计算机研究与发展
Journal of Computer Research and Development
DOI:10.7544/issn1000-1239.2017.20160715
54(11):2558-2566, 2017

基于超结构的 BN 随机搜索学习算法

吕亚丽[1,2] 武佳杰[2] 梁吉业[1] 钱宇华[1]
[1](计算智能与中文信息处理教育部重点实验室(山西大学) 太原 030006)
[2](山西财经大学信息管理学院 太原 030006)
(sxlvyali@163.com)

Random Search Learning Algorithm of BN Based on Super-Structure

Lü Yali[1,2], Wu Jiajie[2], Liang Jiye[1], and Qian Yuhua[1]
[1](*Key Laboratory of Computational Intelligence and Chinese Information Processing (Shanxi University), Ministry of Education, Taiyuan* 030006)
[2](*School of Information Management, Shanxi University of Finance & Economics, Taiyuan* 030006)

Abstract Recently, Bayesian network (BN) plays a vital role in knowledge representation and probabilistic inference. BN structure learning is crucial to research on BN inference. However, there are some disadvantages in the most two-stage hybrid learning method of BN structure: it is easy to lose edges with weak relationship in the first stage, when we learn the super-structure; hill climbing search method is easily plunged into local optimum in the second stage. To avoid the two disadvantages, the super-structure of BN is firstly learned by Opt01ss algorithm, which makes the result miss few edges as much as possible. Secondly, based on super-structure, three search operators are given to analyze the random selection rule of the initial network and address the random optimization strategy for the initial network. Further, SSRandom learning algorithm of BN structure is proposed. The algorithm is a good way to jump out of local optimum extremum to a certain extent. Finally, the learning performance of the proposed SSRandom algorithm is verified by the experiments on the standard Survey, Asia and Sachs networks, by comparing with other three hybrid algorithms according to four evaluation indexs, such as the sensitivity, specificity, Euclidean distance and the percentage of overall accuracy.

Key words Bayesian network (BN); structure learning; random search; super-structure; hybrid algorithm

摘 要 近年来,贝叶斯网络(Bayesian network, BN)在不确定性知识表示与概率推理方面发挥着越来越重要的作用.其中,BN 结构学习是 BN 推理中的重要问题.然而,在当前 BN 结构的 2 阶段混合学习算法中,大多存在一些问题:第 1 阶段无向超结构学习中存在容易丢失弱关系的边的问题;第 2 阶段的

收稿日期:2016-09-20;**修回日期**:2017-02-21
基金项目:国家自然科学基金重点项目(61432011);军民共用重大研究计划联合基金重点项目(U1435212);国家自然科学基金优秀青年科学基金项目(61322211);国家自然科学基金项目(61672332);中国博士后科学基金项目(2016M591409);山西省自然科学基金项目(2013011016-4,2014011022-2)
This work was supported by the Key Program of the National Natural Science Foundation of China (61432011), the Key Program of Plan Joint Foundation for Military and Civilian Sharing Major Research (U1435212), the National Natural Science Foundation of China for Excellent Young Scientists (61322211), the National Natural Science Foundation of China (61672332), the China Postdoctoral Science Foundation (2016M591409), and the Natural Science Foundation of Shanxi Province (2013011016-4, 2014011022-2).

爬山搜索算法存在易陷入局部最优的问题.针对这 2 个问题,首先采用 Opt01ss 算法学习超结构,尽可能地避免出现丢边现象;然后给出基于超结构的搜索算子,分析初始网络的随机选择规则和对初始网络随机优化策略,重点提出基于超结构的随机搜索的 SSRandom 结构学习算法,该算法一定程度上可以很好地跳出局部最优极值;最后在标准 Survey, Asia, Sachs 网络上,通过灵敏性、特效性、欧几里德距离和整体准确率 4 个评价指标,并与已有 3 种混合学习算法的实验对比分析,验证了该学习算法的良好性能.

关键词 贝叶斯网络;结构学习;随机搜索;超结构;混合算法

中图法分类号 TP18

贝叶斯网络(Bayesian network, BN)模型[1]是人工智能学科表示并处理概率知识的一种重要方法.该模型已被广泛应用于基因调控、医疗诊断、故障检测、金融预测分析等方面.其中,BN 学习[2-10]是目前 BN 研究的主要热点之一,也是该模型推向应用的前提基础.

BN 学习主要包括结构学习[2-7]和参数学习[8-10] 2 部分.本文主要集中于对其结构学习的研究.目前,对于 BN 结构学习算法大致可分为 3 类:

1) 基于约束(constraint based, CB)或者依赖分析的算法[3],即利用统计学或者信息论中的知识,对变量之间的依赖性进行测试,分析出各变量之间的关系,然后利用这些依赖性关系构建 BN 模型.该类算法过程比较复杂,但在一些假设下可以得到最优的 BN 结构,并且算法可具有多项式时间复杂度,该类算法比较适合于建立变量多、网络稀疏的 BN 结构学习.

2) 基于评分搜索的算法,即对所有可能的 BN 结构利用某种优化算法进行搜索,然后对搜索到的 BN 进行评分[4-5],以期待找出得分较高的 BN 结构.该类算法过程简单规范,但打分函数的运算复杂程度和结构搜索空间大小都随变量增加呈指数增加,通常以解决时间复杂性高的算法有:①给定节点顺序,进行局部打分,如经典的 K2 算法[6];②选择启发式搜索策略进行整体打分等.该类算法比较适合于变量较少、网络较稠密的 BN 结构学习.

3) 目前较为流行的混合学习算法[7,11-12],该类算法结合上述 2 类算法的优点,首先基于约束的方法系统检查条件独立性关系,得出网络的基本无向图框架(skeleton)[7,13],然后基于此无向图框架采用评分搜索算法来学习整个 BN 结构.其中较典型的混合学习算法有 MMHC(max-min hill-climbing)[7].在 MMHC 算法中,首先利用 MMPC(max-min parents and children)算法得到网络的无向图框架,然后利用爬山算法得到 BN 结构.但由于 MMPC 得到的无向图框架中会丢失应有的一些边,即存在丢边问题.另外,对于经典的爬山算法众所周知的缺点是易陷入局部最优.

2014 年,Villanueva 和 Maciel[14]提出一种有效学习 BN 超结构(super-structure, SS)的 Opt01ss 算法,该算法在有限数据情况下,一定程度上可以避免条件独立性(conditional independence, CI)测试不一致性和近似确定关系(approximate deterministic relationships, ADRs)等问题,尽可能减少了弱关系下边的丢失问题.因此,本文借鉴文献[14]中的超结构学习算法,重点研究随机搜索算法,并考虑解决陷入局部极值问题,提出一种基于超结构的随机搜索 BN 结构的混合学习算法.

1 基础知识

本节主要介绍 BN 模型与其评价准则、超结构框架及其 Opt01ss 学习算法.

1.1 BN 模型与其评价准则

BN 模型是图论和概率论的"结合体",主要由有向无环图(directed acyclic graph, DAG)结构和每个节点相对应的条件概率分布表(conditional probability table, CPT)组成.该模型可形式化表示成 $\mathcal{M}=(\mathcal{G},\mathcal{P})$,其中 $\mathcal{G}=(V,E)$表示 BN 的 DAG 结构.V 表示节点变量;E 表示边集,反映变量间的相互依赖关系;$\mathcal{P}$ 表示每个节点相对应的 CPT 集合,反映父子节点变量之间的依赖程度.

给定数据集 $\mathcal{D}$,BN 结构学习指从数据中获得与数据集 $\mathcal{D}$ 拟合程度较好的 DAG 结构.评价网络结构与样本数据拟合程度的函数称为评分函数.常见的评分函数有多种,如贝叶斯信息评分准则(BIC)[15]、CH 评分[6]、最小描述长度(MDL)[5]等,这里简单介绍本文所采用的 BIC 评分函数,该函数

的评分越高,说明从数据中获得的 DAG 网络结构越好. 其具体形式为

$$f_{\mathrm{BIC}}(\mathcal{D}|\mathcal{M})=\log P(\mathcal{D}|\mathcal{M},\hat{\theta})-\frac{d}{2}\log N,$$

其中,$\hat{\theta}$ 表示给定数据 $\mathcal{D}$ 对模型 $\mathcal{M}$ 的最大似然参数;$d=\sum_{i=1}^{n}q_i(r_i-1)$ 表示模型的参数个数,q_i 是节点变量 V_i 的父节点组合数,r_i 是节点变量 V_i 的取值个数;N 是样本量的大小.

1.2 超结构

对 BN 结构的混合学习算法,一般来说,首先基于约束关系构建出其无向框架图,即其超结构 SS;然后在 SS 的基础上运用搜索评分方法优化 BN 结构,得到评分较高的网络.

定义 1. 超结构 SS[13-14]. 对于一个 DAG $\mathcal{G}$,若一个无向图包含 $\mathcal{G}$ 去掉方向之后的所有边,即完全包含该 DAG 的基本架构,则称该无向图是 $\mathcal{G}$ 的一个完全超结构,否则称为不完全超结构. 若无特别说明,本文均指完全超结构.

近年来,许多学者研究了 BN 超结构的学习算法[13-14,16]. 其中,Villanueva 和 Maciel[14] 提出的 Opt01ss 算法取得了较好的效果. 该算法解决了在样本量有限的情况下,其他超结构学习算法容易出现的 2 个问题:1)CI 测试不一致的问题;2)如图 1 所示的 ADRs 问题,即避免了存在较弱关系的边的丢失问题,保留了尽可能多的边.

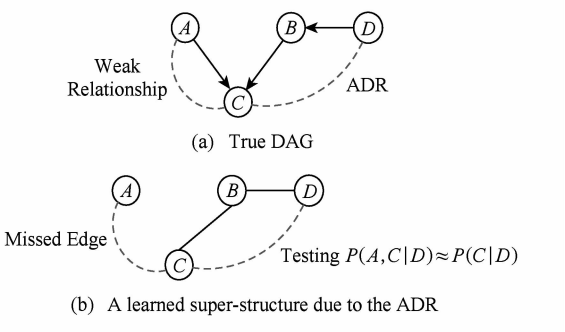


Fig. 1 An example to illustrate ADR problem

图 1 ADR 问题解释示例

由此,本文采用该 Opt01ss 算法思想获得 BN 超结构,然后重点进行随机搜索找出较优的 BN 结构.

2 基于 SS 的随机搜索学习算法 SSRandom

在基于 Opt01ss 算法获得 SS 的基础上,本节主要针对爬山算法中网络最终结果对初始结构的依赖度大,导致易陷入局部极值的问题,研究基于 SS 的随机搜索算法. 具体地,首先介绍基于 SS 的 3 种搜索算子,然后分析基于 SS 的初始网络的随机选取规则,接着讨论使得初始网络避免陷入局部极值的随机优化策略,进一步提出基于 SS 的随机搜索(random search based on SS, SSRandom)的学习算法.

2.1 基于 SS 的搜索算子

由于本文第 1 阶段得出的 SS 已包含最优 BN 去掉方向后的所有边,因此,候选 DAG 的 3 个搜索算子主要指:

1) 添方向算子,记作 O_{Add}. 该算子不同于爬山法中的加边算子,它指将超结构中的边加一个方向后添加到当前 DAG 中.

2) 减边算子,记作 O_{Sub}. 即在当前 DAG 中减去一条边.

3) 转方向算子,记作 O_{Rev}. 即将当前 DAG 中的一条边的方向进行翻转.

需要注意的是:与爬山法相同,加边和转边的前提是保证该图仍保持无环,即保持候选网络仍是 DAG 结构.

2.2 基于 SS 的初始网络的随机选取规则

定义 2. 基于 SS 的随机 DAG——SSRDAG. 通过基于 SS 的 3 种搜索算子,对某 DAG 随机进行一步或多步 O_{Add},O_{Sub},O_{Rev} 操作后得到的 DAG,即称为该 DAG 基于 SS 的随机 DAG,记作 SSRDAG.

在爬山法中,其初始网络是一个无边图,在无边图的基础上进行一步搜索选择得分较高的候选网络作为当前网络,并在此基础上继续优化. 可知,初始无边网络相对距离最优网络较远. 本文探讨的初始网络是距离最优网络要较近的网络. 为了加快搜索,我们初始网络的选择是保留 SS 的所有边,只是在不形成环路的前提下,给每条无向边通过 O_{Add} 算子随机添加一个方向,并随机搜索 m 次,得出 m 个 SSRDAGs,并将 BIC 得分最高的 SSRDAG 作为初始模型 $\mathcal{G}^0$. 可知,该初始网络的选择较爬山法来说有 2 个好处:1)离最优网络更近了一步;2)初始网络是 m 个中根据得分函数选出的,自然这时选择的初始网络相对较优.

2.3 对初始网络的随机优化策略

由于初始网络 $\mathcal{G}^0$ 中已包含 SS 中的所有无向边,因而对其优化时只需要考虑减边 O_{Sub} 和转方向 O_{Rev} 操作,不需要添方向 O_{Add} 操作. 初始优化时,首先将初始 $\mathcal{G}^0$ 作为当前需优化的模型 $\mathcal{G}^*$,然后随机选择 O_{Sub} 和 O_{Rev} 操作 m 次,得到 m 个 SSRDAGs,即得出 m 个候选的 $\mathcal{G}_j^*$ $(1\leqslant j\leqslant m)$,并对当前 $\mathcal{G}^*$ 和候选

的 $\mathcal{G}_j^*$ $(1\leqslant j\leqslant m)$共 $m+1$ 个 SSRDAGs 按 BIC 得分进行非升序排列,将得分最高网络记为当前网络模型 $\mathcal{G}^*$.

为了继续优化当前网络 $\mathcal{G}^*$,并考虑避免结果陷入局部最优,文中引入 2 个 SSRDAGs 随机交叉的概念.

定义 3. 2 个 SSRDAGs 的随机交叉. 对于 2 个 SSRDAGs 模型,其随机交叉指随机选择 1 对节点或多对节点,并互相交叉选定的边后得到 2 个新 SSRDAGs 的操作.

例 1. 对于常见的一个关于年龄 A、性别 G、抽烟 S 和肺癌 LC 的 BN 实例,已知其 DAG 结构包括的有向边有$\langle A,S\rangle$,$\langle G,S\rangle$,$\langle S,LC\rangle$,下面以该网络的一个包含无向边(A,S),(A,G),(G,S),(S,LC),(G,LC)的 SS 为例来详细说明其 2 个 SSRDAGs 的随机交叉操作.

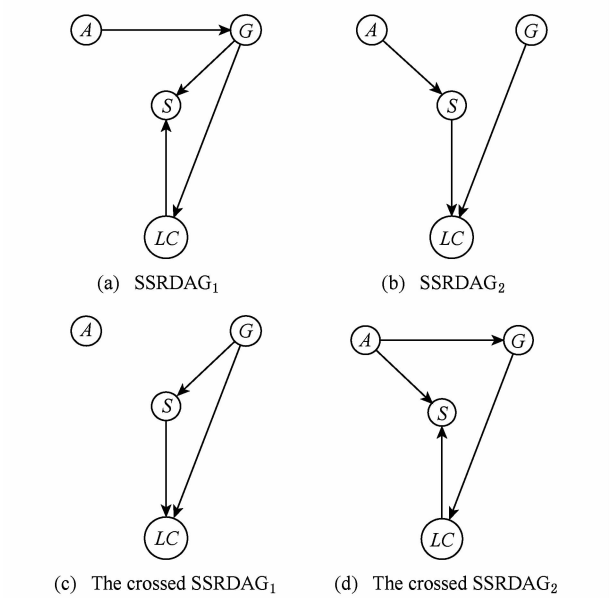


Fig. 2 Two SSRDAGs and their random cross results

图 2 两个 SSRDAGs 与其随机交叉结果

对于图 2(a)(b)中的 2 个 SSRDAGs,假设我们随机选择了节点对(A,G)和(S,LC),并互相交叉它们边的情况,得出如图 2(c)(d)所示的 2 个新 SSRDAGs.

进一步在 $m+1$ 个排好序的模型中,随机选取排序的前 2 个模型中的一个,记为 $\mathcal{G}^*$,并与后面 $m-1$ 个中的一个模型随机交叉. 这样与得分较低网络的随机交叉的目的是以一定的概率接受低分网络,跳出对当前网络的过度依赖,避免陷入局部极值. 并且每次交叉后保留 2 个 SSRDAGs 中得分较高的那个模型,这样的随机循环搜索共进行 m 次,并将每次搜索结果记为 $\mathcal{G}_k^*$ $(1\leqslant k\leqslant m)$. 接着,对 m 个 $\mathcal{G}_k^*$ $(1\leqslant k\leqslant m)$模型中最高得分与 $\mathcal{G}^*$ 的得分相比较,用高分网络替代当前网络 $\mathcal{G}^*$. 一直循环,直至达到循环结束条件,最后得出优化的 BN 模型的 DAG 结构.

2.4 SSRandom 学习算法

BN 结构学习的 SSRandom 算法的目标是基于 SS 随机搜索出 BIC 得分较高的 BN 模型的 DAG. 大致分为 3 个步骤:

Step1. 基于 Opt01ss 算法获得 BN 的 SS;

Step2. 在 SS 基础上,对所有无向边随机使用 O_{Add}算子添加方向,获得得分最高的初始模型 $\mathcal{G}^0$ 作为当前模型 $\mathcal{G}^*$;

Step3. 通过减边算子 O_{Sub}和转方向算子 O_{Rev}对 $\mathcal{G}^*$ 随机进行 m 次优化,并将高分网络与低分网络进行随机交叉,允许以一定的概率接受低分网络,避免结果陷入局部最优,循环得出优化的 BN 结构.

其伪代码描述如算法 1 所示:

算法 1. SSRandom 算法.

输入:数据集 $\mathcal{D}$、循环次数 n、随机搜索次数 m;

输出:优化后 BN 的 DAG $\mathcal{G}$结构.

```
通过 Opt01ss 算法从数据集 D 中获得超结构 SS;
while 循环次数 n 或循环条件满足 do
  for i=1 to m do
    通过 O_Add 算子,随机获得 m 个 SSRDAGs 模型(G_1,G_2,…,G_m),这些模型去掉边后均包含 SS 中的所有边;
  end for
  初始模型 G^0 ← arg max_{1≤i≤m} f_BIC(G_i);
  当前模型 G^* ← G^0;
  for j=1 to m do
    随机通过 O_Sub 和 O_Rev 算子,优化 G^*,获得 SSRDAGs G_j^*;
  end for
  Sort(G^*,G_1^*,G_2^*,…,G_m^*),即按 BIC 得分,对 m+1 个 SSRDAGs 非升序排列;
  G^* ← Sort 中最高得分的网络;
  for k=1 to m do
    G^* ← 从 G^* 和 Sort 中次高得分的网络中随机选一个网络;
    G^* 与剩余 m-1 个中的任意一个进行随机交叉,得 2 个 SSRDAGs 网络 G_k^{1*} 和 G_k^{2*};
    G_k^* ← arg max(f_BIC(G_k^{1*}),f_BIC(G_k^{2*}));
```

end for

if $f_{\mathrm{BIC}}(\mathcal{G}_k^*) < f_{\mathrm{BIC}}(\arg\max_{1\leqslant k\leqslant m} f_{\mathrm{BIC}}(\mathcal{G}_k^*))$ then

$\mathcal{G}^* \leftarrow \arg\max_{1\leqslant k\leqslant m} f_{\mathrm{BIC}}(\mathcal{G}_k^*)$;

end if

end while

return $\mathcal{G}\leftarrow\mathcal{G}^*$.

2.5 时间复杂性分析

在 SSRandom 算法中,在 SS 基础上的随机搜索部分,第 1 个 for 循环是随机选择初始网络,其时间复杂度为 $O(m)$,m 表示随机搜索次数. Sort 排序算法用快速排序时,其时间复杂性为 $O(m\log m)$,其中 m 表示 m 次随机搜索后获得的 m 个网络. 第 2 个和第 3 个 for 循环是随机优化初始网络,其时间复杂度均为 $O(m)$. 因而,整个随机搜索算法部分时间复杂度为 $O(m\log m)$.

而在 SS 的基础上,若考虑用其他算法优化 BN 结构,如蛮力法(brute force)和爬山法(hill climbing)等,那么需要对照分析其时间复杂性情况. 首先假设 SS 得出的无向边数为 e,最坏情况下 $e=\frac{|V|(|V|+1)}{2}$,其中 $|V|$ 表示网络节点变量个数. 下面分别对其具体情况进行分析:

1) 蛮力法. 由于要对无向图中的 e 条边定向,而每条边的定向情况有 2 种,因而最终的网络搜索空间为 2^e,故其时间复杂度为 $O(2^e)$.

2) 爬山法. 由于其搜索网络空间为搜索算子对当前网络的一次加边、或减边、或转边后得到的网络集合,易知,在 SS 的基础上,有 $i(0\leqslant i\leqslant e)$ 条有向边的当前网络的候选网络个数为

$$2e=2(e-i)+i+i,$$

其中,$(e-i)$ 表示一次加边产生的网络个数,2 个 i 分别表示一次减边和一次转边后产生的网络个数,因而 i 取 e 种情况下共有 $2e^2$ 个候选网络,故其时间复杂度为 $O(e^2)$.

由此,在 SS 基础上 3 种算法的时间复杂度分别如表 1 所示:

Table 1 The Time Complexity Comparison

表 1 时间复杂性对比

Algorithms	Time Complexity
SSRandom	$O(m\log m)$
Brute Force	$O(2^e)$
Hill Climbing	$O(e^2)$

Notes: m is the number of random search, and e is the number of undirected edge in SS.

通常,大多情况下 BN 的有向边数均会大于节点数 $|V|$,因而一般混合学习算法第 1 阶段得出的 SS 的无向边数 $e>|V|$,或至少也会接近 $|V|$. 随着网络变量数增加,蛮力法和爬山法的搜索空间会较随机搜索算法的搜索空间快速增加,从本文第 3 节实验也可得到,当 $|V|$ 分别为 6,8,11 且本文的随机搜索次数 $m=10$ 时,大多情况下已比其他 BN 结构算法学到的结果要好. 可见,该随机搜索策略要比通常基于评分搜索的算法高效.

3 实验结果与分析

本节在标准数据集上,通过将本文所提算法以及已有的 3 种 BN 结构学习算法,分别与标准结果在 4 方面评估指标上的对比分析,验证本文所提算法的学习性能.

3.1 实验环境与数据集

实验运行环境为:操作系统 Windows 7、处理器 Intel® Core™ i7-4510U CPU @ 2.00 GHz、内存 4.00 GB.

所有实验在 R 语言环境中借助于 R 语言自带的 bnlearn 工具包[17]实现. 实验选用标准的 Survey, Asia, Sachs 三个网络[17],其网络结构情况如表 2 所示. 实验中具体所用数据均是通过这些基准网络经过逻辑抽样后得出的模拟数据.

Table 2 Three Standard Structures of BNs Used in the Experiments

表 2 实验所采用的 3 个标准 BN 结构

Standard BNs	Number of Nodes	Number of Edges	Minimum Degree	Maximum Degree	Average Degree
Survey	6	6	1	4	2
Asia	8	8	1	4	2
Sachs	11	17	2	7	3.09

3.2 评价指标

BN 结构的学习结果通过灵敏性 Sn(sensitivity)、特效性 Sp(specificity)、欧几里德距离 Ed(Euclidean distance)[14]以及本文定义的整体准确率 Pa(percentage of overall accuracy)四个指标进行评价. 各指标具体描述如下:

1) 灵敏性 Sn. 反映正确学到的边占实际边的比率,该指标值越大越好. 其格式化表示为

$$Sn=\frac{TP}{TP+FN},$$

其中,TP 表示基准网络中有边而且也学出的边的

个数,即正确学出的边的个数;FN 表示基准网络中有边但没有学出的边的个数,即丢失边的个数.

2) 特效性 Sp. 反映正确学出的无边占实际无边的比率,该指标值越大越好. 其格式化表示为

$$Sp=\frac{TN}{FP+TN},$$

其中,FP 表示基准网络中无边但学出有边的个数,即冗余边的个数;TN 表示基准网络中无边而且学出无边的个数,即正确学出无边的个数.

3) 欧几里德距离 Ed. 反映学出网络与真实网络之间在欧氏空间的差距,该指标值越小越好. 其格式化表示为

$$Ed=\sqrt{(1-Sn)^2+(1-Sp)^2}.$$

4) 整体准确率 Pa. 主要通过正确学出的有向边和正确学习的无边的比率来度量学习所得的网络结构的整体准确度,该指标值越大越好. 其格式化表示为

$$Pa=\frac{Sn+Sp}{2}.$$

3.3 结果对比与分析

实验主要对比本文所提的 SSRandom 算法和已有的 MMHC,Hiton+HC (bnlearn 工具包自带的 Hiton 和 HC 搜索策略的混合)、GS + HC (bnlearn 自带的 Grow-Shrink 和 HC 的混合)三种结构学习算法在 Survey,Asia,Sachs 网络上的实验结果,对比指标主要有 Sn,Sp,Ed,Pa 四个.

由于本文所提方法还涉及到算法的循环次数 n 和随机搜索次数 m 的设置. 因而实验中,对于 n 的取值,考虑 $n=100$ 和 $n=300$ 这 2 种情况;对于 m 的取值,考虑 $m=10$ 和 $m=30$ 这 2 种情况.

对于 SSRandom 算法,每种情况下学习 5 次进行平均;对于其他 3 种算法,并不涉及这些参数,所以不需要分情况讨论. 因而在每种情况下,在每次运行 SSRandom 算法的相同数据集上,同时运行其他 3 种算法学习 BN 结构,即它们是 20(即 4×5)次或 5 次的平均. 具体实验结果如表 3~5,粗体数字表示每个指标下最好的实验结果.

Table 3 Comparison Between SSRandom and Other Three Algorithms on the Dataset of Size 5 000

表 3 样本容量为 5 000 时 SSRandom 与其他 3 种算法的实验结果对比

Algorithms	n	m	Survey				Asia				Sachs			
			Sn	Sp	Ed	Pa	Sn	Sp	Ed	Pa	Sn	Sp	Ed	Pa
SSRandom (the average of 5 times)	100	10	**0.967**	0.992	**0.042**	**0.979**	**0.900**	0.938	**0.124**	**0.919**	**0.647**	**0.940**	**0.358**	**0.793**
		30	**0.933**	0.992	**0.068**	**0.963**	**0.925**	0.938	**0.111**	**0.931**	**0.659**	**0.933**	**0.348**	**0.796**
	300	10	**0.967**	0.983	**0.043**	**0.975**	**0.900**	0.942	**0.123**	**0.921**	**0.671**	**0.951**	**0.333**	**0.811**
		30	**0.967**	0.992	**0.034**	**0.979**	**0.925**	0.946	**0.101**	**0.935**	**0.682**	**0.946**	**0.322**	**0.814**
MMHC (the average of 20 times)			0.725	**0.994**	0.275	0.859	0.644	0.978	0.357	0.811	0.197	0.896	0.810	0.546
Hiton+HC (the average of 20 times)			0.725	**0.994**	0.275	0.859	0.650	0.999	0.350	0.824	0.329	0.920	0.675	0.625
GS+HC (the average of 20 times)			0.742	**0.994**	0.259	0.868	0.513	**1.000**	0.488	0.756	0.241	0.935	0.762	0.588

Table 4 Comparison Between SSRandom and Other Three Algorithms on the Dataset of Size 2 000

表 4 样本容量为 2 000 时 SSRandom 与其他 3 种算法的实验结果对比

Algorithms	n	m	Survey				Asia				Sachs			
			Sn	Sp	Ed	Pa	Sn	Sp	Ed	Pa	Sn	Sp	Ed	Pa
SSRandom	300	30	**0.667**	0.975	**0.335**	**0.821**	**0.875**	0.946	**0.150**	**0.910**	**0.624**	**0.944**	**0.381**	**0.784**
MMHC			0.500	**1.000**	0.500	0.750	0.650	0.979	0.351	0.815	0..141	0.925	0.862	0.533
Hiton+HC			0.500	**1.000**	0.500	0.750	0.650	**1.000**	0.350	0.825	0.188	0.908	0.817	0.548
GS+HC			0.500	**1.000**	0.500	0.750	0.525	**1.000**	0.475	0.763	0.118	0.925	0.886	0.521

Notes: The results of each algorithm are the average of 5 times.

Table 5 Comparison Between SSRandom and Other Three Algorithms on the Dataset of Size 10 000
表 5 样本容量为 10 000 时 SSRandom 与其他 3 种算法的实验结果对比

Algorithms	n	m	Survey				Asia				Sachs			
			Sn	*Sp*	*Ed*	*Pa*	*Sn*	*Sp*	*Ed*	*Pa*	*Sn*	*Sp*	*Ed*	*Pa*
SSRandom	100	10	**1.000**	**1.000**	**0.000**	**1.000**	**1.000**	0.925	**0.075**	**0.963**	**0.729**	**0.948**	**0.276**	**0.839**
MMHC			0.900	**1.000**	0.100	0.950	0.725	0.971	0.277	0.848	0.200	0.892	0.807	0.546
Hiton+HC			0.900	**1.000**	0.100	0.950	0.725	**0.996**	0.275	0.860	0.282	0.912	0.723	0.597
GS+HC			0.900	**1.000**	0.100	0.950	0.550	**0.996**	0.450	0.773	0.200	0.903	0.806	0.552

Note: The results of each algorithm are the average of 5 times.

从表 3 可看出,样本量达 5 000 时,分别在(n=100,m=10),(n=100,m=30),(n=300,m=10),(n=300,m=30)四种情况下,除 Survey 网络上的 Sp 和 Asia 网络上 GS + HC 方法的 Sp 外,SSRandom 算法在 4 种指标下均比其他 3 种算法效果好,其 Pa 结果在 Survey 和 Asia 网络上可达 90%以上,在 Sachs 网络上可达近 80%,但其他 3 种算法在 Sachs 网络上的 Pa 效果却在 63%及其以下. 而且可统计出 SSRandom 算法循环次数 n=300 时的结果中有 83.3%(10/12)的要比 n=100 时好,具体可通过每个网络上 n=300 与 n=100 在 m=10 和 m=30 共 4 种组合下的比较统计得出. 同理得出 m=30 时中有 75%(9/12)的结果要比 m=10 时的好.

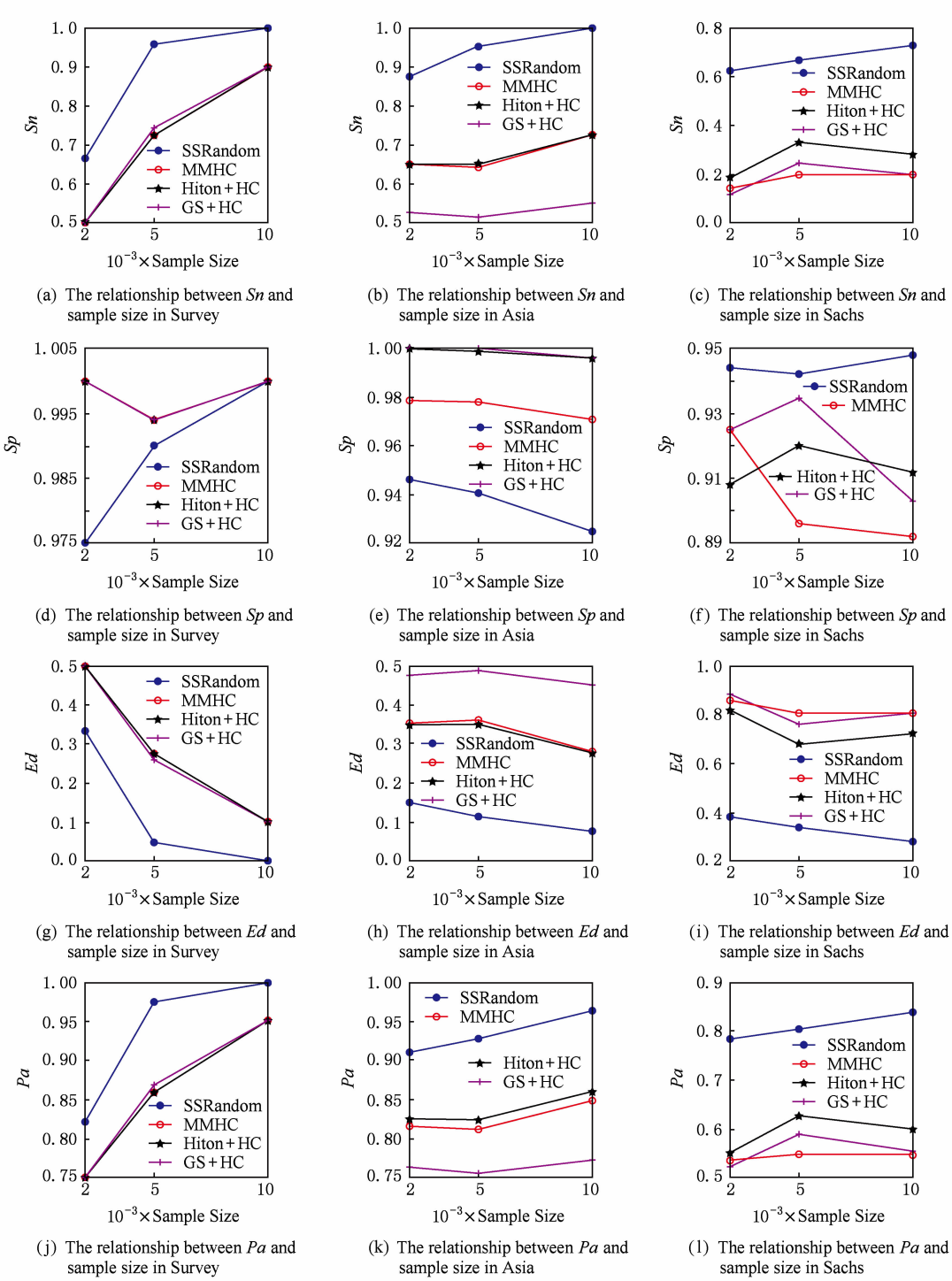
下面进一步对样本量减少和增加时,4 种学习算法的性能进行对比分析. 首先看样本量减少的情况,由于样本量为 5 000 时,SSRandom 算法在 4 种指标下的学习结果大多好于其他 3 种算法,而且在 n=300 时的整体效果要比 n=100 时的好,在 m=30 时的整体效果要比 m=10 时的好. 因而,当样本量减少为 2 000 时,考虑 SSRandom 算法在(n=300,m=30)情况时与其他 3 种算法的实验结果进行对比分析,同样,每次在样本量为 2 000 的相同数据集上运行,共实验 5 次后求平均. 具体如表 4 所示.

从表 4 可知,除部分 Sp 外,本文 SSRandom 算法性能在其他指标下均比样本量为 5 000 时有所下降,但仍比其他 3 种算法要好. 而且从实验数据上看,所有 Pa 均可达 78%以上,但其他 3 种算法,尤其在 Sachs 网络上的 Pa 效果却在 55%以下. 因此,本文算法与其他算法相比,仍较适合样本量 2 000 时进行 BN 结构的学习. 接着,看样本量增加的情况. 同理,由于样本量为 5 000 时,SSRandom 算法的学习效果较好,所以当样本量增加为 10 000 时,我们只考虑 SSRandom 算法在(n=100,m=10)情况时与其他 3 种算法 5 次平均的实验结果进行对比分析,具体如表 5 所示.

从表 5 可知,除 Asia 网络上的 Sp 外,SSRandom 算法在其他情况下的学习结果比其他 3 种算法的结果要好或一样好. 而且从实验数据整体上来看,4 种算法的学习结果较之前样本量为 2 000 和 5 000 时均有所提高. 但对于其他 3 种算法,在 Sachs 网络上的 Pa 效果仍在 60%以下,而本文 SSRandom 算法可提高到 83%以上.

进一步,从表 3~5 中,分析 3 个样本量情况下 Sachs 网络的学习结果. 由表 2 可知,Sachs 网络最大度为 7,较 Survey 和 Asia 网络来说,该网络更易产生 ADRs 问题,丢失较多弱关系边;而本文算法在有限样本情况下,可以较好处理这一问题,而且在后续学习阶段该算法可以一定概率较好地避免使学习结果陷入局部最优等不足,从而使得学习结果相对理想,提高了实验结果的 Pa. 另外,从表 3~5 可得出,4 种算法在不同网络上的各指标效果与不同样本量的关系,具体如图 3 所示.

从图 3(a)~(l)明显可知,随着样本量的增加,除特效性 Sp 外,3 个网络在各评价指标下 4 种算法的性能均在增加. 虽然反映正确学出的无边的特效性 Sp 的性能有所下降,但关联到 Sp 的整体准确率的 Pa 和欧几里德距离 Ed 指标下的性能仍是增加的. 并且可看出本文的 SSRandom 算法在每个样本量下与其他方法相比,整体上效果较好. 更重要的是,当样本量为 2 000 时,较其他网络而言,就可学出很好的网络结构,尤其对于 Survey 网络;当样本量从 2 000 增加至 5 000 时,SSRandom 算法的其各项性能增加得较为明显,很快就可增加至近似最好结果;当样本量从 5 000 增加至 10 000 时,其增长趋势相对平缓一些,这说明了本文 SSRandom 算法与其他 3 种算法相比,在较少样本量的情况下,可以较好地学得网络结构.



(a) The relationship between *Sn* and sample size in Survey

(b) The relationship between *Sn* and sample size in Asia

(c) The relationship between *Sn* and sample size in Sachs

(d) The relationship between *Sp* and sample size in Survey

(e) The relationship between *Sp* and sample size in Asia

(f) The relationship between *Sp* and sample size in Sachs

(g) The relationship between *Ed* and sample size in Survey

(h) The relationship between *Ed* and sample size in Asia

(i) The relationship between *Ed* and sample size in Sachs

(j) The relationship between *Pa* and sample size in Survey

(k) The relationship between *Pa* and sample size in Asia

(l) The relationship between *Pa* and sample size in Sachs

Fig. 3 The relationship between each evaluation index and sample size in Survey, Aisa and Sachs networks, respectively

图 3 各评价指标与样本量分别在 Survey, Aisa, Sachs 网络上的关系

4 结束语

本文针对 BN 结构混合学习算法中超结构构建时容易丢失边和空间搜索时爬山法易陷入局部最优等问题,通过采用 Opt01ss 算法学习超结构,以尽可能地避免了出现丢失边的问题;接着重点研究了基于超结构随机搜索的 SSRandom 学习算法,该随机算法可以一定的概率跳出局部最优. 通过在标准网络上多种样本量情况下 4 种评价指标上的实验,验证了该算法的良好学习性能. 进一步的研究工作将面向高维小样本数据探讨 BN 结构的随机学习方法.

参 考 文 献

[1] Pearl J. Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference [M]. San Mateo, CA: Morgan Kaufmann, 1988

[2] Bouhamed H, Masmoudi A, Lecroq T, et al. Structure space of Bayesian networks is dramatically reduced by subdividing it in sub-networks [J]. Journal of Computational and Applied Mathematics, 2015, 287: 48-62

[3] Cheng J, Greiner R, Kelly J. Learning Bayesian networks from data: An information-theory based approach [J]. Artificial Intelligence, 2002, 137(1/2): 43-90

[4] De Campos L M. A scoring function for learning Bayesian networks based on mutual information and conditional independence tests [J]. Journal of Machine Learning Research, 2006, 7(2): 2149-2187

[5] Lam W, Bacchus F. Learning Bayesian belief networks: An approach based on the MDL principle [J]. Computational Intelligence, 1994, 10(3): 269-293

[6] Cooper G F, Herskovits E. A Bayesian method for the induction of probabilistic networks from data [J]. Machine Learning, 1992, 9(4): 309-347

[7] Tsamardinos I, Brown L E, Aliferis C F. The max-min hill-climbing Bayesian network structure learning algorithm [J]. Machine Learning, 2006, 65(1): 31-78

[8] Masegosa A R, Feelders A J, Gaag L C. Learning from incomplete data in Bayesian networks with qualitative influences [J]. International Journal of Approximate Reasoning, 2015, 69: 18-34

[9] Yao Tiansheng, Choi A, Darwiche A. Learning Bayesian network parameters under equivalence constraints [J]. Artificial Intelligence, 2015, 29(9): 1349-1354

[10] Liao Wenhui, Ji Qiang. Learning Bayesian network parameters under incomplete data with domain knowledge [J]. Pattern Recognition, 2009, 42(11): 3046-3056

[11] Masegosa A R, Moral S. New skeleton-based approaches for Bayesian structure learning of Bayesian networks [J]. Applied Soft Computing, 2013, 13(2): 1110-1120

[12] GaSSe M, AuSSem A, Elghazel H. A hybrid algorithm for Bayesian network structure learning with application to multi-label learning [J]. Expert Systems with Applications 2014, 41(15): 6755-6772

[13] Perrier E, Imoto S, Miyano S. Finding optimal Bayesian network given a super-structure [J]. Journal of Machine Learning Research, 2008, 9(4): 2251-2286

[14] Villanueva E, Maciel C. Efficient methods for learning Bayesian network super-structures [J]. Neurocomputing, 2014, 123: 3-12

[15] Lähdesmäki H, Shmulevich I. Learning the structure of dynamic Bayesian networks from time series and steady state measurements [J]. Machine Learning, 2008, 71(2): 185-217

[16] Villanueva E, Maciel C. Optimized algorithm for learning Bayesian network super-structures [C/OL] //Proc of the ICPRAM'12. 2012 [2016-07-01]. https://www.researchgate.net/publication/289765566_Optimized_algorithm_for_learning_Bayesian_network_super-structures?ev=auth_pub

[17] Scutari M. Learning Bayesian networks with the bnlearn R package [J]. Journal of Statistical Software, 2010, 35(3): 1-22

Lü Yali, born in 1975. PhD. Associate professor. Member of CCF. Her main research interests include probabilistic reasoning, data mining and machine learning.

Wu Jiajie, born in 1989. Master candidate. Student member of CCF. His main research interests include machine learning and probabilistic reasoning (sxcjdxjsjw@163.com).

Liang Jiye, born in 1962. Professor and PhD supervisor. Distinguished member of CCF. His main research interests include granular computing, data mining and machine learning(ljy@sxu.edu.cn).

Qian Yuhua, born in 1976. Professor and PhD supervisor. Member of CCF. His main research interests include granular computing, social computing and machine learning for big data (jinchengqyh@126.com).