神经网络
Neural Networks
第十章

概率神经网络

史忠植

中国科学院计算技术研究所
http://www.intsci.ac.cn/

内容提要

- 10.1 概述
- 10.2 贝叶斯定理
- 10.3 概率密度函数
- 10.4 模式分类的贝叶斯判定策略
- 10.5 密度估计的一致性
- 10.6 概率神经网络
- 10.7 激活函数
- 10.8 贝叶斯阴阳系统理论

样本空间和事件

考虑一个事先不知道输出的试验：

- 试验的样本空间：所有可能输出 W的集合

  如果抛掷两次硬币，则样本空间为 Ω= H H ,H T , TH , T T

- 事件*A*是样本空间Ω 的子集

  上述试验中第一次正面向上的事件为A= HH, HT

概率

- 对每个事件*A*，我们定义一个数字P(*A*)，称为A的概率。概率根据下述三条公理：

  1、事件A 的概率是一个非负实数：$\mathbb{P}(A) \geq 0$

  2、合法命题的概率为1：$\mathbb{P}(\text{true}) = 1$

  3、对两两不相交（互斥）事件*A1, A2, …,*

$$\mathbb{P}\left(\bigcup_{i=1}^{\infty} A_i\right) = \sum_{i=1}^{\infty} \mathbb{P}(A_i)$$

从上述三个公理，可推导出概率的所有的其他性质。频率学派和贝叶斯学派都满足该公理

分布函数

- 令X为一随机变量， x为X的一具体值（数据）
- 则随机变量X的累积分布函数 $F: \mathbb{R} \to [0,1]$ (cumulative distribution function, CDF) 定义为

$$F(x) = \mathbb{P}(X \leq x)$$

- 若X的取值为一些可数的数值{x1，x2, …}则称其为离散型随机变量。

统计概率

统计概率:若在大量重复试验中，事件A发生的频率稳定地接近于一个固定的常数p，它表明事件A出现的可能性大小，则称此常数p为事件A发生的概率，记为P(A)，即

$$p=P(A)$$

可见概率就是频率的稳定中心。任何事件A的概率为不大于1的非负实数，即

$$0<P(A)<1$$

概率分布

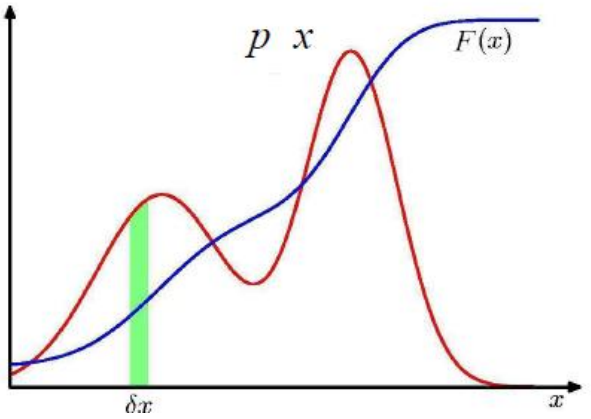
- 对连续型随机变量X，如果存在一个函数P ，使得对所有的x，p $\geq$0 ，且对任意a $\leq$b 有

$$\mathbb{P}\ a<X<b\ =\int_a^b p\ x\ dx$$

- 则函数 被称为概率密度函数 **(probability density function, pdf)**。
- 当F可微时

$$F\ x\ =\int_{-\infty}^{x} p\ t\ dt$$

$$p\ x\ =F'\ x$$



Bernoulli分布

- **一些离散分布的例子：**

  1：*X为一次抛硬币的输出，*

$$\mathbb{P}\ X=1\ =\theta,\quad \mathbb{P}\ X=0\ =1-\theta,\quad \theta\in\ 0,1$$

- **我们称X服从参数为θ的*Bernoulli*分布，记为**$X \sim \mathrm{Ber}(\theta)$

$$p\ \ x|\theta\ =\theta^{x}\ \ 1-\theta\ ^{1-x},\ \text{for}\ x\in\ 0,1$$

二项分布

与Bernoulli分布相关的一些分布：

- **二项(Binomial)分布**：n次试验中硬币输出（如正面）数目X满足二项分布，记为 $X \sim \mathrm{Bin}(n,\theta)$

$$p\ x \mid n,\theta\ = \binom{n}{x} \theta^{x}\ 1-\theta\ ^{n-x}, \quad \binom{n}{x} = \frac{n!}{n-x\ !x!}$$

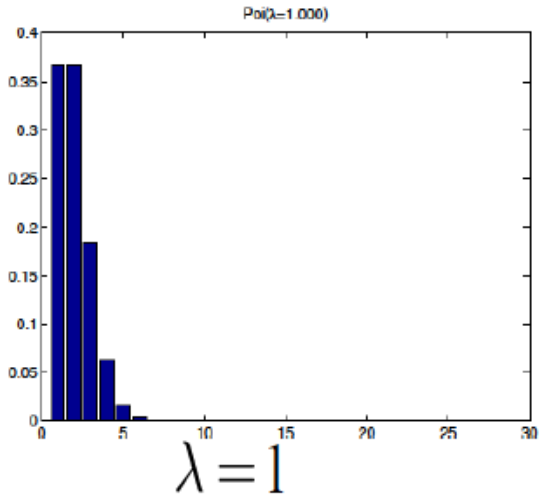
- **多项(Multinomial)分布**：假设一个有K面的骰子，其中抛掷到第j面的概率为θ_j，假设我们一共抛掷n次。令 $\mathbf{x} = x_1,\ldots,x_K$ 表示随机向量，其中x_j 表示抛掷到第j面的次数，则x的分布为多项分布，记为 $X \sim \mathrm{Mu}(n,\theta)$

$$p\ \mathbf{x} \mid n,\theta\ = \binom{n}{x_1\ \ldots x_K} \prod_{j=1}^{K} \theta_j^{x_j}, \quad \binom{n}{x_1\ \ldots x_K} = \frac{n!}{x_1!\ \ldots\ x_K!}$$

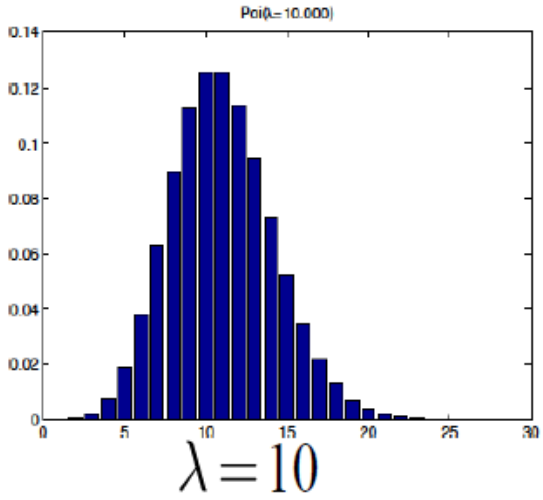
泊松分布

- 2 **泊松分布**： 泊松分布通常用来对稀有事件（如交通事故）的次数进行建模。若 $X \in 0,1,2,....$ 满足参数为 λ 的泊松分布 $X \sim \mathrm{Poi}(\lambda)$ ，其pmf为

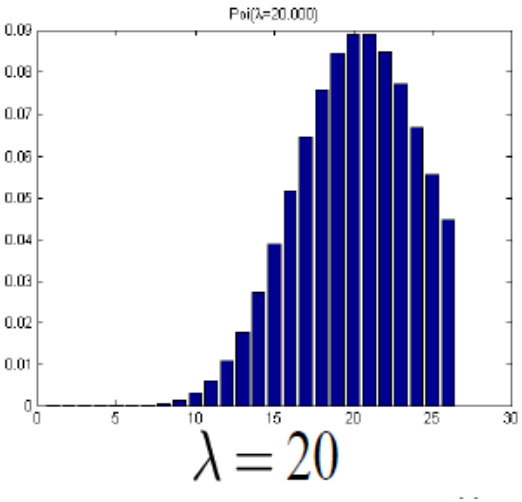
$$p\ x|\lambda = e^{-\lambda}\frac{\lambda^x}{x!}$$



$\lambda = 1$



$\lambda = 10$



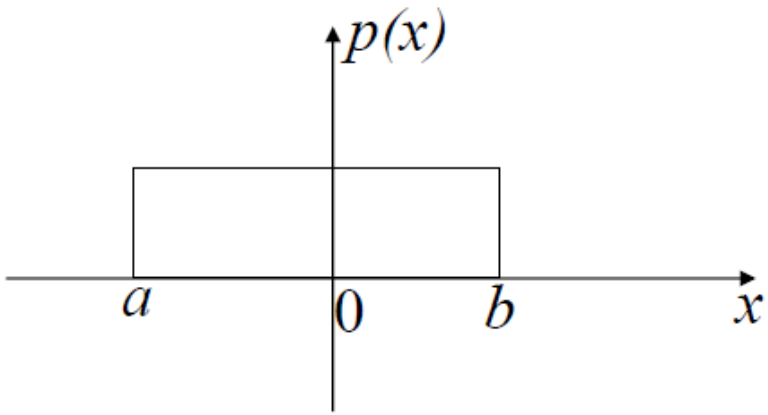
$\lambda = 20$

均匀分布

一些连续分布的例子：

- 1: **均匀分布**：$X \sim \mathrm{U}(a,b)$

$$p\ x\ = \begin{cases} \dfrac{1}{b-a} & x \in\ a,b \\ 0 & otherwise \end{cases}$$

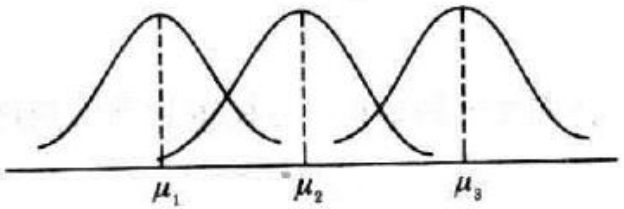


高斯分布

- 2 **高斯分布/正态分布**：$X \sim N(\mu, \sigma^2)$
  - 最重要的分布之一

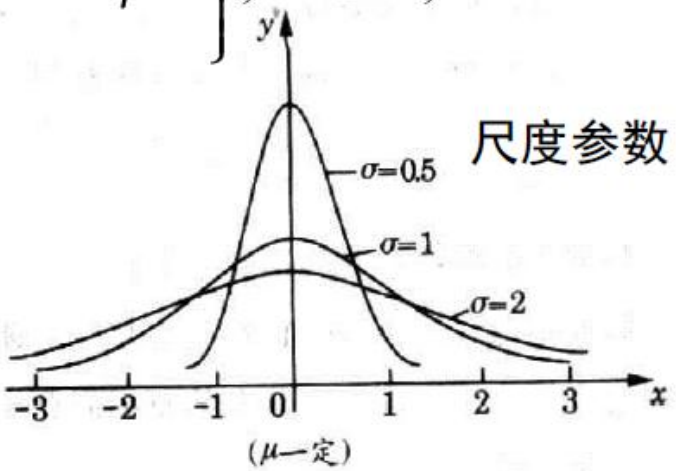
$$p(x) = \frac{1}{\sqrt{2\pi}\sigma} \exp\left\{-\frac{1}{2\sigma^2}(x-\mu)^2\right\}, \quad x \in \mathbb{R}, \ \sigma > 0$$

位置参数



(σ一定)

尺度参数



(μ一定)

- 在实际遇到的许多随机现象都近似服从正态分布
  - 如考试成绩

标准正态分布

- 当 $\mu = 0, \sigma = 1$ 时，称为**标准正态分布**，通常用 Z 表示服从标准正态分布的变量，记为 $Z \sim \mathcal{N}(0,1)$
- pdf和CDF分别记为 $\phi(z)$,$\Phi(z)$

- 标准化：
  - 若 $X \sim \mathcal{N}(\mu, \sigma^2)$ ，则 $Z = (X - \mu)/\sigma \sim \mathcal{N}(0,1)$
  - 若 $Z \sim \mathcal{N}(0,1)$ ，则 $X = \mu + \sigma Z \sim \mathcal{N}(\mu, \sigma^2)$

退化的高斯分布

- 当$\sigma^2 \to 0$时，高斯分布退化为无限高无限窄、中心位于μ的“针”状分布

$$\lim_{\sigma^2 \to 0} \mathcal{N}(x \mid \mu, \sigma^2) = \delta\ x - \mu$$

- 其中Dirac delta 函数

$$\delta\ x\ = \begin{cases} \infty & if\ x = 0 \\ 0 & f\ x \neq 0 \end{cases}, \quad 使得 \int_{-\infty}^{\infty} \delta\ x\ dx = 1$$

- 有用的性质：将某个信号从求和/积分中筛选出来

$$\int_{-\infty}^{\infty} f\ x\ \delta\ x - u\ dx = f\ u$$

- 别将Dirac delta函数与Kronecker delta 函数混淆：

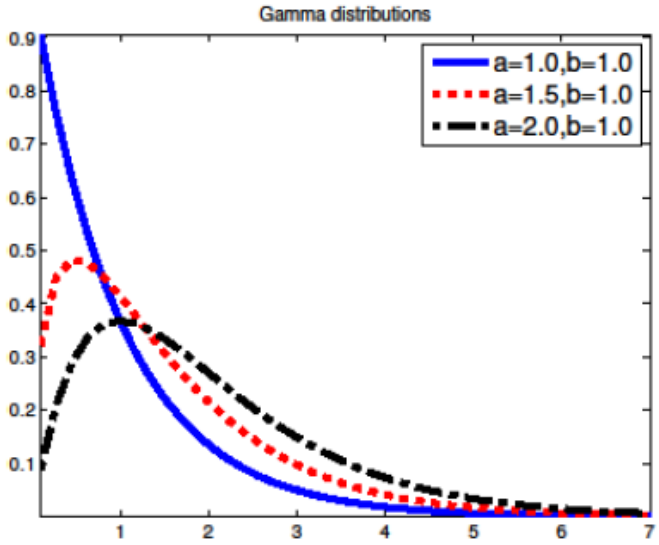
$$\delta_{ij} = \mathbb{I}\ i = j\ = \begin{cases} 1 & if\ i = j \\ 0 & f\ i \neq j \end{cases}$$

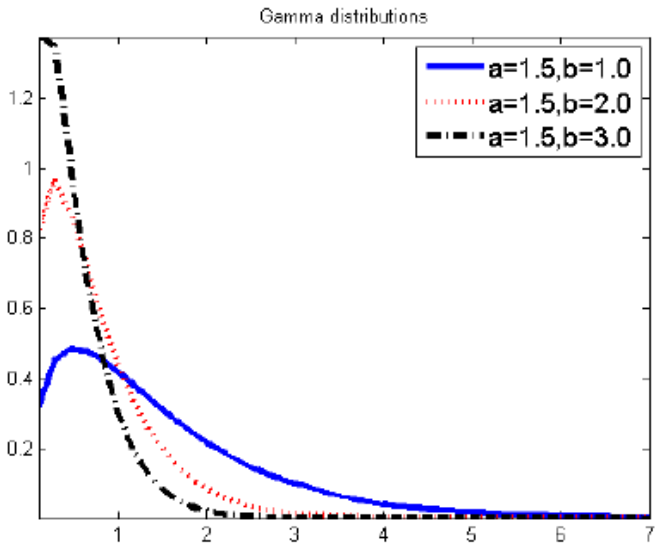
Gamma分布

- 3 **Gamma分布**：对任意正实数随机变量 $x>0$，Gamma分布为 $x \sim \mathrm{Ga}(\mathrm{shape}=a, \mathrm{rate}=b)$

$$p(x \mid a,b) = \frac{b^a}{\Gamma(a)} x^{a-1} e^{-xb}$$

- 其中a为**形状**参数，b为**比率**度参数





Gamma分布

- Gamma分布的另一种表示：用**尺度**参数代替比率参数

$$\mathrm{Ga}\left(x \mid \mathrm{shape}=\alpha, \mathrm{scale}=\beta\right)=\frac{1}{\beta^{\alpha}\Gamma\left(\alpha\right)}x^{\alpha-1}e^{-x/\beta}$$

$$=\mathrm{Ga}\left(x \mid \mathrm{shape}=\alpha, \mathrm{rate}=\frac{1}{\beta}\right)$$

- Gamma分布的几个特例请见2.4.5

- **反Gamma分布**：若 $X \sim \mathrm{Ga}(a,b)$，则 $\frac{1}{X} \sim \mathrm{IG}(a,b)$

$$\mathrm{IG}\left(x \mid \mathrm{shape}=\alpha, \mathrm{scale}=\beta\right)=\frac{\beta^{\alpha}}{\Gamma\left(\alpha\right)}x^{-\alpha-1}e^{-x/\beta}$$

Gaussian Scale Mixtures

- 一些有意思的分布可以表示为一组无限个高斯的**加权和**，其中每个高斯的**方差不同**

$$p(x) = \int \mathcal{N}(x \mid \mu, \tau^2)\pi(\tau^2)\, d\tau^2$$

某个分布

- 如**Student**分布：

$$\mathcal{T}(x \mid \mu, \sigma^2, \nu) = \int_0^\infty \mathcal{N}(x \mid \mu, \sigma^2/\lambda)\mathrm{Ga}\left(\lambda \mid \frac{\nu}{2}, \frac{\nu}{2}\right) d\lambda$$

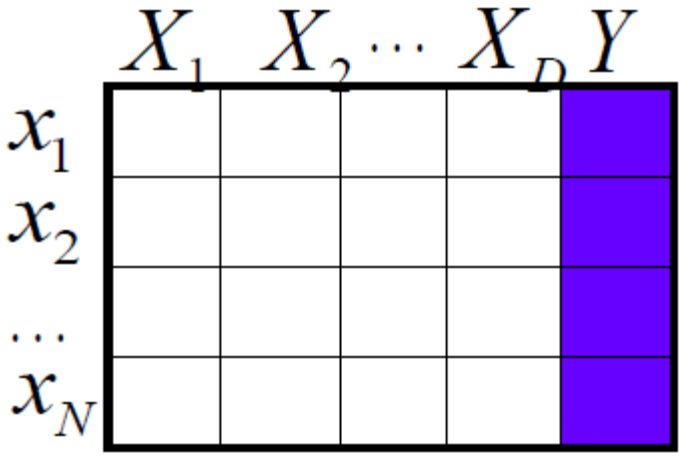
  - **Student** 分布和高斯分布很像，但尾巴更长
  - 当自由度 $\nu \to \infty$ 时，极限分布为高斯分布

$$\mathcal{T}(x \mid \mu, \sigma^2, \infty) = \lim_{\nu \to \infty} \int_0^\infty \mathcal{N}(x \mid \mu, \sigma^2/\lambda)\mathrm{Ga}\left(\lambda \mid \frac{\nu}{2}, \frac{\nu}{2}\right) d\lambda = \mathcal{N}(x \mid \mu, \sigma^2)$$

18

多元随机向量的分布

- 我们可以在多个随机变量组成的向量上定义分布， 称之为多元随机向量的分布。
- 机器学习中我们的数据集通常由多元随机向量分布的样本组成。每一列为一个随机变量，也可以将整个随机向量看成一个随机变量。



边缘分布

➢对离散型随机变量，如果(X, Y)有联合密度函数

则X的边缘密度函数定义为

$$p_X(x) = \mathbb{P}(X = x) = \sum_y \mathbb{P}(X = x, Y = y) = \sum_y p(x, y)$$

➢对连续情况：

$$p_X(x) = \int p(x, y)\, dy$$

联合分布包含了随机向量概率分布的信息 联合分布唯一确定了边缘分布，但反之通常不成立

条件概率

条件概率：我们把事件B已经出现的条件下，事件A发生的概率记做为P(A|B)。并称之为在B出现的条件下A出现的条件概率，而称P(A)为无条件概率。

若事件A与B中的任一个出现，并不影响另一事件出现的概率，即当P(A)＝P(A·B)或P(B)＝P(B·A)时，则称A与B是相互独立的事件。

条件分布

- 对离散随机变量*X, Y,* 给定Y=y时X的分布为条件分布。当y ＞ 0 时：

$$p_{X|Y}\ x \mid y = \mathbb{P}\ X = x \mid Y = y = \frac{\mathbb{P}\ X = x, Y = y}{\mathbb{P}\ Y = y} = \frac{p_{X,Y}\ x, y}{p_Y\ y}$$

- 当对X, Y为连续随机变量时，当y>0时，条件分布为：

$$p_{X|Y}\ x \mid y = \frac{p_{X,Y}\ x, y}{p_Y\ y}$$

- 注意：$p_{X,Y}\ x, y = p_{X|Y}\ x \mid y\ p_Y\ y = p_{Y|X}\ y \mid x\ p_X\ x$

条件概率链规则

- $p\ x,y\ = p\ x|y\ \ p\ \ y$

- **链规则**

$$p\ x,y,z\ = p\ x|y,z\ \ p\ \ y,z$$

$$= p\ x|y,z\ \ p\ \ y|z\ \ p\ \ z$$

- 或

$$p\ x,y,z\ = p\ x\ \ p\ y|x\ \ p\ z|x,y$$

加法定理

两个不相容(互斥)事件之和的概率，等于两个事件概率之和，即

$$P(A+B) = P(A) + P(B)$$

若A、B为两任意事件，则：

$$P(A+B) = P(A) + P(B) - P(AB)$$

乘法定理

设A、B为两个任意的非零事件，则其乘积的概率等于A(或B)的概率与在A(或B)出现的条件下B(或A)出现的条件概率的乘积。

$$P(A \cdot B) = P(A) \cdot P(B|A)$$

或 $$P(A \cdot B) = P(B) \cdot P(A|B)$$

贝叶斯网络是什么

- 贝叶斯网络是用来表示变量间连接概率的图形模式，它提供了一种自然的表示因果信息的方法，用来发现数据间的潜在关系。在这个网络中，用节点表示变量，有向边表示变量间的依赖关系。
- 贝叶斯方法正在以其独特的不确定性知识表达形式、丰富的概率表达能力、综合先验知识的增量学习特性等成为当前数据挖掘众多方法中最为引人注目的焦点之一。

贝叶斯网络是什么

- 贝叶斯（**Reverend Thomas Bayes 1702-1761**）学派奠基性的工作是贝叶斯的论文“关于几率性问题求解的评论”。或许是他自己感觉到它的学说还有不完善的地方，这一论文在他生前并没有发表，而是在他死后，由他的朋友发表的。著名的数学家拉普拉斯（**Laplace P. S.**）用贝叶斯的方法导出了重要的“相继律”，贝叶斯的方法和理论逐渐被人理解和重视起来。但由于当时贝叶斯方法在理论和实际应用中还存在很多不完善的地方，因而在十九世纪并未被普遍接受。

贝叶斯网络是什么

- 二十世纪初，意大利的菲纳特（B. de Finetti）以及英国的杰弗莱（Jeffreys H.）都对贝叶斯学派的理论作出重要的贡献。第二次世界大战后，瓦尔德（Wald A.）提出了统计的决策理论，在这一理论中，贝叶斯解占有重要的地位；信息论的发展也对贝叶斯学派做出了新的贡献。1958年英国最悠久的统计杂志Biometrika全文重新刊登了贝叶斯的论文，20世纪50年代，以罗宾斯（Robbins H.）为代表，提出了经验贝叶斯方法和经典方法相结合，引起统计界的广泛注意，这一方法很快就显示出它的优点，成为很活跃的一个方向。

贝叶斯网络是什么

- 随着人工智能的发展，尤其是机器学习、数据挖掘等兴起，为贝叶斯理论的发展和应用提供了更为广阔的空间。贝叶斯理论的内涵也比以前有了很大的变化。80年代贝叶斯网络用于专家系统的知识表示，90年代进一步研究可学习的贝叶斯网络，用于数据采掘和机器学习。近年来，贝叶斯学习理论方面的文章更是层出不穷，内容涵盖了人工智能的大部分领域，包括因果推理、不确定性知识表达、模式识别和聚类分析等。并且出现了专门研究贝叶斯理论的组织和学术刊物ISBA

贝叶斯网络的应用领域

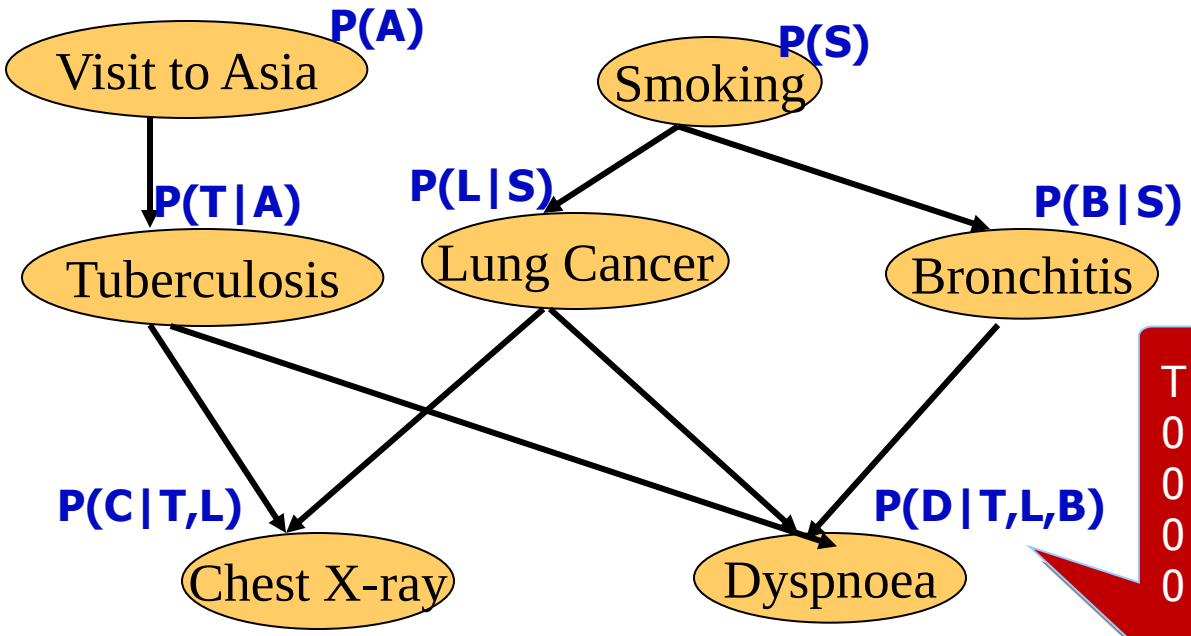
- 辅助智能决策
- 数据融合
- 模式识别
- 医疗诊断
- 文本理解
- 数据挖掘

贝叶斯网络定义

贝叶斯网络是表示变量间概率依赖关系的有向无环图，这里每个节点表示领域变量，每条边表示变量间的概率依赖关系，同时对每个节点都对应着一个条件概率分布表(CPT)，指明了该变量与父节点之间概率依赖的数量关系。

贝叶斯网的表示方法

贝叶斯网络是表示变量间概率依赖关系的有向无环图



CPT:

T	L	B	D=0	D=1
0	0	0	0.1	0.9
0	0	1	0.7	0.3
0	1	0	0.8	0.2
0	1	1	0.9	0.1
			...	

$$P(A, S, T, L, B, C, D) = P(A)\,P(S)\,P(T|A)\,P(L|S)\,P(B|S)\,P(C|T,L)\,P(D|T,L,B)$$

条件独立性假设 ➡ 有效的表示

先验概率

先验概率是指根据历史的资料或主观判断所确定的各事件发生的概率，该类概率没能经过实验证实，属于检验前的概率，所以称之为先验概率。先验概率一般分为两类，一是客观先验概率，是指利用过去的历史资料计算得到的概率；二是主观先验概率，是指在无历史资料或历史资料不全的时候，只能凭借人们的主观经验来判断取得的概率。

后验概率

后验概率一般是指利用贝叶斯公式，结合调查等方式获取了新的附加信息，对先验概率进行修正后得到的更符合实际的概率。

联合概率

联合概率也叫乘法公式，是指两个任意事件的乘积的概率，或称之为交事件的概率。

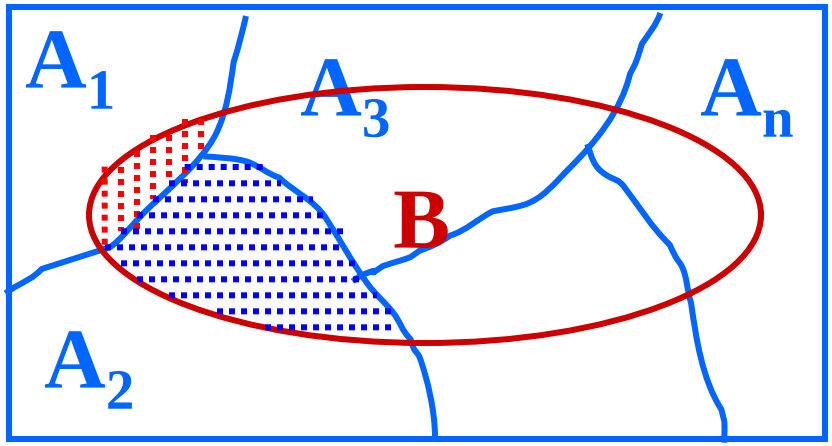
全概率公式

设$A_1,A_2,\ldots,A_n$是两两互斥的事件，且

$P(A_i)>0$，　$i=1,2,\ldots,n$，　$A_1+A_2+\ldots,+A_n=\Omega$

另有一事件$B=BA_1+BA_2+\ldots,+BA_n$

$$P(B)=\sum_{i=1}^{n}P(A_i)P(B\mid A_i)$$



称满足上述条件的

$A_1,A_2,\ldots,A_n$为完备事件组.

全概率

例:某汽车公司下属有两个汽车制造厂,全部产品的40%由甲厂生产,60%由乙厂生产.而甲乙二厂生产的汽车的不合格率分别为1%,2%.求从公司生产的汽车中随机抽取一辆为不合品的概率.

解:设A_1,A_2分别表示{甲厂汽车} {乙厂汽车},B表示{不合格品}

$$P(A_1)=0.4,\ P(A_2)=0.6$$

$$P(B/A_1)=0.01,\ P(B/A_2)=0.02$$

$$\because A_1A_2=\varphi$$

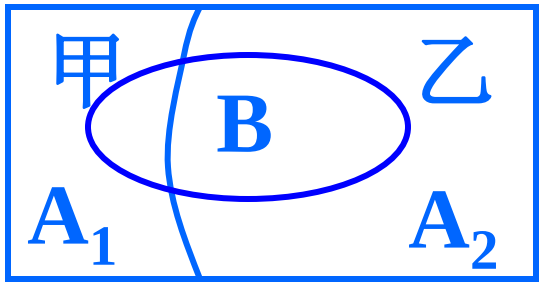
$$P(B)=P(A_1B+A_2B)$$

$$=P(A_1B)+P(A_2B)$$

$$=P(A_1)P(B/A_1)+P(A_2)P(B/A_2)$$

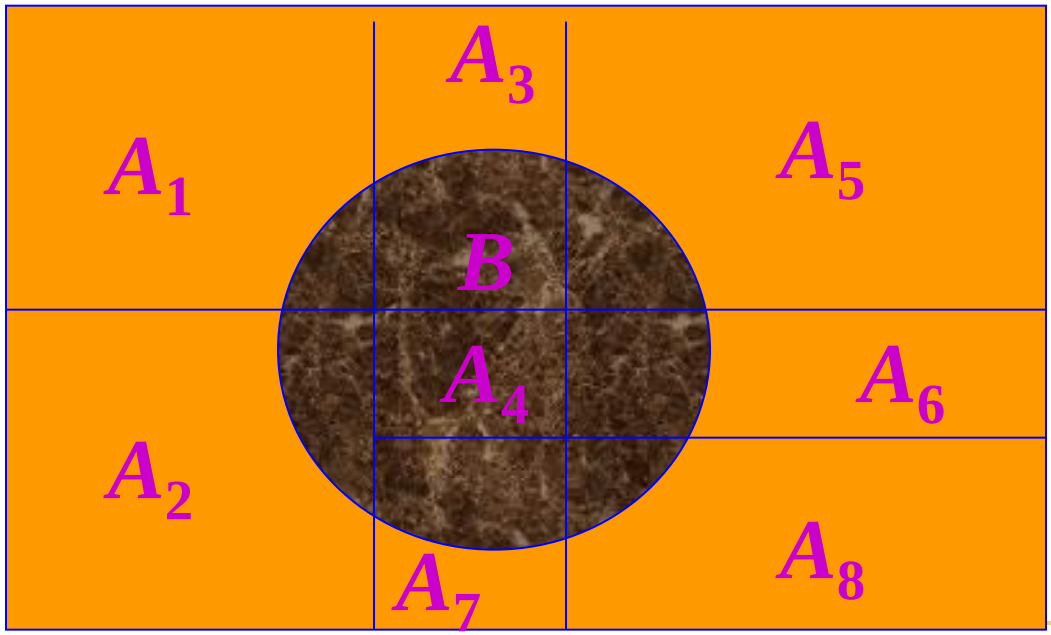
$$=0.4\times0.01+0.6\times0.02$$

$$=0.016$$



全概率

诸A_i是原因
B是结果



由此可以形象地把全概率公式看成为“由原因推结果”，每个原因对结果的发生有一定的“作用”，即结果发生的可能性与各种原因的“作用”大小有关. 全概率公式表达了它们之间的关系 .

贝叶斯公式

设$A_1,A_2,\ldots,A_n$是样本空间中的完备事件组且$P(A_i)>0$，$i=1,2,\ldots,n$, 另有一事件B，则有

$$P(A_i \mid B) = P(A_i)P(B \mid A_i) \Big/ \sum_{j=1}^{n} P(A_j)P(B \mid A_j)$$

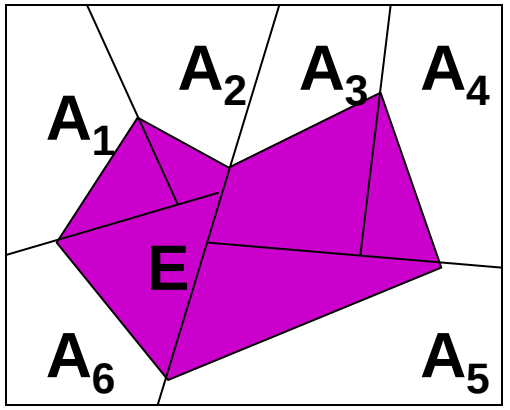
$$i = 1,2,\cdots,n$$

该公式于1763年由贝叶斯(Bayes)给出．它是在观察到事件B已发生的条件下，寻找导致B发生的每个原因的概率.

贝叶斯规则

$$p(A \mid B) = \frac{p(A,B)}{p(B)} = \frac{p(B \mid A)p(A)}{p(B)}$$

$$p(A_i \mid E) = \frac{p(E \mid A_i)p(A_i)}{p(E)} = \frac{p(E \mid A_i)p(A_i)}{\sum_i p(E \mid A_i)p(A_i)}$$



- 基于条件概率的定义
- $p(A_i|E)$ 是在给定证据下的后验概率
- $p(A_i)$ 是先验概率
- $P(E|A_i)$ 是在给定A_i下的证据似然
- $p(E)$ 是证据的预定义后验概率

贝叶斯网络的概率解释

- 任何完整的概率模型必须具有表示（直接或间接）该领域变量联合分布的能力。完全的枚举需要指数级的规模（相对于领域变量个数）
- 贝叶斯网络提供了这种联合概率分布的紧凑表示：分解联合分布为几个局部分布的乘积：

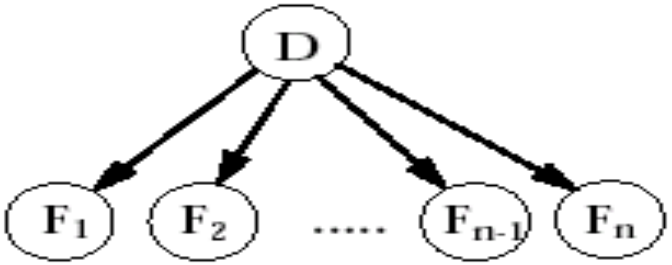
$$P(x_1, x_2, \cdots x_n) = \prod_i P(x_i \mid pa_i)$$

- 从公式可以看出，需要的参数个数随网络中节点个数呈线性增长，而联合分布的计算呈指数增长。
- 网络中变量间独立性的指定是实现紧凑表示的关键。这种独立性关系在通过人类专家构造贝叶斯网中特别有效。

简单贝叶斯学习模型

简单贝叶斯学习模型（Simple Bayes 或 Naïve Bayes ）将训练实例*I*分解成特征向量*X*和决策类别变量*C*。简单贝叶斯模型假定特征向量的各分量间相对于决策变量是相对独立的，也就是说各分量独立地作用于决策变量。尽管这一假定一定程度上限制了简单贝叶斯模型的适用范围，然而在实际应用中，不仅以指数级降低了贝叶斯网络构建的复杂性，而且在许多领域，在违背这种假定的条件下，简单贝叶斯也表现出相当的健壮性和高效性[111]，它已经成功地应用到分类、聚类及模型选择等数据挖掘的任务中。目前，许多研究人员正致力于改善特征变量间独立性的限制[54]，以使它适用于更大的范围。

简单贝叶斯
Naïve Bayesian



$$P(D \mid F_1 \ldots F_n) = P(D) * \prod_i P(F_i \mid D)$$

- 结构简单 – 只有两层结构
- 推理复杂性与网络节点个数呈线性关系

简单贝叶斯学习模型

设样本A表示成属性向量，如果属性对于给定的类别独立，那么P(A|C_i)可以分解成几个分量的积：

$$P(a_1 \mid C_i) * P(a_2 \mid C_i) * \cdots P(a_m \mid C_i)$$

a_i是样本A的第i个属性

简单贝叶斯学习模型

简单贝叶斯分类模型

$$P(C_i \mid A) = \frac{P(C_i)}{P(A)} \prod_{j=1}^{m} P(a_j \mid C_i)$$

$$P(C_i) \qquad P(a_j \mid C_i)$$

这个过程称之为简单贝叶斯分类 (SBC: Simple Bayesian Classifier)。一般认为，只有在独立性假定成立的时候，SBC才能获得精度最优的分类效率；或者在属性相关性较小的情况下，能获得近似最优的分类效果。

简单贝叶斯模型的提升

- 基于Boosting简单贝叶斯模型

提升方法（Boosting）总的思想是学习一系列分类器，在这个序列中每一个分类器对它前一个分类器导致的错误分类例子给与更大的重视。尤其是，在学习完分类器Hk之后，增加了由Hk导致分类错误的训练例子的权值，并且通过重新对训练例子计算权值，再学习下一个分类器H_{k+1}。这个过程重复T次。最终的分类器从这一系列的分类器中综合得出。

Boosting背景

- 来源于:PAC-Learning Model
  Valiant 1984 -11
- 提出问题:
  - 强学习算法: 准确率很高的学习算法
  - 弱学习算法: 准确率不高,仅比随机猜测略好
  - 是否可以将弱学习算法提升为强学习算法

Boosting背景

- 最初的boosting算法
  Schapire 1989
- AdaBoost算法
  Freund and Schapire 1995

Boosting—concepts(3)

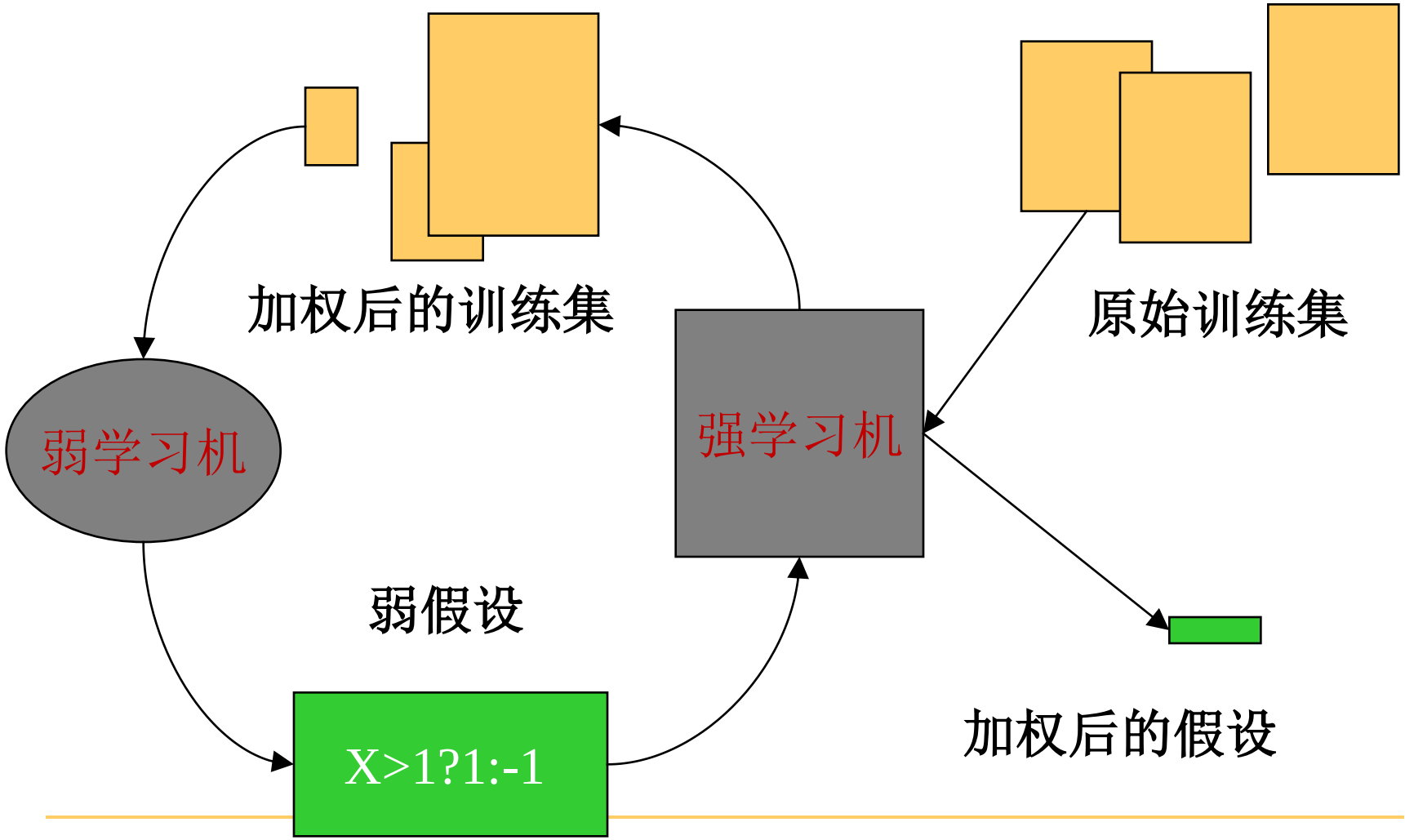
- 弱学习机（**weak learner):** 对一定分布的训练样本给出假设（仅仅强于随机猜测）

  根据有云猜测可能会下雨
- 强学习机（**strong learner):** 根据得到的弱学习机和相应的权重给出假设（最大程度上符合实际情况：**almost perfect expert)**

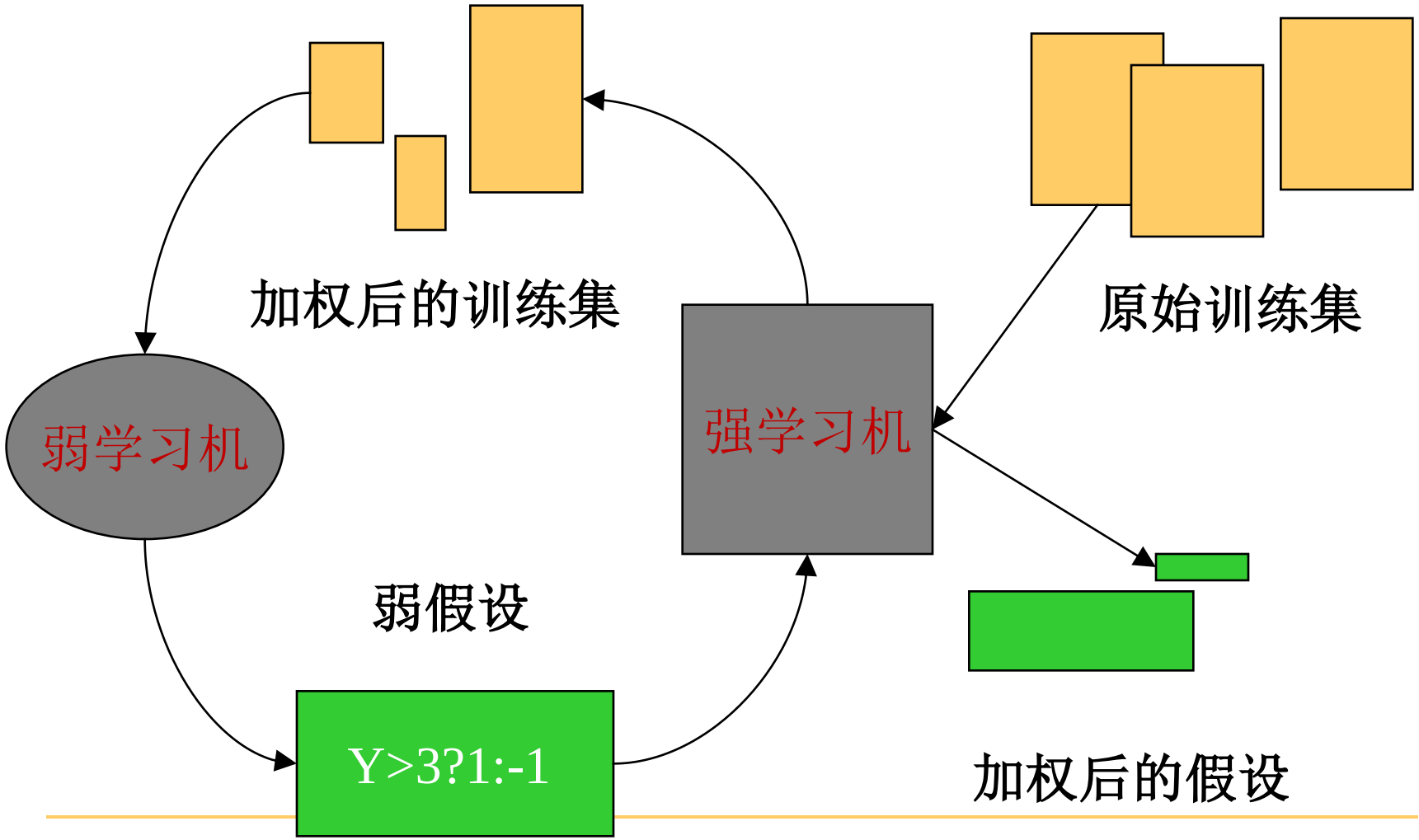
  根据**CNN,ABC,CBS**以往的预测表现及实际天气情况作出综合准确的天气预测

- 弱学习机 —Boosting→ 强学习机

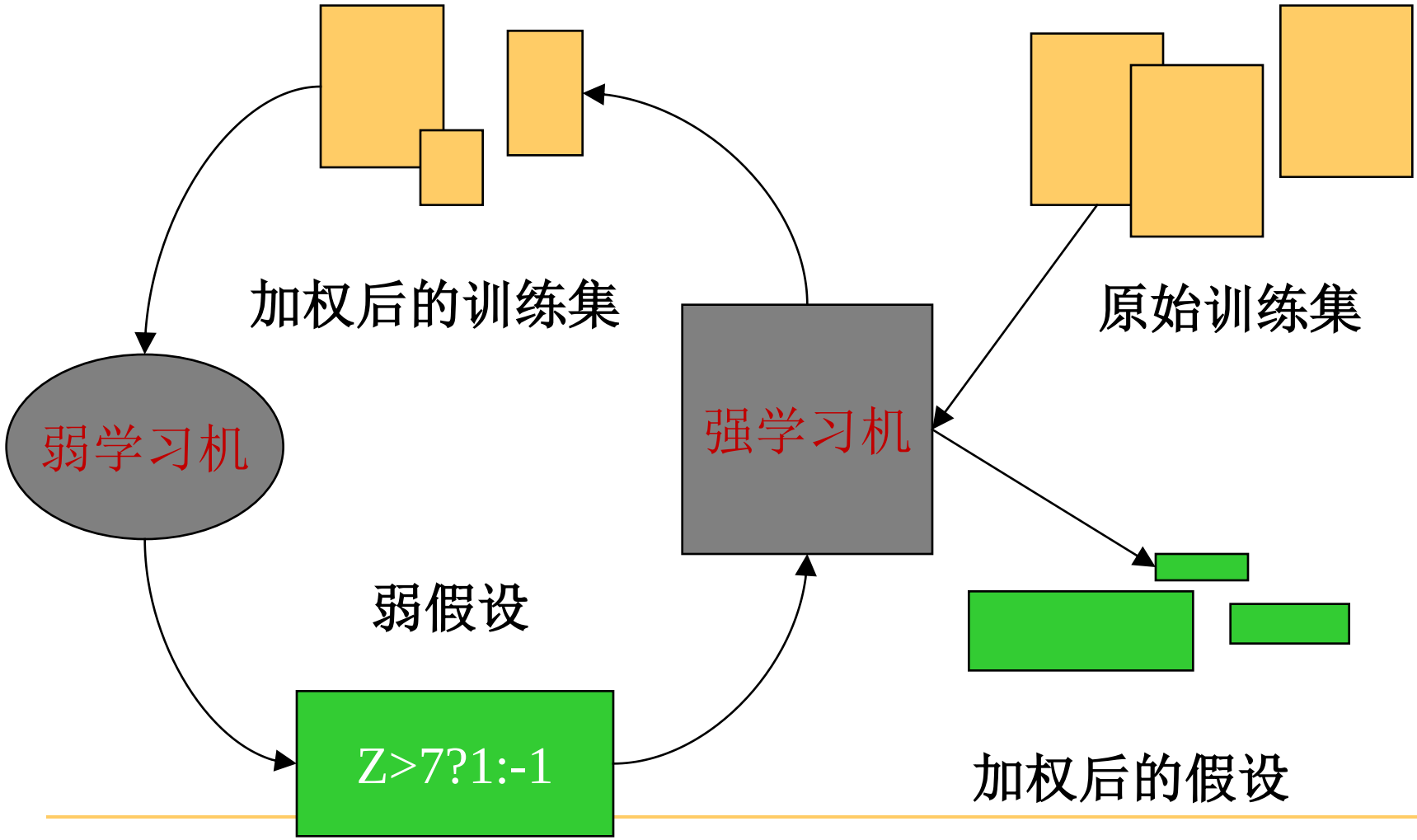
Boosting流程(loop1)



Boosting流程(loop2)

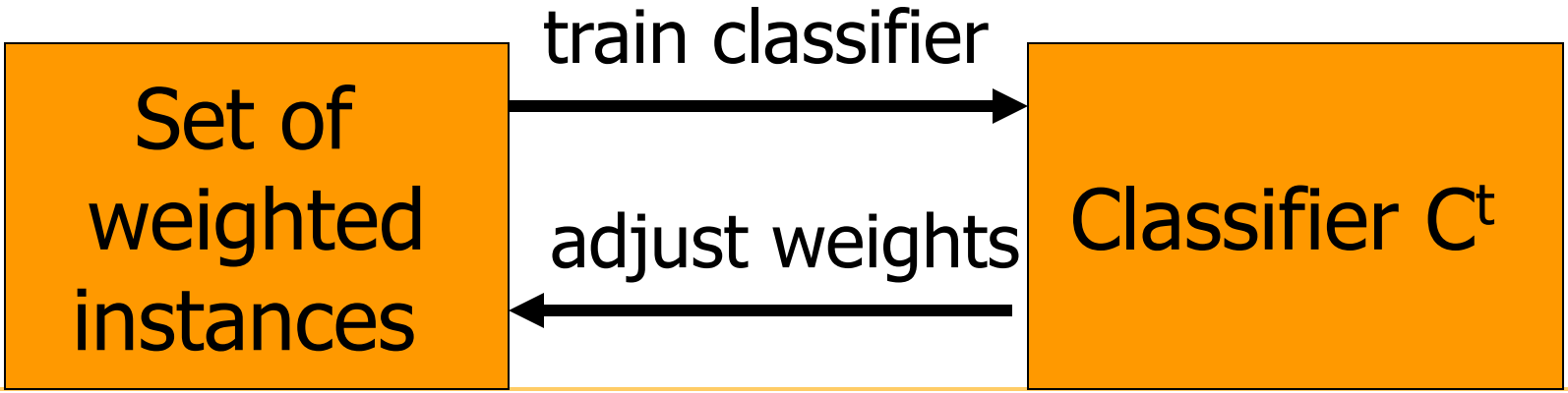


Boosting流程(loop3)



Boosting

- 过程:
  - 在一定的权重条件下训练数据，得出分类法Ct
  - 根据Ct的错误率调整权重



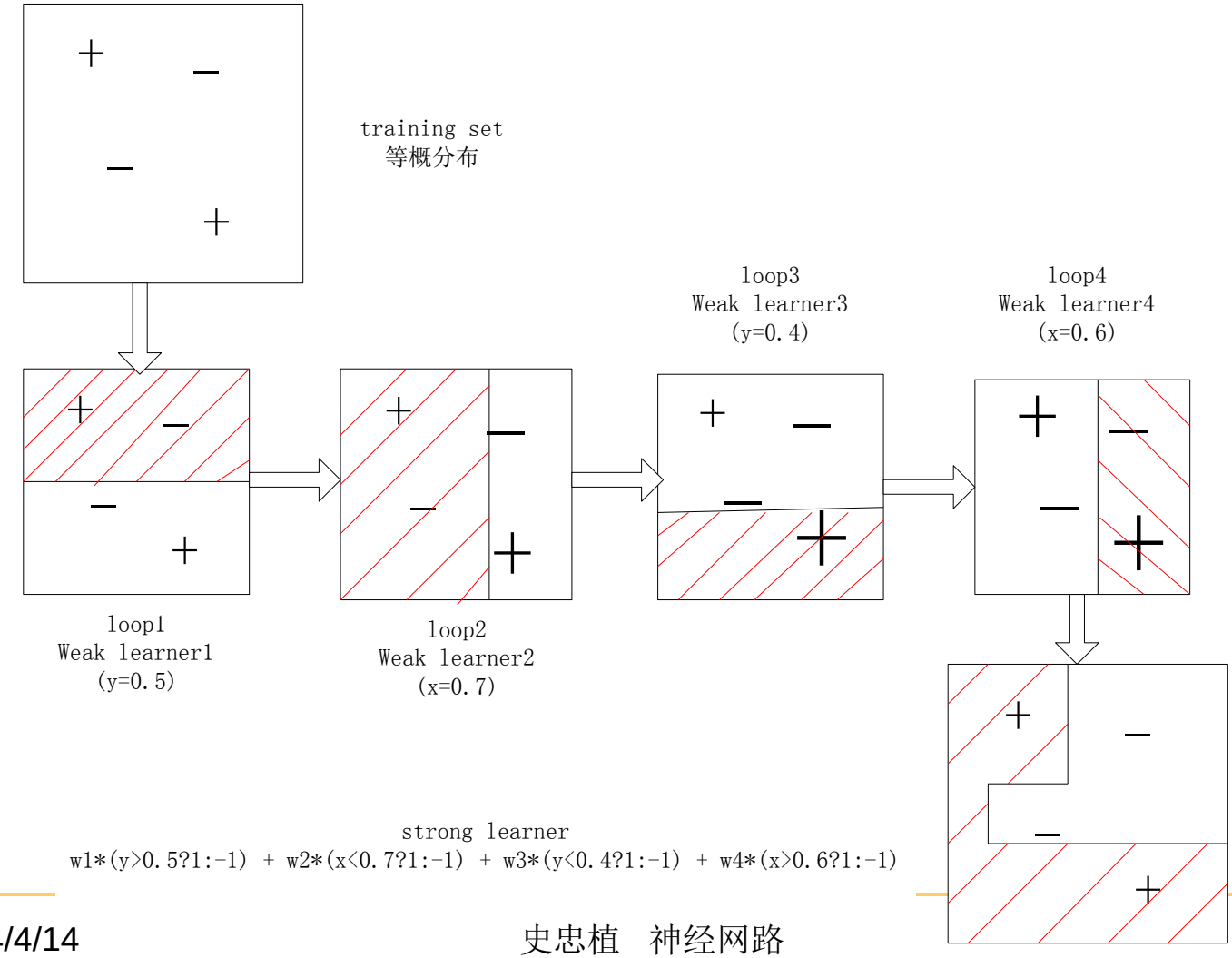
流程描述

- Step1: 原始训练集输入，带有原始分布
- Step2: 给出训练集中各样本的权重
- Step3: 将改变分布后的训练集输入已知的弱学习机，弱学习机对每个样本给出假设
- Step4: 对此次的弱学习机给出权重
- Step5: 转到Step2, 直到循环到达一定次数或者某度量标准符合要求
- Step6: 将弱学习机按其相应的权重加权组合形成强学习机

核心思想

- 样本的权重
  - 没有先验知识的情况下，初始的分布应为等概分布，也就是训练集如果有N个样本，每个样本的分布概率为1/N
  - 每次循环一后提高错误样本的分布概率，分错样本在训练集中所占权重增大， 使得下一次循环的弱学习机能够集中力量对这些错误样本进行判断。
- 弱学习机的权重
  - 准确率越高的弱学习机权重越高
- 循环控制：损失函数达到最小
  - 在强学习机的组合中增加一个加权的弱学习机，使准确率提高，损失函数值减小。

简单问题演示（Boosting训练过程）



算法—问题描述

- 训练集 $\{(x_1,y_1),(x_2,y_2),\cdots,(x_N,y_N)\}$
- $x_i \in R^m$，$y_i \in \{-1,+1\}$
- D_t 为第t次循环时的训练样本分布（每个样本在训练集中所占的概率，D_t总和应该为1）
- $h_t:X\rightarrow\{-1,+1\}$ 为第t次循环时的Weak learner，对每个样本给出相应的假设，应该满足强于随机猜测：

$$P_{(x,y)\in D_t}[y=h_t(x)]>\frac{1}{2}+\varepsilon$$

- w_t为h_t的权重
- $H_t(X_i)=sign(\sum_{i=1}^{t} w_i\cdot h_t(X_i))$为t次循环得到的Strong learner

算法——样本权重

- 思想：提高分错样本的权重

- $y_i \cdot H_t(\mathrm{X}_i)$ 反映了strong learner对样本的假设是否正确

$$y_i \cdot H_t(\mathrm{X}_i) \begin{cases} > 0 & right \\ < 0 & wrong \end{cases}$$

- 采用什么样的函数形式？

$$\exp\left(-y_i \cdot H_t(\mathrm{X}_i)\right)$$

算法—弱学习机权重

- 思想：错误率越低，该学习机的权重应该越大
- $\varepsilon_t = P_{(x,y)\in D_t}[y \neq h_t(x)]$ 为学习机的错误概率
- 采用什么样的函数形式？

和指数函数遥相呼应：

$$w_t = \frac{1}{2}\ln\left(\frac{1-\varepsilon_t}{\varepsilon_t}\right)$$

Boosting算法问题

- 如何调整训练集，使得在训练集上训练的弱分类器得以进行；
- 如何将训练得到的各个弱分类器联合起来形成强分类器。

Adaboost 算法

针对以上两个问题，AdaBoost算法进行了调整：

- 使用加权后选取的训练数据代替随机选取的训练样本，这样将训练的焦点集中在比较难分的训练数据样本上；
- 将弱分类器联合起来，使用加权的投票机制代替平均投票机制。让分类效果好的弱分类器具有较大的权重，而分类效果差的分类器具有较小的权重。

Adaboost 算法

与Boosting算法不同的是，AdaBoost算法不需要预先知道弱学习算法学习正确率的下限即弱分类器的误差，并且最后得到的强分类器的分类精度依赖于所有弱分类器的分类精度，这样可以深入挖掘弱分类器算法的能力。

Adaboost 算法

AdaBoost算法中不同的训练集是通过调整每个样本对应的权重来实现的。开始时，每个样本对应的权重是相同的，即其中n为样本个数，在此样本分布下训练出一弱分类器。对于分类错误的样本，加大其对应的权重；而对于分类正确的样本，降低其权重，这样分错的样本就被突显出来，从而得到一个新的样本分布。在新的样本分布下，再次对样本进行训练，得到弱分类器。依次类推，经过T次循环，得到T个弱分类器，把这T个弱分类器按一定的权重叠加（boost）起来，得到最终想要的强分类器。

Adaboost 算法步骤

AdaBoost算法的具体步骤如下：

1. 给定训练样本集S，其中X和Y分别对应于正例样本和负例样本；T为训练的最大循环次数；
2. 初始化样本权重为1/n ，即为训练样本的初始概率分布；
3. 第一次迭代：(1)训练样本的概率分布相当，训练弱分类器;(2)计算弱分类器的错误率;(3)选取合适阈值，使得误差最小；(4)更新样本权重；

经T次循环后，得到T个弱分类器，按更新的权重叠加，最终得到的强分类器。

Adaboost 算法步骤

Adaboost算法是经过调整的Boosting算法，其能够对弱学习得到的弱分类器的错误进行适应性(Adaptive)调整。上述算法中迭代了T次的主循环，每一次循环根据当前的权重分布对样本x定一个分布P，然后对这个分布下的样本使用弱学习算法得到一个弱分类器，对于这个算法定义的弱学习算法，对所有的样本都有错误率，而这个错误率的上限并不需要事先知道，实际上。每一次迭代，都要对权重进行更新。更新的规则是：减小弱分类器分类效果较好的数据的概率，增大弱分类器分类效果较差的数据的概率。最终的分类器是个弱分类器的加权平均

算法--Adaboost

Given : $(x_1, y_1), \cdots, (x_N, y_N)$ where $x_i \in \mathrm{X}, y_i \in \{-1, +1\}$

Initialize $D_1(i) = \frac{1}{N}$

For $t = 1, \cdots T$:

- Train weak learner using D_t
- Get weak learner $h_t : \mathrm{X} \longrightarrow R$
- Compute w_t
- Update $D_{t+1}(i) = \frac{D_t(i)\exp(-y_i w_t h_t(x_i))}{Z_t}$

Z_t is a normalization factor

Output thefinal classifier(strong learner) :

$$H(x) = sign(\sum_{t}^{T} w_t h_t(x))$$

AdaBoost.M1

- 初始赋予每个样本相等的权重*1/N*；
- *For t = 1, 2, ..., T Do*
  - 学习得到分类法C_t；
  - 计算该分类法的错误率E_t

    E_t=所有被错误分类的样本的权重和；
  - $\beta_t = E_t/(1 - E_t)$
  - 根据错误率更新样本的权重；

    正确分类的样本：$W_{new} = W_{old} * \beta_t$

    错误分类的样本：$W_{new} = W_{old}$
  - 调整使得权重和为1；
- 每个分类法C_t的投票价值为$\log [1 / \beta_t]$

AdaBoost 训练误差

- 将$\gamma_t=1/2-E_t$；
- Freund and Schapire 证明:
  最大错误率为:

$$\prod\left[2\sqrt{\varepsilon_t(1-\varepsilon_t)}\right]=\prod\sqrt{1-4\gamma_t^{\,2}}$$

$$\leq \exp(-2)\sum\gamma_t^{\,2}$$

- 即训练错误率随γ_t的增大呈指数级的减小.

AdaBoost 泛化误差 (1)

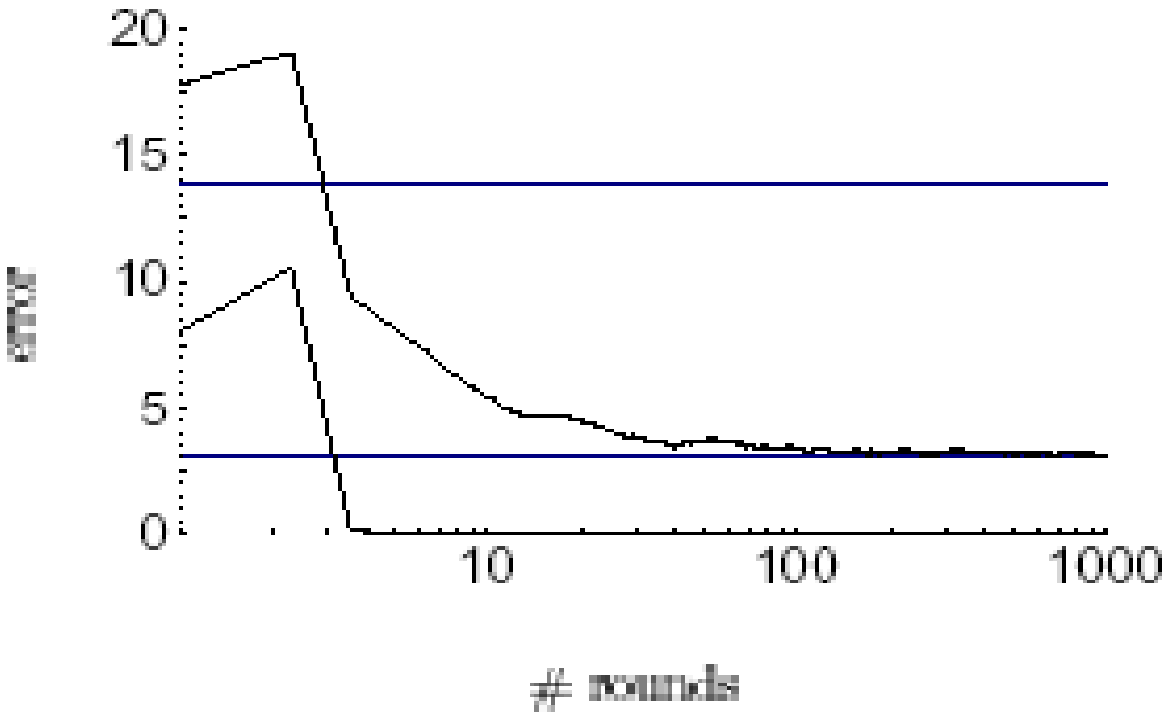
- 最大总误差:

$$\hat{p}_r(H(x) \neq y) + \hat{O}(\sqrt{\frac{Td}{m}})$$

  - m： 样本个数
  - d： VC维
  - T : 训练轮数
  - Pr: 对训练集的经验概率
- 如果T值太大,Boosting会导致过适应（overfit）

AdaBoost泛化误差 (2)

- 许多的试验表明:
  Boosting不会导致overfit



AdaBoost泛化误差(3)

- 解释以上试验现象;
- 样本(X,Y)的margin:

$$\text{margin}(x,y)= y\sum_t \alpha_t h_t(x)$$

  - $\alpha_t=1/2 \ln ((1- E_t)/ E_t)$
  - 较大的正边界表示可信度高的正确的预测
  - 较大的负边界表示可信度高的错误的预测

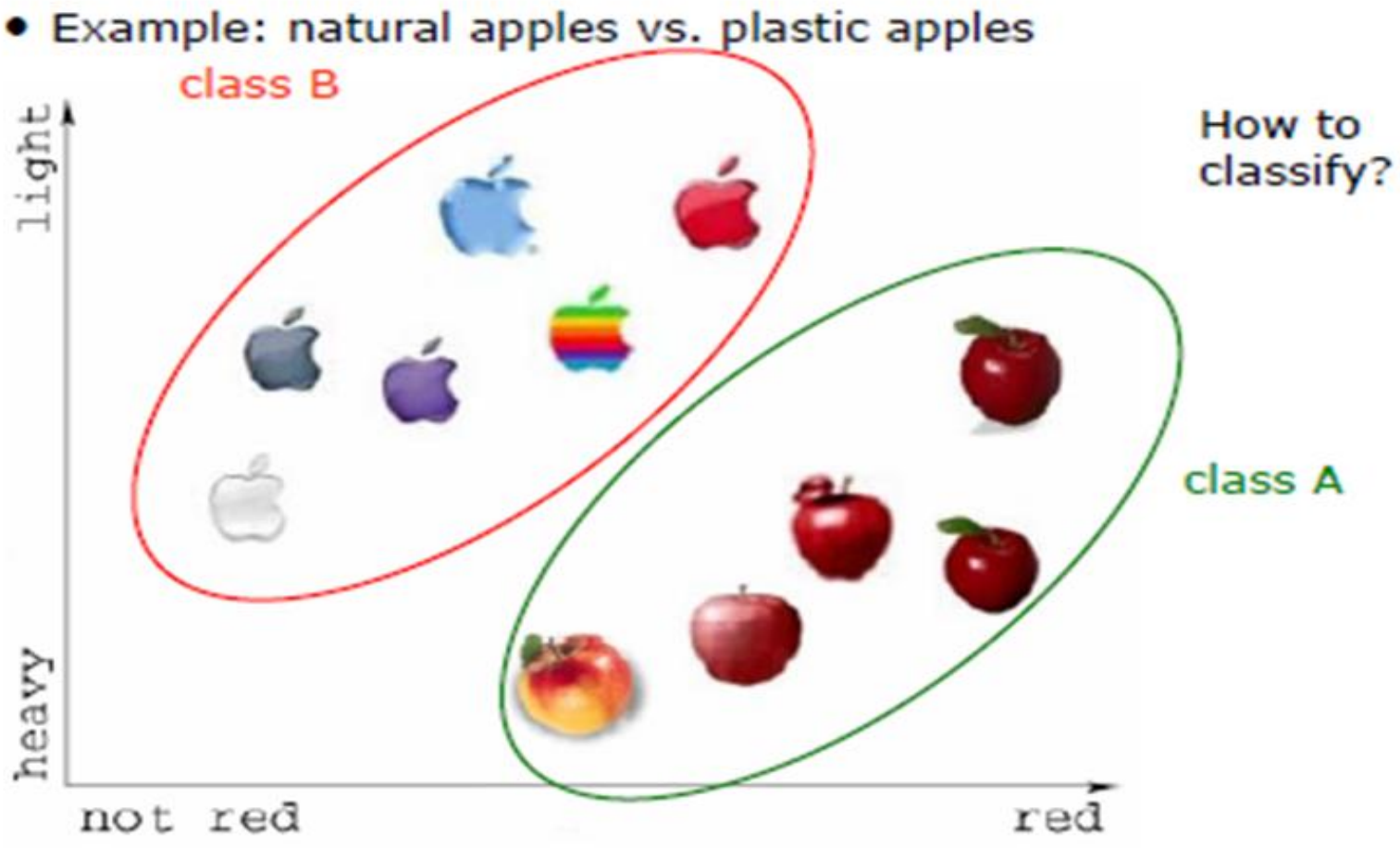
AdaBoost泛化误差 (4)

- 解释: 当训练误差降低后,Boosting继续提高边界,从而增大了最小边界,使分类的可靠性增加,降低总误差.
- 总误差的上界:

$$\hat{P}_r(margin(x,y) \le \theta) + \hat{O}\left(\sqrt{\frac{d}{m\theta^2}}\right)$$

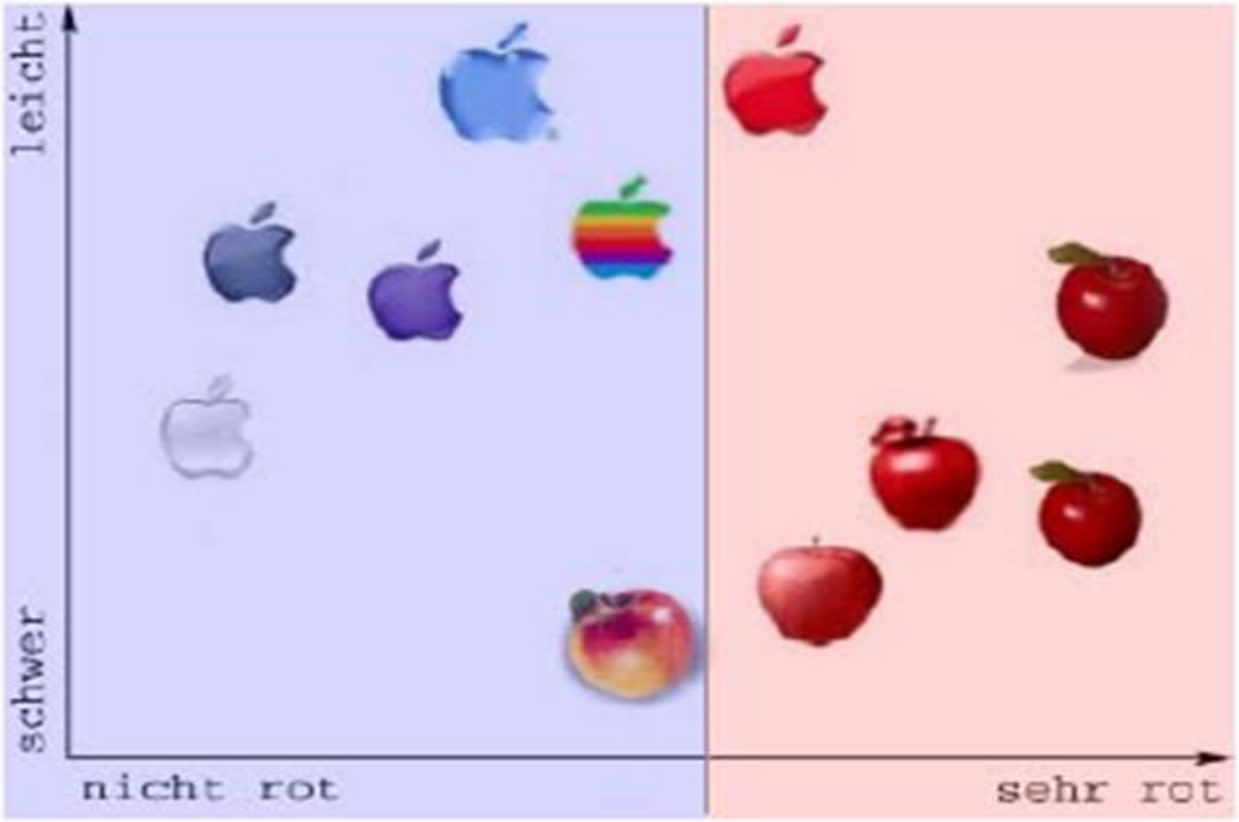
- 该公式与T无关

Adaboost 应用实例



Adaboost 应用实例

- Example: natural apples vs. plastic apples

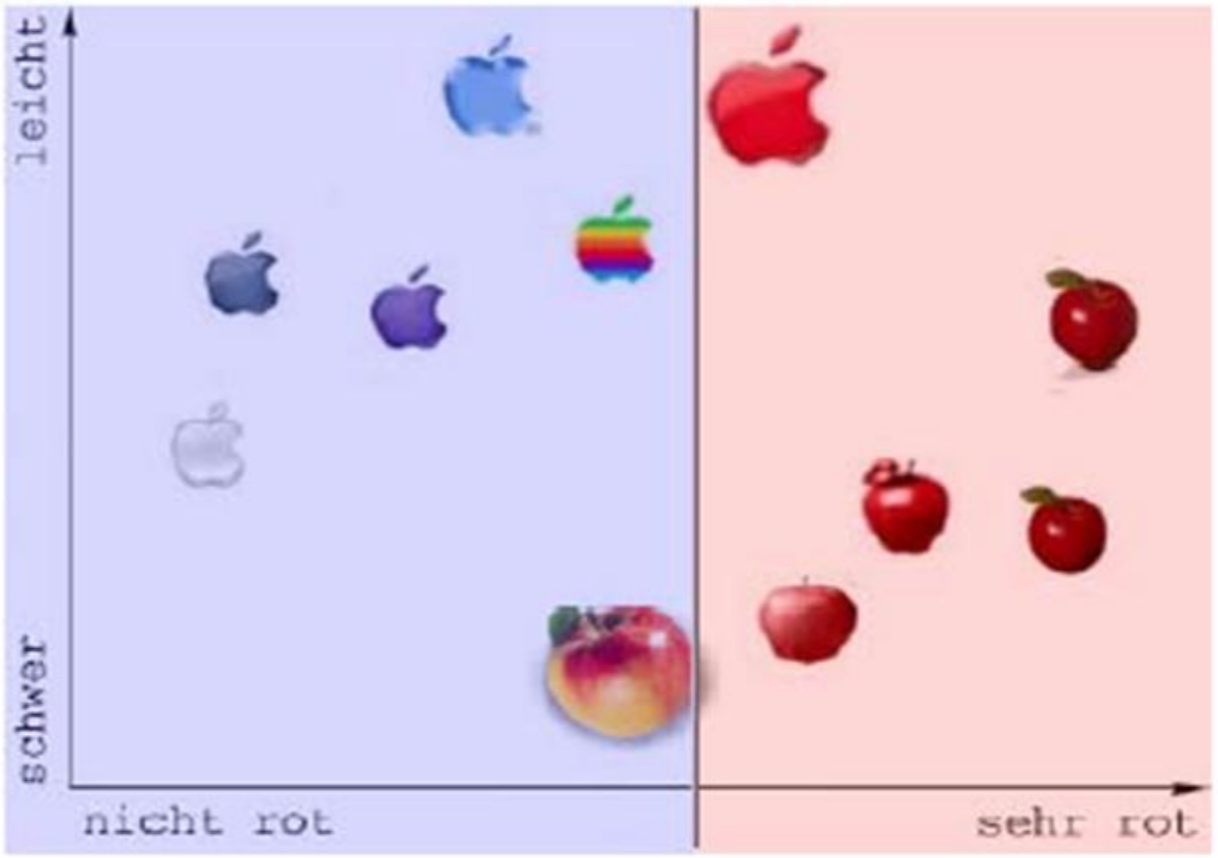


1st hypothesis

Weak classifier (cuts on coordinate axes)

Adaboost 应用实例

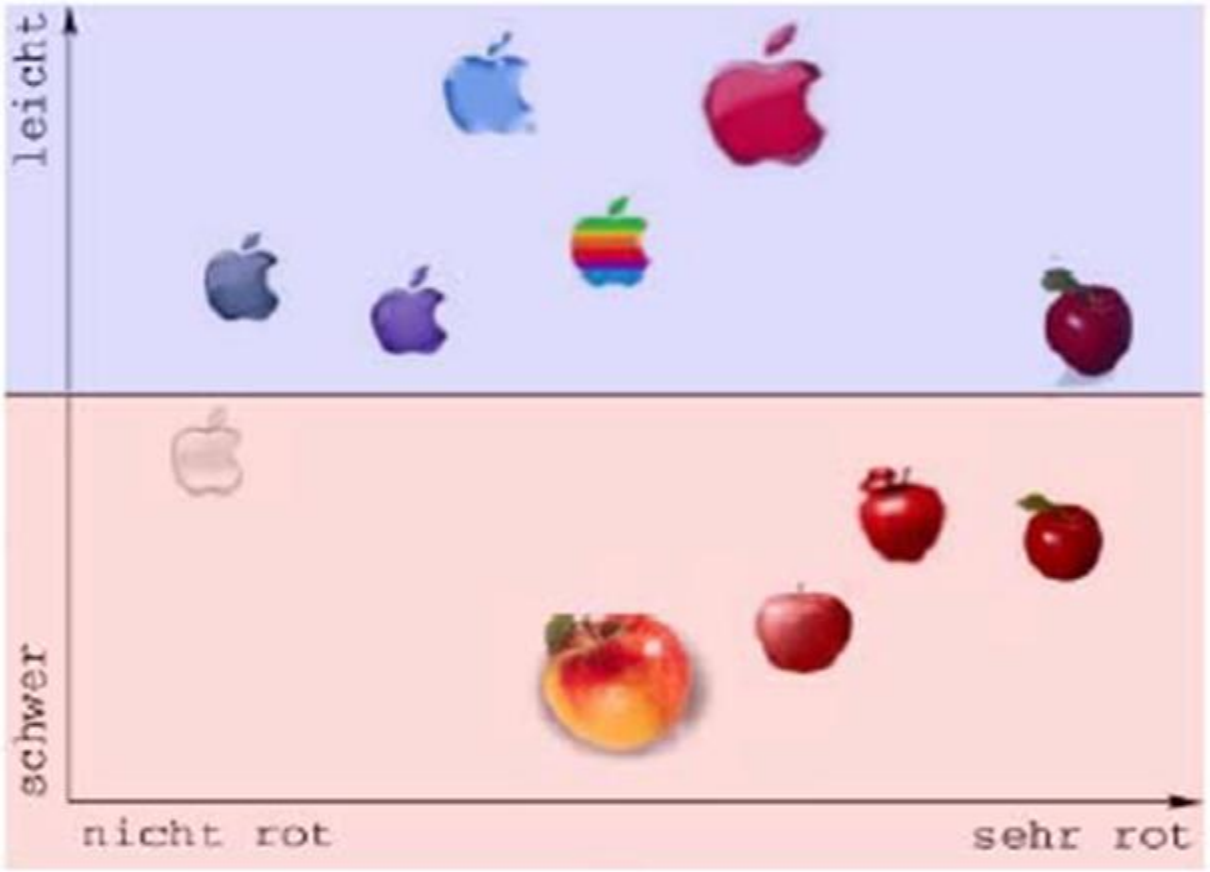
- Example:



Recomputing weightings of the training patterns

Adaboost 应用实例

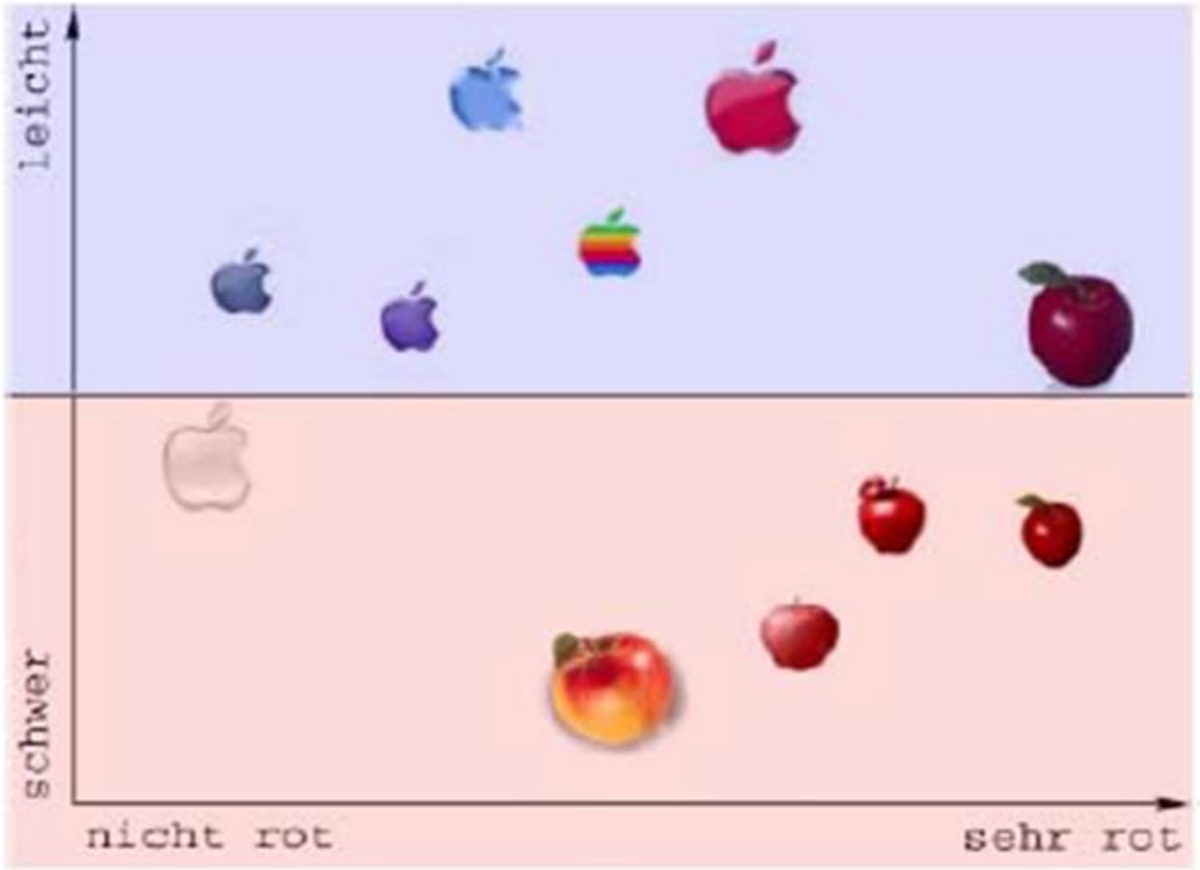
- Example:



2nd hypothesis

Adaboost 应用实例

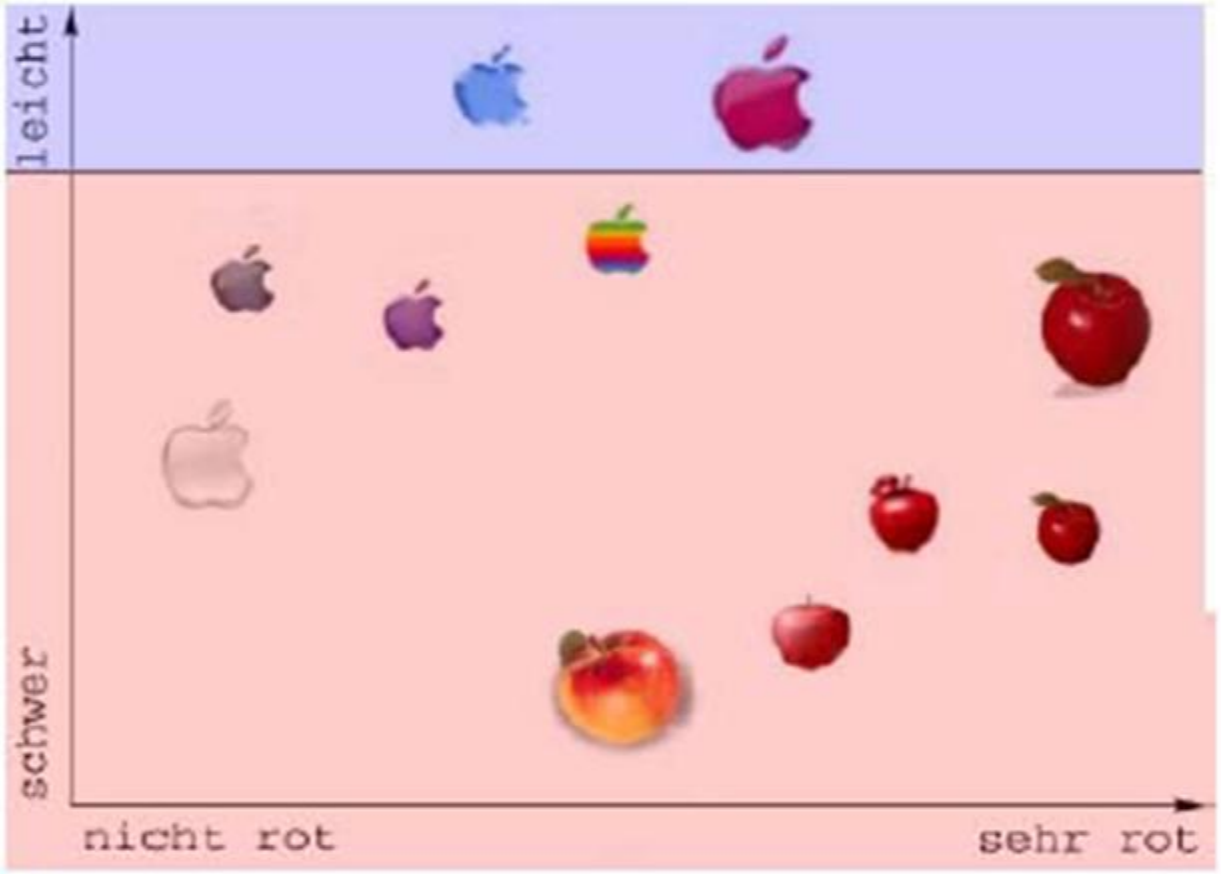
- Example:



Recompute weighting

Adaboost 应用实例

- Example:



3rd hypothesis

Adaboost 应用实例

- Example:

Recompute weighting

4th hypothesis

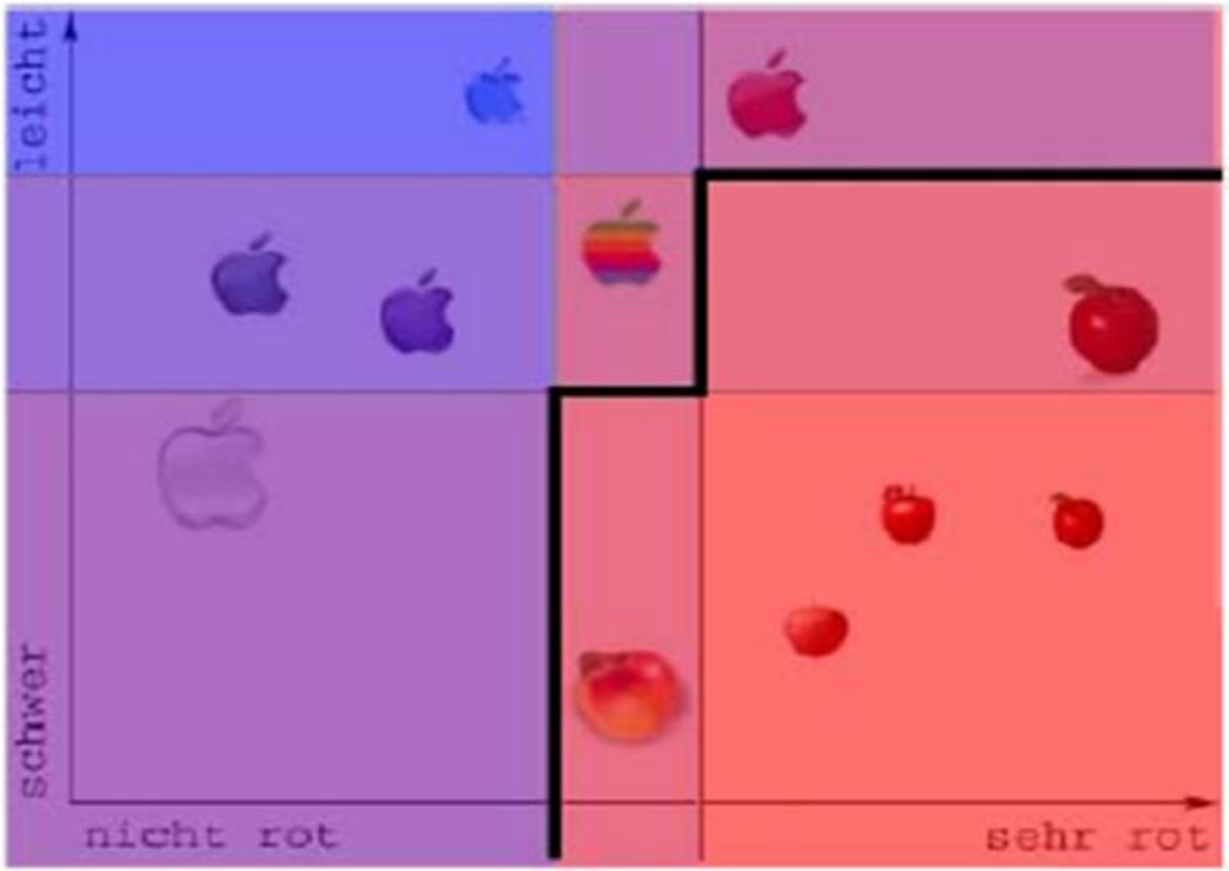
Adaboost 应用实例

- Example:

Combination of hypotheses

Adaboost 应用实例

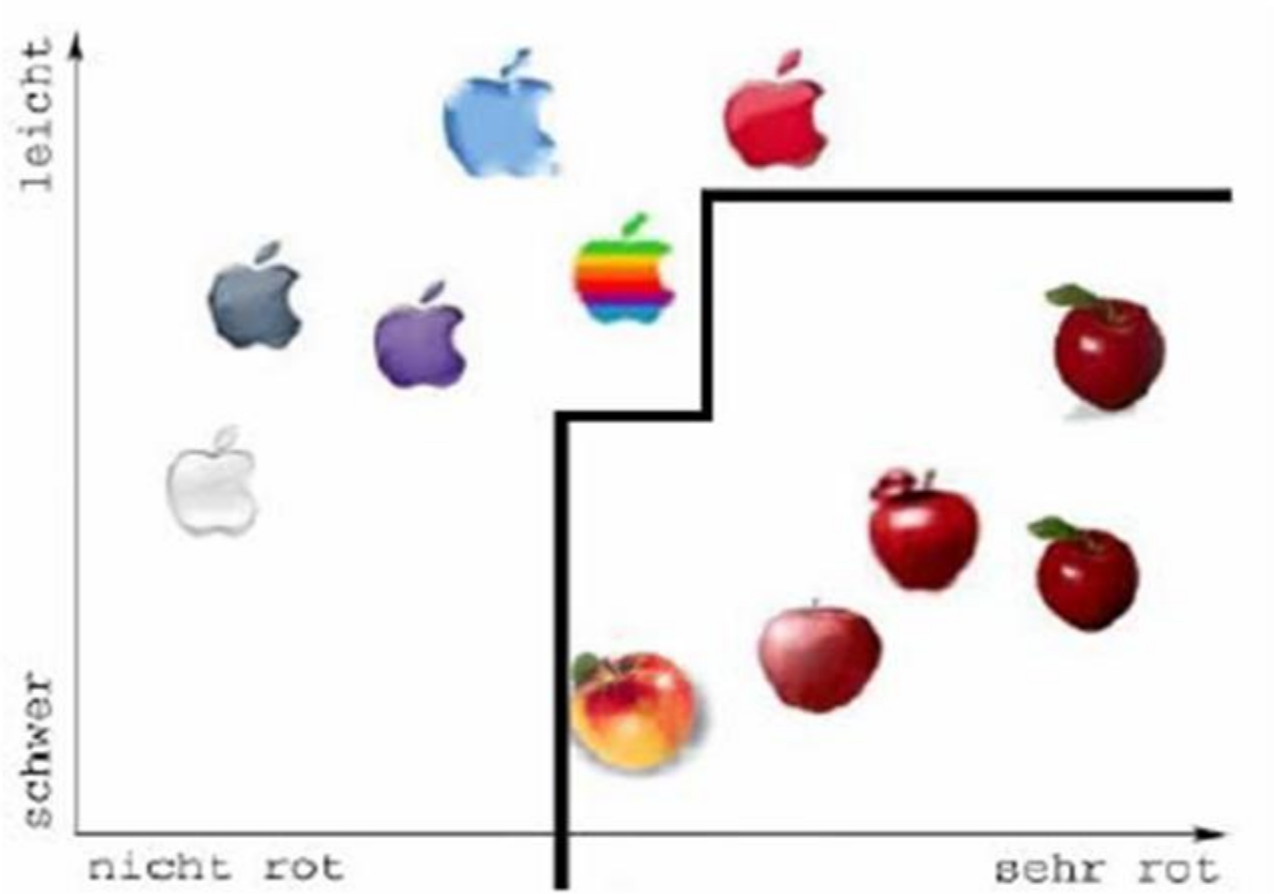
- Example:



Decision surface

Adaboost 应用实例

- Example



构建贝叶斯网络

$$B =< N, A, \Theta >$$

是表示变量间连结关系的有向无环图

- 贝叶斯网络的学习

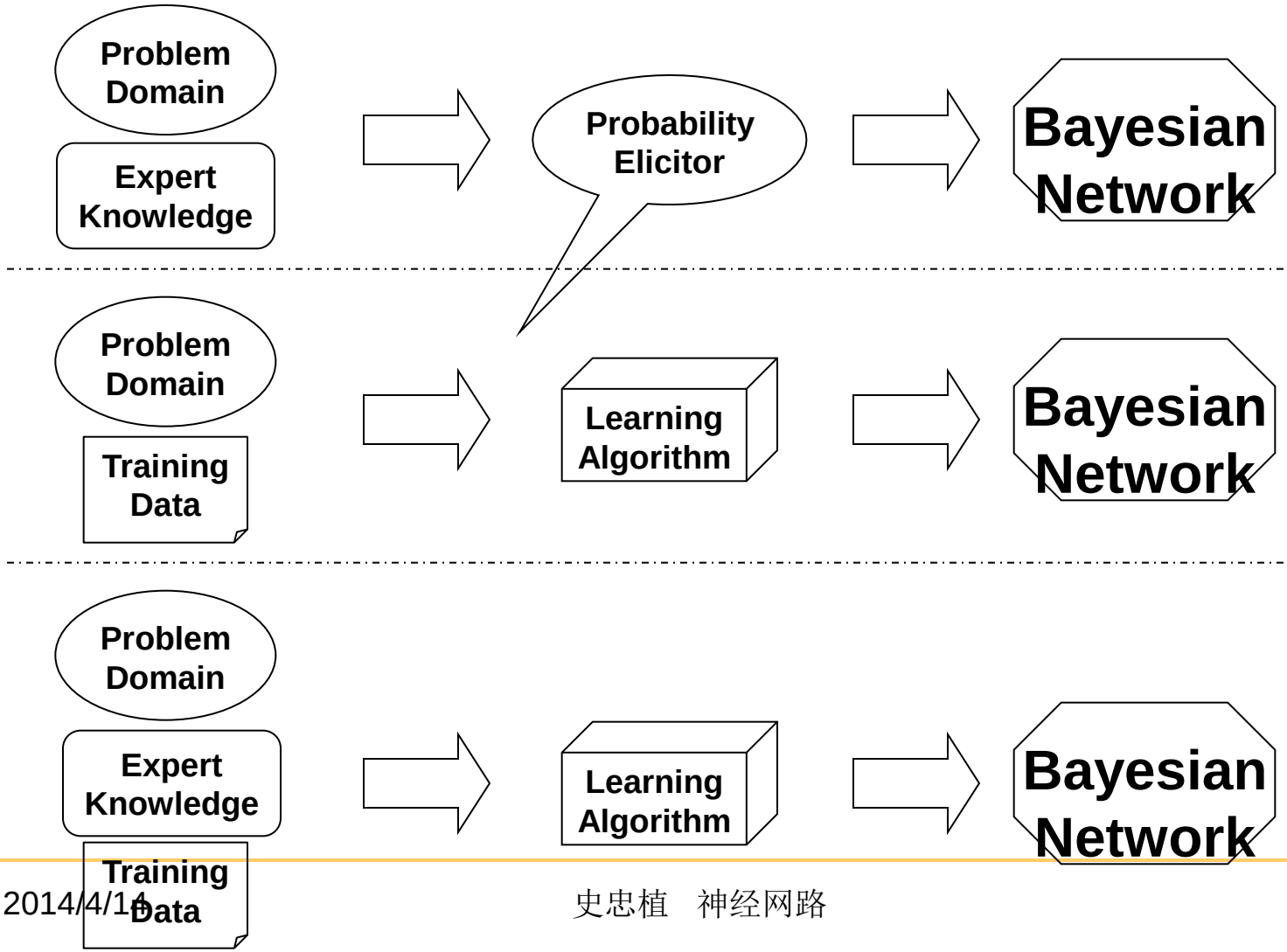
基于评分函数的结构学习

结构学习

基于条件独立性检验的结构学习

参数学习

构建贝叶斯网络


Problem Domain
Expert Knowledge
Probability Elicitor
Bayesian Network
Problem Domain
Training Data
Learning Algorithm
Bayesian Network
Problem Domain
Expert Knowledge
Training Data
Learning Algorithm
Bayesian Network

贝叶斯概率(密度估计)

贝叶斯学习理论利用先验信息和样本数据来获得对未知样本的估计，而概率（联合概率和条件概率）是先验信息和样本数据信息在贝叶斯学习理论中的表现形式。如何获得这些概率（也称之为密度估计）是贝叶斯学习理论争议较多的地方。研究如何根据样本的数据信息和人类专家的先验知识获得对未知变量（向量）的分布及其参数的估计。它有两个过程：一是确定未知变量的先验分布；二是获得相应分布的参数估计。如果以前对所有信息一无所知，称这种分布为无信息先验分布；如果知道其分布求它的分布参数，称之为有信息先验分布。

密度估计

- 先验分布的选取原则
  - ✓共轭分布
  - ✓杰弗莱原则
  - ✓最大熵原则

从数据中学习

- 共轭分布族
  - 先验与后验属于同一分布族
  - 预先给定一个似然分布形式
- 对于变量定义在0-1之间的概率分布，存在一个离散的样本空间
  - Beta 对应着 2 个似然状态
  - 多变量 Dirichlet 分布对应 2个以上的状态

共轭分布

Raiffa和Schaifeer提出先验分布应选取共轭分布，即要求后验分布与先验分布属于同一分布类型。它的一般描述为：

设样本$X_1, X_2, \cdots, X_n$对参数θ的条件分布为$p(x_1, x_2, \cdots, x_n \mid \theta)$，如果先验分布密度函数 $\pi(\theta)$

决定的后验密度 $\pi(\theta \mid x)$ 与 $\pi(\theta)$

同属于一种类型，则称 $\pi(\theta)$

为$p(x \mid \theta)$的共轭分布。

杰弗莱原则

- 杰弗莱对于先验分布的选取做出了重大的贡献，它提出一个不变原理，较好地解决了贝叶斯假设中的一个矛盾，并且给出了一个寻求先验密度的方法。杰弗莱原则由两个部分组成：一是对先验分布有一合理要求；一是给出具体的方法求得适合于要求的先验分布。先验分布的选取原则

最大熵原则

熵是信息论中描述事物不确定性的程度的一个概念。
如果一个随机变量只取与两个不同的值，比较下面两种情况：

（1） $p(x=a)=0.98$ $p(x=a)=0.02$

（2） $p(x=a)=0.45$ $p(x=a)=0.55$

很明显，（1）的不确定性要比（2）的不确定性小得多，而且从直觉上也可以看得出当取的两个值得概率相等时，不确定性达到最大。

最大熵原则

设随机变量x是离散的，它取 $a_1, a_2 \cdots a_k \cdots$

至多可列个值，且 $p(x = a_i) = p_i \quad i = 1,2,\cdots$

则

$$H(x) = -\sum_i p_i \ln p_{i\,p(x)}$$

称为x的熵

对连续型随机变量x，它的概率密度函数为p(x)，若积分

$$H(x) = -\int p(x) \ln p(x)$$

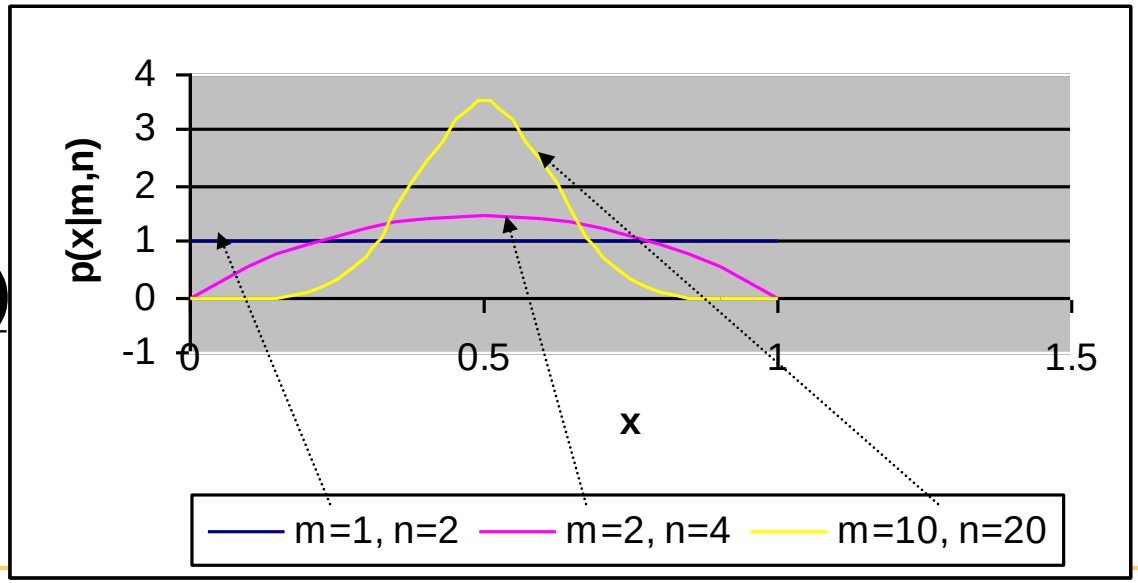
有意义，称它为连续型随机变量的熵

先验分布的选取－beta分布

$$p_{Beta}(x|m,n) = \frac{\Gamma(n)}{\Gamma(m)\Gamma(n-m)} x^{m-1}(1-x)^{n-m-1}$$

$$mean = \frac{m}{n}$$

$$variance = \frac{m(1-m/n)}{n(n+1)}$$



先验分布的选取－多项Dirichlet分布

$$p_{\text{Dirichlet}}(x \mid m_1, m_2, \ldots, m_N) = \frac{\Gamma(\sum_{i=1}^{N} m_i)}{\Gamma(m_1)\Gamma(m_2)\ldots\Gamma(m_N)} x^{m_1-1} x^{m_2-1} \ldots x^{m_N-1}$$

$$\text{mean of the } i^{\text{th}} \text{ state} = \frac{m_i}{\sum_{i=1}^{N} m_i}$$

$$\text{variance of the } i^{\text{th}} \text{ state} = \frac{m_i(1 - m_i / \sum_{i=1}^{N} m_i)}{\sum_{i=1}^{N} m_i (\sum_{i=1}^{N} m_i + 1)}$$

不完全数据的密度估计

- ✓期望最大化方法（Expectation Maximization EM）
- ✓Gibbs抽样（Gibbs Sampling GS）
- ✓Bound and Collapse (BC)

期望最大化方法

分为以下几个步骤：

（1）含有不完全数据的样本的缺项用该项的最大似然估计代替；

（2）把第一步中的缺项值作为先验信息，计算每一缺项的最大后验概率，并根据最大后验概率计算它的理想值。

（3）用理想值替换（1）中的缺项。

（4）重复（1—3），直到两次相继估计的差在某一固定阈值内。

Gibbs抽样

- Gibbs抽样（Gibbs Sampling GS）

  GS是最为流行的马尔科夫、蒙特卡罗方法之一。GS把含有不完全数据样本的每一缺项当作待估参数，通过对未知参数后验分布的一系列随机抽样过程，计算参数的后验均值的经验估计。

贝叶斯网络的结构学习

- 基于搜索评分的方法:
  - 初始化贝叶斯网络为孤立节点
  - 使用启发式方法为网络加边
  - 使用评分函数评测新的结构是否为更好
    - ✓ 贝叶斯评分（Bayesian Score Metric）
    - ✓ 基于熵的评分
    - ✓ 最小描述长度MDL(Minimal Description Length)
  - 重复这个过程，直到找不到更好的结构
- 基于依赖分析的方法:
  - 通过使用条件独立性检验conditional independence (CI)找到网络的依赖结构

基于MDL的贝叶斯网结构学习

- 计算每一点对之间的互信息：

$$I_P(X;Y) = \sum_{x,y} P(x,y) log \frac{p(x,y)}{p(x)p(y)}$$

- 建立完全的无向图，图中的顶点是变量，边是变量之间的互信息
- 建立最大权张成树
- 根据一定的节点序关系，设置边的方向

基于条件独立性的贝叶斯网络学习

- 假定：节点序已知
- 第一阶段 (Drafting)
  - 计算每对节点间的互信息，建立完整的无向图.
- 第二阶段 (Thickening)
  - 如果接点对不可能d-可分的话，把这一点对加入到边集中。
- 第三阶段 (Thinning)
  - 检查边集中的每个点对，如果两个节点是d-可分的，那么移走这条边。

基于条件独立性检验(CI)的贝叶斯网络结构学习

1）初始化图结构B=<N,A,Θ>,A=Φ,R=Φ,S=Φ;
2）对每一节点对，计算它们的互信息，并将互信息大于某一域值的节点对按互信息值的大小顺序加入到S中；
3）从S中取出第一个点对，并从S中删除这个元素，把该点对加入到边集A中；
4） 从S中剩余的点对中，取出第一个点对，如果这两各界点之间不存在开放路径，再把该点对加入A到中，否则加入到R中；
5）重复4),直到S为空；
6）从R中取出第一个点对；
7）找出该点对的某一块集，在该子集上做独立性检验，如果该点对的两个节点，仍然相互依赖，则加入到A中；
8） 重复6),直到R为空；
9） 对A中的每一条边，如果除这条边外，仍旧含有开放路径，则从A中临时移出，并在相应的块集上作独立

树增广的朴素贝叶斯网TAN的结构学习

1） 计算各属性节点间的条件互信息;
2） 以属性变量为节点，以条件互信息为节点之间的连接权，构造无向完全图;
3） 生成最大权张成树;
4） 转换无向的最大权张成树为有向树;
5） 从类别变量向各属性节点引一条有向变，生成 TAN 模型。

$$I(A_i, A_j \mid C) = \sum_{A_i, A_j, C} P(A_i; A_j \mid C) \log \frac{P(A_i; A_j \mid C)}{P(A_i \mid C) P(A_j \mid C)}$$

主动贝叶斯网络分类器

- 主动学习:

  主动在候选样本集中选择测试例子，并将这些实例以一定的方式加入到训练集中。

- 选择策略

  抽样选择
  - 随机抽样
  - 相关抽样
  - 不确定性抽样

  投票选择

主动贝叶斯网络分类器

•学习过程

输入：带有类别标注的样本集L，未带类别标注的候选样本集UL,选择停止标准e，每次从候选集中选择的样本个数M

输出：分类器C.

过程：

```
While not e
{
  TrainClassifer(L,C) //从L中学习分类器C;
  For each  x计算ES;
  SelectExampleByES(S,UL,M,ES)  //根据ES从UL中选择M个例子的子集S.
  LabeledAndAdd(S,L); //用当前的分类器C标注S中的元素，并把它加入到L中。
  Remove(S,UL);  //从UL中移走S.
```

主动贝叶斯网络分类器

- 基于最大最小熵的主动学习

  首先从测试样本中选择出类条件熵最大和最小的候选样本（MinExample, MaxExample），然后将这两个样本同时加入到训练集中。类条件熵最大的样本的加入，使得分类器能够对具有特殊信息的样本的及早重视；而类条件熵最小的样本是分类器较为确定的样本，对它的分类也更加准确，从而部分地抑制了由于不确定性样本的加入而产生的误差传播问题

$$H(C \mid x) = -\sum_{i=1}^{|C|} p(c_i \mid x) \ln(p(c_i \mid x))$$

主动贝叶斯网络分类器

- 基于分类损失与不确定抽样相结合的主动学习

分类损失：

$$L_D = \int_x L(p(c \mid x), \hat{p}_D(c \mid x)\, p(x) dx$$

选择过程：

从测试样本中选择个熵较大的样本，组成集合maxS，然后对此集合中每个元素计算相对于该集合的分类损失和，选择分类损失和最小的样本做标注并加入到训练样本集中。

主动贝叶斯网络分类器

ALearnerByMaxMinEntropy测试结果

初始标注
样本数：96

未标注训练
样本数：500

	A	B	C	D	E	F	精度评定(%)	
							精度	召回率
A	**645**	6	5	0	0	0	0.7670	0.9832
B	140	**132**	0	0	0	0	0.9429	0.4853
C	25	2	**50**	0	0	0	0.8475	0.6494
D	5	0	2	**33**	1	0	0.9167	0.8049
E	9	0	0	3	**51**	0	0.9623	0.8095
F	17	0	2	0	1	**64**	1.0000	0.7619

ALearnerByUSandCL测试结果

测试集
样本数：1193

	A	B	C	D	E	F	精度评定(%)	
							精度	召回率
A	**641**	11	4	0	0	0	0.8412	0.9771
B	81	**191**	0	0	0	0	0.8565	0.7022
C	8	21	**48**		0	0	0.8571	0.6234
D	6	0	2	**32**	1	0	0.9143	0.7273
E	9	0	0	3	**51**	0	0.9623	0.8095
F	17	0	2	0	1	**64**	1.0000	0.7619

Thank You

Intelligence Science

http://www.intsci.ac.cn/