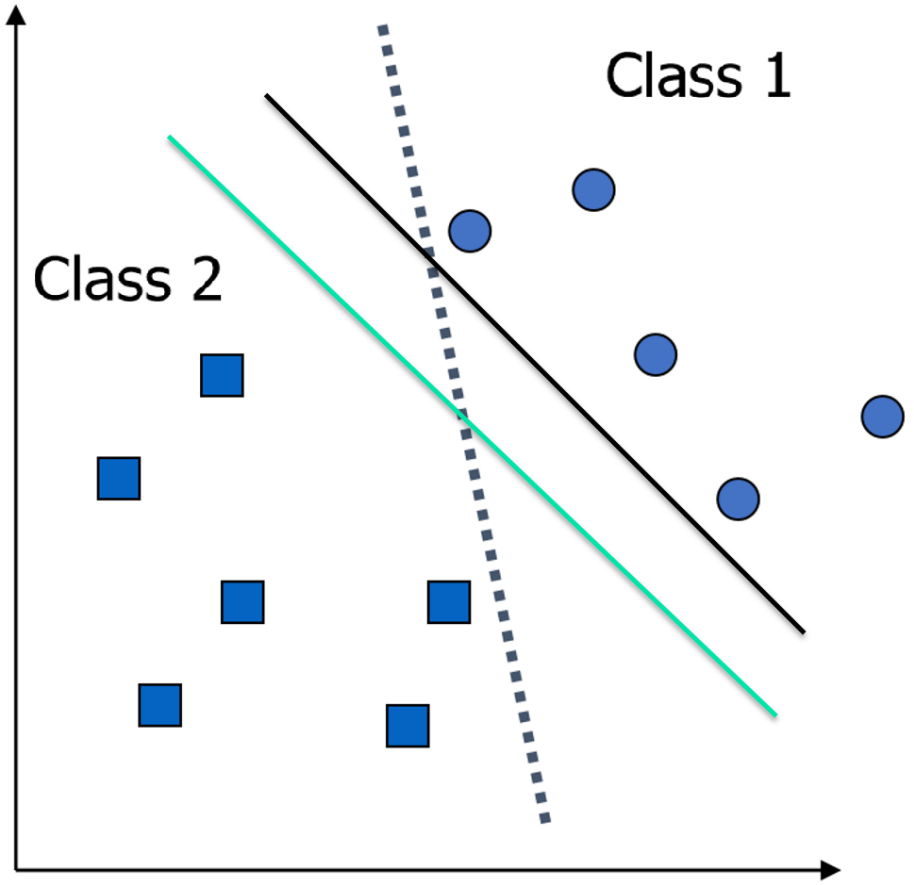
$$f(\mathbf{x}) = \mathbf{w}^{\cdot}\,\mathbf{x} + b = 0,$$

其中，w是超平面的法向量，标量b是偏差参数。
训练完成得到参数以后

$$h(\mathbf{w}) = sign[\mathbf{w}^{\cdot}\,\mathbf{x} + b],$$

可分情况下，分界线不是唯一的，怎样的分界线具有最好泛化性

右边的三条分界线均可正确分类训练集，哪个具有最好泛化性？



感知机可能训练得到任何可能的分界线，受控于初始和样本次序

支持向量机（SVM）

- 函数距离

虽然SVM分类中，属于同一类的节点分类结果都为+1或者都为-1，同一类的数据样本没有体现出差别。

但是$|w^{T}x+b|$能够反映出数据样本距离超平面的远近。距离超平面越远，则分类正确的确定性程度越高。

此外，如果分类正确，则$w^{T}x+b$的正负号应该与y一致，所以可以用$y(w^{T}x+b)$表示一个样本分类正确的确定性程度，这个量被称为函数距离（间隔），记为γ。

支持向量机（SVM）

- 几何距离

需要注意的是，把w 和b等比例缩放，超平面其实没有发生变化，但是函数间隔却会变大变小。

此时，可以定义任意一点x到超平面的**几何距离（间隔）**为

$$d = \frac{|\mathbf{w}^{\text{T}}\mathbf{x} + b|}{\|\mathbf{w}\|} = \frac{y(\mathbf{w}\ \mathbf{x} + b)}{\|\mathbf{w}\|}.$$

函数距离与几何距离的关系为$d = \frac{\gamma}{||w||}$。其中$||w||$是w的欧几里得范数。

可以认为距离d的大小与w的长度无关，只与w的方向有关系。因此可以固定参数向量w的长度为1，即

支持向量机（SVM）

由于训练数据线性可分，我们希望能找到一个线性函数$f(\mathbf{x})$对所有的样本都满足$y_i f(\mathbf{x}_i) > 0$，而且可以确定这样的函数一定存在。那么，分类模型的优化表达为

$$\begin{aligned} \max_{\mathbf{w},b} \quad & M \\ s.t. \quad & y_i(\mathbf{w}^{\cdot}\mathbf{x}_i + b) \geq M, \\ & \|\mathbf{w}\| = 1. \end{aligned}$$

其中M 也可以理解为距离超平面最近的数据点，到超平面的几何距离。也就是

$$\frac{y(\mathbf{w}^{\cdot}\mathbf{x} + b)}{\|\mathbf{w}\|} = M.$$

与超平面距离最接近的数据点被称为支持向量。

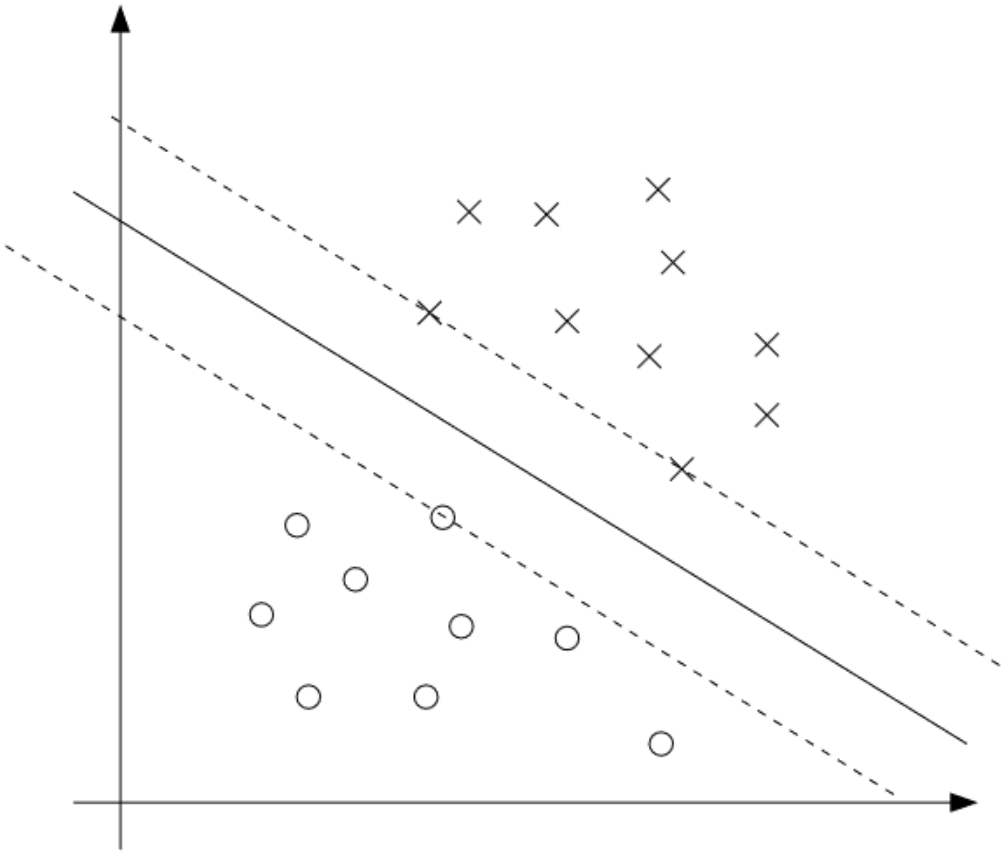
支持向量机（SVM）

- 支持向量和间隔边界

支持向量所在的两个与分类超平面平行的超平面，之间的距离称为间隔。

最优超平面就是使得间隔达到最大的超平面。

在SVM中，最优超平面只由支持向量决定。就算间隔之外加入更多的点，也不会改变结果。



支持向量机（SVM）

由于函数距离$y(w^{\top}x+b)$的取值大小，并不影响优化问题的解，所以为了简化计算，可以设定$y(w^{\top}x+b)=1$。所以优化问题就变为了：

$$\max_{w,b} \quad \frac{2}{\|\mathbf{w}\|}$$
$$s.t. \quad y_i(\mathbf{w}^{\cdot}\mathbf{x}_i+b)\geq 1.$$

$$\min_{w,b} \quad \frac{1}{2}\|\mathbf{w}\|^2$$
$$s.t. \quad y_i(\mathbf{w}^{\cdot}\mathbf{x}_i+b)\geq 1\,(i=1,2,\ldots,N).$$

支持向量机（SVM）

- 拉格朗日对偶优化

引入拉格朗日乘子向量$\boldsymbol{\alpha} = [\alpha_1, \alpha_2, \cdots, \alpha_N]$，通过拉格朗日函数将约束条件融入到目标函数中，得到优化问题对应的拉格朗日函数为

$$L(\mathbf{w}, b, \boldsymbol{\alpha}) = \frac{1}{2} \| \mathbf{w} \|^2 - \sum_{i=1}^{N} \alpha_i (y_i (\mathbf{w} \cdot \mathbf{x}_i + b) - 1),$$

其中各乘子变量$\alpha_i (i = 1, 2, \cdots, N)$均为非负值。

拉格朗日对偶函数

$$g(\boldsymbol{\alpha}) = \min_{\mathbf{w}, b} L(\mathbf{w}, b, \alpha)$$

可以证明，$g(\boldsymbol{\alpha})$一定小于或等于原优化问题的最优值。

支持向量机（SVM）

求解拉格朗日对偶优化问题：

$$\max_{\alpha_i \geq 0} \min_{\mathbf{w},b} L(\mathbf{w},b,\boldsymbol{\alpha}).$$

首先，固定α，关于w和b最小化拉格朗日函数$L(\mathbf{w},b,\boldsymbol{\alpha})$。对w和$b$求导，我们可以得出

$$\frac{\partial L}{\partial \mathbf{w}} = \mathbf{0} \Rightarrow \mathbf{w} = \sum_{i=1}^{N} \alpha_i y_i \mathbf{x}_i$$

$$\frac{\partial L}{\partial b} = 0 \Rightarrow \sum_{i=1}^{N} \alpha_i y_i = 0$$

将上面两式带入函数$L(\mathbf{w},b,\boldsymbol{\alpha})$中，得到对偶函数为

$$g(\boldsymbol{\alpha}) = \frac{1}{2}\|\mathbf{w}\|^2 - \sum_{i=1}^{N} \alpha_i y_i \mathbf{w}\cdot\mathbf{x}_i + \sum_{i=1}^{N} \alpha_i$$

$$= \sum_{i=1}^{N} \alpha_i + \frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N} \alpha_i \alpha_j y_i y_j \mathbf{x}_i \cdot \mathbf{x}_j - \sum_{i=1}^{N}\sum_{j=1}^{N} \alpha_i \alpha_j y_i y_j \mathbf{x}_i \cdot \mathbf{x}_j$$

$$= \sum_{i=1}^{N} \alpha_i - \frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N} \alpha_i \alpha_j y_i y_j \mathbf{x}_i \cdot \mathbf{x}_j .$$

支持向量机（SVM）

其次，求解对偶优化函数，计算对偶变量α的最优解，即使得对偶函数取极大值的解。根据对偶公式得到对偶优化问题的具体表示为

$$\max_{\alpha} \quad \sum_{i=1}^{N}\alpha_i - \frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N}\alpha_i\alpha_j y_i y_j \mathbf{x}_i^{\cdot}\mathbf{x}_j$$

$$s.t. \quad \alpha_i \geq 0$$

$$\sum_{i=1}^{N}\alpha_i y_i = 0.$$

参数b可以通过如下条件求得

$$\alpha_i(y_i(\mathbf{w}^{\cdot}\mathbf{x}_i + b) - 1) = 0, \quad i = 1,2,...,N.$$

支持向量机（SVM）

对于所有的支持向量，由于相应的对偶变量为正值，原优化问题中的不等式约束变成等式约束。因此，根据任一个支持向量$\mathbf{x}_j$，可得出

$$b = y_j - \mathbf{w}^{\cdot} \mathbf{x}_j .$$

至此，对于待分类样本$\mathbf{x}$，支持向量机分类器表示为

$$h(\mathbf{x}) = sign[\sum_{i=1}^{N} y_i a_i \mathbf{x}^{\cdot} \mathbf{x}_i + b],$$

其中运用了变换 $\mathbf{w}^{\text{唵}}\mathbf{x} = \sum_{i=1}^{N} y_i \alpha_i \mathbf{x}\ \mathbf{x}_i .$

支持向量机（SVM）

- 随堂练习

设正例点有$x_1 = (3,3)^T$，$x_2 = (4,3)^T$；负例点有$x_3 = (1,1)^T$。请求解最优分类超平面，并指出哪些点是支持向量。

支持向量机（SVM）

- 随堂练习

$$\begin{aligned} \max_{\alpha} \quad & \sum_{i=1}^{N}\alpha_i - \frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N}\alpha_i\alpha_j y_i y_j \mathbf{x}_i^{\cdot}\mathbf{x}_j \\ s.t. \quad & \alpha_i \geq 0 \\ & \sum_{i=1}^{N}\alpha_i y_i = 0. \end{aligned}$$

代入对偶问题，

$$\textbf{max}\ \alpha_1 + \alpha_2 + \alpha_3 - \frac{1}{2}(18\alpha_1^2 + 25\alpha_2^2 + 2\alpha_3^2 + 42\alpha_1\alpha_2 - 12\alpha_1\alpha_3 - 14\alpha_2\alpha_3)$$

$$\textbf{s.t.}\quad \alpha_1 + \alpha_2 - \alpha_3 = 0;\ \alpha_i \geq 0, i = 1,2,3$$

支持向量机（SVM）

- 随堂练习

将 $\alpha_3 = \alpha_1 + \alpha_2$ 代入优化目标，得 $2\alpha_1 + 2\alpha_2 - 4{\alpha_1}^2 - \frac{13}{2}{\alpha_2}^2 - 10\alpha_1\alpha_2$；

对α_1和α_2求偏导数并令其等于0，可求出极值点$\left(\frac{3}{2}, -1\right)$。极值点不满足约束条件，所以最大值应该在边界上达到，也就是$\alpha_1 = 0$或者$\alpha_2 = 0$的情况（代入后分别对另一个乘子求导并令导数等于0）。代入分别为$\left(0, \frac{2}{13}\right)$和$\left(\frac{1}{4}, 0\right)$。

代入后$\left(\frac{1}{4}, 0\right)$取值更大。然后，$\alpha_3 = \frac{1}{4}$。

则$w_1 = w_2 = \frac{1}{2}$，$b = -2$。

分类超平面为$\frac{1}{2}x^{(1)} + \frac{1}{2}x^{(2)} - 2$。支持向量为$x_1 = (3,3)^T$和$x_3 = (1,1)^T$

线性不可分的支持向量机

- 核方法

设x和z来自空间Γ（不一定是线性空间），满足下式的函数κ被称为核函数

$$\kappa(x,z)=\langle\phi(x),\phi(z)\rangle,$$

其中ϕ是从空间Γ到希尔伯特空间F的映射

$$\phi: x\in\Gamma \text{ a} \quad \phi(x)\in F,$$

空间F通常被称为特征空间。

对于支持向量机，设映射函数为$\phi(\mathbf{x})$，那么 $\kappa(\mathbf{x}_i,\mathbf{x}_j)=\phi(\mathbf{x}_i)\cdot\phi(\mathbf{x}_j)$
对偶问题的优化目标变为

$$\sum_{i=1}^{N}\alpha_i-\frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N}\alpha_i\alpha_j y_i y_j\kappa(\mathbf{x}_i,\mathbf{x}_j).$$

原来的$\mathbf{w}\cdot\mathbf{x}$ 变为$\mathbf{w}\cdot\phi(\mathbf{x})$，同时考虑到参数w可以由对偶变量和输入数据表达

$$\mathbf{w}\cdot\phi(\mathbf{x})=\sum_{i=1}^{N}\alpha_i y_i\kappa(\mathbf{x}_i,\mathbf{x}).$$

线性不可分的支持向量机

常见的基本核函数：

- **线性核**

$$\kappa(\mathbf{x}_i, \mathbf{x}_j) = \mathbf{x}_i^{\cdot} \mathbf{x}_j .$$

- **多项式核**

$$\kappa(\mathbf{x}_i, \mathbf{x}_j) = (\mathbf{x}_i^{\cdot} \mathbf{x}_j + 1)^d ,$$

其中参数d是多项式次数。

- **高斯核**

$$\mathbf{x}_i^{\cdot} \mathbf{x}_j = \exp\left\{ -\frac{\| \mathbf{x}_i - \mathbf{x}_j \|^2}{2\sigma^2} \right\},$$

也称为径向基函数核，其中参数σ和多项式核函数中的参数d一样，通过需要通过模型选择来确定具体取值。