8 图模型

使用概率分布的图形表示被称为概率图模型，该模型提供了几个有用的性质：

- 提供了简单的概率模型可视化的简单方式，可用于设计新的模型
- 通过观察模型，可以深刻认识模型的性质，包括条件独立性质
- 高级模型的推断和学习过程中的复杂计算可以根据图计算表达，图隐式承载了背后的数学表达式。

在概率图模型中，每个结点表示一个随机变量，链接表示这些变量之间的概率关系，图描述了联合概率分布在所有随机变量上能够分解为一组因子的乘积的方式，每个因子只依赖于随机变量的一个子集。

- 贝叶斯网络，也叫有向图模型，图之间的连接有特定的方向，使用箭头表示。
- 马尔科夫随机场，也叫无向图模型，连接没有箭头，没有方向性质

有向图对于表达随机变量之间的因果关系是很有用的，而无向图对于表示随机变量之间的软限制比较有用。为了求解推断问题，较方便的方法是将有向图和无向图都转化为不同的表示形式，被称为因子图。

8.1 贝叶斯网络

首先对于三个变量a, b, c上的任意的一个联合分布$p(a, b, c)$。他们可以是离散或连续的，图模型的强大之处在于一个具体的图可以表示一大类的概率分布，可以写成下面的形式：

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

可以表示为下图

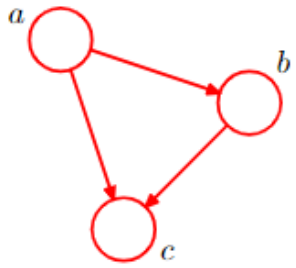

图 8.1: 一个有向图模型，表示三个变量a, b, c上的联合概率分布，对应于公式（8.2）右侧的分解。

如果存在一个从结点a到结点b的链接，那么我们说结点a是结点b的父节点，节点b是a的子节点。
我们可以将其拓展到K个变量的联合概率分布$p(x_1, \cdots, x_K)$,可以表示为：

$$p(x_1, \cdots, x_K) = p(x_K|x_1, \cdots, x_{K+1}) \cdots p(x_2|x_1)p(x_1)$$

对于一个给定的K，我们可以将其表示为K个节点的有向图，每一个节点对英语公式右侧的一个条件概率分布，每一个节点的输入链接包括所有以编号低于当前节点编号为起点的链接，我们说这个图是全连接的，因为每对节点之间都存在一个链接。
而真正传递出图表示的概率分布的性质的信息是图中链接的缺失。
我们对于图8.2写出其对应的联合概率密度表达式：

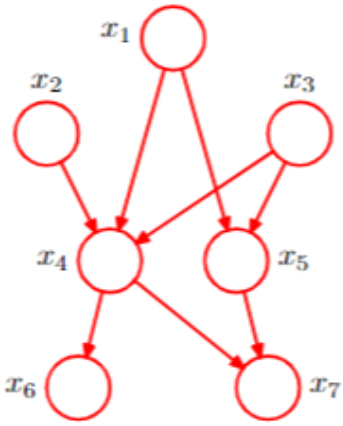

图 8.2: 有向无环图描述变量$x_1, \ldots, x_7$。联合概率分布的对应的概率分解由公式（8.4）给出。

$$p(x_1)p(x_2)p(x_3)p(x_4|x_1,x_2,x_3)p(x_5|x_1,x_3)p(x_6|x_4)p(x_7|x_4,x_5)$$

所以在给定的有向图和变量上对应的一般关系：
在图的所有节点上定义的联合概率分布由一系列的条件概率的乘积组成，每一项对应图中的一个节点。每一个这样的条件概率分布的条件都是图中对应节点的父节点所对应的变量。因此对于有K个节点的图，联合概率为：

$$p(x) = \prod_{k=1}^{K} p(x_k|\mathbf{pa_k})$$

其中$\mathbf{pa_K}$表示节点x_k的父节点的集合，$x = \{x_1, \cdots, x_K\}$。
这个关键的方程表示有向图模型的联合概率分布的分解属性，我们可以很容易地推广到让图的每一个节点关联一个变量的集合，或者关联向量值的变量，如果右侧的每一个条件概率分布都是归一化的，则这个表示方法整体总是归一化的。
我们考虑的有向图要满足一个重要的限制：不能存在有向环，所以这样的图一定是有向无环图，或者**DAG**。

8.1.1 例子：多项式回归

对于一个贝叶斯多项式拟合模型，随机变量为多项式系数向量w和观测数据$\mathbf{t} = (t_1, \cdots, t_N)^T$。此外，该模型包含输入数据$\mathbf{x} = (x_1, \cdots, x_N)^T$,噪声方差$\sigma^2$以及表示w的高斯先验分布的精度的超参数$\alpha$,这些都是模型的参数而非随机变量，所以我们看到联合概率分布等于先验分布与N个条件概率分布的乘积：

$$p(\mathbf{t}|w) = p(w)\prod_{n=1}^{N} p(t_n|w)$$

我们在图模型中引入图结构称为板，标记为N，表示有N个同类型的点。
可以显式写出模型的参数和随机变量：

$$p(\mathbf{t}, w|\mathbf{x}, \alpha, \sigma^2) = p(w|\alpha) \prod_{n=1}^{N} p(t_n|w, x_n, \sigma^2)$$

我们可以在图表示中显示写出$\mathbf{x}, \alpha$,随机变量可以用空心圆表示，确定性参数由小的实心圆表示。
当我们使用图模型于模式识别问题当中时，我们将某些随机变量设置为具体的值，比如将变量$\{t_n\}$根据多项式曲线拟合问题中的训练集进行设置。在图模型中，通过给特定的节点加上阴影来表示观测变量。

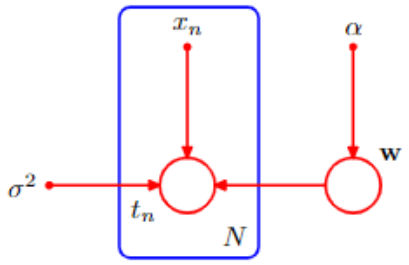

图 8.5: 本图给出了与图8.4相同的模型，但是显式地画出了确定性参数，用小的实心圆表示。

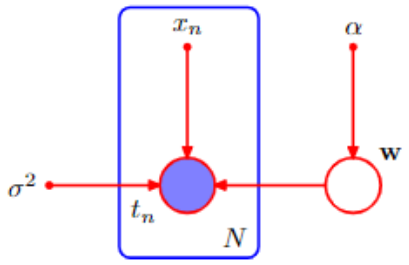

图 8.6: 与图8.5相同，但是结点$\{t_n\}$被标记为阴影，表示对应的随机变量被设置成它们在训练集里的观测值。

图中的例子里，w不是观测变量，而是潜在变量的一个例子。
当观测到$\{t_n\}$的值，如果有必要的话，可以计算参数w的后验概率：

$$p(w|\mathbf{t}) \propto p(w) \prod_{n=1}^{N} p(t_n|w)$$

其中省略了确定性参数。
一般情况下，我们不关心w这样的参数，而是要对输入变量进行预测，给定一个输入值$\hat{x}$,我们希望得到以观测数据为条件的$\hat{t}$的概率分布。以确定性参数为条件，模型的所有随机变量的联合概率分布为：

$$p(\hat{t}, \mathbf{t}, w|\hat{x}, \mathbf{x}, \alpha, \sigma^2) = \left[\prod_{n=1}^{N} p(t_n|x_n, w, \sigma^2) \right] p(w|\alpha) p(\hat{t}|\hat{x}, w, \sigma^2)$$

然后根据概率的加和原则，对模型参数w进行积分，可以得到$\hat{t}$的预测分布：

$$p(\hat{t}|\hat{x},\mathbf{x},\mathbf{t},\alpha,\sigma^2) \propto \int p(\hat{t},\mathbf{t},w|\hat{x},\mathbf{x},\alpha,\sigma^2)\mathrm{d}w$$

其中我们隐式地将$\mathbf{t}$中的随机变量设置为数据集中观测到的具体值。

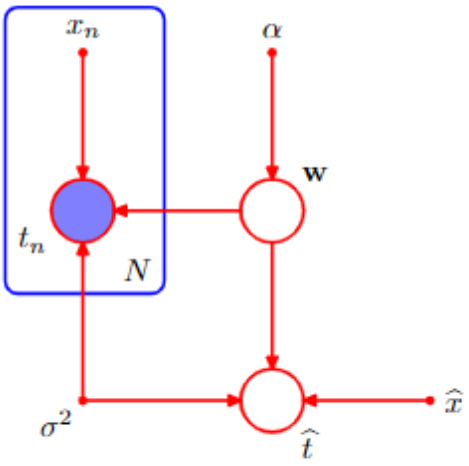

图 8.7: 多项式回归模型，对应于图8.6。同时画出了一个新的输入值$\hat{x}$以及对应的模型精度$\hat{t}$。

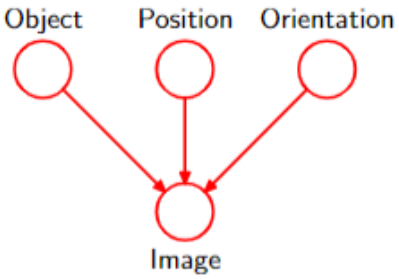

图 8.8: 一个图模型，表示物体的图像的创建过程。其中，物体的种类（一个离散变量）以及物体的位置和方向（连续变量）具有独立的先验概率。图像（一个像素灰度值的向量）的概率分布与物体的种类以及它的位置和方向无关。

8.1.2 生成式模型

我们希望从给定的概率分布中抽取样本，使用图模型可以采用祖先采样的方式：考虑对于K个变量的一个联合概率分布$p(x_1,\cdots,x_N)$,它根据公式$p(x)=\prod_{k=1}^{K}p(x_k|\mathbf{pa_k})$进行分解，对应于一个有向无环图，我们假设每一个节点的序号都大于他的父节点，我们的目标是从这样的联合概率分布中采样$\hat{x}_1,\cdots,\hat{x}_K$。

我们首先可以选出序号最小的节点，按照概率$p(x_1)$进行采样，记作$\hat{x}_1$,然后顺序计算每一个节点，使得对于节点n，可以根据条件概率分布$p(x_n|\mathbf{pa_k})$进行采样，其中父节点的变量被设置为他们的取样值。在每一个阶段，父节点的变量总是可以得到的。因为它们对应于已采样过的序号较小的节点。一旦我们对最后的变量x_K采样完毕，我们就达到了根据联合概率密度分布取样的目标。为了得到对应于变量的子集的边缘概率分布中进行取样，我们简单的对联合概率分布进行取样，简单地取要求的节点的取样值，忽略剩余节点的取样值。例如，为了对于$p(x_2,x_4)$进行取样，我们简单地对联合概率分布取样，

然后保留$\hat{x}_2, \hat{x}_4$，丢弃剩余的值$\{\hat{x}_{j\neq\{2,4\}}\}$.
对于概率模型的实际应用，通常的情况是，数量众多的变量对应于图的终端结点（表示观测值），较少的变量对应于潜在变量。潜在变量的主要作用是使得观测变量上的复杂分布可以表示为由简单条件分布（通常是指数族分布）构建的模型。
图模型描述了生成观测数据的一种因果关系（causal）过程,因此这种模型也被称为生成式模型。之前描述的多项式回归模型不是生成式模型，因为没有与输入变量x相关联的概率分布，因此无法从该模型中人工生成数据点。通过引入合适的先验概率分布$p(x)$，我们可以将模型转变为生成式模型，代价是增加了模型复杂度。
概率模型中的隐含变量不必具有显式的物理含义。它的引入可以仅仅为了从更简单的成分中建立一个更复杂的联合概率分布。在任何一种情况下，应用于生成式模型的祖先取样方法都模拟了观测数据的创造过程，因此可以产生“幻想的”数据，它的概率分布（如果模型完美地表示现实）与观测数据的概率分布相同。在实际应用中，从一个生成式模型中产生人工生成的观测数据，对于理解模型所表示的概率分布形式很有帮助。

8.1.3 离散变量

如果将有向图中所有的父节点-子节点对的关系选为共轭的，那么这样的模型会有一些较好的性质，如果父节点和子节点都是对应于离散变量的，对于有K个可能状态的一元离散变量x,概率$p(x|\mu) = \prod_{k=1}^{K} \mu_k^{x_k}$,并且由参数$\mu = (\mu_1, \cdots, \mu_K)^T$控制。由于$\sum_k \mu_k = 1$,所以为了定义概率分布，只需要指定$K-1$个$\mu_k$的值即可。
如果对于两个离散变量x_1, x_2,每一个都有K个状态，联合概率分布为：

$$p(x_1, x_2|\mu) = \prod_{k=1}^{K}\prod_{l=1}^{K} \mu_{kl}^{x_{1k}x_{2l}}$$

由于参数μ_{kl}满足限制条件$\sum_k \sum_l \mu_{kl} = 1$,所以该分布由$K^2-1$个参数控制，所以对于M个变量的任意一个联合概率分布，需要确定的参数数目为$K^M - 1$,因此，随M的数量指数增长。我们使用$p(x_1, x_2) = p(x_2|x_1)p(x_1)$,它对应于一个只有两个节点的图。边缘概率分布$p(x_1)$需要K-1个参数确定，类似的，对于条件概率$p(x_2|x_1)$需要对$x_1$的K个取值，分别指定K-1个参数，因此联合概率分布的参数总数值为：$(K-1) + K(K-1) = K^2 - 1$.
现在假设两个变量之间是独立的，这样每一个变量由一个独立的多项式概率分布描述得到，参数总数为$2(K-1)$,对于M个独立的离散变量上的概率分布，参数总数为$M(K-1)$，因此参数总数随着变量数量线性增长，所以从图的角度来看，我们通过删除节点之间的链接，减小了参数的数目，代价是类别的概率分布受到限制。
更一般的，对于M个离散变量，可以使用有向图对联合概率分布建模，每一个变量对应于一个节点，如果图是全连接的，那么得到了完全一般的概率分布，由$K^M - 1$个参数，若不存在链接，则参数总数为$M(K-1)$，为了减小参数数目，可以使用参数共享的方式。

图 8.10: M个离散结点组成的链，每个结点有K个状态，要求指定$K-1+(M-1)K(K-1)$个参数，它随着链的长度M线性增长。相反，M个结点的一个完全连接的图具有K^M-1个参数，它随着M指数增长。

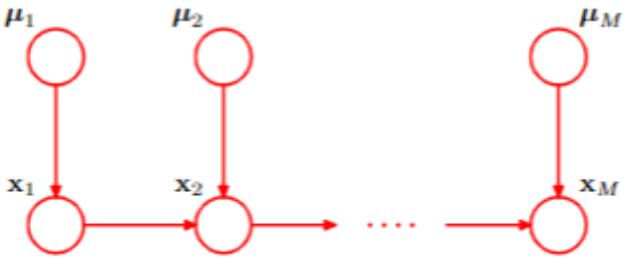

图 8.11: 图8.10的模型的扩展，包含了控制离散分布的参数的狄利克雷先验分布。

通过引入参数的狄利克雷先验，我们可以将离散变量上的图模型转化为贝叶斯模型。从图的观点来看，每个结点需要额外的父结点表示对应于每个离散结点的参数。

另一种控制离散变量模型参数数量的指数增长的方式是对条件概率分布使用参数化的模型，而不使用条件概率值的完整表格。

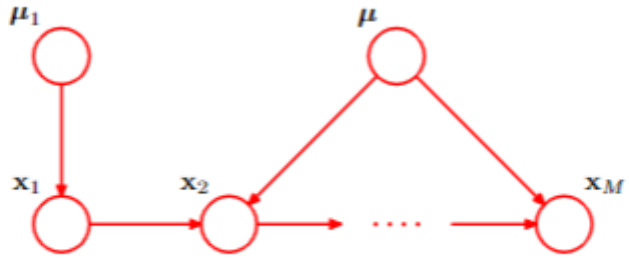

图 8.12: 与图8.11相同，但是所有的条件概率分布$p(\boldsymbol{x_i} \mid \boldsymbol{x_{i-1}})$共享一个单一的参数$\boldsymbol{\mu}$的集合。

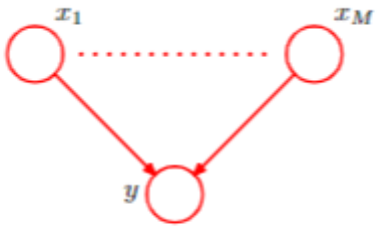

图 8.13: 一个由M个父结点$x_1,\ldots,x_M$和一个单一子结点y组成的图，用来说明离散变量的参数化条件概率分布的思想。

8.1.4 线性高斯模型

对于D个变量上的任意的有向无环图，其中结点i表示服从高斯分布的一元连续随机变量x_i。这个分布的均值是结点i的父结点$\mathbf{pa}_i$的状态的线性组合，即:

$$p(x_i|\mathbf{pa}_i)=\mathcal{N}\left(x_i\Big|\sum_{j\in\mathbf{pa}_i}w_{ij}x_j+b_i,v_i\right)$$

所以联合概率分布的对数是图中所有节点上的条件概率分布的乘积的对数，形式为：

$$\begin{aligned}\ln p(x)&=\sum_{i=1}^{D}\ln p(x_i|\mathbf{pa_i})\\&=-\sum_{i=1}^{D}\frac{1}{2v_i}\left(x_I-\sum_{j\in\mathbf{pa_i}}w_{ij}x_i-b_i\right)^2+\text{const}\end{aligned}$$

所以联合概率分布是一个多元高斯分布，我们可以递归确定联合概率分布的均值和方差，每一个变量的概率分布都是高斯分布，因此：

$$x_i=\sum_{j\in\mathbf{pa_i}}w_{ij}x_j+b_i+\sqrt{v_i}\epsilon_i$$

其中ϵ_i是一个零均值单位方差的高斯随机变量，满足$\mathbb{E}[\epsilon_i]=0,\mathbb{E}[\epsilon_i\epsilon_j]=I_{ij}$,对上面的式子取期望，可以得到：

$$\mathbb{E}[x_i]=\sum_{j\in\mathbf{pa}_i}w_{ij}\mathbb{E}[x_j]+b_i$$

所以从一个序号最低的节点开始，沿着图递归进行计算，就可以求出$\mathbb{E}[x]$的各个元素，同样地，我们可以以递归的方式得到$p(x)$的协方差矩阵的第i,j个元素，即：

$$\begin{aligned}\mathbf{cov}(x_i,x_j)&=\mathbb{E}[(x_i-\mathbb{E}[x_i])]\mathbb{E}[(x_j-\mathbb{E}[x_j])]\\&=\mathbb{E}\left[(x_i-\mathbb{E}[x_i])\Big\{\sum_{j\in\mathbf{pa_j}}w_{jk}(x_k-\mathbb{E}[x_k])+\sqrt{v_j}\epsilon_j\Big\}\right]\\&=\sum_{k\in\mathbf{pa_j}}w_{jk}\mathbf{cov}[x_i,x_k]+I_{ij}v_j\end{aligned}$$

因此协方差可以从序号最低的结点开始，递归地计算。
所以对于极端情况，如果图中不存在链接，在这种情况下不存在参数w_{ij}，只有D个参数b_i和D个参数v_i,所以根据递归关系，可以得到$p(x)$的均值为$(b_1,\cdots,b_D)^T$,协方差矩阵为对角矩阵$\text{diag}(v_1,\cdots,v_D)$。联合概率分布总共有$2D$个参数，表示由D个独立一元高斯分布组成的集合。
对于全连接图，由于每一个节点的序号都低于其父节点的序号。w_{ij}的数量为$\frac{D(D-1)}{2}$,加上协方差矩阵

的参数个数，总共$\frac{D(D+1)}{2}$,对应于一个一般的对称协方差矩阵。
所以对于图8.14中的变量，使用递归关系，可以得到联合高斯分布的均值和协方差为：

$$\mu = (b_1, b_2 + w_{21}b_1, b_3 + w_{32}b_2 + w_{32}w_{21}b_1)^T$$
$$\Sigma = \begin{pmatrix} v_1 & w_{21}v_1 & w_{32}w_{21}v_1 \\ w_{21}v_1 & v_2 + w_{21}^2 v_1 & w_{32}(v_2 + w_{21}^2 v_1) \\ w_{32}w_{21}v_1 & w_{32}(v_2 + w_{21}^2 v_1) & v_3 + w_{32}^2(v_2 + w_{21}^2 v_1) \end{pmatrix}$$

因此节点i的条件概率分布形式为：

$$p(x_i|\mathbf{pa_i}) = \mathcal{N}\left(x_i \middle| \sum_{j \in \mathbf{pa_i}} W_{ij}x_j + b_i, \Sigma_i\right)$$

注意，我们已经看到高斯变量x的均值μ的共轭先验本身是μ上的一个高斯分布。此时我们已经遇到了线性高斯关系的一个具体的例子。因此x和μ的联合分布就是高斯分布。这对应于一个简单的具有两个结点的图，其中表示μ和结点是表示x的结点的父结点。μ上的概率分布的均值是控制先验分布的参数，因此它可以被看做超参数。由于超参数的值本身是未知的，因此我们可以再一次使用贝叶斯的观点，引入一个超参数上的先验概率分布。这个先验概率分布有时被称为超先验（hyperprior），它还是一个高斯分布。这种构造过程原则上可以延伸到任意层次。这个模型是层次贝叶斯模型（hierarchical Bayesian model）的一个例子.

8.2 条件独立

多变量概率分布的一个重要的概念就是条件独立，考虑变量a, b, c，并且假设在给定b, c的条件下a的条件概率分布不依赖于b的值，即：

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

则在给定c的条件下，a条件独立于b。如果考虑以c为条件下的a，b的联合分布，则：

$$\begin{aligned} p(a, b|c) &= p(a|b, c)p(b|c) \\ &= p(a|c)p(b|c) \end{aligned}$$

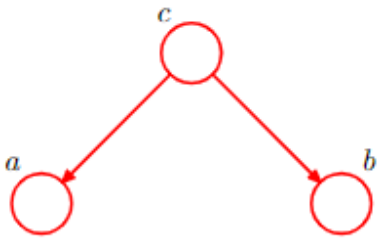

图 8.15: 三个变量a, b, c上的图模型的三个例子中的第一个，这些例子用来讨论有向图模型的条件独立性质。

因此，我们看到了，以c为条件，a和b的联合概率分布分解为了a的边缘概率分布和b的边缘概率分布的乘积（全部以c为条件）。
如果这性质对于c的所有可能值都成立，则，可用记号：

$$a \perp\!\!\!\perp b|c$$

表示在给定c的条件下，a和b独立。
在使用概率模型时，条件独立性起着重要的作用：简化了模型，降低了模型的训练和推断的计算量。
如果一组变量的联合概率分布的表达式是根据条件概率分布的乘积表示的（即有向图的数学表达形式），那么原则上我们可以通过重复使用概率的加和规则和乘积规则测试是否具有潜在的条件独立性。
在实际应用中，这种方法非常耗时。图模型的一个重要的优雅的特征是，联合概率分布的条件独立性可以直接从图中读出来，不用进行任何计算。我们可以使用"d-划分"来完成该应用，其中"d"表示“有向”

8.2.1 图的三个例子

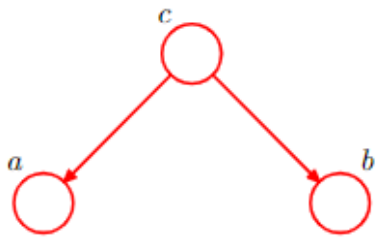

图 8.15: 三个变量a, b, c上的图模型的三个例子中的第一个，这些例子用来讨论有向图模型的条件独立性质。

对于这个图的联合概率分布形式：

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

如果没有变量是观测变量，则可以对两侧进行积分或求和，考察a和b是否独立：

$$p(a,b) = \sum_c p(a|c)p(b|c)p(c)$$

一般的，不能分解为$p(a)p(b)$,所以：

$$a \not\perp\!\!\!\perp b|\emptyset$$

在给定c的条件下：

$$\begin{aligned} p(a,b|c) &= \frac{p(a,b,c)}{p(c)} \\ &= p(a|c)p(b|c) \end{aligned}$$

所以：

$$a \perp\!\!\!\perp b|c$$

结点c被称为关于这个路径“尾到尾”（tail-to-tail），因为结点与两个箭头的尾部相连。
这样的一个连接结点a和结点b的路径的存在使得结点相互依赖。然而，当我们以结点c为条件时（如图8.16所示），被用作条件的结点“阻隔”了从a到b的路径，使得a和b变得（条件）独立了。

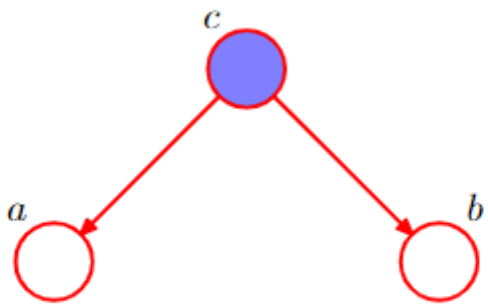

图 8.16: 与图8.15相同，但是我们以变量c为条件。

对应于图8.17的联合概率分布：

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

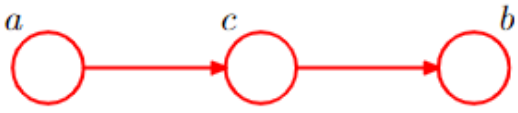

图 8.17: 3结点图的三个例子中的第二个，这些例子用来说明有向图模型的条件独立框架。

首先，如果所有变量都不是观测变量，对c积分或求和：

$$p(a,b) = p(a)\sum_c p(c|a)p(b|c) = p(a)p(b|a)$$

这不能分解为$p(a)p(b)$，所以：

$$a \not\perp\!\!\!\perp b|\emptyset$$

如果以c为条件，使用贝叶斯定理：

$$\begin{aligned} p(a,b|c) &= \frac{p(a,b,c)}{p(c)} \\ &= \frac{p(a)p(c|a)p(b|c)}{p(c)} \\ &= p(a|c)p(b|c) \end{aligned}$$

所以：

$$a \perp\!\!\!\perp b|c$$

与之前一样，我们可以用图表示这个结果。结点c被称为关于从结点a到结点b的路径“头到尾”（head-to-tail）。这样的一个路径连接了结点a和结点b，并且使它们互相之间存在依赖关系。如果我们现在观测结点c，如图8.18所示，那么这个观测“阻隔”了从a到b的路径，因此我们得到了条件独立性质$a \perp\!\!\!\perp b|c$。

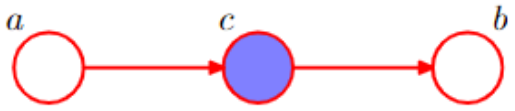

图 8.18: 与图8.17相同，但是现在以c为条件。

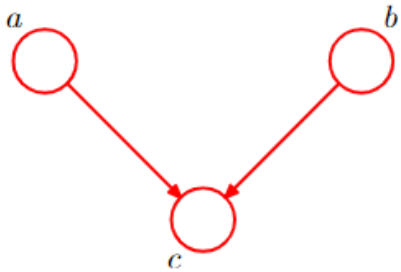

图 8.19: 3结点图的三个例子中的最后一个例子，这些例子用来研究图模型中的条件独立性质。这张图与前两个例子的性质相当不同。

对于8.19可以得到联合概率分布：

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

当考虑没有变量是观测变量时，我们有：

$$p(a,b) = p(a)p(b)$$

所以

$$a \perp\!\!\!\perp b|\emptyset$$

当以c为条件时，可以得到：

$$\begin{aligned} p(a,b|c) &= \frac{p(a,b,c)}{p(c)} \\ &= \frac{p(a)p(b)p(c|a,b)}{p(c)} \end{aligned}$$

这通常不能被分解为乘积$p(a|c)p(b|c)$,所以：

$$a \not\perp\!\!\!\perp b|c$$

图形上，我们说结点c关于从a到b的路径是“头到头”（head-to-head），因为它连接了两个箭头的头。当结点c没有被观测到的时候，它“阻隔”了路径，从而变量a和b是独立的。然而，以c为条件时，路径被“解除阻隔”，使得a和b相互依赖了。
我们引入一些新的概念。如果存在从结点x到结点y的一条路径，其中路径的每一步都沿着箭头的方向，那么我们说结点y是结点x的后继（descendant）。这样，可以证明，在一个头到头的路径中，如果任意结点或者它的任意一个后继被观测到，那么路径会被“解除阻隔”。
总之，一个尾到尾结点或者头到尾结点使得一条路径没有阻隔，除非它被观测到，之后它就阻隔了路径。相反，一个头到头结点如果没有被观测到，那么它阻隔了路径，但是一旦这个结点或者至少一个后继被观测到，那么路径就被“解除阻隔”了。

8.2.2 d-划分

考虑一个一般的有向图，其中A, B, C是任意无交集的结点集合（它们的并集可能比图中结点的完整集合要小）。
为了知道一个有向无环图是否暗含了一个特定的条件依赖$A \perp\!\!\!\perp B|C$,我们考虑从A中任意结点到B中任意结点的所有可能的路径。我们说这样的路径被“阻隔”，如果它包含一个结点满足下面两个性质中的任何一个。

- 路径上的箭头以头到尾或者尾到尾的方式交汇于这个结点，且这个结点在集合C中。
- 箭头以头到头的方式交汇于这个结点，且这个结点和它的所有后继都不在集合C中。

如果所有的路径都被“阻隔”，那么我们说C把A从B中d-划分开，且图中所有变量上的联合概率分布将会满足$A \perp\!\!\!\perp B|C$。
我们已经看到一个特定的有向图表示将联合概率分布分解为条件概率分布乘积形式的一个具体的分解方式。图也表示一组条件独立的性质，这些性质通过d-划分的方式得到，并且d-划分定理实际上是一个等价于这两个性质的表示。为了让这一点更明显，将有向图想象成滤波器是很有帮助的。
假设我们考虑x上的一个特定的联合概率分布p(x)，其中x对应于图中的（未观测）结点。一个概率分布能够通过滤波器当且仅当它能够用与图对应的公式（8.5）给出的分解方式进行分解。如果我们将变量x的集合上的所有可能的概率分布p(x)输入到滤波器中，那么通过滤波器的概率分布的子集被记作$\mathcal{DF}$，表示有向分解.
我们还可以将图用作另一种滤波器，首先将d-划分准则应用到图中，列出所有得到的条件独立性质，然后只有当一个概率分布满足所有这些性质时才允许通过。如果我们将所有可能的概率分布输入到这一类滤波器中，那么d-划分定理告诉我们，允许通过的概率分布的集合就是$\mathcal{DF}$。
一种极限的情况下，我们有一个全连接的图，它不表示任何的条件独立性质，可以表示给定变量上的任何可能的联合概率分布。集合$\mathcal{DF}$将包含所有可能的概率分布p(x)。在另一种情况下，我们有一个完全非连接的图，即一张不存在任何链接的图。这对应的联合概率分布可以分解为图结点组成的变量上的边缘概率分布的乘积。

对于任意给定的图，分布的集合DF中的概率分布还会具有图中未描述的独立性质。例如，一个完全分解的概率分布总会通过由对应变量组成的任意图结构表示的滤波器。

对于马尔科夫随机毯：对于一个联合概率分布$p(x_1, \cdots, x_D)$,它由一个D节点的有向图表示，使用分解性质，可以得到：

$$\begin{aligned} p(x_i|x_{\{j\neq i\}}) &= \frac{p(x_1, \cdots, x_D)}{\int p(x_1, \cdots, x_D)\mathrm{d}x_i} \\ &= \frac{\prod_k p(x_k|\mathbf{pa}_k)}{\int \prod_k p(x_k|\mathbf{pa}_k)\mathrm{d}x_i} \end{aligned}$$

我们现在观察到任何与x_i没有函数依赖
关系的因子都可以提到x_i的积分外面，从而在分子和分母之间消去。
唯一剩余的因子是节点x_i本身的条件概率分布$p(x_i|\mathbf{pa}_i)$,以及满足下面性质的节点x_k的条件概率分布： 结点x_i在$p(x_k|pa_k)$的条件集合中，即x_i是x_k的父结点。条件概率分布$p(x_i|pa_i)$依赖于结点x_i的父结点，而条件概率分布$p(x_k|pa_k)$依赖于x_i的子结点以及同父结点（co-parents），即那些对应于x_k（而不是x_i）的父结点的变量。由父结点、子结点、同父结点组成的结点集合被称为马尔科夫毯，如图8.26所示。我们可以将结点x_i的马尔科夫毯想象成将x_i与图的剩余部分隔离开的最小结点集合。注意，只包含x_i的父结点和子结点是不够的，因为之前的例子表明，子结点的观测不会阻隔某个结点到同父结点的路径。因此我们必须也观测同父结点。

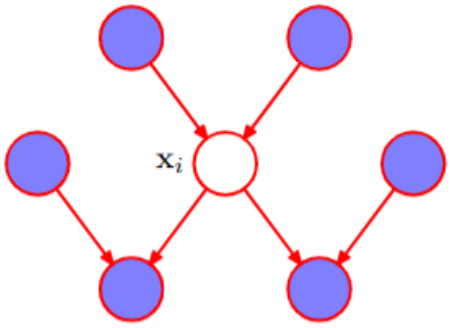

图 8.26: 结点$\boldsymbol{x}_i$的马尔科夫毯由父结点、子结点、同父结点组成的集合构成。它的性质为：以图中所有剩余结点为条件，$\boldsymbol{x}_i$的条件概率分布值依赖于马尔科夫毯中的变量。

8.3 马尔科夫随机场

使用无向图描述的图模型表示一个分解关系，也表示一种条件独立关系。
一个马尔科夫随机场，也叫马尔科夫网络，或者叫无向图模型，包含一组节点，每一个节点都对应一个变量或一组变量，链接无向，不包含箭头。

8.3.1 条件独立性质

通过移除图中链接的方向性，父子节点的非对称性也被移除了，因此不存在头对头这种形式。
为了检验条件独立性质

$$A \perp\!\!\!\perp B | C$$

我们考虑链接A的节点和B的节点的所有可能路径，如果所有路径都通过C中一个或多个节点，那么所有路径都被阻隔，则条件独立性质成立，若至少存在一条未被阻隔的路径，则存在至少某些对应于图的概率分布不满足条件独立性质。注意，这与d划分的准则完全相同，唯一的差别在于没有头到头的现象。
因此，无向图的条件独立性的检测比有向图简单。
无向图的马尔科夫毯的形式相当简单，因为结点只条件依赖于相邻结点，而条件独立于任何其他的结点

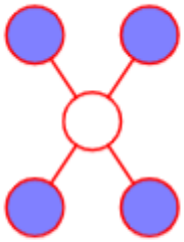

图 8.28: 对于一个无向图，结点x_i的马尔科夫毯由相邻结点的集合组成。它的性质为：以图中所有剩余变量为条件，x_i的条件概率分布只依赖于马尔科夫毯中的变量。

8.3.2 分解性质

考虑两个节点x_i, x_j,他们不存在链接，则给定图中所有的其他节点，这两个节点一定是条件独立的，条件独立性可表示为：

$$p(x_i, x_j|x_{\setminus\{i,j\}}) = p(x_i|x_{\setminus\{i,j\}})p(x_j|x_{\setminus\{i,j\}})$$

于是，联合概率分布的分解一定要让x_i和x_j 不出现在同一个因子中，从而让属于这个图的所有可能的概率分布都满足条件独立性质。

团块 为图中结点的一个子集，使得在这个子集中的每对结点之间都存在链接。所以团块中的节点都是全连接的。此外，一个 最大团块 （maximal clique）是具有下面性质的团块：不可能将图中的任何一个其他的结点包含到这个团块中而不破坏团块的性质。

我们将联合概率分布分解的因子定义为团块中变量的函数，我么们可以考虑最大团块的函数而不失一般性，因为其他团块一定是最大团快的子集。

我们将团块记作C，其中的变量集合记作x_C这样联合概率分布可以写作图的最大团块的势函数$\phi_C(x_C)$的乘积:

$$p(x) = \frac{1}{Z}\prod_C \phi_C(x_C)$$

这里的Z是划分函数，也是归一化常数：

$$Z = \sum_x \prod_C \phi_C(x_C)$$

通过只考虑满足$\phi_C(c_C) \geq 0$的势函数，我们确保$p(x) \geq 0$.

我们不把势函数的选择限制为具有具体的概率含义（例如边缘概率分布或者条件概率分布）的函数。这与有向图的情形相反。在有向图的情形中，每个因子表示对应变量以它的父结点为条件的条件概率分布。然而，在特殊情况下，例如无向图是通过有向图构建的情况，势函数可能确实有这样的意义，正如我们将要看到的那样.

归一化常数的存在是无向图的一个主要的缺点。如果我们有M个离散节点，每个节点有K个状态，则归一化项的计算涉及到对K^M个状态进行求和，因此在最坏情况下，计算量是模型大小的指数形式。对于局部条件概率函数来说，划分函数是不需要计算的，因为涉及到两个边缘函数的比值，类似地，对于计算局部边缘概率，我们可以计算未归一化的联合概率分布，然后在计算的最后阶段显式地归一化边缘概率。假设边缘概率只涉及到少量的变量，那么归一化系数的计算是可行的。

为了给出精确的关系，我们再次回到作为滤波器的图模型的概念中。考虑定义在固定变量集合上的所有可能的概率分布，其中这些变量对应于一个具体的无向图的节点。我们可以将UI定义为满足下面条件的概率分布的集合：从使用图划分的方法得到的图中可以读出条件独立性质，这个概率分布应该与这些条件独立性质相容。类似地，我们可以将UF 定义为满足下面条件的概率分布的集合：可以表示为关于

图中最大团块的分解的形式的概率分布，其中分解方式由公式（8.39）给出。Hammersley-Clifford定理（Clifford, 1990）表明，集合UI和UF是完全相同的。
由于我们的势函数被限制为严格大于零，因此将势函数表示为指数的形式更方便，即:

$$\phi_C(x_C) = \exp\{-E(x_C)\}$$

其中$E(x_C)$被定义为能量函数，指数表示为波尔兹曼分布，联合概率分布被定义为势函数的乘积，总能量即各个最大团块的加和。
与有向图的联合分布的因子不同，无向图中的势函数没有一个具体的概率意义。虽然这使得选择势函数具有更大的灵活性，因为没有归一化的限制，但是这确实产生了一个问题，即对于一个具体的应用来说，如何选择势函数。可以这样做：将势函数看成一种度量，它表示了局部变量的哪种配置优于其他的配置。具有相对高概率的全局配置对于各个团块的势函数的影响进行了很好的平衡。

8.3.4 与有向图的关系

对于一个有向图8.32，有向图的联合概率分布：

$$p(x) = p(x_1)p(x_2|x_1)p(x_3|x_2)\cdots p(x_N|x_{N-1})$$

图 8.32: (a)有向图的例子。(b)等价的无向图。

然后我们将其转化为无向图的表示方法，联合概率分布为：

$$p(x) = \frac{1}{Z}\phi_{1,2}(x_1, x_2)\phi_{2,3}(x_2, x_3)\cdots\phi_{N-1,N}(x_{N-1}, x_N)$$

我们只需要令：

$$\begin{aligned}\phi_{1,2}(x_1, x_2) &= p(x_1)p(x_2|x_1)\\ \phi_{2,3}(x_2, x_3) &= p(x_3|x_2)\\ &\cdots\\ \phi_{N-1,N}(x_{N-1}, x_N) &= p(x_N|x_{N-1})\end{aligned}$$

此时Z=1.

当推广到一般情况时，如果无向图的团块势函数由有向图的条件概率分布给出，那就可以完成，我们要确保出现在每个条件概率分布中变量的集合是无向图中至少一个团块的成员，对于有多个父节点的鸡诶但来说，我们对于这种头对头的路径节点，要在两个父节点之间添加链接，这样去掉箭头后的无向图被叫做道德图。
我们看到从一个有向图表示转化为无向图表示的过程中，我们必须从图中丢弃掉一些条件独立性质。当然，通过简单地使用全连接的无向图，我们可以很容易地将有向图上的任意概率分布转化为无向图上的概率分布。但是，这会丢弃掉所有的条件独立性质，因此没有意义。“伦理”过程增加了最少的额外链接，因此保持了最大的条件独立性质。
对于一个具体的用作滤波器的有向图或者无向图，从而给定变量上的所有可能的概率分布的集合都可以被化简为一个子集，这个子集保持了图给出的条件独立性质。
如果一个概率分布所有的条件独立性质都可以被一个图表现出来，则该图是这个概率分布的D图，因此一个完全非连接的图（不存在链接）是任意概率分布的平凡D图；
我们还可以考虑一个具体的概率分布，判断哪些图具有适当的条件独立性质。如果一个图的每个条件独立性质都可以由一个具体的概率分布满足，那么这个图被称为这个概率分布的I图（I map，表示“独立图”（independence map））。显然，一个完全连接的图是任意概率分布的平凡I图。
如果概率分布的每个条件独立性质都由可以由图反映出来，反之也成立，那么这个图被称为是概率分布的完美图（perfect map）。于是，一个完美图既是I图又是D图。
对于每个概率分布，都可能有有向图的完美图，也可能有无向图的完美图存在，也可能不存在完美图。
图框架可以用一种相容的方式，扩展为同时包含有向链接和无向链接的图。这种图被称为链图（chain graphs）。
虽然与有向图或者无向图自身相比，这种图可以表示更多的概率分布，但是仍然存在概率分布，使得链图也无法给出一个完美图。

8.4 图模型中的推断

对于贝叶斯定理的图表示，将两个变量x和y上的联合概率分布$p(x,y)$表示为$p(x)p(y|x)$，现在假设观测到了y，则将x的边缘概率分布看作是在潜在变量x上的先验概率分布，现在要推断x上对应的后验概率分布,可以得到

$$p(y)=\sum_{x'}p(y|x')p(x')$$

$$p(x|y)=\frac{p(y|x)p(x)}{p(y)}$$

从图模型的角度来看，图中箭头的方向反转了。

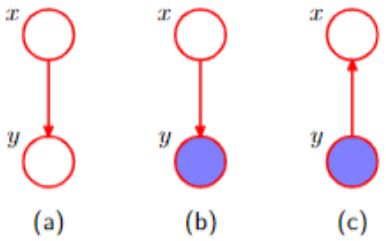

图 8.37: 贝叶斯定理的图表示。详细讨论见正文。

8.4.1 链推断

图 8.32: (a)有向图的例子。(b)等价的无向图。

对于8.32中的节点链，可以看到联合概率分布形式为：

$$p(x) = \frac{1}{Z}\phi_{1,2}(x_1, x_2)\phi_{2,3}(x_2, x_3)\cdots\phi_{N-1,N}(x_{N-1}, x_N)$$

对于具体的情形来说，N个节点，每个变量有K个状态，此时势函数$\phi_{n-1,n}(x_{n-1}, x_n)$是一个K*K的表，因此联合概率分布有$(N-1)K^2$个参数。
对于推断问题，寻找边缘概率分布$p(x_n)$，其中x_n是链上具体的节点，可以通过对联合概率分布在除x_n上的所有变量进行积分或求和得到：

$$p(x_n) = \sum_{x_1}\cdots\sum_{n-1}\sum_{n+1}\cdots\sum_{N}p(x)$$

我们可以计算联合概率分布，然后显式进行求和，x有K^N个可能的值，从而联合概率的计算和存储以及得到p(xn)所需的求和过程，涉及到的存储量和计算量都会随着链的长度N 而指数增长。
我们可以重新整理上式，利用乘法对加法的分配律，得到：

$$p(x_n) = \frac{1}{Z} \underbrace{\left[\sum_{x_{n-1}} \phi(_{n-1,n}(x_{n-1}, x_n)) \cdots \left[\sum_{x_1} \phi_{1,2}(x_1, x_2)\right] \cdots\right]}_{\mu_\alpha(x_n)} \underbrace{\left[\sum_{x_{n+1}} \phi(_{n,n+1}(x_n, x_{n+1})) \cdots \left[\sum_{x_N} \phi_{N-1,N}(x_{N-1}, x_N)\right] \cdots\right]}_{\mu_\beta(x_n)}$$

所以对x_1只涉及到$\phi_{1,2}(x_1, x_2)$，所以计算代价是$O(K^2)$,得到K个数字与$\phi_{2,3}(x_2, x_3)$相乘，代价依然是$O(K^2)$,所以总代价是$O(NK^2)$,如果图是全连接的，那么将不存在条件独立性质，我们就必须直接计算完整的联合概率分布。
根据上面的公式，可以看到:

$$p(x_n) = \frac{1}{Z}\mu_\alpha(x_n)\mu_\beta(x_n)$$

$\mu_\alpha(x_n)$可以看作是从节点x_{n-1}到x_n沿着链向前传播的信息，$\mu_\beta(x_n)$可以看作是从节点x_{n+1}到x_n沿着链向后传播的信息。每条信息由K个值组成，乘积即二者的点积，得到K个值的集合。
μ_α和μ_β都可以递归计算，因为:

$$\begin{aligned}\mu_\alpha(x_n) &= \sum_{x_{n-1}} \phi_{n-1,n}(x_{n-1}, x_n)\left[\sum_{x_{n-2}\cdots}\right] \\ &= \sum_{x_{n-1}} \phi_{n-1,n}(x_{n-1}, x_n)\mu_\alpha(x_{n-1})\end{aligned}$$

归一化常数Z可以很容易地通过对

$$p(x_n) = \frac{1}{Z}\mu_\alpha(x_n)\mu_\beta(x_n)$$

右侧所有状态求和得到。
所以再求每一个节点时的边缘概率时，可以将每一个μ_α, μ_β都求出来之后再另外计算。
如果需要计算$p(x_{n-1,x_n})$,可以得到:

$$p(x_{n-1}, x_n) = \frac{1}{Z}\mu_\alpha(x_{n-1})\phi_{n-1,n}(x_{n-1}, x_n)\mu_\beta(x_n)$$

这是一个很有用的结果，因为在实际应用中，我们可能希望使用团块势函数的参数形式，或者等价地，使用条件概率分布的参数形式（在有向图中）。为了在并非所有的变量都被观测到的情况下学习势函数的参数，我们可以使用EM算法。可以证明，以任意观测数据为条件，团块的局部联合概率分布恰好是E步骤中所需要的。

8.4.2 树

可以看到，由节点链组成的图的精确推断可以在关于节点数量的线性时间内完成，更一般的，通过局部信息在更大的一类图中传递，我们可以高效进行推断，可以对之前在节点链的情形中得到的信息传递公式进行简单推广，得到加和-乘积算法，这为树结构图的精确推断提供了高效的框架。
在无向图的情形中，树被定义为满足下面性质的图：任意一对结点之间有且只有一条路径。
于是这样的图没有环。在有向图的情形中，树的定义为：有一个没有父结点的结点，被称为根（root），其他所有的结点都有一个父结点。如果我们将有向树转化为无向图，我们会看到“伦理”步骤不会增加任何链接，因为所有的结点至多有一个父结点，从而对应的道德图是一个无向树。无向树和有向树的例子如图8.39(a)和8.39(b)所示。注意，一个表示为有向树的概率分布可以很容易地转化为一个由无向树表示的概率分布，反之亦然。

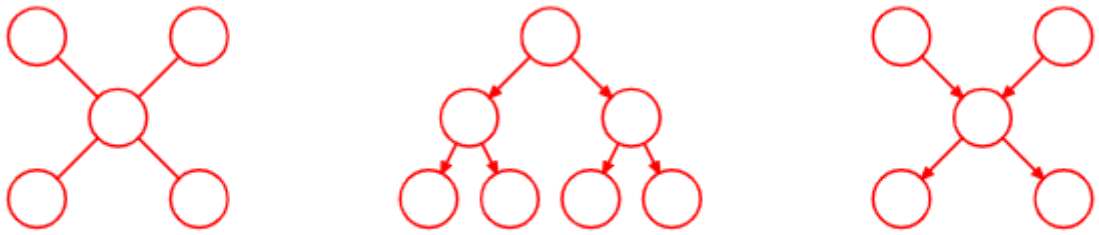

图 8.39: 三个树结构的例子。(a)一个无向树，(b)一个有向树，(c)一个有向多树。

若有向图中有多个父节点的节点存在，但在任意两个节点中仍然只有一条路径，则称为多树，对应的道德图会出现环。

8.4.3 因子图

有向图和无向图都使得若干个变量的一个全局函数能够表示为这些变量的子集上的因子的乘积。因子图显式地表示出了这个分解，方法是：在表示变量的结点的基础上，引入额外的结点表示因子本身。
可以将一组变量上的联合概率分布写成因子乘积形式：

$$p(x) = \prod_s f_s(x_s)$$

其中x_s表示变量的一个子集，将单独的变量记作x_i，每个因子是对应的变量集合x_s的函数。
有向图的分解中，因子$f_s(x_s)$是局部条件概率分布，无向图中因子是最大团块上的势函数，此时归一化

系数而可看作是定义在空变量集合上的因子。
在因子图中，概率分布中的每个变量都有一个结点（与之前一样，用圆圈表示），这与有向图和无向图的情形相同。还存在其他的结点（用小正方形表示），表示联合概率分布中的每个因子$f_s(x_s)$。最后，在每个因子结点和因子所依赖的变量结点之间，存在无向链接。例如，考虑一个表示为因子图形式的概率分布:

$$p(x) = f_a(x_1, x_2) f_b(x_1, x_2) f_c(x_2, x_3) f_d(x_3)$$

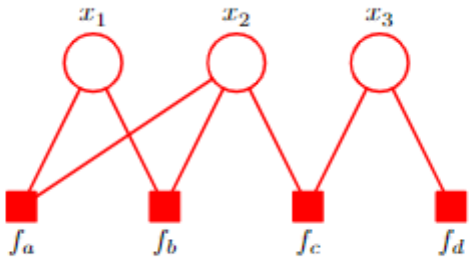

图 8.40: 因子图的例子，对应于公式（8.60）的分解。

这可以表示8.40表示的因子图。
关于8.40所示的因子图若是无向图，则可以将f_a, f_b合并，f_c, f_d合并，但是在因子图中显式表示出这些因子，更能表达出关于分解本身的更细节的信息。
由于因子图由两类节点组成，且所有链接都在两类不同节点之间，所以是一个二部图，对于无向图，可以转化为因子图，可以构造变量节点，对应于原始无向图，然后构造额外的因子节点，对应于最大团块x_s,因子被设置为势函数，对于同一个无向图，可能有多个因子图，如8.42:

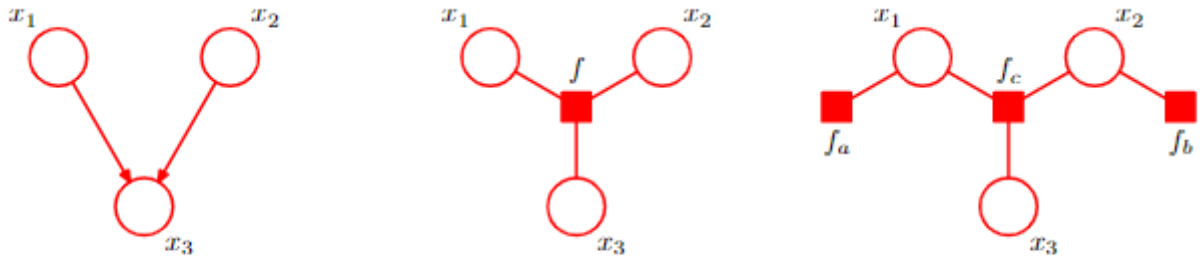

图 8.42: (a)一个有向图，可以分解为$p(x_1)p(x_2)p(x_3 \mid x_1, x_2)$。(b)一个因子图，表示与有向图相同的概率分布，它的因子满足$f(x_1, x_2, x_3) = p(x_1)p(x_2)p(x_3 \mid x_1, x_2)$。(c)一个不同的因子图，表示同样的概率分布，因子为$f_a(x_1) = p(x_1), f_b(x_2) = p(x_2), f_c(x_1, x_2, x_3) = p(x_3 \mid x_1, x_2)$。

而当将树结构图，无论是有向树还是无向树，转化为因子图后还是树（即，因子图没有环，且任意两个结点之间有且只有一
条路径)
在有向多树中，由于存在“伦理”步骤，转化为无向图会引入环，但是转化后的因子图仍然是树，见8.43.

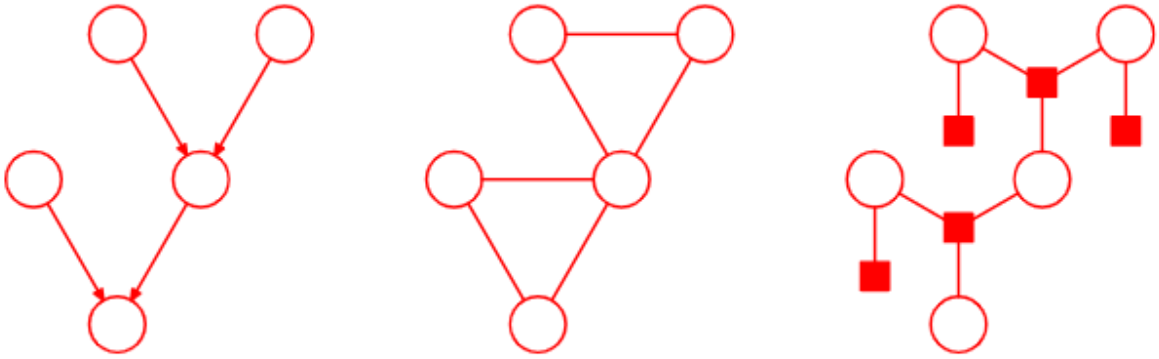

图 8.43: (a)一个有向多树。(b)将多树转化为无向图的结果，展示了环的形成。(c)将多树转化为因子图的结果，保留了树形结构。

事实上，有向图中由于链接父结点和子结点产生的局部环可以在转换到因子图时被移除，只需定义合适的因子函数即可，如图8.44所示

图 8.44: (a)具有局部环的有向图的片段。(b)转化得到的因子图的片段，具有树形结构，其中$f(x_1, x_2, x_3) = p(x_1)p(x_2 \mid x_1)p(x_3 \mid x_1, x_2)$。

8.4.4 加和-乘积算法

为了进行在树结构的图上的精确推断，我们把注意力集中于计算结点或者结点子集上的局部边缘概率分布，这会引出加和-乘积算法（sum-product algorithm）。稍后，我们会修改这个方法，使得概率最大的状态被找到，这就引出了最大加和算法（max-sum algorithm）。
关于有向无环图的精确推断，有一个被称为置信传播（belief propagation）的算法，它等价于加和-乘积算法的一个具体情形。这里，我们只考虑加和-乘积算法，因为它的推导和使用都更容易，并且更一般。我们假设原始的图是一个无向树或者有向树或者多树，从而对应的因子图有一个树结构。首先，我们将原始的图转化为因子图，使得我们可以使用同样的框架处理有向模型和无向模型。
我们的目标是利用图的结构完成两件事：

- （1）得到一个高效的精确推断算法来寻找边缘概率，
- （2）在需要求解多个边缘概率的情形，计算可以高效地共享。

首先，对于特定的变量节点x，我们寻找边缘概率$p(x)$,我们假设所有变量都是隐含变量，根据定义，边沿概率分布通过对所有x之外的变量在进行求和的方式得到：

$$p(x) = \sum_{\mathbf{x}\setminus x} p(\mathbf{x})$$

我们可以用

$$p(\mathbf{x}) = f_a(x_1, x_2) f_b(x_1, x_2) f_c(x_2, x_3) f_d(x_3)$$

来替换$p(\mathbf{x})$然后交换加和和乘积的顺序，得到一个高效的算法，我们看到图的树结构使得我们可以将联合概率分布中的因子划分为若干组，每组对应于变量结点x的相邻结点组成的因子结点集合。我们看到联合概率分布可以写成乘积的形式：

$$p(\mathbf{x}) = \prod_{s\in \mathrm{ne}(x)} F_s(x, X_s)$$

其中$\mathrm{ne}(x)$表示，X_s表示子树中通过因子节点f_s与变量节点x相连的所有变量的集合。$F_s(x, X_s)$表示分组中与因子f_s相关联的所有因子的乘积。
所以交换加和和乘积的顺序得到：

$$\begin{aligned} p(\mathbf{x}) &= \prod_{s\in \mathrm{ne}(x)} \left[\sum_{X_s} F_s(x, X_s) \right] \\ &= \prod_{s\in \mathrm{ne}(x)} \mu_{f_s \to x}(x) \end{aligned}$$

其中

$$\mu_{f_s \to x}(x) = \sum_{X_s} F_s(x, X_s)$$

这可以被看做从因子结点fs到变量结点x的信息,需要求解的边缘概率分布p(x)等于所有到达结点x的输入信息的乘积。

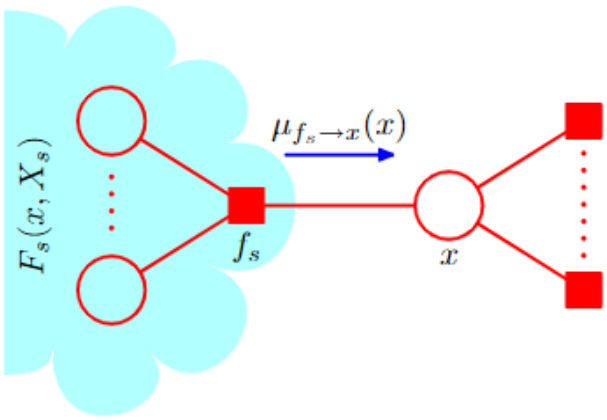

图 8.46: 因子图的片段，说明了边缘概率分布$p(x)$的计算。

对于8.46，每一个因子$F_s(x, X_s)$可以被分解：

$$F_s(x, X_s) = f_s(x, x_1, \cdots, x_M)G_1(x_1, X_{s1}) \cdots G_M(x_M, X_{sM})$$

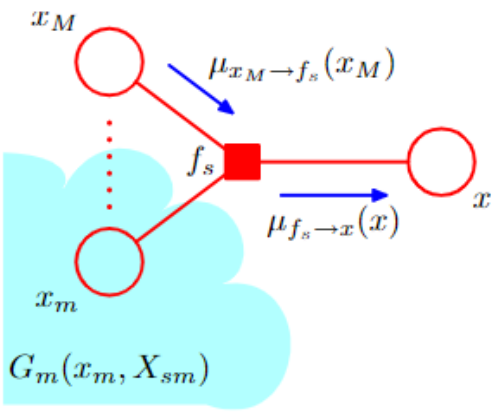

图 8.47: 与因子结点f_s关联的子图的分解。

我们将x之外的与因子f_s相关变量记作$x_1, \cdots, x_M$，变量集合$\{x, x_1, \cdots, x_M\}$是因子f_s依赖变量的集合，也可记作x_s,所以根据上面的公式，可以得到：

$$\begin{aligned}\mu_{f_s \to x}(x) &= \sum_{x_1} \cdots \sum_{x_M} f_s(x_1, \cdots, x_M) \prod_{m \in \mathrm{ne}(f_s) \setminus x} \left[\sum_{X_{sm}} G_m(x_m, X_{sm}) \right] \\ &= \mu_{f_s \to x}(x) = \sum_{x_1} \cdots \sum_{x_M} f_s(x_1, \cdots, x_M) \prod_{m \in \mathrm{ne}(f_s) \setminus x} \mu_{x_m \to f_s}(x_m)\end{aligned}$$

其中

$$\mu_{x_m \to f_s}(x_m) = \sum_{X_{sm}} G_m(x_m, X_{sm})$$

于是，我们引入了两类不同的信息。一类信息是从因子结点到变量结点的信息，记作$\mu_{f \to x}(x)$，另一类信息是从变量结点到因子结点的信息，记作$\mu_{x \to f}(x)$。在任何一种情况下，我们看到沿着一条链接传递的信息总是一个函数，这个函数是与那个链接相连的变量结点相关的变量的函数。

所以根据上面的公式可以知道，一个变量节点通过链接发送一个因子节点的信息可以按以下方式计算：计算沿着所有进入因子节点的其他链接的输入信息的乘积，乘以关联的那个因子，然后对所有与输入信息相关的变量进行求和，如8.47 。一旦因子节点得到从所有其他相邻变量节点的输入信息，这个因子节点就可以向变量节点发送信息。
变量结点到因子结点的信息的表达式：再次使用图分解（子图分解）

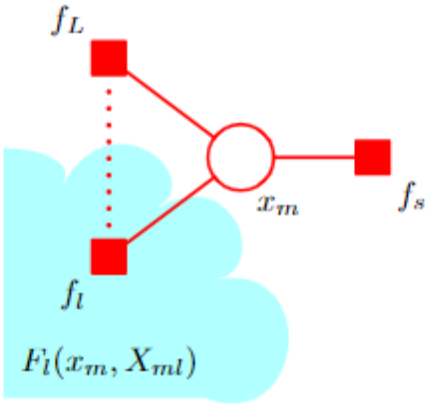

图 8.48: 由一个变量结点向一个相邻因子结点发送的信息的计算。

由8.48，可以看到$G_m(x_m, X_{sm})$由项$F_l(x_m, X_{lm})$的乘积组成，每一个项都与连接到x_m的一个因子节点f_l相关联，不包含f_s,即：

$$G_m(x_m, X_{sm}) = \prod_{l \in \mathrm{ne}(x_m) \setminus f_s} F_l(x_m, X_{ls})$$

其中求乘积的对象是结点x_m的所有相邻结点，排除结点f_s。
所以：

$$\begin{aligned} \mu_{x_m \to f_s}(x_m) &= \prod_{l \in \mathrm{ne}(x_m) \setminus f_s} \left[\sum_{X_{lm}} F_l(x_m, X_{ls}) \right] \\ &= \prod_{l \in \mathrm{ne}(x_m) \setminus f_s} \mu_{f_l \to x_m}(x_m) \end{aligned}$$

因此，为了计算从一个变量结点到相邻因子结点沿着链接传递的信息，我们只需简单地在其他所有结点上对输入信息取乘积。注意，任何只有两个相邻结点的变量结点无需参与计算，只需将信息不变地传递过去即可。此外，我们注意到，一旦一个变量结点接收到了来自所有其他相邻因子结点的输入信息，那么这个变量结点就可以给因子结点发送信息。
我们的目标是计算变量结点x的边缘概率分布，这个边缘概率分布等于沿着所有到达这个结点的链接的输入信息的乘积。
这些信息中的每一条信息都可以使用其他的信息递归地计算。为了开始这个递归计算的过程，我们可以将结点x看成树的根结点，然后从叶结点开始计算。根据上面公式的定义，我们看到如果一个叶结点是一个变量结点，那么它沿着与它唯一相连的链接发送的信息为：

$$\mu_{x \to f}(x) = 1$$

类似地，如果叶结点是一个因子结点，那么我们可以看到，发送的信息的形式为:

$$\mu_{f \to x}(x) = f(x)$$

所以，加和-乘积算法：
首先，我
们将变量结点x看成因子图的根结点，使用上面两个公式，初始化图的叶结点的
信息。之后，递归地应用信息传递步骤，直至信息沿着每一条链接传递完毕，并且根节点收到所有相邻节点的信息，每个结点都可以向根结点发送信息。一旦结点收
到了所有其他相邻结点的信息，那么它就可以向根结点发送信息。一旦根结点收到了所有相邻结点的信息，需要求解的边缘概率分布就可以使用公式

$$
\begin{aligned}
p(\mathbf{x}) &= \prod_{s \in \mathrm{ne}(x)} \left[\sum_{X_s} F_s(x, X_s) \right] \\
&= \prod_{s \in \mathrm{ne}(x)} \mu_{f_s \to x}(x)
\end{aligned}
$$

进行计算。
证明算法正确性：

为了说明每个结点总会收到足够的信息来使得发送信息变得可能，我们可以使用归纳法简单地说明如下。很明显，对于一个由变量根结点直接与几个因子叶结点相连的图，算法仅仅涉及到直接从叶结点向根结点发送形如（8.71）的信息。现在，假设通过每次添加一个结点的方式构建一个一般的图，并且假设对于某个特定的图，我们有一个合法的算法。当添加了一个更多的结点（变量结点或因子结点）之后，这个结点只能通过一个单一的链接与图相连，因为整体的图必须仍然是树，因此新结点是一个叶结点。于是，这个结点向它连接的结点发送一个信息，反过来会收到为了将自己的信息送往根结点所需的所有的信息，因此与之前一样，我们得到了一个合法的算法，从而完成了证明。

现在假设我们想寻找图中每个变量结点的边缘概率分布。这可以通过简单地对每个结点独立地运行上述算法的方式完成。然而，这会相当浪费计算结果，因为许多需要进行的计算被重复了多次。通过“叠加”多个信息传递算法，我们可以得到一个更加高效的步骤，从而得到一般的加和-乘积算法，如下所述。任意选择一个结点（变量结点或因子结点），然后将其指定为根结点。
然后我们从叶节点向根节点传递信息，根节点受到所有来自相邻节点的信息之后，可以向所有的相邻节点发送信息，反过来，这些结点之后会接收到来自所有相邻结点的信息，因此可以沿着远离根结点的链接发送出信息，以此类推。通过这种方式，信息可以从根结点向外传递到叶结点。现在，信息已经在两个方向上沿着图中所有的链接传递完毕，并且每个结点都已经接收到了来自所有相邻结点的信息。
因为每个变量结点会收到来自所有相邻结点的信息，
所以我们可以计算图中每个变量的边缘概率分布。必须计算的的信息的数量等于图中链接数量的二倍，因此所需的计算量仅仅是计算一个边缘概率分布的二倍。作为对比，如果我们对每个结点分别运行加和-乘积算法，那么计算量会随着图的规模以二次函数的形式增长。注意，这个算法实际上与哪个结点被选择为根结点无关。
假设我们想找到边缘概率分布$p(x_s)$,则：

$$p(x_s) = f_s(x_s) \prod_{i \in \mathrm{ne}(f_s)} \mu_{x_i \to f_s}(x_i)$$

这与变量结点的边缘概率分布十分相似.如果因子是参数化的函数，我们希望使用EM算法学习到参数的值，那么这些边缘概率分布恰好就是我们在E步骤中需要计算的值.

正如我们已经看到的那样，一个变量结点发送到一个因子结点的信息仅仅其他链接上的输入信息的乘积。如果必要的话，我们可以用一个稍微不同的形式考查加和-乘积算法，即消去从变量结点到因子结点的信息，仅考虑由因子结点发送出的信息。

我们始终忽略了归一化系数的问题。如果因子图是从有向图推导的，那么联合概率分布已经正确地被归一化了，因此通过加和-乘积算法得到的边缘概率分布会类似地被正确归一化。如果我们开始于一个无向图，那么通常会存在一个未知的归一化系数。

我们可以求出任意一个未归一化的边缘概率分布$\tilde{p}(x_i)$，然后很容易算出归一化系数，只需要对x_i积分即可。

下面是一个例子：

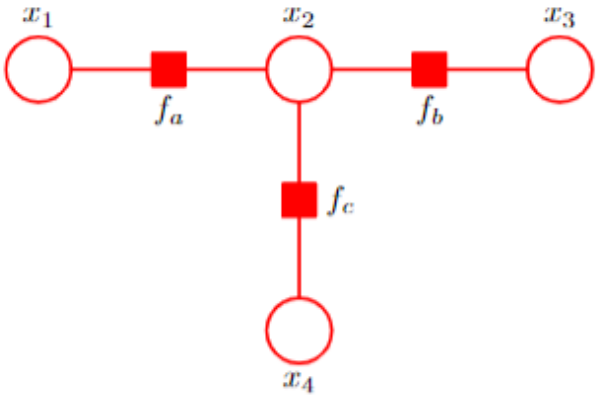

图 8.51: 一个简单的因子图，用来说明加和-乘积算法。

现在，考虑一个简单的例子来说明加和-乘积算法是很有帮助的。图8.51给出了一个简单的4节点因子图，它的未归一化联合概率分布为

$$\tilde{p}(\boldsymbol{x}) = f_a(x_1, x_2) f_b(x_2, x_3) f_c(x_2, x_4) \tag{8.73}$$

为了对这个图应用加和-乘积算法，让我们令结点x_3为根结点，此时有两个叶结点x_1和x_4。从叶结点开始，我们有下面六个信息组成的序列。

$$\mu_{x_1 \to f_a}(x_1) = 1 \tag{8.74}$$

$$\mu_{f_a \to x_2}(x_2) = \sum_{x_1} f_a(x_1, x_2) \tag{8.75}$$

$$\mu_{x_4 \to f_c}(x_4) = 1 \tag{8.76}$$

$$\mu_{f_c \to x_2}(x_2) = \sum_{x_4} f_c(x_2, x_4) \tag{8.77}$$

$$\mu_{x_2 \to f_b}(x_2) = \mu_{f_a \to x_2}(x_2)\mu_{f_c \to x_2}(x_2) \tag{8.78}$$

$$\mu_{f_b \to x_3}(x_3) = \sum_{x_2} f_b(x_2, x_3)\mu_{x_2 \to f_b}(x_2) \tag{8.79}$$

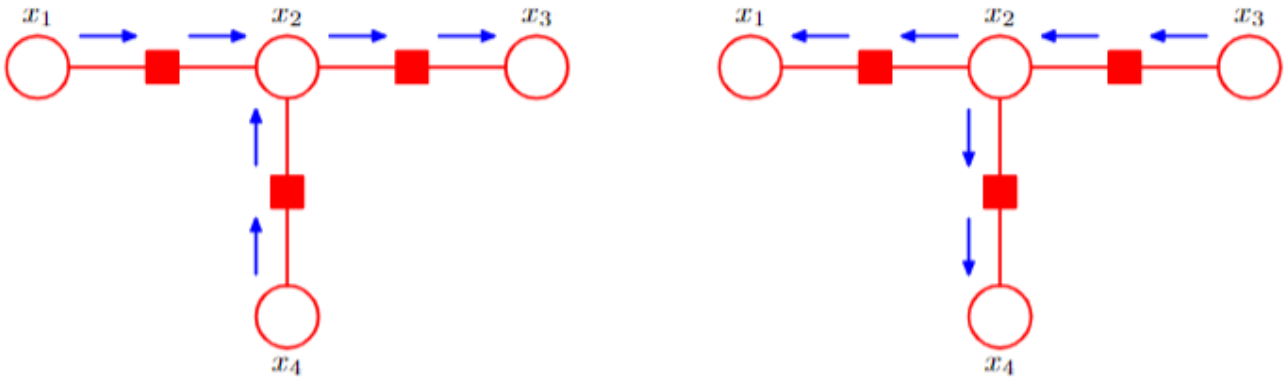

图 8.52: 应用于图8.51给出的图的加和-乘积算法的信息流。(a)从叶结点x_1和x_4向根结点x_3传递。(b)从根结点向叶结点传递。

信息流的方向如图8.52所示。一旦信息传播完成，我们就可以将信息从根结点传递到叶结点，这些信息为

$$\mu_{x_3 \to f_b}(x_3) = 1 \tag{8.80}$$

$$\mu_{f_b \to x_2}(x_2) = \sum_{x_3} f_b(x_2, x_3) \tag{8.81}$$

$$\mu_{x_2 \to f_a}(x_2) = \mu_{f_b \to x_2}(x_2)\mu_{f_c \to x_2}(x_2) \tag{8.82}$$

$$\mu_{f_a \to x_1}(x_1) = \sum_{x_2} f_a(x_1, x_2)\mu_{x_2 \to f_a}(x_2) \tag{8.83}$$

$$\mu_{x_2 \to f_c}(x_2) = \mu_{f_a \to x_2}(x_2)\mu_{f_b \to x_2}(x_2) \tag{8.84}$$

$$\mu_{f_c \to x_4}(x_4) = \sum_{x_2} f_c(x_2, x_4)\mu_{x_2 \to f_c}(x_2) \tag{8.85}$$

现在一个信息已经在两个方向上通过了每个链接，因此我们现在可以计算边缘概率分布。作为一个简单的检验，让我们验证边缘概率分布$p(x_2)$由正确的表达式给出。使用公式（8.63），使用上面的结果将信息替换掉，我们有

$$\begin{aligned}\tilde{p}(x_2) &= \mu_{f_a\to x_2}(x_2)\mu_{f_b\to x_2}(x_2)\mu_{f_c\to x_2}(x_2) \\ &= \left[\sum_{x_1} f_a(x_1,x_2)\right]\left[\sum_{x_3} f_b(x_2,x_3)\right]\left[\sum_{x_4} f_c(x_2,x_4)\right] \\ &= \sum_{x_1}\sum_{x_3}\sum_{x_4} f_a(x_1,x_2)f_b(x_2,x_3)f_c(x_2,x_4) \\ &= \sum_{x_1}\sum_{x_3}\sum_{x_4} \tilde{p}(\boldsymbol{x})\end{aligned} \tag{8.86}$$

这与我们预期的结果相同。

目前为止，我们已经假定图中所有的变量都是隐含变量。在大多数实际应用中，变量的一个子集会被观测到，我们希望计算以这些观测为条件的后验概率分布。
观测结点在加和-乘积算法中很容易处理，如下所述。假设我们将$\mathbf{x}$划分为隐含变量$\mathbf{h}$和观测变量$\mathbf{v}$，且$\mathbf{v}$的观测值被记作$\hat{\mathbf{v}}$。
然后，我们简单地将联合概率分布p(x)乘以$\prod_i I(v_i,\hat{v}_i)$,乘积对应于$p(\mathbf{h},\mathbf{v}=\hat{\mathbf{v}})$,是$p(\mathbf{h},\mathbf{v}=\hat{\mathbf{v}})$的一个未归一化版本，使用加和-乘积算法可以高效计算后验边缘概率$p(h_i|\mathbf{v}=\hat{\mathbf{v}})$,忽略归一化系数。

8.4.5 最大加和算法

有两个其他的比较常见的任务，即找到变量的具有最大概率的一个设置，以及找到这个概率的值。
这两个任务可以通过一个密切相关的算法——最大加和，可以视作动态规划在图模型的一个应用。
可以使用加和-乘积算法，找到每一个变量的$p(x_i)$然后得到使边缘概率最大的x_i^*,然而这只能得到对每个值单独取最大值的结果，我们希望找到联合概率最大的一组，即：

$$\mathbf{x}^{max} = \arg\max_{\mathbf{x}} p(\mathbf{x})$$

联合概率分布的对应值为：

$$p(\mathbf{x}^{max}) = \max_{\mathbf{x}} p(\mathbf{x})$$

于是，我们寻找一个高效的算法，来求出最大化联合概率分布p(x)的x的值，这会使得我们得到在最大值处的联合概率分布的值。为了解决第二个问题，我们只需简单地写出分量的最大值算符，即:

$$\max_{\mathbf{x}} = \max_{x_1}\cdots\max_{x_M} p(\mathbf{x})$$

然后，使用乘法分配律，可以交换乘积与最大值的顺序。
所以：

$$\max_{\mathbf{x}} p(\mathbf{x}) = \frac{1}{Z} \max_{x_1} \cdots \max_{x_M} \left[\phi_{1,2}(x_1, x_2) \cdots \phi_{N-1,N}(x_{N-1}, x_N)\right]$$
$$= \frac{1}{Z} \max_{x_1} \left[\max_{x_2} \left[\phi_{1,2}(x_1, x_2) \left[\cdots \max_{x_N} \phi_{N-1,N}(x_{N-1}, x_N)\right] \cdots \right] \right]$$

正如边缘概率的计算一样，我们看到交换最大值算符和乘积算法会产生一个更高效的计算，并且更容易表示为从结点x_N 沿着结点链传递回结点x_1的信息。

同理，可以将其推广到树结构图上，计算的结构与加和-乘积算法完全相同。特别地，假设我们令图中的一个特定的变量结点为根结点。之后，我们计算起始的一组信息，然后从树的叶结点向内部传递到根结点。

对于每个结点，一旦它接收到来自其他相邻结点的输入信息，那么它就向根结点发送信息。最后对所有到达根结点的信息的乘积进行最大化，得出p(x)的最大值。这可以被称为最大化乘积算法（max-produce algorithm），与加和-乘积算法完全相同，唯一的区别是求和被替换为了求最大值。注意，现阶段，信息被从叶结点发送到根结点，而没有相反的方向。

在实际应用时，由于小概率乘积会产生数值下溢，所以一般要取对数，取对数和取最大值的操作是可交换的。

所以取对数的唯一效果是把最大化乘积算法中的乘积替换成了加和，因此我们得到了最大化加和算法（max-sum algorithm）。根据之前在加和-乘积算法中得到的公式（8.66）和公式（8.69）给出的结果，我们可以基于信息传递写出最大化加和算法，只需把“加和”替换为“最大化”，把“乘积”替换为对数求和即可。结果为

$$\mu_{f\to x}(x) = \max_{x_1,\ldots,x_M} \left[\ln f(x, x_1, \ldots, x_M) + \sum_{m\in \mathrm{nc}(f)\setminus x} \mu_{x_m\to f}(x_m) \right] \tag{8.93}$$

$$\mu_{x\to f}(x) = \sum_{l\in \mathrm{nc}(x)\setminus f} \mu_{f_l\to x}(x) \tag{8.94}$$

最开始的由叶结点发送的信息可以通过类比公式（8.70）和公式（8.71）得到，结果为

$$\mu_{x\to f}(x) = 0 \tag{8.95}$$

$$\mu_{f\to x}(x) = \ln f(x) \tag{8.96}$$

而在根结点处的最大概率可以通过类比公式（8.63）得到，结果为

$$p^{\text{最大}} = \max_x \left[\sum_{s\in \mathrm{nc}(x)} \mu_{f_s\to x}(x) \right] \tag{8.97}$$

目前为止，我们已经看到了如何通过从叶结点到任意选择的根结点传递信息的方式找到联合概率分布的最大值。这个结果与根结点的选择无关。现在，我们转向第二个问题，即寻找联合概率达到最大值的变量的配置。目前，我们已经将信息从叶结点发送到了根结点。计算公式（8.97）的过程也会得到根结点变量的概率最高的值$x^{\text{最大}}$，定义为

$$x^{\text{最大}} = \arg\max_{x} \left[\sum_{s \in \text{ne}(x)} \mu_{f_s \to x}(x) \right] \tag{8.98}$$

现在，我们可能试图简单地继续使用信息传递方法，使用公式（8.93）和公式（8.94），将信息从根结点传回叶结点，然后将公式（8.98）应用于所有剩余的变量结点。然而，由于我们现在进行的是最大化过程而不是求和过程，因此有可能存在多个$\boldsymbol{x}$的配置，它们都会给出$p(\boldsymbol{x})$的最大值。在这种情况下，这个策略就失效了，因为通过对属于不同的最大化配置的每个结点处的信息的乘积进行最大化得到的各个变量值可能给出一个并不对应于最大值的整体配置。

通过使用一个从根结点到叶结点的一个相当不同的信息传递方式，这个问题可以得到解决。为了说明工作原理，让我们再次回到简单的结点链的例子中，其中有N个变量$x_1, \ldots, x_N$，每个变量有K个状态，对应于图8.38所示的图。假设我们令结点x_N是根结点。那么在第一阶段，我们从叶结点x_1开始，将信息传递到根结点，使用下面的公式

$$\mu_{x_n \to f_{n,n+1}}(x_n) = \mu_{f_{n-1,n} \to x_n}(x_n)$$

$$\mu_{f_{n-1,n} \to x_n}(x_n) = \max_{x_{n-1}} [\ln f_{n-1,n}(x_{n-1}, x_n) + \mu_{x_{n-1} \to f_{n-1,n}}(x_{n-1})]$$

将公式（8.94）和公式（8.93）应用到这个特定的图上即可得到上面的结果。叶结点发送的初始信息为

$$\mu_{x_1 \to f_{1,2}}(x_1) = 0 \tag{8.99}$$

这样，x_N的概率最高的值为

$$x_N^{\text{最大}} = \arg\max_{x_N} \left[\mu_{f_{N-1,N} \to x_N}(x_N) \right] \tag{8.100}$$

8.4.6 一般图的精确推断

对于许多实际应用，我们必须处理带有环的图。

信息传递框架可以被推广到任意的图拓扑结构，从而得到一个精确的推断步骤，被称为联合树算法：

信息传递框架可以被推广到任意的图拓扑结构，从而得到一个精确的推断步骤，被称为联合树算法（junction tree algorithm）（Lauritzen and Spiegelhalter, 1988; Jordan, 2007）。这里，我们简短地给出算法的关键步骤。这里不打算给出算法的细节，而是给出各个阶段的大致思想。如果我们的起始点是一个有向图，那么我们首先通过“伦理”步骤，将其转化为无向图。而如果起始点是无向图，那么这个步骤就不需要了。接下来，图被三角化（triangulated），这涉及到寻找包含四个或者更多结点的无弦环，然后增加额外的链接来消除无弦环。例如，在图8.36所示的图中，环$A-C-B-D-A$是一个无弦环，从而一个连接应该添加到在A和B之间或者C和D之间。注意，三角化后的图的联合概率分布仍然由同样的势函数乘积定义，但是这些势函数现在被看做是扩展的变量集合上的势函数。接下来，三角化的图被用于构建新的树结构无向图，被称为联合树（junction tree），它的结点对应于三角化的图的最大团块，它的链接将具有相同变量的团块对连接在了一起。这种方法中连接哪对团块是很重要的问题。正确的做法是选择能得到最大生成树（maximal spanning tree）的连接方式，如下所述。对于连接了某个团块的所有可能的树，被选择的树是树的权值最大的一个，其中链接的权值是由它所连接的两个团块所共享的结点的数量，树的权值是链接的权值之和。由于三角化步骤的存在，得到的树满足运行相交性质（running intersection property），意思是如果一个变量被两个团块所包含，那么它必须也被连接这两个团块的路径上的任意团块所包含。这确保了变量推断在图之间是相容的。最后，一个二阶段的信息传递算法，或者等价的加和-乘积算法，现在可以被应用于这个联合树，得到边缘概率分布和条件概率分布。虽然联合树算法听起来比较复杂，但是它的核心是一个简单的想法。我们已经利用这个想法研究了概率的分解性质，使得加和与乘积能够相互交换，从而可以进行部分求和，避免了直接对联合概率分布的操作。联合树的作用是提供一种组织这些计算的精确高效的方法。值得注意的是，这些完全是通过图操作实现的！

联合树对于任意的图都是精确的、高效的。对于一个给定的图，通常不存在计算代价更低的算法。不幸的是，算法必须对每个结点的联合概率分布进行操作（每个结点对应于三角化的图的一个团块），因此算法的计算代价由最大团块中的变量数量确定。在离散变量的情形中，计算代价会随着这个数量指数增长。一个重要的概念是图的树宽度（treewidth）（Bodlaender, 1993），它根据最大团块中变量的数量进行定义。事实上，它被定义为最大团块的规模减一，来确保一个树的树宽度等于1。由于通常情况下，从一个给定的起始图开始，可以构建出多种不同的联合树，因此树宽度由最大团块具有最少变量的联合树来定义。如果原始图的树宽度比较大，那么联合树算法就变得不可行了。

8.4.7 循环置信传播

在实际应用中，不能使用精确推断，因此使用近似推断，比如变分方法。
循环置信传播就是将加和-乘积算法应用到存在环的图中，信息会绕着图流动多次，对于某些模型会收敛，某些不会，所以需要定义信息传递时间表，让我们假设在任意给定的链接以及任意给定的方向上，每次传递一条信息。从一个结点发送的每条信息替换了之前发送的任何沿着同一链接的同一方向的信息，并且本身是一个函数，这个函数只与算法的前一步的结点接收到的最近的信息有关。

8.4.8 学习图结构

有一些研究超出了推断问题的范围，关注于从数据推断图结构本身，这需要我们定义一个可能结构的空间，以及用于对每个结构评分的度量。

从贝叶斯的观点来看，理想情况下，我们需要计算图结构上的后验概率分布，然后关于概率分布求平均，进行预测。如果我们有一个关于第m个图的先验概率分布p(m)，那么后验概率分布为：

$$p(m|D) \propto p(m)p(D|m)$$

其中D是一个观测数据集。模型证据$p(D|m)$提供了每个模型的分数。然而，计算模型证据涉及到对潜在变量的积分或求和，这对于许多模型来说是一个计算量相当大的问题。