概率图模型

一、背景：

高维随机变量 $P(x_1,x_2,\dots,x_p)$

- 边缘概率 $P(x_i)$
- 条件概率 $P(x_j|x_i)$

在计算过程中的重要法则：

1. 加法法则：$P(x_1)=\int P(x_1,x_2)\,dx_2$
2. 乘法法则：$P(x_1,x_2)=P(x_1)P(x_2|x_1)=P(x_2)P(x_1|x_2)$

Chain Rule：$P(x_1,x_2,\dots,x_p)=P(x_1)P(x_2|x_1)\cdot P(x_3|x_1,x_2)\cdots P(x_p|x_1,x_2,\dots,x_{p-1})$

贝叶斯定理：$P(x_2|x_1)=\frac{P(x_1,x_2)}{P(x_1)}=\frac{P(x_1,x_2)}{\int P(x_1,x_2)dx_2}=\frac{P(x_2)P(x_1|x_2)}{\int P(x_2)P(x_1|x_2)dx_2}$

困境：维度高，计算复杂，$P(x_1,x_2\cdots x_p)$ 计算量太大。

↓

简化 $\xrightarrow{\text{假设独立}}$ $P(x_1,x_2\cdots x_p)=\prod_{i=1}^{p}P(x_i)$

Naive Bayes：$P(x|y)=\prod_{i=1}^{p}P(x_i|y)$

依赖过强，需减弱

$\xrightarrow{\text{Markov Property}}$ $x_j \perp x_{j+1}|x_i \quad j<i$

HMM（齐次Markov假设）

给定x_i的情况下，x_j与x_{i+1}独立

依赖仍然过强，过于单调。

↓

条件独立性 ☆

$x_A \perp x_B | x_C$，x_A, x_B, x_C是集合且不相交。

要把概率的概念引伸到图的结构上，需要图能够较好地表现条件独立性。

概率图

- Representation 表示
  - 有向图 Bayesian
  - 高斯图（连续）
    - Gaussian BN
    - Gaussian MN
  - 无向图 Markov Network
- Inference 推断
  - 精确推断
  - 近似推断
    - 确定性近似（变分推断）
    - 随机近似（MCMC）
- Learning 学习
  - 参数学习
    - 完备数据 {有向, 无向}
    - 隐变量 → EM
  - 结构学习

如何构建有向图：拓扑排序

二. 贝叶斯网络 (有向图)

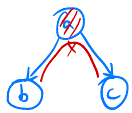
$$P(x_1, x_2, \cdots, x_p) = P(x_1) \cdot \prod_{i=2}^{p} P(x_i | x_{1:i-1})$$

条件独立性：$x_A \perp x_C | x_B$

因子分解：联合概率 $P(x_1, x_2, \cdots, x_p) = \prod_{i=1}^{p} P(x_i | x_{pa(i)})$

$x_{pa(i)}$ 是 x_i 的父亲集合.

例如有向图：



(tail to tail)

$c \perp b | a$

$\Updownarrow$

若 a 被观测，则路径被阻塞.

由因子分解式：$P(a,b,c) = P(a)P(b|a)P(c|a)$

由链式法则：$P(a,b,c) = P(a)P(b|a)P(c|a,b)$

$\Rightarrow P(c|a) = P(c|a,b) \Rightarrow$ b 与 c 独立.

$\Updownarrow$

$P(c|a)P(b|a) = P(c|a,b)P(b|a) = P(b,c|a)$

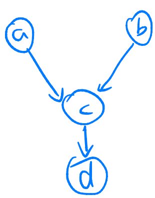
$P(c|a) \cdot P(b|a) = P(b,c|a) \Rightarrow$ b 与 c 独立

(head to tail)

$a \perp c | b$

$\Updownarrow$

若 b 被观测，则路径被阻塞.



(head to head)

a 和 b 先天独立，若 c 被观测，则路径是通的. (c 的后续节点被观测，路径也是通的.

$P(a,b,c) = P(a)P(b)P(c|a,b)$

$P(a,b,c) = P(a)P(b|a)P(c|a,b)$

$\Rightarrow P(b) = P(b|a) \Rightarrow$ a, b 独立

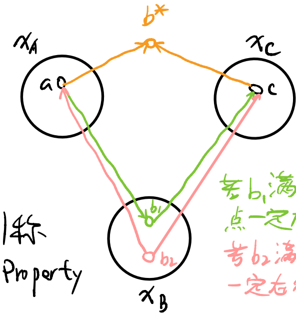
若 b^* 满足 head to head，则该节点不能在 x_B 内部，包括其后继节点

D-Sepration (D划分)

划分规则:

① head to tail　　tail to tail

② head to head



若 b_1 满足 head to tail，则该节点一定在 x_B 内部

若 b_2 满足 tail to tail，则该节点一定在 x_B 内部

这种划分规则称为全局 Markov Property

除 x_i 外所有节点。

$$P(x_i|x_{-i}) = \frac{p(x_i, x_{-i})}{p(x_{-i})} = \frac{p(x)}{\int_{x_i} p(x)\,dx_i} = \frac{\prod_{j=1}^{p} p(x_j|x_{pa(j)})}{\int_{x_i} \prod_{j=1}^{p} p(x_j|x_{pa(j)})\,dx_i}$$

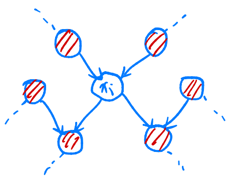
其中与 x_i 无关的信息可以提到积分号前面，与分子中的 x_i 无关项约掉

将与 x_i 有关的设为 Δ，与 x_i 相关的设为 $\boxed{\Delta}$，

则 $P(x_i|x_{-i}) = f(\boxed{\Delta})$

$p(x_i|x_{pa(i)})$

$p(x_{child(i)}|x_i, x_{parent(child(i))})$



马尔可夫毯 (Markov Blanket)

Bayesian Network

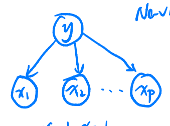
Bayesian Network

- 单一: $\boxed{\text{Naive Bayes}}$ → $P(x|y) = \prod_{i=1}^{p} P(x_i|y=1)$
- 混合: $\boxed{\text{高斯混合模型}}$ (GMM, 用于聚类)
- 时间: $\boxed{\text{马尔可夫链}}$, $\boxed{\text{高斯过程}}$ (无限维高斯分布) → $\boxed{\text{动态模型}}$: $\boxed{\text{HMM}}$ $\boxed{\text{LDS}}$ (Kalman Filter) $\boxed{\text{Particle Filter}}$
- 连续: $\boxed{\text{Gaussian Bayesian Network}}$

从单一到混合

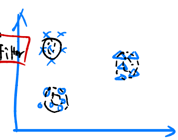
从有限到无限: 空间 (离散→连续)、时间

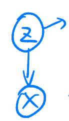
Navie Bayes.



$x_i \perp x_j | y$

GMM





z 是离散的 $z = 1, 2, \cdots, k$

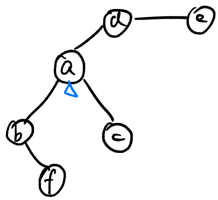
$x|z \sim N(\mu, \Sigma)$

三、马尔可夫随机场(马尔可夫网络)(无向图)

① $x_A \perp x_C \mid x_B$. x_A 到 x_C 路径上至少有一个点在 x_B 中，则 x_A 与 x_C 条件独立.

全局马尔可夫

② 局部马尔可夫



$a \perp \{\text{全集} - a - \text{邻居}\} \mid \text{邻居}$

$a \perp \{e, f\} \mid \{b, c, d\}$

③ 成对马尔可夫

$x_i \perp x_j \mid x_{-i-j} \quad (i \neq j)$

条件独立性体现在以上三个方面 ①全局 ②局部 ③成对.

且 ① $\Leftrightarrow$ ② $\Leftrightarrow$ ③

因子分解.

团：团是关于节点的集合，集合中的节点之间相互都是连通的.

最大团：再添加任何一个节点就不是团的团.

$P(X) = \frac{1}{Z} \prod_{i=1}^{k} \phi(x_{C_i})$ C_i：最大团

x_{C_i} 最大团随机变量集合

$\phi(x_{C_i})$ 势函数，必须为正.

$Z = \sum_{x} \prod_{i=1}^{k} \phi(x_{C_i}) = \sum_{x_1} \sum_{x_2} \cdots \sum_{x_p} \prod_{i=1}^{k} \phi(x_{C_i})$，归一化因子.

一般来讲 $\varphi(x_{C_i}) = \exp\{-E(x_{C_i})\}$，此时 $P(X)$ 被称做吉布斯分布 (Gibbs) 或玻尔兹曼分布.

能量函数.

$$P(x) = \frac{1}{Z} \prod_{i=1}^{k} \phi(x_{C_i}) = \frac{1}{Z} \prod_{i=1}^{k} \exp\{-E(x_{C_i})\} = \frac{1}{Z} \exp\{-\sum_{i=1}^{k} E(x_{C_i})\}$$

Markov Random Field $\Leftrightarrow$ Gibbs Distribution

Inference 推断

P1 推断的目的就是求概率

求概率. $P(X)=P(x_1,x_2,\cdots x_p)$

边缘概率: $P(x_i)=\sum_{x_1}\sum_{x_2}\cdots\sum_{x_{i-1}}\sum_{x_{i+1}}\cdots\sum_{x_p}P(X)$

条件概率: $P(x_A|x_B)$ $x=x_A\cup x_B$

最大后验概率. MAP Inference: $\hat{z}=\arg\max_z P(z|x)\propto\arg\max_z P(z,x)$

Inference
- 精确推断
  - variable Elimination (VE) 变量消除法
  - $\boxed{\text{Belief Propagation 信念传播 (BP)} \rightarrow \text{Sum-Product Algorithm}}$ (针对树结构)
  - Junction tree Algorithm (普通图结构)
- 近似推断
  - Loop Belief Propagation (有环图)
  - Monte Carlo Inference: Importance Sampling, MCMC
  - Variational Inference 变分推断.

P2. Variable Elimination.

例如有马氏链: ⓐ → ⓑ → ⓒ → ⓓ ? (假设 a,b,c,d 为二值 r.v.) $a,b,c,d\in\{0,1\}$

$$
\begin{aligned}
P(d)&=\sum_{abc}P(a,b,c,d)\\
&=\sum_{a,b,c}P(a)P(b|a)P(c|b)P(d|c)\\
&=P(a=0)P(b=0|a=0)P(c=0|b=0)P(d|c=0)\\
&\quad+P(a=1)P(b=0|a=1)P(c=0|b=0)P(d|c=0)\\
&\quad\vdots\\
&\quad+P(a=1)P(b=1|a=1)P(c=1|b=1)P(d|c=1)\\
&=8\times 因子积
\end{aligned}
$$

可以发现, 此时仅有 a, b, c, d 4个二值变量, 若有 N个变量, 每个变量取值为K, 则共 K^p 项

P2. Variable Elimination

例如有马氏链： $a \rightarrow b \rightarrow c \rightarrow d$ (假设a,b,c,d为二值r.v.) $a,b,c,d \in \{0,1\}$

$$P(d) = \sum_{a,b,c} P(a,b,c,d)$$

$$= \sum_{a,b,c} P(a)P(b|a)P(c|b)P(d|c)$$

$$= P(a=0)P(b=0|a=0)P(c=0|b=0)P(d|c=0)$$

$$+ P(a=1)P(b=0|a=1)P(c=0|b=0)P(d|c=0)$$

$$\vdots$$

$$+ P(a=1)P(b=1|a=1)P(c=1|b=1)P(d|c=1)$$

$$= 8 \times \text{因子积}$$

$ab+ac=a(b+c)$

可以发现，此时仅有a,b,c,d 4个二值变量，若有N个变量，每个变量取值为K，则共K^N项

观察式子，并不是所有的因子都与某个变量相关，因此，可以写为：

$$= \sum_{b,c} P(c|b)\cdot P(d|c) \cdot \underbrace{\sum_a P(a)P(b|a)}_{m_a(b)}$$

$$= \sum_c P(d|c) \underbrace{\sum_b P(c|b)\, m_a(b)}_{m_b(c)}$$

$$= \sum_c P(d|c)\, m_b(c)$$

$$= m_c(d)$$

这种思想可以看做是乘法对加法的分配律

对于无向图：

$P(X) = \frac{1}{Z}\prod_{i=1}^{k} \phi_{c_i}(x_{c_i})$, $\frac{1}{Z}\phi_{c_i}(x_{c_i})$ 可以作为因子.

缺点：① 重复计算，计算步骤无法存储，每次计算一个边缘概率就要重新计算一遍整个图。

② ordering，变量消除次序，是一个NP-hard问题

P3. VE To Belief Propagation

$$a \xrightarrow{m_{a\to b}(b)} b \xrightarrow{m_{b\to c}(c)} c \xrightarrow{m_{c\to d}(d)} d \xrightarrow{m_{d\to e}(e)} e$$

计算P(e):

$$P(e)=\sum_{a,b,c,d}P(a,b,c,d,e)=\underbrace{\sum_d P(e|d)\underbrace{\sum_c P(d|c)\underbrace{\sum_b P(c|b)\underbrace{\sum_a P(b|a)P(a)}_{m_{a\to b}(a)}}_{m_{b\to c}(c)}}_{m_{c\to d}(d)}}_{m_{d\to e}(e)}$$

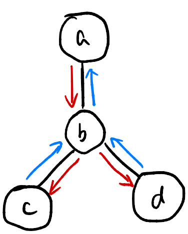
计算P(c):

$$P(c)=\sum_{a,b,d,e}P(a,b,c,d,e)=\sum_{a,b,c,d}P(a)P(b|a)P(c|b)P(d|c)P(e|d)$$
$$=\left(\sum_b P(c|b)\sum_a P(b|a)P(a)\right)\cdot\left(\sum_d P(d|c)\sum_e P(e|d)\right)$$

可以发现在计算P(c)时，前半部分的计算与P(e)有重复，若计算其它边缘概率，也会有重复。
而BP就是用来解决这个问题。

Belief Propagation引出

上面我们一直计算的是有向图的Markov Chain，现在将问题从链结构引申到树结构，
从有向图引申到无向图，例如有如下无向树：



其联合概率因子分解可以写为:

$$P(a,b,c,d)=\frac{1}{Z}\psi_a(a)\psi_b(b)\psi_c(c)\psi_d(d)\psi_{ab}(a,b)\psi_{b,c}(b,c)\psi_{b,d}(b,d)$$

我们只需要求6个边，就能求出图中任意变量的边缘概率

$$P(a)=\sum_{b,c,d}P(a,b,c,d)=\psi_a\underbrace{\sum_b\psi_b\cdot\psi_{ab}\underbrace{\left(\sum_c\psi_c\cdot\psi_{b,c}\right)}_{m_{c\to b}(b)}\cdot\underbrace{\left(\sum_d\psi_d\cdot\psi_{bd}\right)}_{m_{d\to b}(b)}}_{m_{b\to a}(a)}$$

$$\left\{\begin{array}{l} m_{b\to a}(x_a) = \sum_{x_b} \psi_{ab}\cdot\psi_b\cdot m_{c\to b}(x_b)\cdot m_{d\to b}(x_b) \\ P(x_a) = \psi_a\, m_{b\to a}(x_a) \end{array}\right.$$

通项

$$\left\{\begin{array}{l} m_{j\to i}(x_i) = \sum_{x_j} \psi_{ij}\cdot\psi_j\cdot \prod_{k\in NB(j)-i} m_{k\to j}(x_j) \\ P(x_i) = \psi_i\cdot \prod_{k\in NB(i)} m_{k\to i}(x_i) \end{array}\right.$$

☆不要直接求边缘概率($P(a), P(b), \cdots$),只需求所有的$m_{i\to j}$即可.

这就是BP要做的.

$$m_{b\to a} = \sum_b \psi_{a,b}\cdot \underbrace{\underbrace{\psi_b}_{self}\cdot \underbrace{m_{c\to b}\cdot m_{d\to b}}_{children}}_{belief(b)}$$

$$\left\{\begin{array}{l} belief(b) = \psi_b\cdot children \\ m_{b\to a} = \sum_b \psi_{ab}\cdot belief(b) \end{array}\right.$$

belief(b)是b所能把握的信息量.

信念传播算法首先求所有的信息传递(收集或分发)的过程得到所有的$m_{j\to i}$(图的遍历),然后套用公式计算边缘概率,即 BP = VE + Caching

BP遍历图的一种方法(Sequential Implementation)如下:

① 选择一个节点作为根节点,假定为a.

② 收集消息(即图中蓝线):

```
for xi in Neighbor(Root):
    collectMsg(xi)
```

③ 分发消息(即图中红线):

```
for xi in Neighbor(Root):
    distributeMsg(xi)
```

遍历完成后可得所有$m_{i\to j}$,则$P(x_k)$可求.