信度网近似推理算法(下)[*]

Approximate Inherence Algorithms of Belief Network(2)

刘启元 张 聪 沈一栋

(重庆大学计算机科学与工程学院 重庆400044)

2 仿真方法[14]

基于信度网的精确推理算法通过利用各节点之间的条件独立性来加快推理计算,前面介绍的基于搜索的方法则不仅利用节点间的条件独立性而且利用概率分布的一些特性进行近似计算,但是在很复杂的问题领域要想通过分析条件独立性和利用概率分布的特征来加速推理计算往往不太现实。基于仿真的方法则另辟蹊径,既不利用条件独立性,也不考虑概率分布的特征,而是通过对样本进行统计以得到待求概率的近似值[9]。根据概率论,当样本量趋近于无穷大的时候,样本的统计值将等于其概率值,但是当样本量越小的时候,样本的统计值与真实概率值差异的可能性越大,即:$p(\tilde{p}-p^{*}>\varepsilon)$的值越大。其中$\tilde{p}$为对样本的统计值,$P^{*}$为真实的概率值,$\varepsilon$为误差量(一个大于0的正数)。随着样本量的增加,统计计算的结果就越精确。但是该精度的给定不象部分精确计算方法那样给一个误差边界,而只能给出一个概率边界。即样本量越大,其统计结果与真实结果的误差小于某误差限的可能性就越高。

基于仿真的方法又名Monte Carlo[9,8]方法,它首先通过采样技术对某概率分布进行采样以得到一组样本,然后对该组样本进行统计计算以得到待求概率值。通常采样阶段和统计计算阶段完全分开。目前提出的采样方法主要有:Reject Sampling、Importance Sampling[9]、Forward Sampling[3]和Markov Chain Sampling(如Gibbs Sampling[6,9,10]等)。这些采样方法的目的是设计一套算法,能够以最快的速度产生满足统计计算要求的样本。而在数据统计阶段的方法主要有:Straight Simulation[10,11]方法、Likelihood Weighting[2,12]方法和Bounded Variance[1]方法三类。这三类方法都是从不同的角度利用了Monte Carlo法的基本思想,而Bounded Variance方法也可以看作是Likelihood Weighting算法的一个变种。

作为这三种方法基础的Monte Carlo法,其计算的主要思想是通过一组样本以计算某函数的数学期望。其计算公式如下[6,9]:

$$E(f(\cdot))=\sum_{X_1,X_2,\cdots,X_n} f(X_1,X_2,\cdots,X_n)*P(X_1,X_2,\cdots,X_n)\approx\frac{1}{M}*\sum_{t=0}^{M-1}f(X_1^t,X_2^t,\cdots,X_n^t)$$

在上面的公式中需要计算函数$f(X_1,\cdots,X_n)$对于概率分布$P(X_1,\cdots,X_n)$的数学期望。如果直接计算,需要对随机变量$X_1,\cdots,X_n$的所有可能组合状态进行求和(如上式中间部分)。采用Monte Carlo方法,只需要将函数$f(X_1,\cdots,X_n)$对所有的样本求平均,就得到该函数的数学期望。为了便于理解和表达,可以对上面的公式进行改写。在上面的公式中我们假设随机变量$X_1,\cdots,X_n$所有可能的组合构成了一个状态空间Ω,其中ω为该状态空间中的一个状态,定义ω^t为第t个样本的状态,且$\omega^t\in\Omega$。另假设在M个样本中出现的所有组合状态构成状态空间Ω_1,且$\Omega_1\subseteq\Omega$。这样上面的计算公式可变换为:

$$E(f(\cdot))=\sum_{\Omega}f(\omega)*P(\omega)\approx\sum_{\Omega_1}f(\omega)*\frac{Num(\omega)}{M}=\frac{1}{M}\sum_{t=0}^{M-1}f(\omega^t)$$

其中函数$Num(\omega)$表示状态ω在样本中出现的次数,$Num(\omega)/M$则表示状态ω在样本中出现的频率。随着样本量的增加,当样本数趋近于无穷大的时候$\Omega_1=\Omega$,且状态出现的频率将等于该状态的概率。由此可知,Monte Carlo方法在计算过程中主要以频率代替概率,以对样本的算术平均代替数学期望。该方法的计算精度将随着样本量的增加而增加,换句话说要达到越高的精度对样本量的需求就越大。Stochastic Simulation方法、Likelihood Weighting方法和Bounded Variance方法的主要区别在于,它们采用了不同的f函数或(和)针对不同的概率分布进行采样[8,9]。下面介绍用Monte Carlo法计算边缘概率。在概率推理中,边缘概

*)本文受国家自然科学基金,教育部跨世纪优秀人才培养基金以及重庆市科技攻关项目《面向工业应用的智能开发平台及系统研究》的支持。

率的计算是所有计算问题的基础，如果将 Monte Carlo 法用在计算边缘概率的问题中，我们必须先定义一个函数 f。假设先计算单个变量 $X_i=a$ 的边缘概率，其函数 f 的定义如下：

$$f(X_1,\cdots,X_n)=\begin{cases}1, X_i=a\\0, X_i\neq a\end{cases}$$

该函数表示当节点 X_i 的取值为 a 的时候函数 f 的值为 1，否则为 0。在该定义下，函数 f 的数学期望将表示边缘概率 $P(X_i=a)$，即：

$$E(f(\cdot))=\sum_{X_1,X_2,\cdots,X_n} f(X_1,X_2,\cdots,X_n)*P(X_1,X_2,\cdots,X_n)$$

$$=\sum_{X_1,X_2,\cdots,X_n} P(X_1,\cdots,X_i=a,\cdots,X_n)=P(X_i=a)$$

采用 Monte Carlo 法将上面的计算公式近似为：

$$E(f(\cdot))=P(X_i=a)\approx\frac{1}{M}\sum_{t=0}^{M-1}f(X_1,\cdots,X_n)$$

$$=\frac{X_i=a\text{ 的样本数}}{M}$$

可见采用该方法避开了指数式的求和运算量。计算多个变量的边缘概率的计算方式与此类似。

Straight Simulation[10,11]方法由 Pearl 于1987年提出，该方法是对 Monte Carlo 法最直接的利用形式。它将边缘后验概率 $P(X_i=a|E=e)$，表达为后验概率分布 $P(X_1,\cdots,X_i=a,\cdots,X_n|E=e)$ 的边缘化。假设 $P'(X_1,\cdots,X_n)=P(X_1,\cdots,X_n|E=e)$，则后验概率分布 $P(X_i=a|E=e)$ 的计算公式为：

$$P(X_i=i|E=e)=\sum_{X_1,\cdots,X_N} P(X_1,\cdots,X_i=1,\cdots,X_n|E=e)$$

$$=\sum_{X_1,\cdots,X_N} f(X_1,\cdots,X_n)*P'(X_1,\cdots,X_N)\approx\frac{1}{M}\sum_{t=0}^{M-1}f(\omega^t)$$

其中函数 f 的定义如下：

$$f(X_1,\cdots,X_n)=\begin{cases}1, X_i=a\\0, X_i\neq a\end{cases}$$

该方法在采样时，基于的概率分布是随机变量间的后验联合概率分布 $P'(X_1,\cdots,X_n)$。在计算过程中首先利用一个采样器，对后验联合概率分布进行采样，以得到一组满足该概率分布的样本；然后计算函数 f 对于该样本集合的算术平均以近似后验概率：$P(X_i=a|E=e)$。该方法对样本的利用并不充分，只是简单地针对样本中包含 $X_i=a$ 的项时 f 函数的取值为1，否则取值为0。因此在达到相同精度下对样本的需求量较大，也就是计算时间复杂度较高。下面介绍的 Likelihood Weighting 算法就避免了这种简单的0、1取值方式，得到了比 Straight Simulation 算法更高的计算效率[1]。

Likelihood Weighting[2,12]是当前在信度网非精确推理领域应用最广泛的近似计算方法，该算法同样采用 Monte Carlo 方法的思想。它首先将联合概率分布分解为两项的乘积：

$$P(Z=z,E=e)=\rho(z,e)*\omega(z,e)$$

其中 Z 为非证据节点集合，E 为证据节点集合，这两个集合的当前取值分别为 z 和 e。上面公式中的两项 $\rho(z,e)$ 和 $\omega(z,e)$ 的计算公式分别为：

$$\rho(z,e)=\prod_{Z_i\in Z}P(Z_i|Pa(Z_i))|Z=z,E=e$$

$$\omega(z,e)=\prod_{E_i\in E}P(E_i|Pa(E_i))|Z=z,E=e$$

其中"Z=z,E=e"表示证据节点集和非证据节点集合中节点的当前取值分别为 z 和 e。由此可得到联合概率分布：

$$P(z,e)=\prod_i P(X_i|Pa_i)|Z=z,E=e$$

$$=(\prod_{Z_i\in Z}P(Z_i|Pa(Z_i))|Z=z,E=e)*(\prod_{E_i\in E}P(E_i|Pa(E_i))|Z=z,E=e)$$

$$=\rho(z,e)*\omega(z,e)$$

在概率推理中，边缘概率的计算是其它概率问题计算的基础。采用 Likelihood Weighting 方法进行推理计算时，需要计算两个边缘概率：$P(E=e)$ 和 $P(X=x,E=e)$。其中 $P(E=e)$ 表示证据的边缘概率，$P(X=x,E=e)$ 表示后验概率节点 X 和证据的联合边缘概率。这两个边缘概率的计算将采用 Monte Carlo 法，变概率平均为基于样本的算术平均。在 Likelihood Weighting 方法中边缘概率 $P(E=e)$ 的计算可以表达为：

$$P(E=e)=\sum_{X_i\in Z}(\prod_i P(X_i|Pa_i)|E=e)$$

$$=\sum_{z\in\Omega}\omega(z,e)*\rho(z,e)$$

其中 Ω 为随机变量集合 Z 中所有节点可能的组合状态所构成的空间。如果把 $\omega(z,e)$ 看作 Monte Carlo 方法中的函数 $f(\cdot)$，将 $\rho(z,e)$ 看作在空间 Ω 上的一个分布，则 $P(E=e)$ 可近似地表达为：

$$P(E=e)\approx\frac{1}{M}*\sum_{t=0}^{M-1}\omega(z^t,e)$$

在上式中 z^t 为满足在空间 Ω 上的分布 $\rho(z,e)$ 的第 t 个样本。而对于该方法中的另外一项边缘分布 $P(X=a,E=e)$ 的计算过程与此类似，即：

$$P(X=x,E=e)=\sum_{X_i\in Z/X}(\prod_i P(X_i|Pa_i)|X=x,E=e)$$

$$=\sum_{z\in\Omega,\text{且}z\text{包含}X=x}\omega(z,e)*\rho(z,e)$$

$$=\sum_{z\in\Omega}\omega(z,e)*\lambda(z,e)*\rho(z,e)$$

其中 $\lambda(z,e)$ 的定义为：

$$\lambda(z,e)=\begin{cases}1,\text{当 }X=x\in z\\0,\text{其余情况}\end{cases}$$

如果将 $\omega(z,e)*\lambda(z,e)$ 看作 Monte Carlo 方法中的函数 $f(\cdot)$，将 $\rho(z,e)$ 看作在空间 Ω 上的一个分布，则 P

$(X=x,E=e)$可近似地表达为：

$$P(X=x,E=e)\approx\frac{1}{M}\sum_{i=0}^{M-1}\omega(z^i,e)*\lambda(z^i,e)$$

通过这两个边缘概率可以直接计算出后验概率 $P(X=x|E=e)$。在采用 Likelihood Weighting 法进行近似计算的过程中，需要首先对概率分布 $\rho(z,e)$进行采样，得到满足该分布的一组样本；然后分别按照上面给出的两个 f 函数，计算它们对于该样本集合的算术平均以近似计算后验概率 $P(X=x|E=e)$。该方法在基于信度网的近似推理领域被广泛采用，但由于在计算边缘概率 $P(X=x,E=e)$的过程中使用了二值函数 $\lambda(z,e)$，它类似于象 Straight Simulation 方法中那样使得在计算 $P(X=x,E=e)$的过程中对样本的利用不够充分，还可以对该算法进行改进。这一思想就导致了 Bounded Variance 方法的产生。

Bounded Variance[1]方法是 Likelihood Weighting 方法的一个变种。它对 Likelihood Weighting 方法作了两点修改。第一点是在计算 $P(X=x,E=e)$的过程中没有引入二值函数 $\lambda(z,e)$，而是将 $X=x$、$E=e$ 作为一个联合证据，采用类似计算 $P(E=e)$的方法直接计算；第二点是在计算边缘概率如：$P(E=e)$的过程中，将原算法中的公式 $\omega(z,e)$改写为：

$$\zeta(z,e)=\omega(z,e)/\prod_{i=1}^{k}u_i$$

其中 k 为节点集合 E 中的节点数。假设节点集合 E 中有节点：$E_1,\cdots,E_k$，每一个节点拥有一张条件概率表：$P(E_i=e_i|Pa_i)$，则 u_i 表示了节点 E_i 的条件概率表中各项的上限，而 l_i 则表示了该条件概率表中各项的下限。在计算时采用如下公式进行计算：

$$P(E=e)\approx\frac{1}{M}*\prod_{i=1}^{k}u_i*\sum_{i=0}^{M-1}\zeta(z^i,e)$$

经证明该算法的收敛速度比 Likelihood Weighting 方法更快，在不存在极端概率的情况下，该算法是一个多项式时间复杂度的算法。

在以上这些算法中，第一步都是通过采样得到一组满足某概率分布的样本，然后再进行统计计算。目前的采样方法有多种，如：Reject Sampling、Importance Sampling、Forward Sampling、Gibbs Sampling 等。这些方法的目的就是尽可能快地得到满足要求的样本。下面简单地介绍一下这些采样方法。

·Reject Sampling[9]方法：不直接针对所需要的概率分布 $P(X)$进行采样，它通过引入另外一个更简单的分布 $P'(X)$，希望能够通过对 $P'(X)$进行采样以得到满足概率分布 $P(X)$的样本。对于 $P'(X)$的选取要求对于 X 的所有取值 x，均满足不等式：$P(x)<=c*P'(x)$。在采样过程中，首先通过对分布 $P'(X)$产生一个采样值 x^*，然后对该采样值以概率 $P(x)/c*P'(x)$接受。如果不接受 x^*，则继续产生另一个 x^*，直到该样本点被接受，然后再产生下一个样本点。经证明该采样算法的结果确实表达了分布：$P(X)$。但该方法对简化分布 $P'(X)$有一定的要求，对于一个复杂的分布要寻找到一个满意的简化几乎不可能。如果选择不当，为了保证在任何取值下不等式 $P(x)<=c*P'(x)$都成立，常量 c 的取值可能很大。它将导致样本的接受率降低，对于有些样本甚至非常低，从而使得样本的产生速度变得很慢。

·Importance Sampling[9]方法：同 Reject Sampling 方法一样，为了避免在复杂的分布 $P(X)$上直接采样，选取了另外一个分布 $P'(X)$，但是它与 Reject Sampling 方法不同之处在于它并不要求 $P'(X)$是 $P(X)$的近似，而仅仅要求当 $P(X=x)\neq0$时，$P'(X=x)\neq0$。采用该方法的原理如下：

$$E(f(X_1,\cdots,X_n))=\sum_{x_1,\cdots,x_n}f(x_1,\cdots,x_n)*P(x_1,\cdots,x_n)$$

$$=\sum_{x_1,\cdots,x_n}f(x_1,\cdots,x_n)*\frac{P(x_1,\cdots,x_n)}{P'(x_1,\cdots,x_n)}*P'(x_1,\cdots,x_n)$$

$$=\sum_{x_1,\cdots,x_n}f'(x_1,\cdots,x_n)*P'(x_1,\cdots,x_n)$$

在上式中将函数 $f(X_1,\cdots,X_n)$对于概率分布 $P(X_1,\cdots,X_n)$的数学期望，等价于求函数 $f'(X_1,\cdots,X_n)$对于概率分布 $P'(X_1,\cdots,X_n)$的数学期望。在采样时需要对分布 $P'(X_1,\cdots,X_n)$进行采样。为了简化计算可以将信度网中所有节点的连接边都去掉，将该网络表达的分布作为 $P'(X_1,\cdots,X_n)$。理论上在该算法中对分布 $P'(X_1,\cdots,X_n)$和原始分布 $P(X_1,\cdots,X_n)$之间的近似程度没作要求。但如果选择不当，在达到相同精度下所需要的样本量同样会大大增加。在原概率分布中，概率值大的状态对最后计算结果的影响也就大。如果在新的分布中这些状态的概率值反而小，那么为了产生这些状态所需要的样本数将会大大增加。因此新概率分布同原概率分布的近似程度越高越好，但同时新概率分布也要越简化越好，因为它有利于样本的产生速度。但是要寻找到这样一个分布非常困难，为此有些学者提出了一种 Adaptive Importance Sampling 的方法。该方法首先给定新分布的函数类型，然后任意给定该函数的参数，并通过在采样过程中不断地修正这些参数以得到一个满意的 $P'(X)$。最后以该 $P'(X)$实施 Importance Sampling。

·Forward Sampling[3]方法：该方法在采样时，为了避免直接在复杂的联合概率分布上采样，就采用了一种局部采样的方法。每次计算只改变一个节点的值，直到所有的节点被处理一次将得到一个样本。其采样过程为，首先从信度网中就是从顶层节点(无输入边的

节点开始，一层一层地往信度网的下层节点进行采样，直到所有节点均被实例化。在每一次采样时，节点取值的分布将满足 $P(X_i|Pa_i)$。由于从顶层节点开始逐步往下层节点进行采样，每一个节点在采样时它的父节点状态均已知，因此每个节点的采样将非常简单。经证明采用 Forward Sampling 将产生满足要求的样本。

·Gibbs Sampling[1,6,9,12]方法。除了上面介绍的方法外，还有一类基于 Markov 链的采样方法，其中最有代表性的就是 Gibbs Sampling 方法。该方法与前面方法的不同之处在于它的每一步采样不是一次性地改变所有节点的取值。而是一次改变一个节点 X_i，该节点的取值将满足概率分布 $P(X_i|X_1,\cdots,X_{i-1},X_{i+1},\cdots,X_n)$。在采样过程中，首先给各个节点随机赋初值，然后依顺序逐个采样信度网节点。每改变一个节点的状态将产生一个样本。信度网中首次采用 Gibbs Sampling 是在 Pearl 于1987年发表的文[10,11]中，后来该方法被 York 进行了增强。采用 Gibbs 方法在初期产生的样本将不满足所要求的概率分布，但是随着采样过程的继续，其采样的结果将收敛于该分布。通常只利用收敛后的样本进行计算。但是采用 Gibbs Sampling 方法对被采样的概率分布有要求，即不能出现概率值为0的状态，否则采用 Gibbs Sampling 将不能收敛到满足应有的分布。为了提高收敛的速度，Gibbs Sampling 方法还出现了许多变种。

以上的这些采样方法都被成功地用于信度网近似推理的计算中。除此以外这些方法还被用于基于其它图形模型的推理计算中。对于这些方法的综述详见文[9]。

结语 除了上述的方法外还出现了一些其它的近似计算方法，如基于 Noisy-OR 的搜索方法、基于变换的方法(Variational Method)[4,5,7]等。这些方法的主要目的都是如何尽可能多地利用信度网的结构特征、分布特征以加快收敛速度或简化计算，得到较快的计算速度。目前最引人注目的是变换方法。该方法主要通过对信度网里面的条件概率表引入额外的参数进行近似，以化简条件概率表内部的联系(解偶)，从而简化信度网以达到快速计算的目的。虽然它不象仿真方法那样简单和普遍适用，但它却能够处理一些其它方法很难处理的大型信度网，比如：QMR[5,13](Quick Medical Reference)。QMR 是一个两层的信度网，第一层600个节点表示600种病，第二层4000多个节点表示4000多种症状。该网络的平均父节点数为160.5个，每个条件概率表平均包含$2^{160.5}$项，采用 Junction Tree 算法其 Clique 节点的平均大小为105个内部节点。如此大的一个信度网采用精确推理显然无法计算，采用其它的近似算法的推理效率也极低。而采用变换方法可以达到很满意的推理速度，目前该算法已不局限于双层信度网，其条件概率表也不局限于用 Noisy-OR 的方法给出，它正在逐步成为一个比较通用的方法[5]。信度网推理虽然永远不可能找到一个多项式时间复杂度的算法，但是对这些各有所长的方法进行组合是一个很有希望的解决问题的办法，比如基于搜索的方法可以与基于仿真的方法相结合，在条件概率表呈现不同的特征的情况下采用不同的方法；另外变换方法和基于搜索的方法也可以结合，因为它们各自利用的概率分布的特征不同，如果混合使用可以弥补一些各自算法的不足；还有变换的方法也可以与仿真的方法相结合，用以检验当前结果的收敛性，以避免不必要的计算……。

参考文献

1 Dagum P, Luby M. An Optimal Approximation for Bayesian Inference. Artificial Intelligence, 1993, 60: 141~153

2 Fung R, Chang K-C. Weighting and integrating evidence for stochastic simulation in Bayesian networks. In: Henrion M, Shachter R, Kanal L, Lemmer J. eds. Uncertainty in Artificial Intelligence 5, Elsevier, Amsterdam, The Netherlands, 1990. 209~219

3 Henrion M. Propagating uncertainty in Bayesian networks by probabilistic logic sampling. In: Lemmer J, Kanal L eds. Uncertainty in Artificial Intelligence 2, North Holland, Amsterdam, The Netherlands, 1988. 149~163

4 Jaakkola T, Jordan M. Recursive algorithms for approximating probabilities in graphical models. In Advances of Neural Information Processing Systems 9. Cambridge, MA: MIT Press

5 Jaakkola T S, Jordan M I. Variational Probabilistic Inference and the QMR-DT Network. JAIR. 1999, 10: 291~322

6 Jensen C S, Kong A, Kjaerulff U. Blokcing Gibbs sampling in very large probabilistic expert systems. International Journal of Human Computer Studies, 1995, 42: 647~666

7 Jordan M I, Jaakkola T S. An Introduction to Variational Method for Graphical Models, Machine Leraning, 1999-10-11

8 MacKay D J C. Introduction to Monte Carlo methods. In: Jordan M I, ed. Learning in Graphical Models. Cambridge, MA: MIT Press

9 Neal R M. Probabilistic inference using Markov chain Monte Carlo methods: [Technical Report CRGTR-93-1]. Department of Computer Science, University of Toronto, 1993

10 Pearl J. Addendum: Evidential reasoning using stochastic simulation of causal models. Artificial Intelligence, 1987, 33: 131

11 Pearl J. Evidential reasoning using stochastic simulation of causal models. Artificial Intelligence, 1987, 32: 245~257

12 Shachter R D, Peot M A. Simulating approaches to general probabilistic inference on belief networks. Uncertainty in Artificial Intelligence, 1990, 5: 221~231

13 Shwe M, et al. Probabilistic diagnosis using a reformulation of the INTERNIST-1/QMR knowledge base I. The probabilistic model and inference algorithms. Methods of Information in Medicine, 1991, 30: 241~255

14 刘启元，张聪，沈一栋. 信度网近似推理算法(上). 计算机科学, 2001, 28(1)