人工智能技术
Artificial Intelligence

——人工智能：经典智能+智能计算+智能学习
AI: Classical Intelligence + Computing Intelligence + Learning Intelligence

王鸿鹏，王润花，韩明静，
许丽，靖智博
南开大学人工智能学院
hanmj@mail.nankai.edu.cn

概率模型学习
PROBABILISTIC MODEL LEARNING

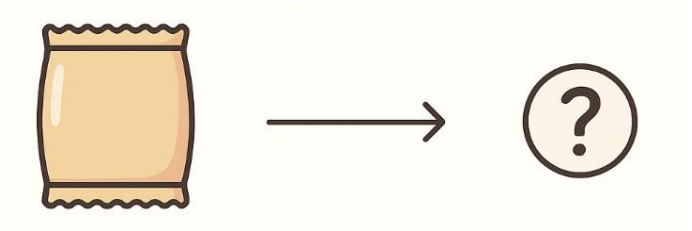
将学习视为一种从观测中进行不确定的推理的形式，
并设计模型来表示不确定的世界

统计学习

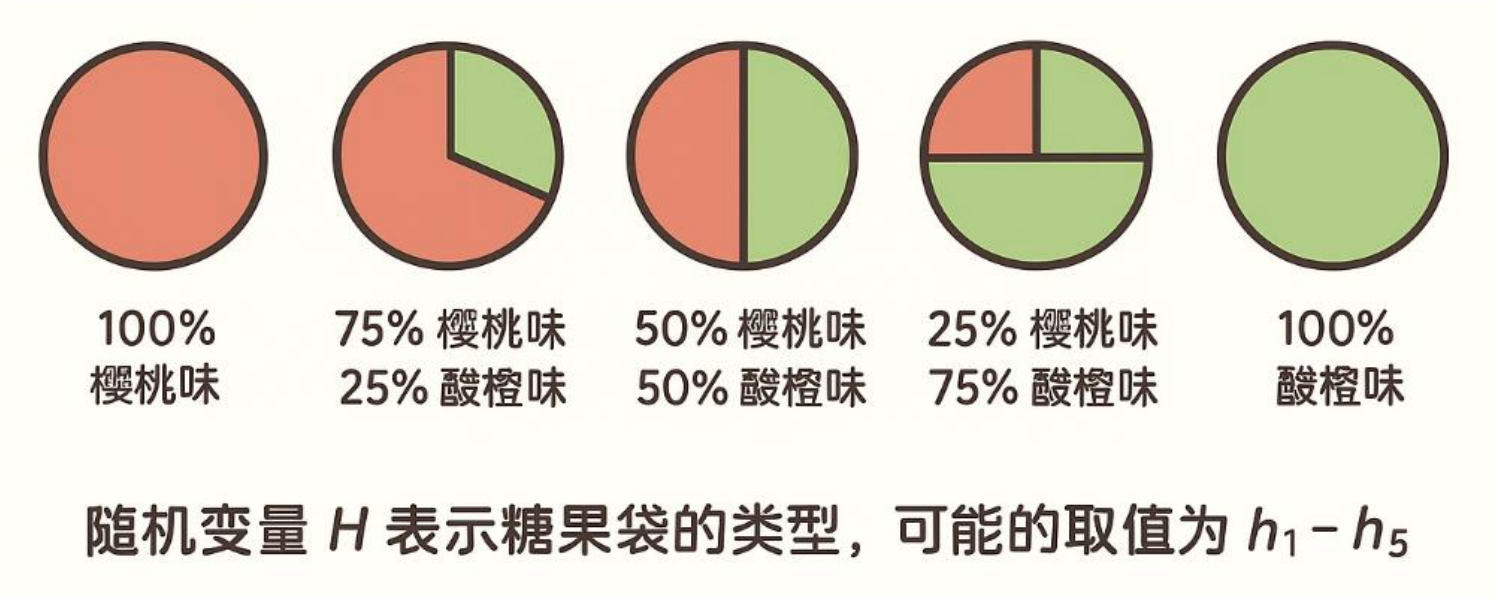
核心概念：数据 + 假设

数据可以看作证据——描述相关领域的一部分随机变量或所有随机变量的实例

假设是关于相关领域如何运作的一些概率理论，逻辑理论是其中的一个特例



糖果袋的 5 种可能类型(假设空间)



任务：预测下一块糖果的口味

统计学习

贝叶斯学习（Bayesian learning）基于给定的数据计算每个假设发生的概率并在此基础上进行预测。

预测是通过对所有假设按概率加权求和所得的，而不是仅仅使用了单个“最佳”假设。**学习就可以归约为概率推断**

- 通过贝叶斯法则得到每个假设的概率：

$$P(h_i \mid \boldsymbol{d}) = \alpha P(\boldsymbol{d} \mid h_i) P(h_i)$$

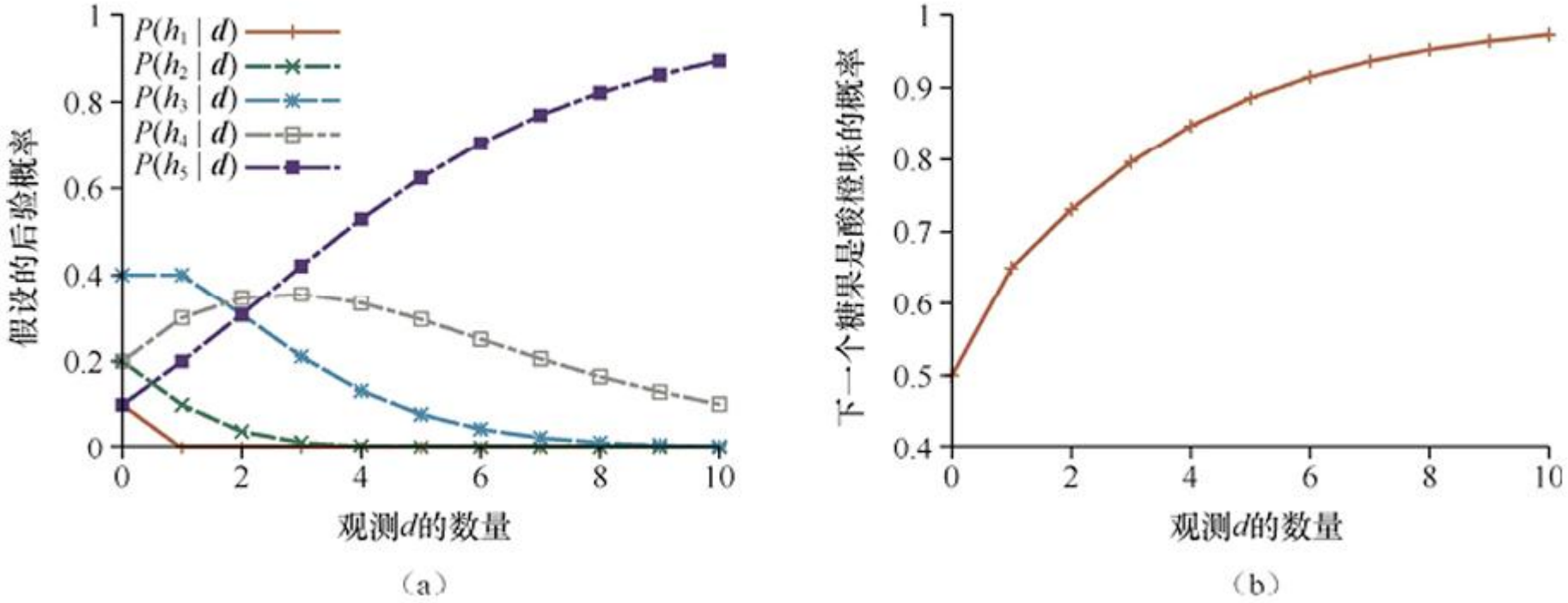
假设先验：P(hi)　　每个假设下数据的似然：$P(\boldsymbol{d} \mid h_i)$

- 对一个未知量X做出预测：

$$\boldsymbol{P}(X \mid \boldsymbol{d}) = \sum_i \boldsymbol{P}(X \mid h_i) P(h_i \mid \boldsymbol{d})$$

其中每一个假设都参与决定了X的分布。这个式子说明预测是通过对每个假设的预测进行加权平均得到的

统计学习



（a）后验概率 $P(h_i \mid d_1, \cdots, d_N)$ 。观测数量N为1～10，且每一个观测都是酸橙味的糖果。
（b）贝叶斯预测 $P(D_{N+1} = \text{lime} \mid d_1, \cdots, d_N)$

贝叶斯预测最终会与真实的假设吻合。

统计学习

最大后验假设

- 对于真实的学习问题, 假设空间通常非常大或无限大，在大多数情况下，我们必须采用近似或简化的方法
- 最大后验(MAP) 假设: 基于单个 可能性最大的假设 h_{MAP} 进行预测，即使得 $P(h_i|\mathrm{d})$最大化的假设
- (例：在糖果例子中，在连续3次观测到酸橙糖之后有$h_{MAP} = h_5$，因此MAP学习器预测第四颗糖果是酸橙糖的概率为1.0，这比贝叶斯预测概率0.8更有风险)
- 即最大化 $P(\mathrm{d}|h_i)P(h_i)$ 或 $\log P(\mathrm{d}|h_i) + \log P(h_i)$
- 随着数据量越来越多，MAP预测和贝叶斯预测将变得越来越接近，因为与MAP假设竞争的其他假设的可能性越来越低。

统计学习

最大后验假设

- 利用假设先验知识P(h_i)约束假设的复杂性：越复杂的假设对应的先验概率越低，其中部分原因是它们数量太多了。但是，越复杂的假设拟合数据的能力越强。
- 假设的先验体现了假设的复杂性与其数据拟合程度之间的权衡

示例1：在逻辑函数的情况下，即H只包含确定性的假设（例如h1表示所有的糖果都是樱桃味）

- 如果假设h_i是一致的，$P(\boldsymbol{d} \mid h_i)$ 则为1，否则为0
- h_{MAP}将是与数据一致的最简单的逻辑理论。因此，最大后验学习自然体现了奥卡姆剃刀

示例2：对$P(\boldsymbol{d} \mid h_i)P(h_i)$取对数，则最大化的$h_{MAP}$等价于最小化：

$$-\log_2 P(\boldsymbol{d} \mid h_i) - \log_2 P(h_i)$$

- 从信息编码和概率之间的联系角度分析，可以看到$-\log_2 P(h_i)$等于说明假设h_i所需的位数，$-\log_2 P(\boldsymbol{d} \mid h_i)$是给定假设时说明数据所需的额外位数（为了更好理解，可以考虑，如果假设确切地预测了数据，就好像假设为h5和一连串出现的酸橙味糖果一样，那么此时不需要任何额外位数，则$\log_2 1 = 0$ 。）
- MAP学习所选择的是能最大程度压缩数据的假设。同样的任务可以通过称为最小描述长度（MDL）的学习方法更直接地阐述。MAP学习通过给更简单的假设赋予更高的概率来体现其简单性，而MDL则通过计算假设和数据在二进制编码中的位数来直接体现简单性。

统计学习

最大似然假设

- 假定假设空间具有均匀先验分布
- 在这种情况下，MAP学习被简化为选择一个使 $P(\boldsymbol{d} \mid h_i)$ 最大的h_i，即是所谓的最大似然（maximum-likelihood）假设，h_{ML}
- 当没有理由采用某个先验或倾向于某个假设（例如所有的假设都同样复杂）时，最大似然是一个合理的方法
- 当数据集很大时，假设的先验分布就不那么重要了，因为来自数据的证据足够强大，足以淹没假设的先验分布。这意味着在大数据集的情况下，最大似然学习是贝叶斯学习和MAP学习的一个很好的近似，但在小数据集上可能会出现问题

完全数据学习

任务设计

密度估计：假设要学习一个概率模型，给定数据是从该概率模型生成的，那么学习这个概率模型的一般性任务被称为密度估计。

- 密度估计是一种无监督学习
- 最简单的情形——即拥有完全数据的情形：当每个数据点包含所学习的概率模型的每个变量的值时，我们称数据是完全的
- 对于结构固定的概率模型，进行参数学习（parameter learning），即寻找其参数数值

完全数据学习

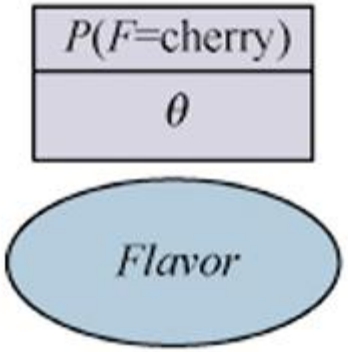
最大似然参数学习：离散模型

- 连续的假设集,参数记为θ，其对应的假设为h_θ

（θ表示樱桃味糖果所占的比例，其对应的假设为h_θ，此时酸橙味糖果所占的比例恰好为1- θ， 假设已经打开了 N 颗糖果，其中有 c 颗为樱桃味，l=N-c 颗为酸橙味。 ）

- 对于大型数据集，先验变得无关紧要。 假设所有的比例有相同的先验可能性，那么采用最大似然估计是合理的。

最大似然(Maximum likelihood ML)学习：选择使得 $P(\mathrm{d}|h_i)$最大的假设h_{ML}



$$P(\boldsymbol{d}\mid h_\theta)=\prod_{j=1}^{N}P(d_j\mid h_\theta)=\theta^c\cdot(1-\theta)^\ell$$

最大对数似然：

$$L(\boldsymbol{d}\mid h_\theta)=\log P(\boldsymbol{d}\mid h_\theta)=\sum_{j=1}^{N}\log P(d_j\mid h_\theta)=c\log\theta+\ell\log(1-\theta)$$

对L关于θ微分使微分结果为0：

$$\frac{\mathrm{d}L(\boldsymbol{d}\mid h_\theta)}{\mathrm{d}\theta}=\frac{c}{\theta}-\frac{\ell}{1-\theta}=0\quad\Rightarrow\quad\theta=\frac{c}{c+\ell}=\frac{c}{N}$$

最大似然假设: 糖果袋中樱桃口味的真实比例是到 目前为止所打开观测到的糖果中樱桃口味的占比

完全数据学习

最大似然参数学习：离散模型

- 最大似然参数学习的标准方法：

（1）将数据的似然写成关于参数的函数的形式。
（2）写下对数似然关于每个参数的导数。
（3）解出使得导数为0的参数。

- 只需找到与数据最匹配的模型；在均匀先验条件下与最大后验估计（MAP）相同（若所有假设具有相同复杂度则合理）

- ML是“标准”（非贝叶斯）统计学习方法

- 问题：当数据集非常小以至于一些事件还未发生时——如，还没有樱桃味的糖果被观测到——最大似然假设将把这些事件的概率置为0。
→解决技巧：将所有事件发生次数的计数初始化为1而不是0

完全数据学习

最大似然参数学习：离散模型

- 多参数推广：假设糖果包装与糖果口味相关，在选定一颗糖果后，其包装在概率上服从某个未知的条件分布，该分布取决于糖果的口味。→3个参数θ，θ_1， θ_2

- 假设打开了 N 颗糖果，其中 c 颗是樱桃味的，l颗是酸橙味的。包装的计数如下：r_c颗樱桃味糖果的包装为红色， g_c颗樱桃味糖果的包装为绿色， r_l颗酸橙味糖果的包装为红色， g_l颗酸橙味糖果的包装为绿色。则该数据的似然为

$$P(\boldsymbol{d} \mid h_{\theta,\theta_1,\theta_2}) = \theta^c(1-\theta)^\ell \cdot \theta_1^{r_c}(1-\theta_1)^{g_c} \cdot \theta_2^{r_l}(1-\theta_2)^{g_l}$$

取对数

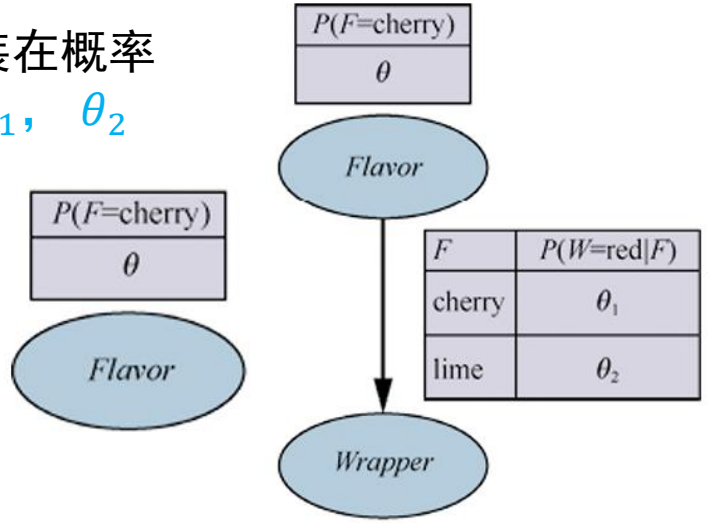
$$L = [c\log\theta + \ell\log(1-\theta)] + [r_c\log\theta_1 + g_c\log(1-\theta_1)] + [r_\ell\log\theta_2 + g_\ell\log(1-\theta_2)]$$

令对数似然对每个参数求导并置为0 时

$$\frac{\partial L}{\partial\theta} = \frac{c}{\theta} - \frac{\ell}{1-\theta} = 0 \quad \Rightarrow \quad \theta = \frac{c}{c+\ell}$$

$$\frac{\partial L}{\partial\theta_1} = \frac{r_c}{\theta_1} - \frac{g_c}{1-\theta_1} = 0 \quad \Rightarrow \quad \theta_1 = \frac{r_c}{r_c+g_c}$$

$$\frac{\partial L}{\partial\theta_2} = \frac{r_\ell}{\theta_2} - \frac{g_\ell}{1-\theta_2} = 0 \quad \Rightarrow \quad \theta_2 = \frac{r_\ell}{r_c+g_\ell}$$



（a）樱桃味糖果和酸橙味糖果比例未知情况下的贝叶斯网络。（b）包装颜色（依概率）与糖果口味相关情况下的模型

- 一旦我们有了完全数据，贝叶斯网络的最大似然参数学习问题 将可以被分解为一些分离的学习问题，每个问题对应一个参数
- 给定其父变量，变量的参数值恰好是该变量值在每一个父变量值下观测到的频率。

完全数据学习

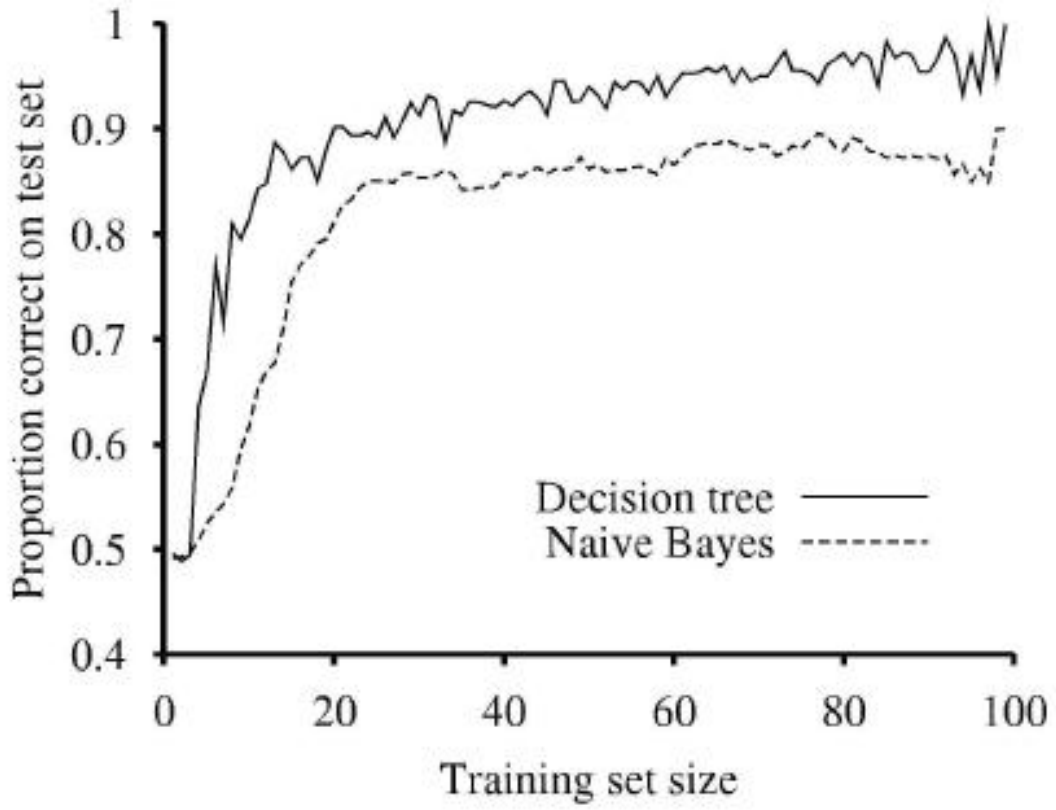
朴素贝叶斯模型

- 假设属性在给定类的情况下是相互条件独立的
- 寻找最大似然参数值。 一 旦模型已经用该方法训练完成，它就可以被用于给类别 C 还未被观测过 的新样例分类。
- 当观测到的属性值为$x_1, \ldots, x_n$, 其属于某一类的概率为

$$\mathbf{P}(C \mid x_1, \ldots, x_n) = \alpha \, \mathbf{P}(C) \prod_i \mathbf{P}(x_i \mid C) \, .$$

- 通过选择可能性最大的类，可以获得一个确定性的预测
- 朴素贝叶斯可以很好地推广到大规模的问题上：当有 n 个布尔属性时，我们只需要2 n + 1 个参数，且不需要任何的搜索就能找到朴素贝叶斯最大似然假设 h_{ML}
- 优点：可以很好地处理噪声或缺失数据
- 缺点：条件独立性假设在实际中通常不成立；该假设会导 致对某些概率做出过度自信的估计，使得它们接近0或1，尤其是在具有大量属性的情况下

完全数据学习



- 该方法学习得相当好，但不及决策树学习：因为真实的假设是一个决策树，而决策树不能被朴素贝叶斯模型准确地表达

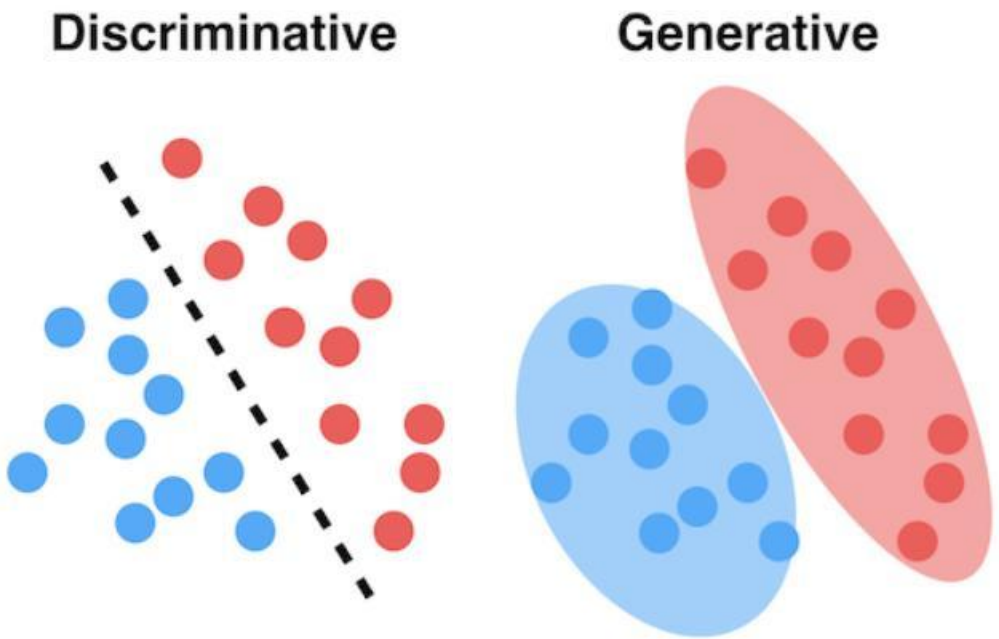
将朴素贝叶斯学习应用于餐厅等待问题得到的学习曲线；决策树的学习曲线也在图中给出，用于比较

完全数据学习

生成模型和判别模型

生成模型 （generative model）对每一类的概率分布进行建模（eg. 朴素贝叶斯文本分类器 ）

判别模型 （discriminative model）直接学习类别之间的决策边界(eg. 逻辑斯谛回归分类器)



完全数据学习

最大似然参数学习：连续模型

例子：单变量高斯密度函数

任务：学习单变量高斯密度函数的参数。即假设数据按如下分布生成：

$$P(x)=\frac{1}{\sigma\sqrt{2\pi}}e^{-\frac{(x-\mu)^2}{2\sigma^2}}$$

模型的参数为均值μ以及标准差σ。

假设我们有观测值 $x_1,\cdots,x_N$，那么其对数似然为：

$$L=\sum_{j=1}^{N}\log\frac{1}{\sigma\sqrt{2\pi}}e^{-\frac{(x_j-\mu)^2}{2\sigma^2}}=N(-\log\sqrt{2\pi}-\log\sigma)-\sum_{j=1}^{N}\frac{(x_j-\mu)^2}{2\sigma^2}$$

令其导数为0

$$\frac{\partial L}{\partial\mu}=-\frac{1}{\sigma^2}\sum_{j=1}^{N}(x_j-\mu)=0 \quad\Rightarrow\quad \mu=\frac{\sum_j x_j}{N}$$

$$\frac{\partial L}{\partial\sigma}=-\frac{N}{\sigma}+\frac{1}{\sigma^3}\sum_{j=1}^{N}(x_j-\mu)^2=0 \quad\Rightarrow\quad \sigma=\sqrt{\frac{\sum_j(x_j-\mu)^2}{N}}$$

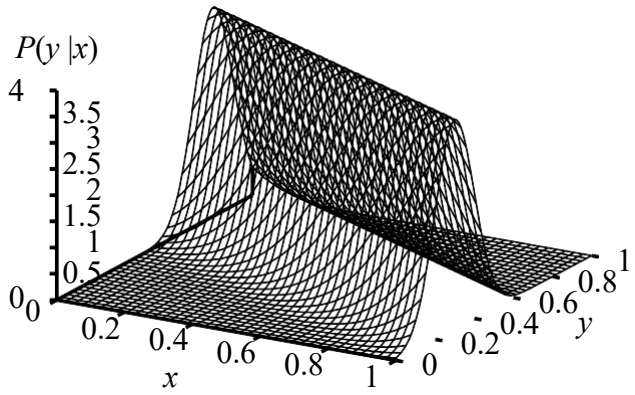
完全数据学习

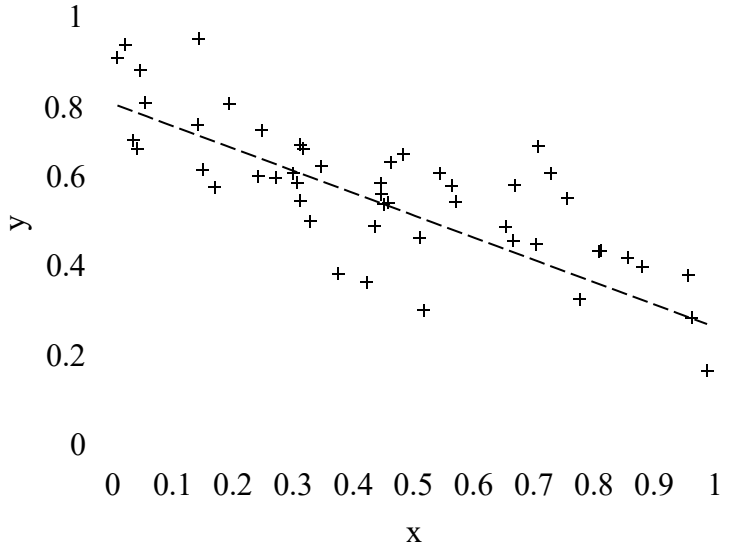
最大似然参数学习：连续模型

例子：线性高斯模型

有一个连续的父变量 X 和一个连续 的子变量 Y 。 Y 服从高斯分布，其均值线性地依赖于 X ，其标准差是固定的。为了学习条件分布P(Y|X)，可以最大化条件似然：

$$P(y \mid x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{(y-(\theta_1 x+\theta_2))^2}{2\sigma^2}}$$





参数为θ_1、 θ_2和 σ

如果仅考虑定义 x 和 y 之间线性关系的参数θ_1、 θ_2，那么等价于最小化：

$$(y-(\theta_1 x+\theta_2))^2$$ →恰好是 L2损失，即实际值 y 和预测值θ_1x+θ_2 之间的平方误差

如果数据的生成过程带有固定方差的高斯噪声，那么最小化误差平方和恰好给出最大似然线性模型

完全数据学习

贝叶斯参数学习

- 最大似然学习方法在小数据集情况下存在严重缺陷
- 基于贝叶斯方法的参数学习过程从一个关于假设的先验分布开 始，随着新数据出现而不断更新该分布

- 从贝叶斯角度看来, θ 是随机变量$\boldsymbol{\Theta}$ 的一个未知值， $\boldsymbol{\Theta}$定义了假设空间
  (糖果例子有一个参数θ ：随机挑选一颗糖果，它为樱桃味的概率)

- 假设的先验是先验分布 $\mathbf{P}(\boldsymbol{\Theta})$.
  ($\mathrm{P}(\boldsymbol{\Theta} = \theta)$ 是糖果袋中含有θ比例的樱桃味糖果的先验概率)

- 如果参数θ可以是介于0和1之间的任意一个值，那么$\mathbf{P}(\boldsymbol{\Theta})$将是一个 连续的概率密度函数，如果对$\theta$的可能的值没有任 何的信息，那么可以采用均匀分布$P(\theta) = Uniform(\theta{:}0,1)$作为先验，它意味着任何取值都是等可能的。

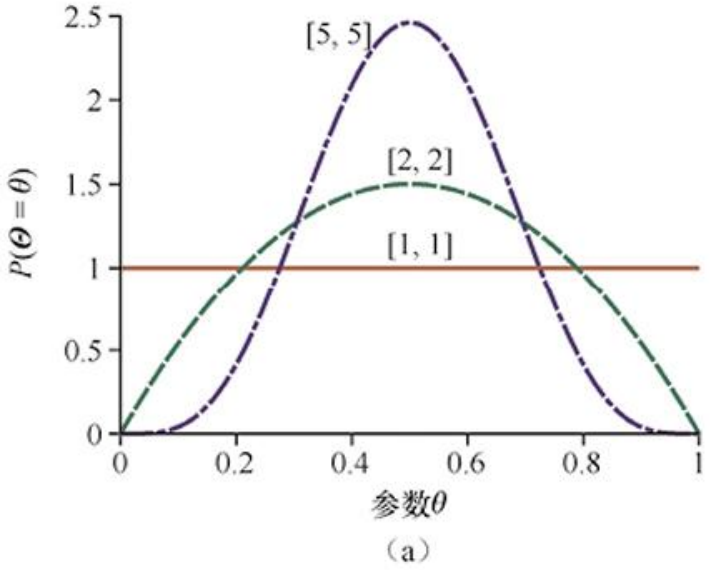
完全数据学习

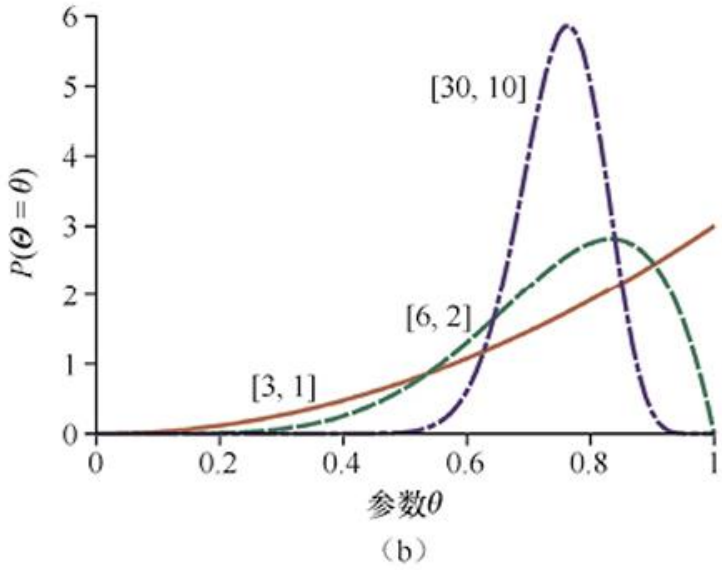
贝叶斯参数学习

β分布是一个灵活的概率密度函数族，每个β分布由两个 超参数 a和b 定义

$$Beta(\theta; a, b) = \alpha\, \theta^{a-1}(1-\theta)^{b-1}$$

- 如果参数Θ有先验 β (a, b)，那么在一个数据点被观测之后，其参数Θ的后验分布仍是一个β分布。换句话说， β分布在这种更新规则下是封闭的。





不同(a, b)下Beta (a, b)分布的例子

完全数据学习

贝叶斯参数学习

假设观测到了一颗樱桃味的糖果，那么有

$$P(\theta \mid D_1 = \text{cherry}) = \alpha\, P(D_1 = \text{cherry} \mid \theta) P(\theta)$$
$$= \alpha' \theta \cdot Beta(\theta; a, b) = \alpha' \theta \cdot \theta^{a-1}(1-\theta)^{b-1}$$
$$= \alpha' \theta^{a}(1-\theta)^{b-1} = \alpha' Beta(\theta; a+1, b)$$

- 在观测完这个樱桃味的糖果后，简单地增大了参数 a 的 值；同样，观测到一颗酸橙味的糖果之后，增大参数 b 的值
- 超参数a和b看作 虚拟计数——因为先验分布 Beta (a, b)可被视为是从均匀分布先验 Beta (1 , 1)出发，并且 已经“虚拟”地观测到 a − 1次樱桃味糖果和 b – 1次酸橙味糖果
- 保持 a 和 b 两者比值不变，不断增大 a 和 b ，通过观测一系列分布， 可以清楚地观测到参数的后验分布随着数据增多的变化情况。

完全数据学习

贝叶斯参数学习

进一步推广至3个参数：θ 、θ_1和θ_2

（θ为每次抽到 cherry 的概率， θ_1代表樱桃味糖果中包装为红色的概率， θ_2代表酸橙味糖果中包装为红色的概率）

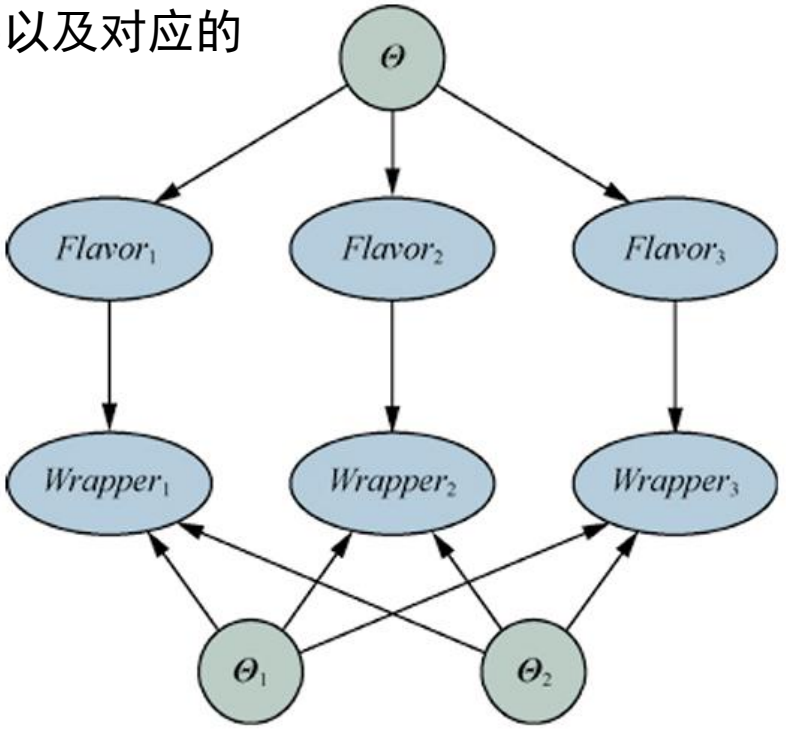
- 参数独立性假设：$\mathbf{P}(\Theta, \Theta_1, \Theta_2) = \mathbf{P}(\Theta)\mathbf{P}(\Theta_1)\mathbf{P}(\Theta_2)$.
- 加入节点 Wrapperi 与 Flavori 用 于表示第 i 个被观测到的糖果包装以及对应的糖果口味
  - Flavori 取决 于口味对应的参数θ：

$$P(Flavor_i = cherry \mid \Theta = \theta) = \theta \ .$$

  - Wrapperi 取决于参数θ_1和θ_2 ：

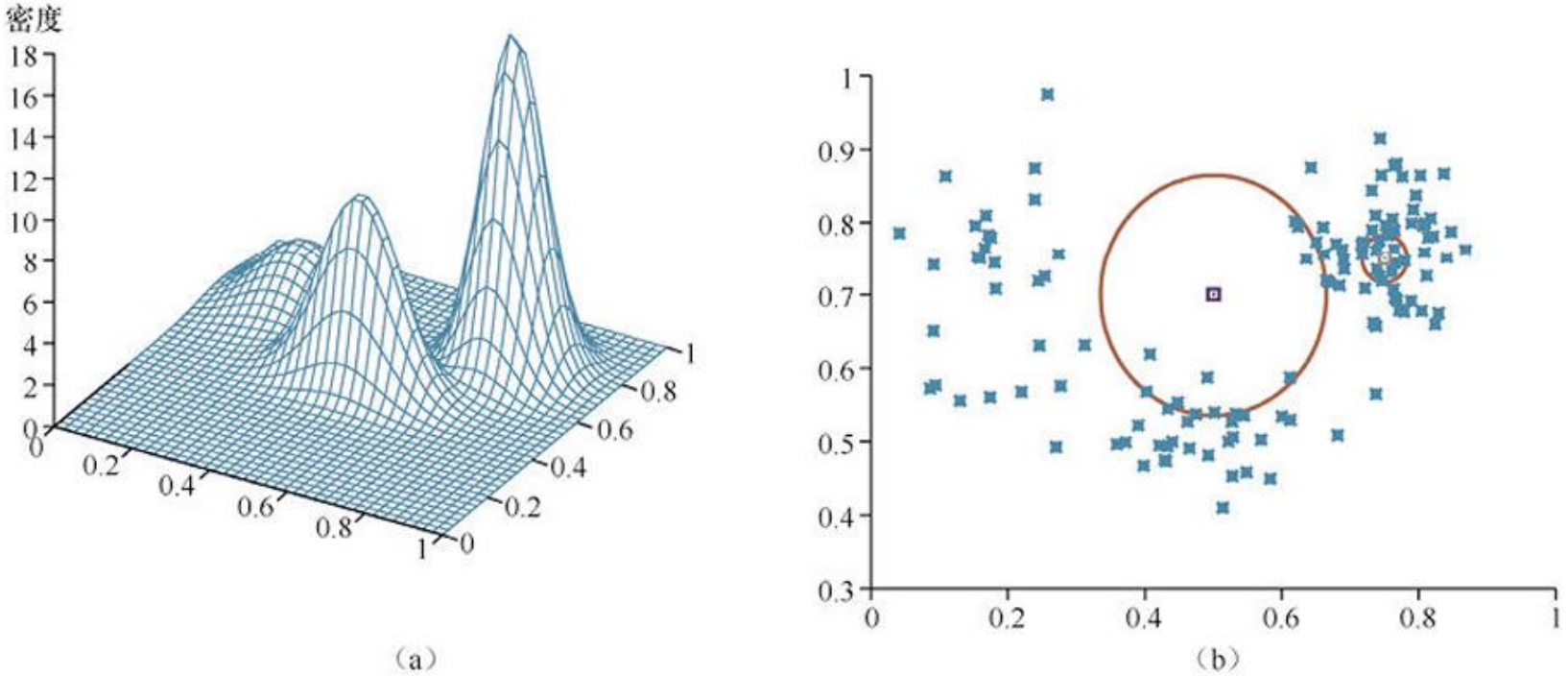
$$P(Wrapper_i = red \mid Flavor_i = cherry, \Theta_1 = \theta_1) = \theta_1$$
$$P(Wrapper_i = red \mid Flavor_i = lime, \Theta_2 = \theta_2) = \theta_2 \ .$$

  - 可以使用贝叶斯网络的推断算法



完全数据学习

非参数模型密度估计——在连续域中学习一个概率模型



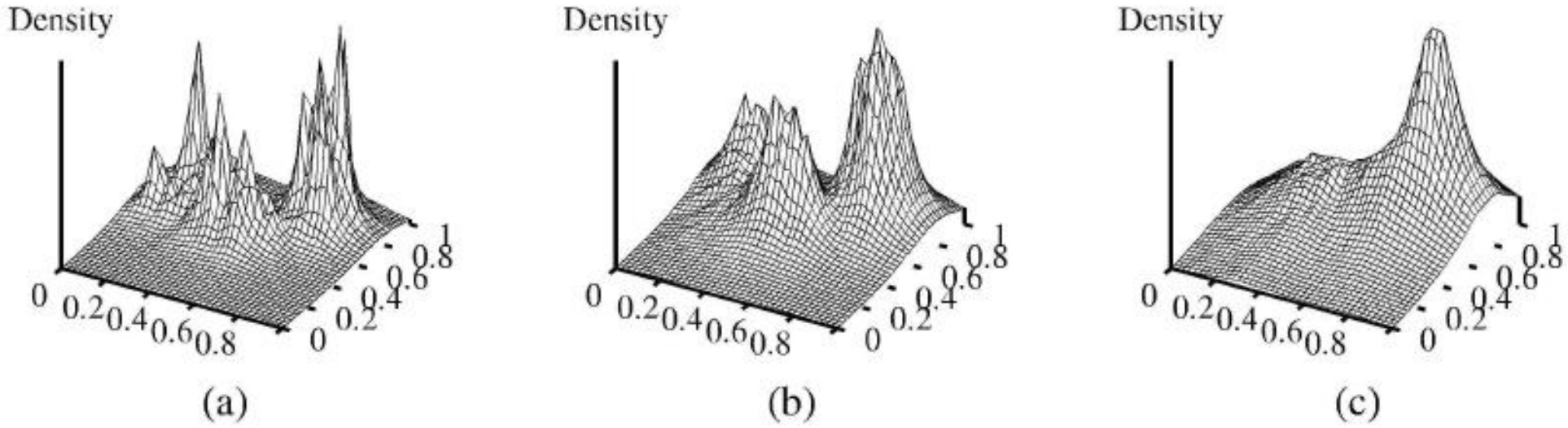
（a）混合高斯模型的三维样貌。
（b）从混合高斯模型中采样的128个数据点、两个查询点（小方块）以及它们的10近邻（大圆圈以及右边的小圆圈）

完全数据学习

非参数模型密度估计

- k近邻模型

给定一组数据样本点，为估计某个查询点x的未知概率密度，我们可以简单地估计数据点落在查询点x附近的密度。



应用k近邻进行密度估计，所用的数据为图20-8b中的数据，分别对应k =3、10和 40。k = 3的结果过于尖锐，40的结果过于光滑，而10的结果接近真实情况。最好的k值可以通过交叉验证进行选择

完全数据学习

非参数模型密度估计

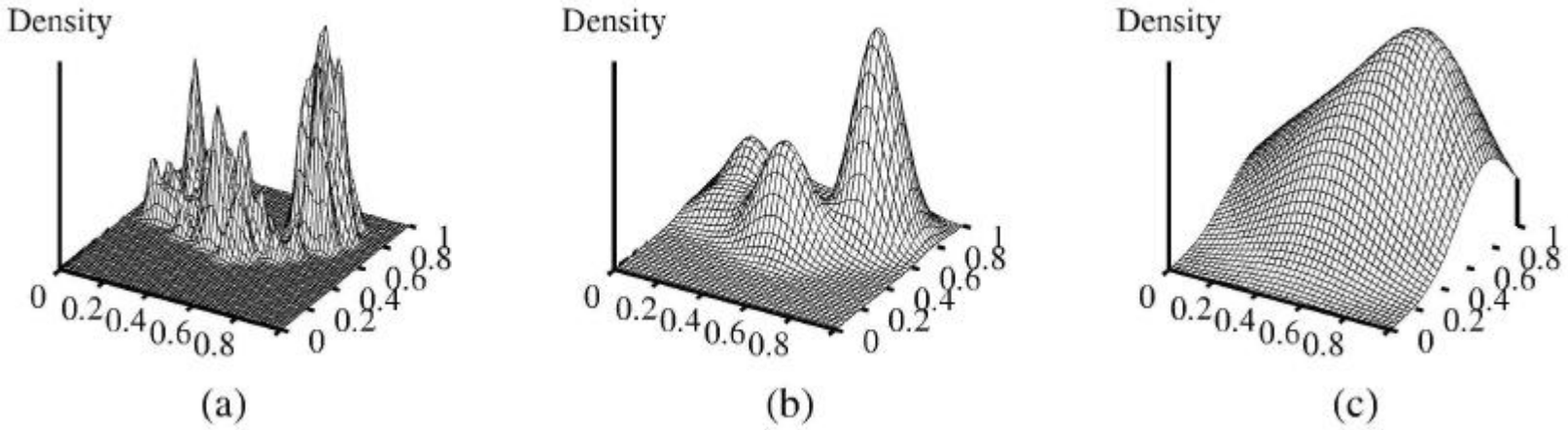
- 核方法

为了在密度估计中应用核函数，假设每个数据点都将生成一个与自己相关的密度函数。例如，可以采用在每个维度上标准差均为w的球形高斯核。那么对于查询点x，给出的密度估计值为数据核函数的均值

$$P(\mathbf{x}) = \frac{1}{N}\sum_{j=1}^{N}\mathcal{K}(\mathbf{x},\mathbf{x}_j)\ .$$

其中

$$\mathcal{K}(\boldsymbol{x},\boldsymbol{x}_j) = \frac{1}{(w^2\sqrt{2\pi})^d}\mathrm{e}^{-\frac{D(\boldsymbol{x},\boldsymbol{x}_j)^2}{2w^2}}$$

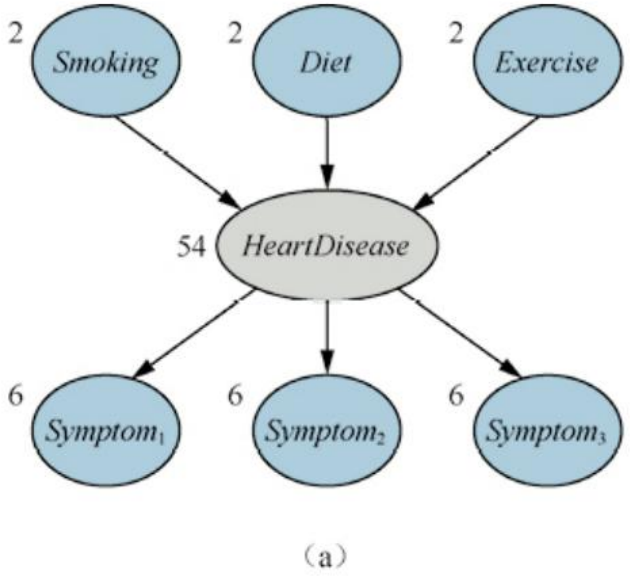


使用核函数进行密度估计，所用数据为图20-8b中的数据，分别采用了w = 0.02、0.07和0.20的高斯核。其中w = 0.07的结果最接近真实情况

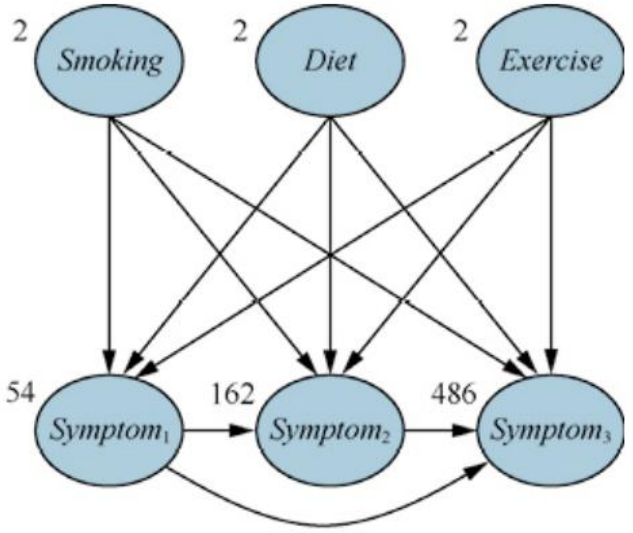
隐变量学习——EM算法

隐变量学习

- 现实生活中，许多问题存在隐变量（hidden variable），有时也称为隐藏变量（latent variable），这些变量在数据中是未被观测的。



（a）



（b）

隐变量可以大大减少确定一个贝叶斯网络所需参数的个数。同样也可 以大大减少所需学习的参数的个数。

（a）一个简单的心脏病诊断网络，其中HeartDisease是一个隐变量。每个变量有3个可能的值，并标明了每个变量对应的条件独立参数的个数，其总数为78。
（b）去除隐变量HeartDisease之后的等效网络。注意，给定了父变量值后，症状对应的变量不再是条件独立的。这个网络有708个参数

隐变量学习——EM算法

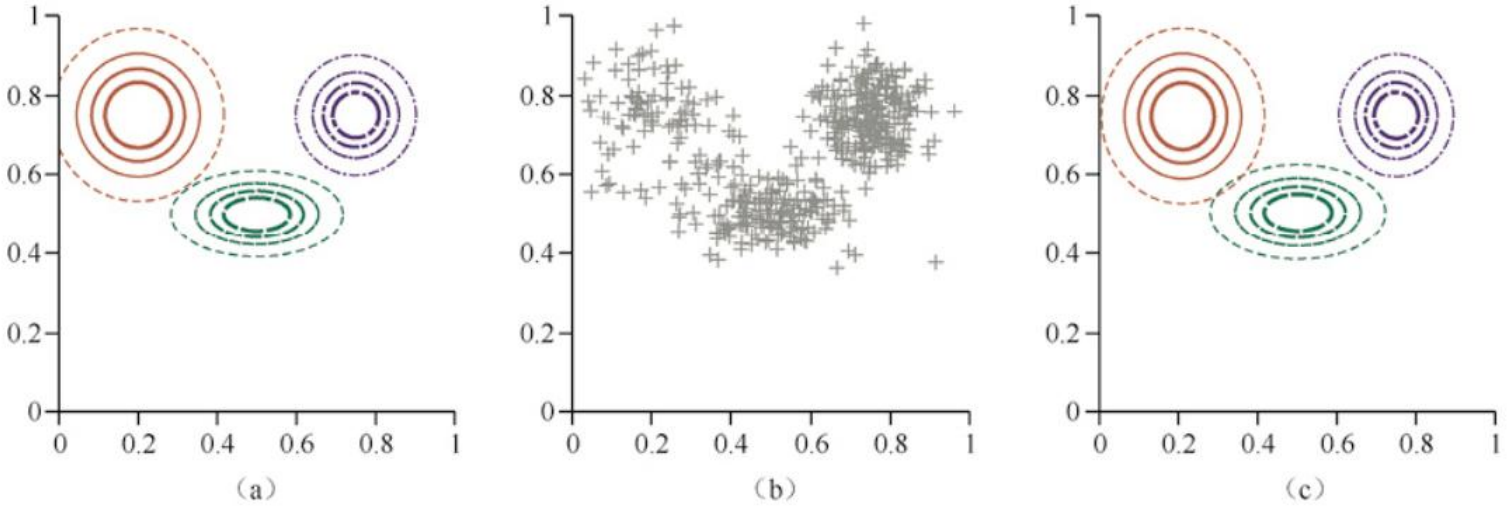
无监督聚类：学习混合高斯

- 无监督聚类（unsupervised clustering）是在一个对象集合中识别多个类别的问题。该问题被称为是无监督的，是因为数据没有被赋予类别标签。
- 无监督聚类以数据为出发点。需要了解什么样的概率分布可能产生这些数据。
- 聚类假设了数据是从某个混合分布P中生成的。该分布由k个分量组成，每个分量本身是一个分布。数据点通过以下方法生成：首先选择其中一个分量，然后从该分量采样一个样
- 本，从而生成一个数据点。令随机变量C为数据对应的分量，其值为1, …, k；那么混合分布将由下式给出：

$$P(\boldsymbol{x}) = \sum_{i=1}^{k} P(C=i)\, P(\boldsymbol{x} \mid C=i)$$

- 对于连续数据，多元高斯分布是各个分量分布的一个自然选择，这就是所谓的混合高斯分布族。混合高斯分布的参数为 $w_i = P(C=i)$（各分量的权重）、 μ_i（各分量的均值），以及 Σ_i（各分量的协方差）
- 相应的无监督聚类问题则是从原始数据中复原出高斯混合模型
  - 如果知道每个数据点由哪个分量生成，那么就很容易复原对应的高斯分布分量
  - 如果知道每个分量的参数，那么可以给出每个数据点属于某个分量的概率

隐变量学习——EM算法



（a）由3个分量组成的混合高斯模型，其权重（从左到右）分别为0.2、0.3和0.5。
（b）采样于（a）中模型的500个数据点。
（c）根据（b）中数据点，使用EM算法重建出的模型

隐变量学习——EM算法

无监督聚类：学习混合高斯

期望最大化（expectation_x0002_maximization，EM）算法

- 假设知道模型的参数，然后推断每个数据点属于各个分量的概率
- 重新使用数据拟合各个分量，其中每个分量的拟合都将用到整个数据集，每个数据点的权重由它属于该分量的概率给出
- 重复以上过程直到算法收敛

本质上，所做的事情是基于当前的模型推断隐变量——数据点属于某个分量——的概率分布，进而“完善”数据。对于混合高斯模型，我们可以任意地初始化混合模型参数，然后进行两个步骤的迭代。

隐变量学习——EM算法

无监督聚类：学习混合高斯

期望最大化（expectation_maximization，EM）算法

（1）E步：计算概率$P_{ij}=\mathrm{P}(\mathrm{C}=\mathrm{i}|X_j)$，即数据点x是由分量i生成的概率。根据贝叶斯法则，我们有$P_{ij}=\alpha\mathrm{P}(X_j|\mathrm{C}=\mathrm{i})\mathrm{P}(\mathrm{C}=\mathrm{i})$。其中$\mathrm{P}(X_j|\mathrm{C}=\mathrm{i})$项是x在第i个高斯分量中的概率，P(C=i)项是第i个高斯分量的权重。定义$n_i=\sum_j p_{ij}$，即分配至第i个分量的数据点的有效个数。

（2）M步：按照以下式子计算新的均值、方差和各分量的权重。

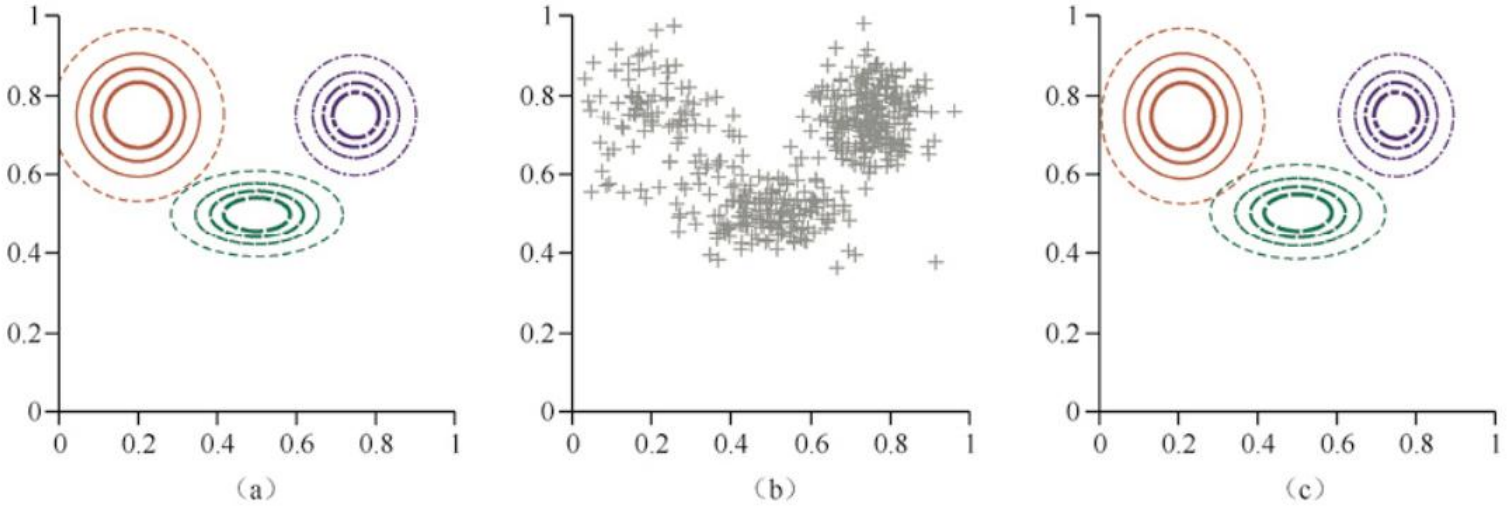
$$\boldsymbol{\mu}_i \leftarrow \sum_j p_{ij}\mathbf{x}_j/n_i$$

$$\boldsymbol{\Sigma}_i \leftarrow \sum_j p_{ij}(\mathbf{x}_j-\boldsymbol{\mu}_i)(\mathbf{x}_j-\boldsymbol{\mu}_i)^{\top}/n_i$$

$$w_i \leftarrow n_i/N$$

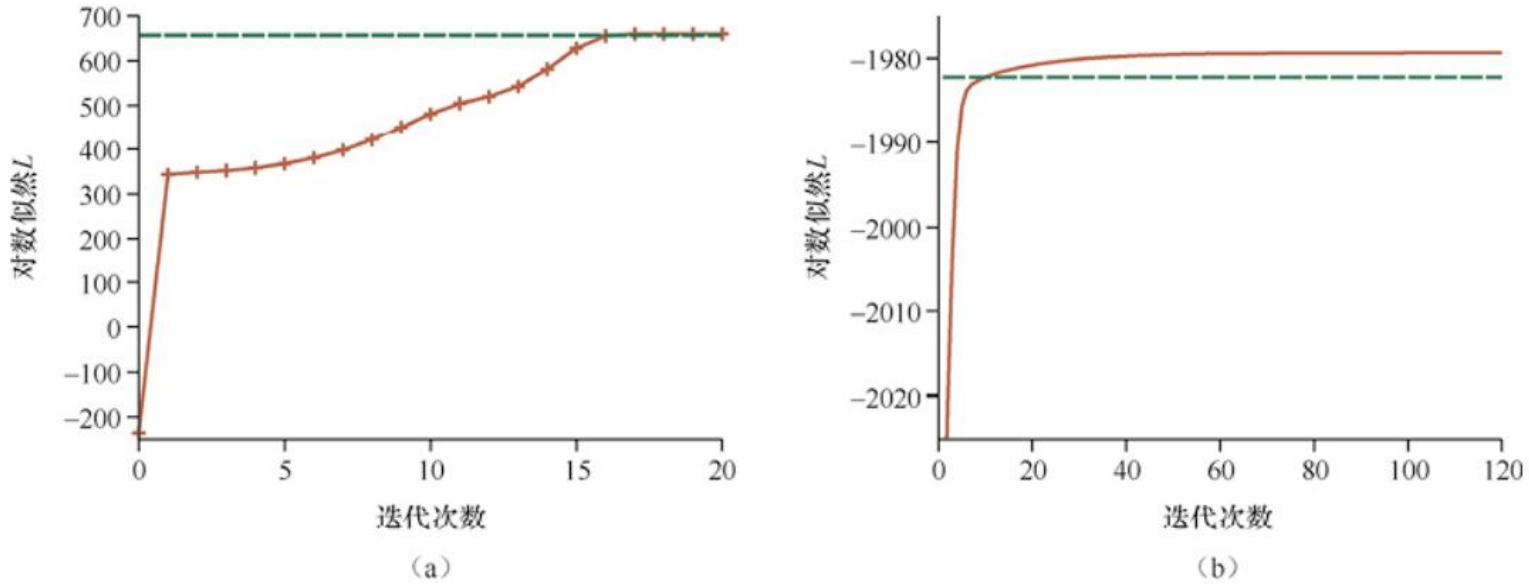
E步也称期望步，它可以视为计算隐指示（hidden indicator）变量Z_{ij}的期望值p_{ij}的步骤，若数据xj由第i个分量生成，则Z_{ij}为1，否则为0。M步也称最大化步，其目标是寻找给定隐指示变量的期望情况下，使数据的似然最大化的新参数

隐变量学习——EM算法



（a）由3个分量组成的混合高斯模型，其权重（从左到右）分别为0.2、0.3和0.5。
（b）采样于（a）中模型的500个数据点。
（c）根据（b）中数据点，使用EM算法重建出的模型

隐变量学习——EM算法



数据的对数似然L关于EM算法迭代次数的函数关系。水平线表示真实模型下数据的对数似然。
（a）混合高斯模型对应的变化图。
（b）贝叶斯网络对应的变化图

隐变量学习——EM算法

无监督聚类：学习混合高斯

需要注意两点。

- 第一，最终学习到的模型的对数似然值略高于用于生成数据的真实模型的对数似然值。它简单地反映了这样一个事实：数据是随机生成的，也许没有精确地反映出真实的模型。
- 第二，在EM算法的进行过程中，数据的对数似然在每一次迭代后都将提升。在大多数情况下，可以证明EM算法将达到似然函数的局部极大值。（在极少数情况下，它可能会达到一个鞍点，甚至一个局部极小值。）从这个意义上说，EM类似于基于梯度的爬山算法，但需要注意的是它没有“步长”这一参数。

EM算法问题：

- 它可能导致某个高斯分量发生退化，使得它仅仅包含一个数据点。那么它的方差将趋向于零，且它的似然将趋向无穷！如果不知道混合模型中有多少个分量，就需要尝试不同的分量个数，即尝试不同的k值，并观测哪个值的效果最好，但这也可能导致发生另一些错误。
- 两个分量可能会“合并”，导致它们有相同的均值和方差，且它们共享数据点。这种退化的局部极大值是一个严重的问题，特别是在高维情况下。

解决方案：

1. 对模型参数赋予先验并采用MAP版本的EM算法。
2. 如果某个分量太小或太接近于另一个分量，则使用新的随机参数重置该分量。一个合理的初始化方法对算法也有帮助。

隐变量学习——EM算法

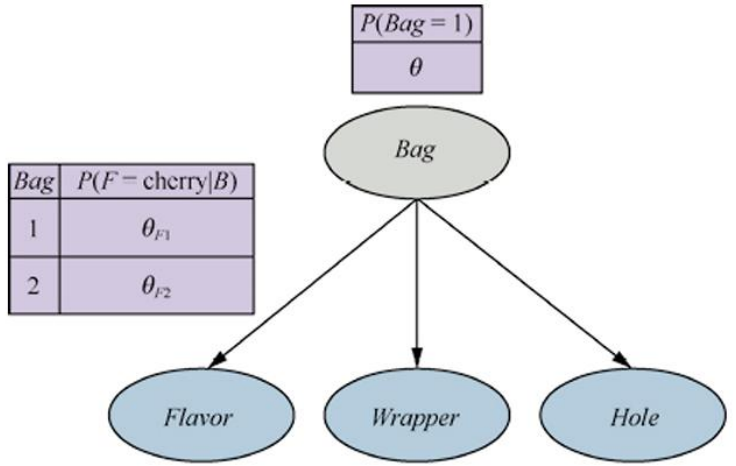
学习带隐变量的贝叶斯网络参数值

例子：有两袋混合在一起的糖果。

- 糖果有3个特征：除口味（Flavor）和包装（Wrapper）外，一些糖果中间还有夹心（Holes），而有些糖果没有。
- 糖果在每个糖果袋中的分布状况可以用朴素贝叶斯模型进行描述：在给定糖果袋的情况下，特征之间是独立的，但每个特征的条件概率取决于这个糖果袋的状况
- θ为糖果取自糖果袋1的先验概率;θ_{F1}与θ_{F2}分别是给定糖果取自于糖果袋1或糖果袋2后,它是樱桃口味的概率;θ_{W1}与θ_{W2}是在同样的给定条件下糖果包装为红色的概率;θ_{H1}和θ_{H2}是在同样的给定条件下糖果有夹心的概率。
- 假设从一个模型中生成了1000个样本，真实参数分布为：

$$\theta = 0.5, \theta_{F1} = \theta_{W1} = \theta_{H1} = 0.8, \theta_{F2} = \theta_{W2} = \theta_{H2} = 0.3$$

	$W = red$		$W = green$	
	$H = 1$	$H = 0$	$H = 1$	$H = 0$
$F = cherry$	273	93	104	90
$F = lime$	79	100	94	167



关于糖果的混合模型。不同口味、包装的比例以及是否有夹心取 决于糖果袋，该变量是不可观测的

隐变量学习——EM算法

学习带隐变量的贝叶斯网络参数值

首先，考虑参数θ 。由于糖果袋是一个隐变量，计算糖果个数的期望。糖果个数的期望 $\hat{N}(Bag=1)$ 是每个糖果来自于糖果袋1的概率之和

$$\theta^{(1)} = \hat{N}(Bag=1)/N = \sum_{j=1}^{N} P(Bag=1 \mid flavor_j, wrapper_j, holes_j)/N \ .$$

利用贝叶斯法则以及条件独立性计算得到

$$\theta^{(1)} = \frac{1}{N}\sum_{j=1}^{N} \frac{P(flavor_j \mid Bag=1)P(wrapper_j \mid Bag=1)P(holes_j \mid Bag=1)P(Bag=1)}{\sum_i P(flavor_j \mid Bag=i)P(wrapper_j \mid Bag=i)P(holes_j \mid Bag=i)P(Bag=i)} \ .$$

在数据完全可观测的情形下，可以直接通过观测到糖果袋1中的樱桃味和酸橙味糖果数量来估计其他参数值。糖果袋1中的樱桃味糖果数量的期望可以由下式给出

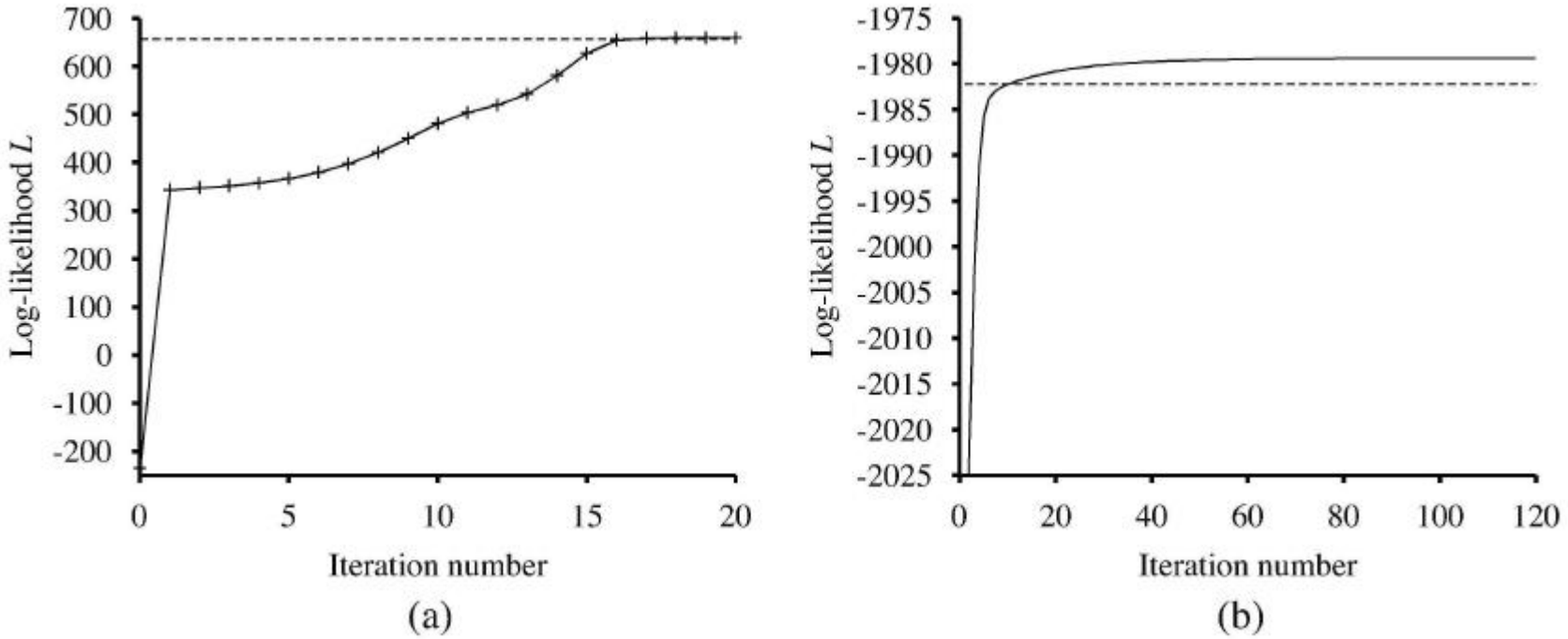
$$\sum_{j:Flavor_j=cherry} P(Bag=1 \mid Flavor_j=cherry, wrapper_j, holes_j) \ .$$

CPT中的参数更新由计数期望归一化后给出

$$\theta_{ijk} \leftarrow \hat{N}(X_i=x_{ij}, \mathbf{U}_i=\mathbf{u}_{ik})/\hat{N}(\mathbf{U}_i=\mathbf{u}_{ik}) \ .$$

一般性的规律，即在带隐变量的贝叶斯网络学习中，参数更新可以从每个样例的推断结果中直接得到。更进一步地，每个参数的估计都只需要用到局部的后验概率。这里的“局部”意味着每个变量Xi的条件概率表（CPT）可以从仅涉及Xi及其父节点Ui的后验概率中学习得到

隐变量学习——EM算法



数据的对数似然L关于EM算法迭代次数的函数关系。水平线表示真实模型下数据的对数似然。（a）混合高斯模型对应的变化图。（b）贝叶斯网络对应的变化图

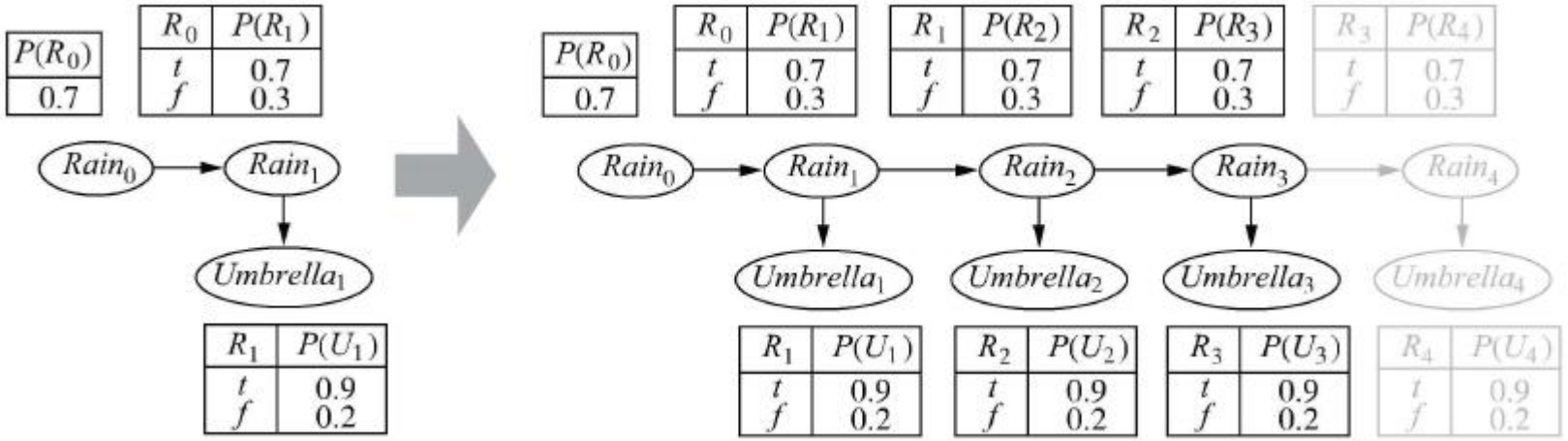
隐变量学习——EM算法

学习隐马尔可夫模型

- 将EM算法应用于学习隐马尔可夫模型（HMM）中的转移概率
- 隐马尔可夫模型可以用一个带有单个离散状态变量的动态贝叶斯网络来表示
- 每个数据点为一个长度有限的观测序列，因此要解决的问题是从一组观测序列（或仅从一个长序列）中学习转移概率
- 在贝叶斯网络中，每个参数都是分离的；但在隐马尔可夫模型中，对于任意时刻t，从状态i到状态j的转移概率 $\theta_{ijt} = P(X_{t+1} = j \mid X_t = i)$ 是相等的，即对任意时刻t有 $\theta_{ijt} = \theta_{ij}$
- 为了估计从状态i到状态j的转移概率，我只需计算系统在状态i经过一次转移后到达状态j的次数比例的期望

$$\theta_{ij} \leftarrow \sum_t \hat{N}(X_{t+1} = j, X_t = i) / \sum_t \hat{N}(X_t = i) .$$

隐变量学习——EM算法



表示隐马尔可夫模型的动态贝叶斯网络展开图

隐变量学习——EM算法

EM算法的一般形式

X:所有样例中的所有观测值
Z: 所有样例中的所有隐变量
θ: 概率模型中的所有参数

- 那么对应的EM算法可以表示为

$$\boldsymbol{\theta}^{(i+1)} = \underset{\boldsymbol{\theta}}{\operatorname{argmax}} \sum_{\mathbf{z}} P(\mathbf{Z}=\mathbf{z} \mid \mathbf{x}, \boldsymbol{\theta}^{(i)}) L(\mathbf{x}, \mathbf{Z}=\mathbf{z} \mid \boldsymbol{\theta}) \ .$$

- E步是求和计算，即计算分布 $P(\boldsymbol{Z}=\boldsymbol{z} \mid \boldsymbol{x}, \theta^{(i)})$ 下“完整”数据对数似然的期望，其中该分布为给定数据后隐变量的后验分布
- M步则是选取参数使得该对数似然的期望达到最大
- 对于混合高斯模型，隐变量为Zij，当样例j由分量i生成时其值为1。对于贝叶斯网络，Zij是样例j中未观测到的变量Xi的值。对于HMM，Zjt是样例序列j在时刻t所处的状态。

小结

- 贝叶斯学习方法将学习表述为概率推断的形式，利用观测值对假设空间的先验分布进行更新。这个方法很好地呈现了奥卡姆剃刀原理，但它很难处理更复杂的假设空间。
- 最大后验（MAP）学习选择给定数据下可能性最大的假设。该方法同样利用了假设的先验分布，并且该方法通常比贝叶斯学习更易处理。
- 最大似然学习选择使得数据的似然最大的假设；它等价于使用均匀分布作为先验的最大后验学习。例如线性回归或完全可观测的贝叶斯网络等简单的情形中，我们容易找到最大似然的闭式解。

朴素贝叶斯学习也是一个运用范围广泛且特别有效的方法。

- 当一些变量被隐藏时，期望最大化（EM）算法可以找到局部最大似然解。其应用包括高斯混合模型的无监督聚类、贝叶斯网络学习和隐马尔可夫模型的学习。
- 学习贝叶斯网络的结构是模型选择的一个例子。它通常涉及结构空间中的离散搜索。我们需要一些方法来权衡模型的复杂性和拟合程度。
- 非参数模型通过一些数据点集合来表示某一分布，因此它的参数数量将随着训练集的增大而增加。最近邻方法寻找离查询点最近的样例，而核方法则考虑所有样例基于距离的加权组合。

应用

从概率模型到生成模型

概率模型学习：通过学习数据的概率分布来理解世界。用概率描述不确定性

$$p(x), \quad p(y|x), \quad p(x,y)$$

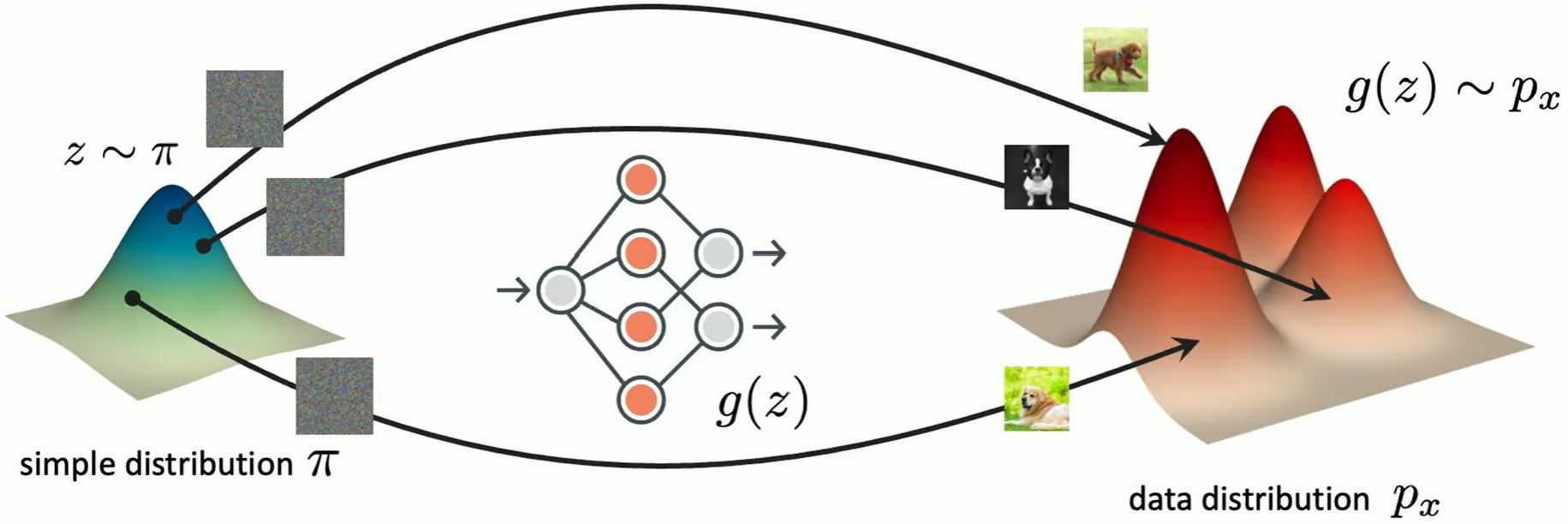
目标是找到一个概率分布$p_\theta(\mathrm{x})$，能解释观测数据是如何产生的,找到最可能产生这些数据的模型

$$\theta^* = \arg\max_\theta p_\theta(x)$$

传统概率模型：显式表示概率密度函数
生成式概率模型：除建模外，还要能够采样并生成数据

生成式学习 = 概率建模 + 采样生成

应用


$z \sim \pi$
$g(z)$
$g(z) \sim p_x$
simple distribution π
data distribution p_x

应用

- Generative models are about $p(x|y)$

What can be y?

- condition
- constraint
- labels
- attributes

- more abstract
- less informative

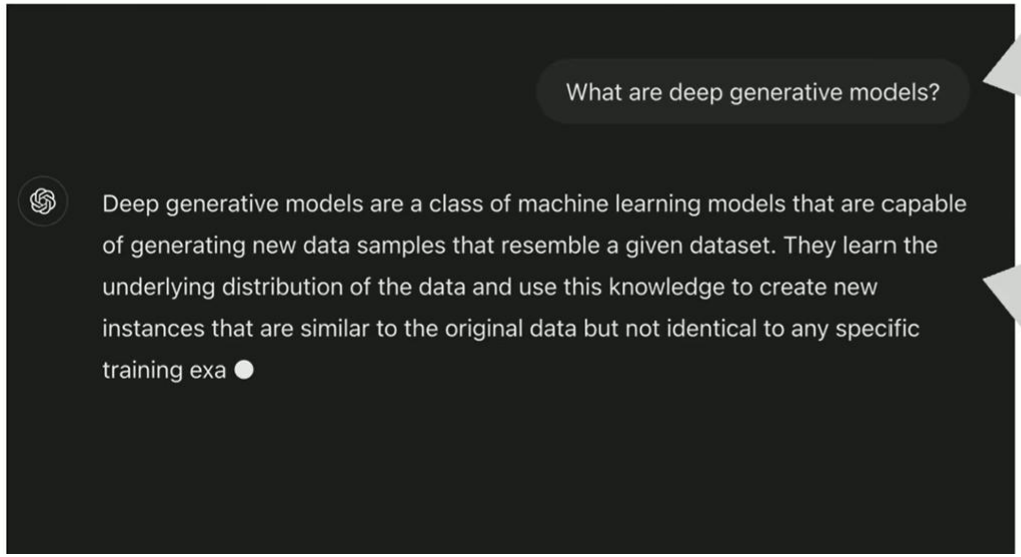
What can be x?

- “data”
- samples
- observations
- measurements

- more concrete
- more informative

应用

- **Natural language conversation**



y: prompt

x: response of the chatbot

应用

- **Text-to-image/video generation**

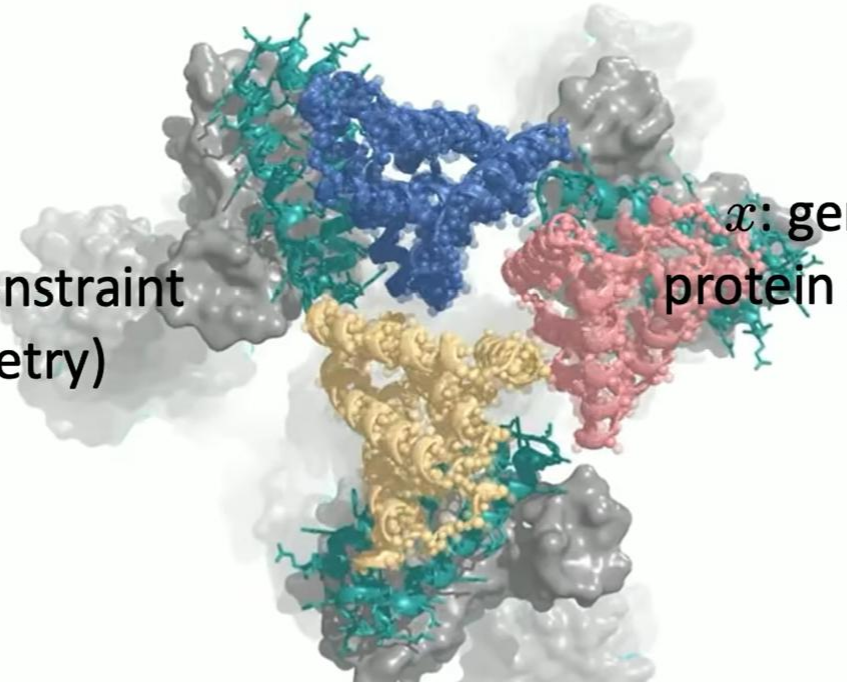
Prompt: teddy bear teaching a course, with "generative models" written on blackboard

y: text prompt

x: generated visual content

应用

- **Protein structure generation**

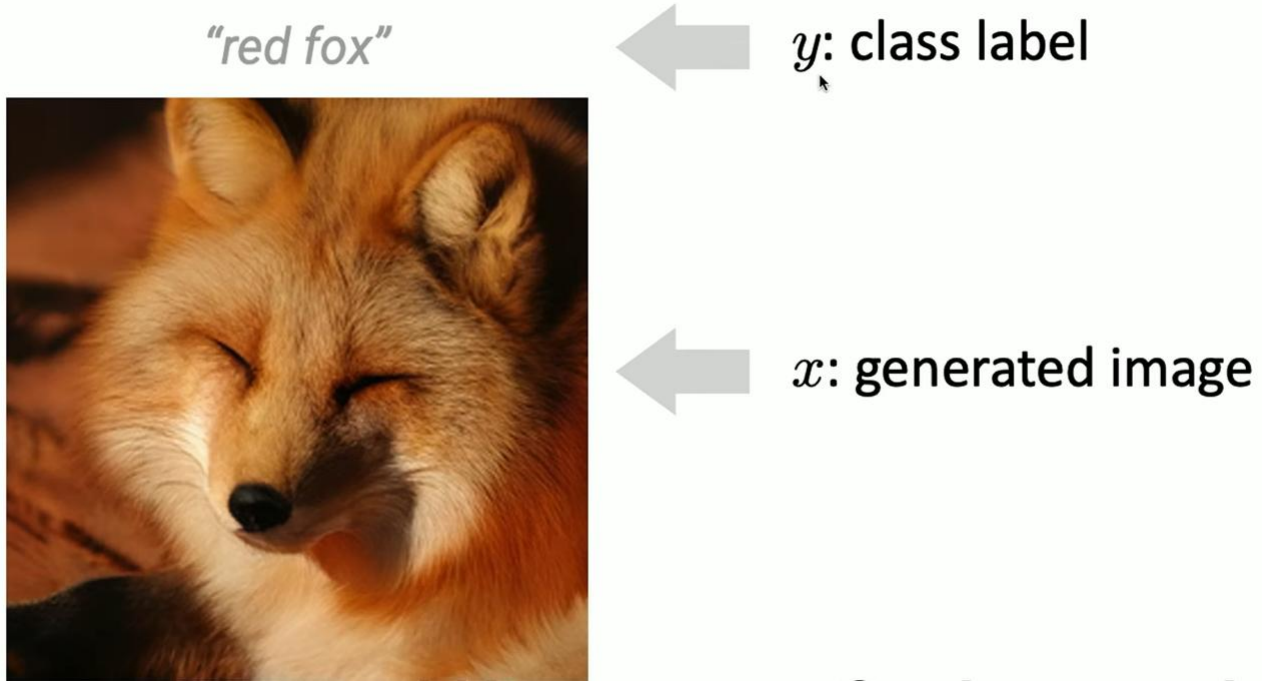


y: condition/constraint
(e.g., symmetry)

x: generated
protein structures

应用

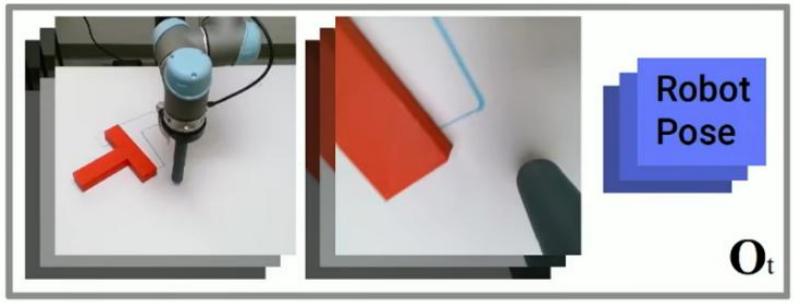
- **Class-conditional image generation**



应用

- **Policy Learning in Robotics**

y: visual and other sensory observations



x: policies (probability of actions)