

基于改进 QMAP 的贝叶斯网络参数学习算法

邸若海¹, 李叶¹, 万开方², 吕志刚¹, 王鹏¹

(1.西安工业大学 电子信息工程学院, 陕西 西安 710021; 2.西北工业大学 电子信息学院, 陕西 西安 710072)

摘 要:小数据集使得贝叶斯网络参数学习中的统计信息不准确,导致只依靠数据难以得到准确的贝叶斯网络参数。定性最大后验估计(QMAP)方法是目前小数据集条件下贝叶斯网络参数学习精度最高的算法。然而,当参数约束数量较多或参数可行域较小时,QMAP 算法中的拒绝-接受采样过程会变得极为耗时甚至难以完成。为了提高 QMAP 算法的学习效率同时又尽量不影响其学习精度,设计了一种约束区域中心点的解析计算方法来替代原有的拒绝-接受采样计算方法。结合参数约束构建一个求解约束区域边界点的目标优化模型;利用凸优化引擎来求解该目标优化模型,获得约束区域的边界点和中心点;通过获得的约束区域中心点改进现有的 QMAP 算法。仿真实验证明,所提出的 CMAP 算法的参数学习精度稍差于 QMAP 算法,但计算效率比 QMAP 算法提高了 2~5 倍。

关 键 词:贝叶斯网络;参数学习;定性最大后验估计;参数约束;目标优化
中图分类号:TP31 **文献标志码:**A **文章编号:**1000-2758(2021)06-1356-12

针对小数据集条件下的贝叶斯网络参数学习问题,目前主要通过引入参数约束来优化参数学习过程。根据约束数量的不同,现有的研究分为单一约束和多种约束 2 种情况。

针对单一约束,Writting 等^[1]利用定性影响约束构造惩罚函数,并将最大熵函数与惩罚函数结合构造目标优化函数,最后利用改进的 APN 算法对参数进行求解;Feelders 等^[2-3]将定性影响条件下的参数学习看作一个保序回归问题,首先采用最大似然估计算法得到参数初始值,然后结合定性影响约束和保序回归法估计参数。Plajner 等^[4]分别利用梯度下降法、保序回归来处理单调性约束,大幅度提高了参数学习的精度。国内学者周堃等^[5]通过引入平滑先验来优化似然函数,并将定性影响约束作为一个惩罚项与改进后的似然函数构成参数目标优化函数,采用梯度下降方法进行求解。邸若海等^[6-7]通过构造单调性约束表示专家经验,然后将单调性约束转化为狄利克雷分布的超参数,进而结合最大后验概率估计获取贝叶斯网络参数。任佳等^[8]采用区间约束限制参数学习,将区间约束转化为贝塔分布的超参数,进而结合贝叶斯估计学习参数。黄政等^[9]将近似相等的先验约束以正态分布的形式融入到单调性约束的贝叶斯参数学习过程中,提高了参数学习的精度。

由于单一约束方法一般只针对某种特殊的专家知识,限制了专家经验的利用,学习精度一般。因此,利用多种约束进行参数学习受到了更多的关注。Niculescu 等^[10]和 Campos 等^[11]提出基于凸优化的参数学习方法。Liao 等^[12]结合 EM 算法,将约束引入到数据缺失条件下的 BN 参数学习。Zhou 等^[13-14]提出约束多项式分布参数学习(multinomial parameter learning with constraints,MPL-C)方法。郭志高等^[15]提出基于双重约束的参数学习方法。Rui 等^[16]提出基于定性最大后验概率(qualitative maximum a posterior,QMAP)的参数学习方法,采用拒绝-接受采样的方法获取满足约束的模型参数,并利

收稿日期:2021-04-14 **基金项目:**国家自然科学基金面上项目(62171360)、电子信息系统复杂电磁环境效应国家重点实验室基金(CEMEE2020Z0202B)、陕西省自然科学基金基础研究计划(2020JQ-816)、陕西省教育厅专项科研计划项目(20JK0680)与西安市科技计划项目(2020KJRC0033)资助

作者简介:邸若海(1986—),西安工业大学讲师,主要从事复杂系统建模、贝叶斯网络学习研究。

通信作者:王鹏(1978—),西安工业大学教授,主要从事复杂系统建模、图像处理研究。e-mail:wp_xatu@163.com

用采样所得参数的均值替代先验参数的分布,进而利用 MAP 计算对应的模型参数的后验估计。与其他算法相比,QMAP 算法不仅给出了一个在多约束条件下的贝叶斯网络参数学习通用框架,而且具有较好的学习效果。在此基础上,Guo 等^[17]利用 QMAP 算法学习参数,并进一步利用参数约束判断参数的合理性,避免了 QMAP 算法所得参数不满足约束的情况。然而,当网络复杂度变大,参数约束较多或约束可行域较小时,由于拒绝-接受采用算法本身的随机性,导致难以获得满足约束的参数。从而导致 QMAP 算法非常耗时。

针对上述问题,本文设计了一种新的约束区域中心点的解析计算方法来替代原有的拒绝-采样方法,在此基础上,提出了 CMAP (constrained maximum a posteriori) 算法。首先,设计一种新的目标函数,结合参数约束构建一个带约束的目标优化问题;其次,利用一种寻优引擎来求解该目标优化问题获得约束区域的边界点和中心点,最终以获得的中心点作为待求参数的先验分布,并利用 MAP 求取得待求参数的后验估计。

1 贝叶斯网络

下面给出贝叶斯网络一般的定义:

定义 1 贝叶斯网络^[18]由结构 G 和参数 Θ 两部分组成。贝叶斯网络的结构为一个有向无环图 $G = (V, E)$, 其中节点变量集合 $V = \{X_1, X_2, \dots, X_n\}$ 表示变量,有向边集合 E 描述变量间的依赖关系。贝叶斯网络的参数 $\Theta = \{\theta_i\}_{i=1,2,\dots,n}$ 为一组条件概率分布,每个节点存储其自身与其父节点集之间的条件概率分布,即 $P(X_i | P_a(X_i))$ 。

即贝叶斯网络包含定性和定量两部分内容。在定性层面,有向无环图描述变量之间的依赖或独立关系。在定量层面,条件概率分布表示变量之间依赖程度的大小。

贝叶斯网络通过引入条件独立性完成了联合概率分布的分解,它将联合概率分布 P 分解为

$$P(X_1, X_2, \dots, X_n) = \prod_{i=1}^n P(X_i | P_a(X_i)) \quad (1)$$

式中, X_i 可取离散值和连续值。

2 参数的先验分布及约束类型

2.1 参数先验分布

贝叶斯公式将先验分布和似然函数结合,得到 θ 后验分布即

$$P(\theta | D) \propto P(\theta) L(\theta | D) \quad (2)$$

式中, 概率分布 $P(\theta)$ 表示 θ 的先验知识, 似然函数 $L(\theta | D) = P(D | \theta)$ 表示数据集 D 的影响。在 i.i.d 条件下, (2) 式中 $L(\theta | D)$ 为多项式似然函数的共轭分布, 假设先验分布 $P(\theta)$ 是 Dirichlet 分布, 即

$$P(\theta) = \prod_{i=1}^n \prod_{j=1}^q \prod_{k=1}^r \theta_{ijk}^{\alpha_{ijk}-1} \quad (3)$$

则后验分布 $P(\theta | D)$ 也是 Dirichlet 分布。在观测样本下, θ 的后验分布 $P(\theta | D)$ 为

$$p(\theta | D) \propto \prod_{i=1}^n \prod_{j=1}^q \prod_{k=1}^r \theta_{ijk}^{N_{ijk} + \alpha_{ijk} - 1} \quad (4)$$

式中: α 为位置参数; $\sum_{k=1}^r \alpha_{ijk} = 1$, N_{ijk} 为符合父节点取值为 j 且子节点取值为 k 时的样本统计量。当样本量 D 很小时, 贝叶斯估计主要依赖于先验知识; 当样本量 D 增大时, 贝叶斯估计越依赖于样本数据, 越接近 (5) 式所示的最大似然估计

$$\theta_{ijk} = \frac{N_{ijk}}{\sum_k N_{ijk}} \quad (5)$$

Dirichlet 先验分布中参数的物理意义不明显, 领域专家难以直接确定, 在实际应用中存在困难。根据 Dirichlet 分布特性, 其条件分布和边缘分布都是 Beta 分布且 Beta 分布为二项式分布的共轭分布族, 因此可以通过确定 Beta 分布中的 2 个形参来近似表达与其相关的领域知识。

根据贝叶斯估计和领域知识建立的先验 Beta 分布模型, 在 i.i.d 条件下 (6) 式成立

$$P(\theta) = B[\alpha_1, \alpha_2] \theta^{\alpha_1-1} (1-\theta)^{\alpha_2-1} \quad (6)$$

式中, $B[\alpha_1, \alpha_2] = \frac{\tau(\alpha_1 + \alpha_2)}{\tau(\alpha_1)\tau(\alpha_2)}$, α_1 和 α_2 均大于 0, 称为 Beta 分布的 2 个超参数, 将领域知识表示为先验分布超参数 α_1 和 α_2 描述的估计值以及估计值的可信度, 能够更加准确地描述领域知识, 该 Beta 分布的期望与方差分别为

$$D_B = \frac{\alpha_1 \alpha_2}{(\alpha_1 + \alpha_2)^2 (\alpha_1 + \alpha_2 + 1)}$$

$$E_B = \frac{\alpha_1}{\alpha_1 + \alpha_2} \quad (7)$$

2.2 参数约束类型

基于小数据集的参数学习,仅依靠数据已经不能获取满足精度要求的参数。因此,现在的方法都

是将先验知识引入到参数学习过程之中,来弥补数据的不足。先验知识一般通过不同形式的参数约束来表示,主要的参数约束如表 1 所示。

表 1 参数约束类型

序号	约束类型	约束表达式	约束意义
1	规范性约束	$\sum_{k=1}^{r_i} \theta_{ijk} = 1, 0 \leq \theta \leq 1, \forall i, j, k$	表示父节点取值相同时的所有参数之和为 1
2	区间约束	$0 \leq \alpha_{ijk} \leq \theta_{ijk} \leq \beta_{ijk} \leq 1$	表示参数属于某一给定的区间
3	分布内约束	$\theta_{ijk} \leq \theta_{ijk'}, \forall k \neq k'$	表示父节点取值相同时的同一分布内参数之间的大小关系
4	分布间约束	$\theta_{ij'k} \leq \theta_{ijk}, \forall j \neq j'$	表示子节点取值相同时的不同分布之间参数的大小关系
5	近似相等约束	$\theta_{i'j'k'} \approx \theta_{ijk}$	表示任意 2 个参数之间近似相等
6	加性协同约束	$\theta_{ij_1k} + \theta_{ij_2k} \leq \theta_{ij_3k} + \theta_{ij_4k}, \forall i, k$	表示子节点相同时不同父节点取值所对应参数之和的大小关系
7	乘性协同约束	$\theta_{ij_1k} \theta_{ij_2k} \leq \theta_{ij_3k} \theta_{ij_4k}, \forall i, k$	表示子节点相同时不同父节点取值所对应参数乘积之间的大小关系

3 基于改进 QMAP 的贝叶斯网络参数学习算法

3.1 低维参数约束域的可视化及中心点计算

首先分析参数约束区域的特点,分别给出二维和三维情况下约束区域中心点的计算方法。在二维状态下,假设参数满足以下约束

$$\begin{cases} \theta_1 + \theta_2 = 1 \\ \theta_1 > \theta_2 \\ 0 \leq \theta_1 \leq 1 \\ 0 \leq \theta_2 \leq 1 \end{cases} \quad (8)$$

通过将约束区域可视化,图 1 中加粗的线条区域就是参数的约束域,即线段 AC。只需获取 A 点和 C 点的坐标,即可获得线段 AC 的中点。本文将求取点 A 和 C 坐标的问题转化为一个带约束的优化问题,通过公式(9)可以求取点 C 的坐标,通过公式(10)可以求取 A 点的坐标。

$$\begin{aligned} &\min [(\theta_1 - 0)^2 + (\theta_2 - 1)^2] \\ &\text{s.t.} \begin{cases} \theta_1 + \theta_2 = 1 \\ \theta_1 > \theta_2 \\ 0 \leq \theta_1 \leq 1 \\ 0 \leq \theta_2 \leq 1 \end{cases} \\ &\min [(\theta_1 - 1)^2 + (\theta_2 - 0)^2] \end{aligned} \quad (9)$$

$$\text{s.t.} \begin{cases} \theta_1 + \theta_2 = 1 \\ \theta_1 > \theta_2 \\ 0 \leq \theta_1 \leq 1 \\ 0 \leq \theta_2 \leq 1 \end{cases} \quad (10)$$

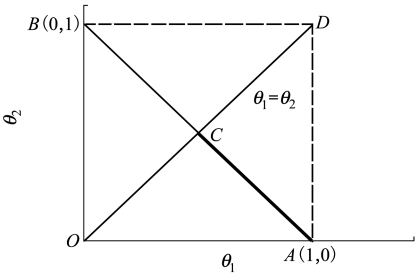


图 1 二维参数的可行域

在三维情况下,假设参数满足以下约束

$$\begin{cases} \theta_1 + \theta_2 + \theta_3 = 1 \\ \theta_1 > \theta_2 \\ \theta_1 < \theta_3 \\ 0 \leq \theta_1 \leq 1 \\ 0 \leq \theta_2 \leq 1 \\ 0 \leq \theta_3 \leq 1 \end{cases} \quad (11)$$

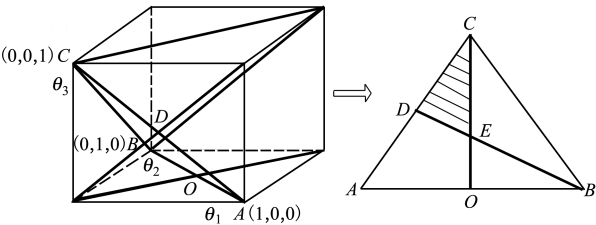


图 2 三维参数的可行域

从图 2 可以看出,参数的约束区域为 $\triangle CDE$,类似于二维的情况,将其转化为一个目标最优化问题,具体如下

$$\begin{aligned} & \min [(\theta_1 - 0)^2 + (\theta_2 - 0)^2 + (\theta_3 - 1)^2] \\ & \text{s.t.} \begin{cases} \theta_1 + \theta_2 + \theta_3 = 1 \\ \theta_1 > \theta_2 \\ \theta_1 < \theta_3 \\ 0 \leq \theta_1 \leq 1 \\ 0 \leq \theta_2 \leq 1 \\ 0 \leq \theta_3 \leq 1 \end{cases} \end{aligned} \quad (12)$$

$$\begin{aligned} & \min [(\theta_1 - 0)^2 + (\theta_2 - 1)^2 + (\theta_3 - 0)^2] \\ & \text{s.t.} \begin{cases} \theta_1 + \theta_2 + \theta_3 = 1 \\ \theta_1 > \theta_2 \\ \theta_1 < \theta_3 \\ 0 \leq \theta_1 \leq 1 \\ 0 \leq \theta_2 \leq 1 \\ 0 \leq \theta_3 \leq 1 \end{cases} \end{aligned} \quad (13)$$

$$\begin{aligned} & \min [(\theta_1 - 1)^2 + (\theta_2 - 0)^2 + (\theta_3 - 0)^2] \\ & \text{s.t.} \begin{cases} \theta_1 + \theta_2 + \theta_3 = 1 \\ \theta_1 > \theta_2 \\ \theta_1 < \theta_3 \\ 0 \leq \theta_1 \leq 1 \\ 0 \leq \theta_2 \leq 1 \\ 0 \leq \theta_3 \leq 1 \end{cases} \end{aligned} \quad (14)$$

下面给出本文参数约束域中心点计算的一般化的方法,设待求参数

$$\boldsymbol{\theta} = (\theta_{ij_1k_1}, \theta_{ij_1k_2}, \cdots, \theta_{ij_qk_m}, \theta_{ij_2k_1}, \theta_{ij_2k_2}, \cdots, \theta_{ij_2k_m}, \cdots, \theta_{ij_qk_1}, \theta_{ij_qk_2}, \cdots, \theta_{ij_qk_m})$$

式中,参数对应的子节点状态数为 m ,父节点状态数为 q 。 n 维空间的单位基向量为 $\mathbf{e}_1, \mathbf{e}_2, \cdots, \mathbf{e}_n$, $\mathbf{e}_1 = \overbrace{(1, 0, 0, \cdots, 0)}^n, \cdots, \mathbf{e}_n = \overbrace{(0, 0, 0, \cdots, 1)}^n$ 。当参数约束中包含分布间约束时,通过公式 (15) 分别利用 n 维参数空间的每一个单位基向量计算相应的边界点,具体的计算方法如下所示:

$$\theta_{\text{rang}} = \underset{\theta}{\operatorname{argmin}} (\boldsymbol{\theta} - \mathbf{e}_i) (\boldsymbol{\theta} - \mathbf{e}_i)^T$$

$$\begin{aligned} & \text{s.t.} \begin{cases} \theta_{ij_1k_1} > \theta_{ij_1k_2} \\ \theta_{ij_2k_1} < \theta_{ij_2k_2} \\ \cdots \cdots \cdots \\ \theta_{ij_qk_1} < \theta_{ij_qk_m} \\ \theta_{ij_1k_1} > \theta_{ij_2k_1} \\ \theta_{ij_1k_1} > \theta_{ij_3k_1} \\ \cdots \cdots \cdots \\ \theta_{ij_qk_m} > \theta_{ij_{q-1}k_m} \\ a_1 \leq \theta_{ij_1k_1} \leq b_1 \\ a_2 \leq \theta_{ij_1k_2} \leq b_2 \\ \cdots \cdots \cdots \\ a_{qm} \leq \theta_{ij_qk_m} \leq b_{qm} \\ 0 \leq \theta_{ij_1k_1} \leq 1 \\ 0 \leq \theta_{ij_1k_2} \leq 1 \\ \cdots \cdots \cdots \\ 0 \leq \theta_{ij_qk_m} \leq 1 \\ \theta_{ij_1k_1} + \theta_{ij_1k_2} + \cdots + \theta_{ij_1k_m} = 1 \\ \theta_{ij_2k_1} + \theta_{ij_2k_2} + \cdots + \theta_{ij_2k_m} = 1 \\ \cdots \cdots \cdots \\ \theta_{ij_qk_1} + \theta_{ij_qk_2} + \cdots + \theta_{ij_qk_m} = 1 \end{cases} \end{aligned} \quad (15)$$

当参数不包含分布间约束时,可根据参数的局部独立性,将上述高维 (mq) 的优化问题化简为低维 (m) 的优化问题。具体如下所示:

$$\begin{aligned} \theta_{\text{rang}} &= \underset{\theta}{\operatorname{argmin}} (\boldsymbol{\theta} - f_{(w-1)m+1}) (\boldsymbol{\theta} - f_{(w-1)m+1})^T \\ &\vdots \\ \theta_{\text{rang}} &= \underset{\theta}{\operatorname{argmin}} (\boldsymbol{\theta} - f_{wm}) (\boldsymbol{\theta} - f_{wm})^T \end{aligned}$$

$$\begin{aligned} & \text{s.t.} \begin{cases} \theta_{ij_wk_1} > \theta_{ij_wk_2} \\ a_1 \leq \theta_{ij_wk_1} \leq b_1 \\ 0 \leq \theta_{ij_wk_1} \leq 1 \\ 0 \leq \theta_{ij_wk_2} \leq 1 \\ \cdots \cdots \cdots \\ 0 \leq \theta_{ij_wk_m} \leq 1 \\ \theta_{ij_wk_1} + \theta_{ij_wk_2} + \cdots + \theta_{ij_wk_m} = 1 \end{cases} \end{aligned} \quad (16)$$

式中, $w = 1, 2, \cdots, q$, 总共需要求解 mq 个 m 维的优化问题。

通过 (15) ~ (16) 式可以求取约束区域的边界点,本文通过边界点直接求平均的方式求取约束区域的中心点,在二维情况下肯定能够获取中心点。

在三维情况下,当约束区域为三角形时,可以获得约束区域的中心点,当约束区域为四边形时,只能获取近似的中心点。在高维情况下,也会存在只能获取近似中心点的情况。这是本文算法精度稍差于 QMAP 算法的原因。

3.2 CMAP 算法

CMAP 算法的具体步骤如下:

步骤 1 记录样本数据中的节点状态统计量 N_{ij} 和 N_{ijk} 。

步骤 2 利用优化引擎求解(15)~(16)式,以求取参数约束的边界点,进而对边界点进行平均,获取参数约束的中心点。利用中心点替代待求参数的先验分布,进而代入最大后验估计公式(17)获取待求参数的后验估计。

$$\hat{\theta}_{ijk} = \frac{N_{ijk} + M_{ijk}}{\sum_{k=1}^{r_i} (N_{ijk} + M_{ijk})} = \frac{N_{ijk} + M_{ijk}}{N_{ij} + M_{ij}} \quad (17)$$

式中: $M_{ijk} = A * P(X_i = k | \Pi_i = j, \Omega)$, A 为等价样本量。 $P(X_i = k | \Pi_i = j, \Omega)$ 为待求参数的先验分布,这里用中心点的坐标替代。这是本文算法与 QMAP 算法最大的不同。QMAP 算法是用得到满足约束的参数均值替代参数的先验分布。

步骤 3 利用交叉验证方法确定系数 A 即等价样本量。

步骤 4 根据公式(17)完成定性最大后验概率估计。

3.3 算例说明

为了更好地解释本文的算法,接下来通过一个简单的算例来说明。

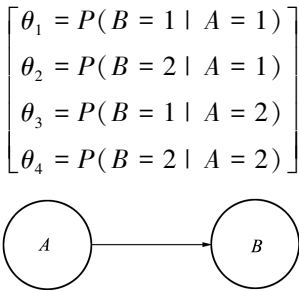


图 3 一个简单的贝叶斯网络

假设图 3 中节点 B 的参数是待求的参数,此时,已知的参数约束为

$$\text{s.t.} \begin{cases} \theta_1 + \theta_2 = 1 \\ \theta_3 + \theta_4 = 1 \\ \theta_1 > \theta_2 \\ \theta_3 < \theta_4 \\ \theta_1 > \theta_3 \\ 0 \leq \theta_1 \leq 1 \\ 0 \leq \theta_2 \leq 1 \\ 0 \leq \theta_3 \leq 1 \\ 0 \leq \theta_4 \leq 1 \\ a_2 \leq \theta_1 \leq b_2 \\ a_3 \leq \theta_3 \leq b_3 \end{cases} \quad (18)$$

$\theta_1, \theta_2, \theta_3, \theta_4$ 组成了 4 维参数空间,首先判断参数的约束类型是否包含第 2 节中的分布间约束,如果不包含,则可以利用参数的局部独立性将待求参数进行分解和降维。如果包含,则无法利用参数的局部独立性将参数进行分解和降维。很明显,公式(18)中包含分布间约束 $\theta_1 > \theta_3$,无法利用参数局部独立性。针对这种情况,本文利用标准单位基向量构建目标函数

$$\begin{aligned} (\theta_1, \theta_2, \theta_3, \theta_4) = \underset{\theta_1, \theta_2, \theta_3, \theta_4}{\operatorname{argmin}} [& (\theta_1 - 1)^2 + (\theta_2 - 0)^2 + \\ & (\theta_3 - 0)^2 + (\theta_4 - 0)^2] \\ \text{s.t.} \begin{cases} \theta_1 + \theta_2 = 1 \\ \theta_3 + \theta_4 = 1 \\ \theta_1 > \theta_2 \\ \theta_3 < \theta_4 \\ \theta_1 > \theta_3 \\ 0 \leq \theta_1 \leq 1 \\ 0 \leq \theta_2 \leq 1 \\ 0 \leq \theta_3 \leq 1 \\ 0 \leq \theta_4 \leq 1 \\ a_2 \leq \theta_1 \leq b_2 \\ a_3 \leq \theta_3 \leq b_3 \end{cases} \quad (19) \end{aligned}$$

$$\begin{aligned} (\theta_1, \theta_2, \theta_3, \theta_4) = \underset{\theta_1, \theta_2, \theta_3, \theta_4}{\operatorname{argmin}} [& (\theta_1 - 0)^2 + (\theta_2 - 1)^2 + \\ & (\theta_3 - 0)^2 + (\theta_4 - 0)^2] \end{aligned}$$

$$\begin{aligned} & \begin{cases} \theta_1 + \theta_2 = 1 \\ \theta_3 + \theta_4 = 1 \\ \theta_1 > \theta_2 \\ \theta_3 < \theta_4 \\ \theta_1 > \theta_3 \end{cases} \\ \text{s.t.} & \begin{cases} 0 \leq \theta_1 \leq 1 \\ 0 \leq \theta_2 \leq 1 \\ 0 \leq \theta_3 \leq 1 \\ 0 \leq \theta_4 \leq 1 \\ a_2 \leq \theta_1 \leq b_2 \\ a_3 \leq \theta_3 \leq b_3 \end{cases} \end{aligned} \quad (20)$$

$$\begin{aligned} & (\theta_1, \theta_2, \theta_3, \theta_4) = \underset{\theta_1, \theta_2, \theta_3, \theta_4}{\operatorname{argmin}} [(\theta_1 - 0)^2 + (\theta_2 - 0)^2 + \\ & (\theta_3 - 1)^2 + (\theta_4 - 0)^2] \\ & \begin{cases} \theta_1 + \theta_2 = 1 \\ \theta_3 + \theta_4 = 1 \\ \theta_1 > \theta_2 \\ \theta_3 < \theta_4 \\ \theta_1 > \theta_3 \end{cases} \\ \text{s.t.} & \begin{cases} 0 \leq \theta_1 \leq 1 \\ 0 \leq \theta_2 \leq 1 \\ 0 \leq \theta_3 \leq 1 \\ 0 \leq \theta_4 \leq 1 \\ a_2 \leq \theta_1 \leq b_2 \\ a_3 \leq \theta_3 \leq b_3 \end{cases} \end{aligned} \quad (21)$$

$$\begin{aligned} & (\theta_1, \theta_2, \theta_3, \theta_4) = \underset{\theta_1, \theta_2, \theta_3, \theta_4}{\operatorname{argmin}} [(\theta_1 - 0)^2 + (\theta_2 - 0)^2 + \\ & (\theta_3 - 0)^2 + (\theta_4 - 1)^2] \\ & \begin{cases} \theta_1 + \theta_2 = 1 \\ \theta_3 + \theta_4 = 1 \\ \theta_1 > \theta_2 \\ \theta_3 < \theta_4 \\ \theta_1 > \theta_3 \end{cases} \\ \text{s.t.} & \begin{cases} 0 \leq \theta_1 \leq 1 \\ 0 \leq \theta_2 \leq 1 \\ 0 \leq \theta_3 \leq 1 \\ 0 \leq \theta_4 \leq 1 \\ a_2 \leq \theta_1 \leq b_2 \\ a_3 \leq \theta_3 \leq b_3 \end{cases} \end{aligned} \quad (22)$$

求得参数约束域的4个边界点,进而将这4个点的坐标进行平均,用平均所得的点来近似替代参数约束域的中心点。

假设参数约束中不包含分布间约束,便可利用参数的局部独立性将其分解。具体如下所示:

$$\begin{aligned} & (\theta_1, \theta_2) = \underset{\theta_1, \theta_2}{\operatorname{argmin}} [(\theta_1 - 0)^2 + (\theta_2 - 1)^2] \\ & \begin{cases} \theta_1 + \theta_2 = 1 \\ \theta_1 > \theta_2 \end{cases} \\ \text{s.t.} & \begin{cases} 0 \leq \theta_1 \leq 1 \\ 0 \leq \theta_2 \leq 1 \\ a_2 \leq \theta_1 \leq b_2 \end{cases} \end{aligned} \quad (23)$$

$$\begin{aligned} & (\theta_1, \theta_2) = \underset{\theta_1, \theta_2}{\operatorname{argmin}} [(\theta_1 - 1)^2 + (\theta_2 - 0)^2] \\ & \begin{cases} \theta_1 + \theta_2 = 1 \\ \theta_1 > \theta_2 \end{cases} \\ \text{s.t.} & \begin{cases} 0 \leq \theta_1 \leq 1 \\ 0 \leq \theta_2 \leq 1 \\ a_2 \leq \theta_1 \leq b_2 \end{cases} \end{aligned} \quad (24)$$

$$\begin{aligned} & (\theta_3, \theta_4) = \underset{\theta_3, \theta_4}{\operatorname{argmin}} [(\theta_3 - 0)^2 + (\theta_4 - 1)^2] \\ & \begin{cases} \theta_3 + \theta_4 = 1 \\ \theta_3 < \theta_4 \end{cases} \\ \text{s.t.} & \begin{cases} 0 \leq \theta_3 \leq 1 \\ 0 \leq \theta_4 \leq 1 \\ a_3 \leq \theta_3 \leq b_3 \end{cases} \end{aligned} \quad (25)$$

$$\begin{aligned} & (\theta_3, \theta_4) = \underset{\theta_3, \theta_4}{\operatorname{argmin}} [(\theta_3 - 1)^2 + (\theta_4 - 0)^2] \\ & \begin{cases} \theta_3 + \theta_4 = 1 \\ \theta_3 < \theta_4 \end{cases} \\ \text{s.t.} & \begin{cases} 0 \leq \theta_3 \leq 1 \\ 0 \leq \theta_4 \leq 1 \\ a_3 \leq \theta_3 \leq b_3 \end{cases} \end{aligned} \quad (26)$$

由(23)~(26)式可知,当参数约束中不存在分布间约束时,可以将原来4维的参数约束空间分解为2个2维的参数约束空间,降低待求问题的复杂度。

4 仿真结果及分析

为了验证本文算法的有效性,分别利用 Asia、Alarm、Win95PTs 和 Andes 网络进行仿真验证。这些网络经常被用于贝叶斯网络的结构和参数学习算法验证,表2给出了这几个网络的基本信息。采用

公式(19)~(22)给出了利用标准基向量构建目标优化函数的方法。利用公式(19)~(22)可以

平均 KL 距离来衡量参数学习的精度,公式 (27) 给出了 KL 距离的计算公式,其中 $P(X)$ 表示求出的样本分布, $Q(X)$ 表示真实的样本分布。采用平均运行时间来衡量算法的效率。本文中的运行时间是通过 Matlab 中的 Tic-Toc 指令获取。算法运行 20 次,求均值和方差。

$$d_{\text{KL}}(P,Q) = \sum_x P(X) \log_2 \frac{P(X)}{Q(X)} \quad (27)$$

表 2 仿真网络信息

仿真网络	网络规模	节点数	边的数目	参数个数
Asia	小	8	8	18
Alarm	中	37	46	509
Win95PTs	大	76	112	574
Andes	超大	223	338	1 157

4.1 仿真实验 1

本节考虑参数的分布间约束,部分仿真结果如表 3 所示。

表 3 不同算法的 KL 距离比较

仿真网络	样本数量	ML	ME	MAP	CML	CME	QMAP	CMAP
Asia	40	15.52±3.50	3.90±0.04	3.36±0.18	3.60±0.57	1.82±0.07	0.82±0.14	1.50±0.04
	80	14.35±3.64	3.73±0.06	2.62±0.44	2.76±0.34	1.52±0.02	0.60±0.06	1.16±0.07
	120	10.17±2.93	2.81±0.24	1.98±0.11	3.67±0.35	1.15±0.13	0.45±0.04	0.95±0.09
	160	4.86±2.94	2.50±0.28	1.56±0.45	3.56±0.18	1.04±0.0 7	0.42±0.06	0.84±0.04
	200	2.56±1.58	2.28±0.41	1.05±0.54	1.79±0.18	0.84±0.09	0.38±0.06	0.71±0.04
Alarm	40	273.04±22.29	139.84±2.81	118.75±7.37	95.88±6.54	60.46±1.72	33.78±1.17	58.14±1.87
	80	275.49±25.36	124.45±1.75	99.27±3.90	102.91±6.29	52.45±1.48	29.78±0.75	49.39±1.59
	120	262.14±22.68	111.18±5.14	89.46±5.95	96.93±5.50	47.84±2.90	27.55±1.19	45.77±3.07
	160	243.75±19.58	106.12±5.25	85.46±3.50	97.37±4.52	45.97±2.84	27.00±1.20	43.86±2.10
	200	229.07±23.50	101.64±3.93	80.39±5.20	94.01±6.24	43.96±2.09	25.67±1.16	42.07±1.80
Win95PTs	40	269.71±22.86	152.27±1.38	134.34±2.66	126.24±6.01	96.66±0.74	57.98±1.51	89.70±1.64
	80	283.48±41.42	142.88±1.39	124.96±5.51	127.38±8.50	89.38±1.11	55.06±0.66	82.96±1.03
	120	308.45±42.10	136.33±1.45	121.07±6.28	118.95±7.32	85.32±0.88	53.44±0.36	79.86±0.27
	160	305.57±15.75	131.81±1.66	114.93±1.34	98.69±6.61	82.41±0.71	51.36±0.19	76.86±0.57
	200	304.90±30.19	127.80±1.49	114.46±2.38	98.19±2.66	80.01±0.12	50.53±0.60	74.90±0.60
Andes	40	1 252.65±37.19	271.30±1.96	228.02±6.60	744.61±30.50	119.50±1.62	43.24±1.97	121.32±3.84
	80	1 046.26±44.69	215.98±1.05	187.57±6.53	636.29±33.50	93.18±1.02	36.40±0.33	92.15±2.75
	120	892.49±33.50	185.69±1.70	160.63±5.56	566.77±15.83	78.12±1.53	31.11±2.28	75.45±4.80
	160	868.22±48.74	165.74±2.35	150.41±5.35	542.73±2.77	68.62±1.30	28.80±2.38	67.45±3.73
	200	780.04±52.41	150.57±1.53	135.50±7.81	515.63±7.81	61.99±1.07	26.44±0.36	60.88±1.17

表 4 不同算法的运行时间比较

仿真网络	样本数量	ML/ 10^{-3}	ME/ 10^{-3}	MAP/ 10^{-3}	CML/ 10^{-3}	CME/ 10^{-3}	QMAP/ 10^{-3}	CMAP/ 10^{-3}
Asia	40	0.1±0.0	82.0±0.4	0.2±0.0	231.8±4.3	218.5±4.3	178.4±0.3	30.3±0.1
	80	0.2±0.0	91.2±0.2	0.3±0.0	236.4±7.4	218.2±7.4	197.8±0.3	49.4±0.1
	120	0.2±0.0	84.9±0.3	0.4±0.0	242.1±4.3	219.7±1.3	217.5±0.3	68.6±0.3
	160	0.3±0.0	82.6±0.2	0.5±0.0	247.5±5.9	223.0±5.9	236.4±0.3	87.4±0.5
	200	0.3±0.0	83.5±0.5	0.6±0.0	247.6±2.4	223.5±2.4	255.9±0.7	106.8±0.4
Alarm	40	0.1±0.0	85.8±1.0	0.1±0.0	89.2±0.9	123.9±0.9	412.0±0.4	81.1±1.8
	80	0.2±0.0	89.8±0.9	0.2±0.0	96.9±1.9	125.8±0.6	432.1±0.7	106.3±2.8
	120	0.3±0.0	92.1±0.8	0.3±0.0	99.7±1.1	126.2±1.4	452.8±0.5	132.6±3.1
	160	0.4±0.0	91.1±0.9	0.4±0.0	103.5±0.2	128.1±1.1	471.6±0.7	154.7±5.1
	200	0.5±0.0	92.0±0.7	0.5±0.0	105.3±1.3	129.6±1.0	492.0±0.7	176.0±8.8
Win95PTs	40	0.1±0.0	90.9±0.6	0.1±0.0	84.9±0.6	117.0±1.1	312.4±33.9	152.6±0.2
	80	0.2±0.0	93.2±0.6	0.2±0.0	89.3±0.3	117.9±0.2	343.9±38.2	172.3±0.3
	120	0.3±0.0	93.7±0.3	0.3±0.0	92.2±0.4	118.3±0.7	381.5±18.0	191.2±0.6
	160	0.4±0.0	95.0±0.8	0.4±0.0	94.4±0.3	118.7±0.5	410.2±18.0	211.3±0.4
	200	0.5±0.0	95.0±0.9	0.5±0.0	95.4±0.4	119.7±0.2	433.6±17.4	230.3±0.2
Andes	40	0.2±0.0	111.7±0.1	0.2±0.0	135.9±0.8	144.8±0.5	179.8±0.3	55.5±0.5
	80	0.3±0.0	109.9±0.3	0.3±0.0	143.2±0.1	144.8±0.4	203.3±0.1	75.7±0.2
	120	0.4±0.0	109.6±0.1	0.4±0.0	147.1±0.6	144.8±0.1	226.9±0.2	96.5±0.2
	160	0.5±0.0	110.0±0.2	0.5±0.0	149.3±0.8	144.0±0.1	250.2±0.2	117.1±0.2
	200	0.6±0.0	110.1±0.5	0.6±0.0	151.7±0.2	144.7±0.5	271.9±0.2	137.7±0.1

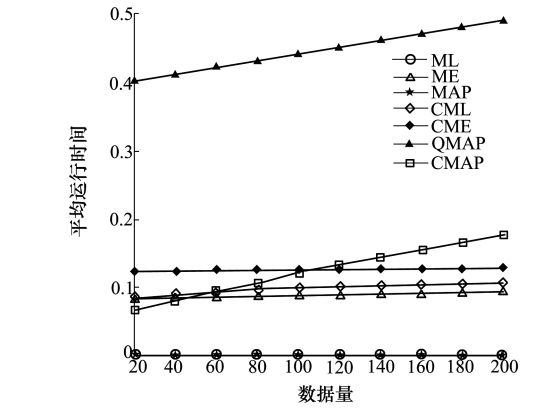


图 4 不同算法的平均运行时间比较 (Alarm 网络)

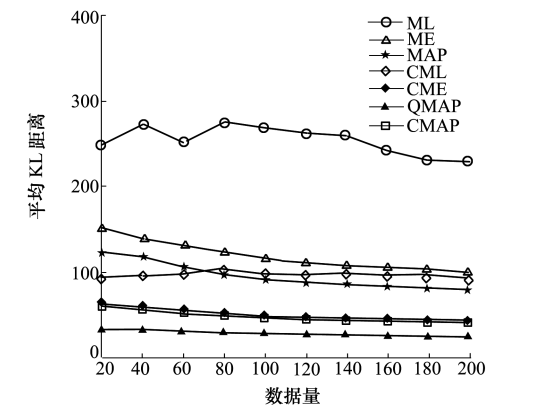


图 5 不同算法的平均 KL 距离比较 (Alarm 网络)

通过对表 3~4 和图 4~5 中的实验结果进行分析,可以得到以下结论:

1) 所有算法的学习精度排序为 QMAP>CMAP>others。

2) 所有算法运行时间的大致排序为 QMAP>CME>CMAP>others。

3) 本文提出的 CMAP 的学习精度稍差于 QMAP 算法,但运行效率高于 QMAP 算法。

4.2 仿真实验 2

本节不考虑参数的分布间约束,则可通过公式(16)对公式(15)进行降维,通过对表 5~6 和图 6~7 中的实验结果进行分析,可以得到以下结论:

1) 所有算法的学习精度排序为 QMAP>CMAP>others。
所有算法的运行时间的大致排序为 QMAP>CME>CMAP>others

表 5 不同算法的 KL 距离比较

仿真网络	样本数量	ML	ME	MAP	CML	CME	QMAP	CMAP
Asia	40	7.58±1.50	3.85±0.28	2.28±0.90	5.77±1.57	1.91±0.35	0.83±0.16	1.49±0.13
	80	8.60±2.52	3.11±0.11	2.16±0.87	6.99±1.34	1.63±0.41	0.73±0.06	1.31±0.27
	120	9.02±2.26	2.82±0.37	1.63±0.27	7.62±1.35	1.36±0.13	0.70±0.08	1.07±0.10
	160	7.39±2.62	2.46±0.14	1.50±0.30	6.54±1.18	1.26±0.07	0.62±0.12	1.02±0.09
	200	6.40±2.58	2.35±0.10	1.47±0.29	6.50±1.18	1.22±0.06	0.58±0.06	1.01±0.14
Alarm	40	291.17±33.07	139.75±3.69	118.26±11.76	119.34±10.73	85.48±3.71	41.09±2.41	79.87±5.17
	80	250.65±25.93	124.12±5.21	97.88±7.70	119.27±10.87	74.33±3.12	37.46±1.81	69.73±3.45
	120	231.28±26.27	113.62±3.91	89.23±6.45	114.79±7.04	68.68±2.65	34.71±1.33	63.94±2.95
	160	243.75±19.58	106.85±3.09	82.50±4.61	105.81±4.52	64.76±1.97	32.64±1.06	59.84±2.50
	200	204.81±15.27	99.78±1.97	78.55±4.62	99.48±4.24	60.95±1.75	30.32±1.10	56.96±2.24
Win95PTs	40	243.21±34.33	153.13±1.20	133.16±6.02	128.15±6.01	106.36±2.50	64.34±1.07	99.38±2.38
	80	314.25±51.42	143.58±1.57	127.32±3.70	152.78±8.50	97.97±1.26	60.56±1.44	91.02±2.30
	120	342.26±52.10	136.31±0.87	126.46±3.90	156.71±11.32	92.79±1.16	57.56±1.79	86.22±2.24
	160	360.02±37.02	130.75±1.24	123.28±2.56	161.05±16.61	86.15±0.57	55.49±1.15	76.86±0.57
	200	355.96±50.76	127.14±0.74	119.28±4.84	145.57±22.66	132.26±1.70	53.56±0.94	79.12±1.11
Andes	40	1 365.36±70.07	272.13±2.01	242.97±7.91	785.21±27.65	132.26±1.70	48.97±1.37	130.08±1.98
	80	1 145.18±63.38	215.28±1.93	201.50±6.03	681.91±30.42	101.50±1.47	40.71±1.07	99.21±1.03
	120	960.89±56.62	185.86±2.79	170.67±7.11	604.48±28.40	85.44±1.13	34.64±0.95	82.79±2.08
	160	900.49±74.10	165.42±2.85	155.44±8.54	566.47±31.05	74.51±1.17	30.61±0.30	72.73±1.20
	200	852.74±67.54	148.97±2.67	143.92±9.74	530.65±29.11	66.20±1.11	28.33±1.04	65.17±0.93

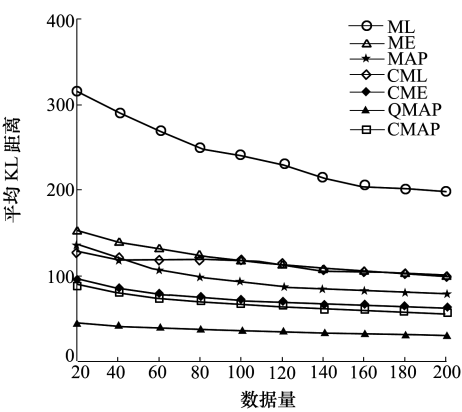


图 6 不同算法的平均 KL 距离比较 (Alarm 网络)

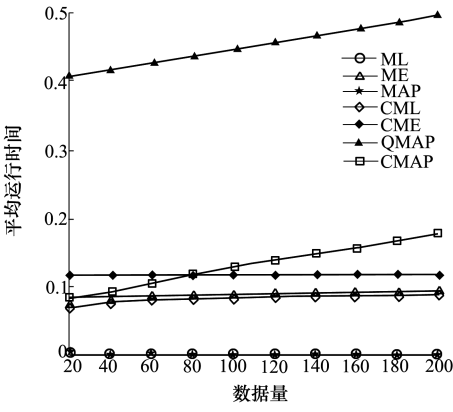


图 7 不同算法的平均运行时间比较 (Alarm 网络)

表 6 不同算法的运行时间比较

仿真网络	样本数量	ML/ 10^{-3}	ME/ 10^{-3}	MAP/ 10^{-3}	CML/ 10^{-3}	CME/ 10^{-3}	QMAP/ 10^{-3}	CMAP/ 10^{-3}
Asia	40	0.2±0.0	192.0±5.2	0.2±0.0	211.5±9.0	209.9±3.4	182.3±0.6	32.0±0.2
	80	0.3±0.0	188.5±5.7	0.3±0.0	214.6±9.1	208.6±4.0	201.4±0.4	51.9±0.2
	120	0.4±0.0	192.8±4.4	0.4±0.0	223.1±4.5	209.5±3.5	223.6±0.6	72.6±0.2
	160	0.5±0.0	191.0±5.2	0.5±0.0	230.1±5.9	209.4±2.8	242.2±0.7	92.6±0.6
	200	0.6±0.0	192.7±2.5	0.6±0.0	235.7±6.8	210.7±2.9	262.4±1.4	113.0±0.9
Alarm	40	0.1±0.0	85.1±0.8	0.1±0.0	75.7±1.3	114.4±1.1	416.6±0.6	81.1±1.8
	80	0.2±0.0	87.4±1.2	0.2±0.0	81.9±1.8	115.2±1.1	436.9±0.5	116.7±11.7
	120	0.3±0.0	90.6±2.2	0.3±0.0	84.5±1.4	116.7±0.9	457.0±0.8	137.5±13.7
	160	0.4±0.0	91.0±1.5	0.4±0.0	86.0±1.2	116.9±0.5	477.1±0.6	155.6±10.6
	200	0.5±0.0	92.4±1.8	0.5±0.0	88.5±1.4	117.0±0.5	497.0±0.4	177.8±13.2
Win95PTs	40	0.1±0.0	93.5±0.5	0.1±0.0	75.9±0.9	111.7±0.6	138.9±0.2	215.2±29.4
	80	0.2±0.0	95.1±0.4	0.2±0.0	80.9±1.4	112.0±0.7	158.6±0.2	272.2±36.4
	120	0.3±0.0	95.9±0.8	0.3±0.0	83.6±1.4	111.8±0.3	178.7±0.1	307.2±25.6
	160	0.4±0.0	96.8±0.8	0.4±0.0	86.0±1.3	112.2±0.4	198.8±0.4	329.1±20.4
	200	0.5±0.0	97.7±1.0	0.5±0.0	87.3±1.2	119.7±0.4	218.8±218.8	353.4±23.7
Andes	40	0.2±0.0	112.7±0.8	0.2±0.0	122.4±0.7	134.4±0.5	171.9±0.2	78.3±7.4
	80	0.3±0.0	111.1±0.5	0.3±0.0	129.2±0.6	133.6±0.3	192.1±0.2	98.7±8.6
	120	0.4±0.0	111.4±0.6	0.4±0.0	133.0±0.4	133.7±0.4	212.4±0.1	120.8±10.5
	160	0.5±0.0	111.2±0.3	0.5±0.0	136.9±0.5	133.4±0.3	232.5±0.3	140.4±8.9
	200	0.6±0.0	111.2±0.2	0.6±0.0	139.0±0.5	133.3±0.3	252.7±0.3	160.6±8.1

5 结 论

为了提高 QMAP 算法的学习效率同时又不影响其学习精度,本文设计了一种新的约束区域中心点的解析计算方法来替代原有的拒绝-接受采样计算方法。仿真实验证明,本文提出的 CMAP 算法的

参数学习精度稍差于 QMAP 算法,但计算效率比 QMAP 算法高出一个量级。缺点是 CMAP 算法仍涉及最优化问题求解的引擎,引擎的优劣直接决定了算法的时效性,下一步研究工作是引入更高效的引擎用于求解参数约束域的中心点,进一步提高本文提出的参数学习方法的效率。

参考文献:

[1] WITTIG F, JAMESON A. Exploiting qualitative knowledge in the learning of conditional probabilities of Bayesian networks [C]//Proceedings of the Sixteenth International Conference on Uncertainty in Artificial Intelligence, California, 2000

[2] FEELDERS A, GAAG L. Learning Bayesian network parameters under order constraints[J]. International Journal of Approximate Reasoning, 2006, 42(1/2): 37-53

[3] MASEGOSA A R, FEELDERS A J, GAAG L C V D. Learning from incomplete data in Bayesian networks with qualitative influences[J]. International Journal of Approximate Reasoning, 2016, 69: 18-34

[4] MARTIN Plajner, JIRÍVomlel. Learning bipartite Bayesian networks under monotonicity restrictions[J]. International Journal of

General Systems, 2020, 49(1): 88-111

- [5] ZHOU Y, FENTON N, ZHU C. An empirical study of Bayesian network parameter learning with monotonic influence constraints [J]. Decision Support Systems, 2016, 87: 69-79
- [6] 邸若海, 高晓光, 郭志高. 基于单调性约束的离散贝叶斯网络参数学习[J]. 系统工程与电子技术, 2014, 18(5): 272-277
- DI Ruohai, GAO Xiaoguang, GUO Zhigao. Parameter learning of discrete Bayesian networks based on monotonicity constraints [J]. Systems Engineering and Electronics, 2014, 18(5): 272-277 (in Chinese)
- [7] DI Ruohai, GAO Xiaoguang, GUO Zhigao. Learning Bayesian networks parameters under new monotonic constraints[J]. Journal of Systems Engineering and Electronics, 2017, 28(6): 1248-1255
- [8] 任佳, 高晓光, 白勇. 信息不完备小样本条件下离散 DBN 参数学习[J]. 系统工程与电子技术, 2012, 34(8): 1723-1728
- REN Jia, GAO Xiaoguang, BAI Yong. Discrete DBN parameter learning under the condition of incomplete information with small samples[J]. Systems Engineering and Electronics, 2012, 34(8): 1723-1728 (in Chinese)
- [9] 曾强, 黄政, 魏曙寰. 融合专家先验知识和单调性约束的贝叶斯网络参数学习方法[J]. 系统工程与电子技术, 2020, 42(3): 642-656
- ZENG Qiang, HUANG Zheng, WEI Shuhuan. Bayesian network parameter learning method based on expert prior knowledge and monotonicity constraint[J]. Systems Engineering and Electronics, 2020, 42(3): 642-656 (in Chinese)
- [10] NICULESCU R, MITCHELL T, RAO B. Bayesian network learning with parameter constraints[J]. Journal of Machine Learning Research, 2006, 7(1): 1357-1383
- [11] CAMPOS C, JI Q. Improving Bayesian network parameter learning using constraints[C]//Proceedings of the Nineteenth International Conference on Pattern Recognition in Florida, 2008
- [12] LIAO Wenhui, JI Qiang. Learning Bayesian network parameters under incomplete data with domain knowledge[J]. Pattern Recognition, 2009, 42(11): 3046-3056
- [13] ZHOU Y, FENTON F, NEIL M. Bayesian network approach to multinomial parameter learning using data and expert judgments [J]. International Journal of Approximate Reasoning, 2014, 55(5): 1252-1268
- [14] ZHOU Y, FENTON N, HOSPADALES T, et al. Probabilistic graphical models parameter learning with transferred prior and constraints[C]//Proceedings of the Thirty First International Conference on Uncertainty in Artificial Intelligence, 2015
- [15] 郭志高, 高晓光, 邸若海. 小数据集条件下基于双重约束的 BN 参数学习[J]. 自动化学报, 2014, 40(7): 1509-1516
- GUO Zhigao, GAO Xiaoguang, DI Ruohai. BN parameter learning based on double constraints under the condition of small data-set[J]. Acta Automatica Sinica, 2014, 40(7): 1509-1516 (in Chinese)
- [16] RUI Chang, Shoemaker R, WEI Wang. A novel knowledge-driven systems biology approach for phenotype prediction upon genetic intervention[J]. IEEE/ACM Transactions on Computational Bioinformatics, 2011, 3(8): 683-697
- [17] GUO Z G, GAO X G, REN H, et al. Learning Bayesian network parameters from small data sets: a further constrained qualitatively maximum a posteriori method[J]. International Journal of Approximate Reasoning, 2017, 91(12): 22-35
- [18] HECKERMAN D, WELLMAN M P. Bayesian networks[J]. Communications of ACM, 1995, 38(3): 27-30

Bayesian network parameter learning algorithm based on improved QMAP

DI Ruohai¹, LI Ye¹, WAN Kaifang², LYU Zhigang¹, WANG Peng¹

(1.School of Electronic and Information Engineering, Xi'an Technological University, Xi'an 710021, China;
2.School of Electronic and Information, Northwestern Polytechnical University, Xi'an 710072, China)

Abstract: Small data sets make the statistical information in Bayesian network parameter learning inaccurate, which makes it difficult to get accurate Bayesian network parameters based on data. Qualitative maximum a posteriori estimation (QMAP) is the most accurate algorithm for Bayesian network parameter learning under the condition of small data sets. However, when the number of parameter constraints is large or the parameter feasible region is small, the rejection-acceptance sampling process in QMAP algorithm will become extremely time-consuming. In order to improve the learning efficiency of QMAP algorithm and not affect its learning accuracy as much as possible, a new analytical calculation method of the center point of constrained region is designed to replace the original rejection-acceptance sampling calculation method. Firstly, a new objective function is designed, and a constrained objective optimization problem for solving the boundary points of the constrained region is constructed. Secondly, a new optimization engine is used to solve the objective optimization problem, and the boundary points and center points of the constrained region are obtained. Finally, the existing QMAP algorithm is improved by the obtained center points. The simulation results show that the CMAP algorithm proposed in this paper has a slightly worse parameter learning accuracy than the QMAP algorithm, but its computational efficiency is 2–5 times higher than that of the QMAP algorithm.

Keywords: Bayesian network; parameter learning; qualitative maximum a posteriori estimation; parameter constraints; objective optimization

引用格式:邸若海, 李叶, 万开方, 等. 基于改进 QMAP 的贝叶斯网络参数学习算法[J]. 西北工业大学学报, 2021, 39(6): 1356-1367

DI Ruohai, LI Ye, WAN Kaifang, et al. Bayesian network parameter learning algorithm based on improved QMAP[J]. Journal of Northwestern Polytechnical University, 2021, 39(6): 1356-1367 (in Chinese)