

推移性を利用したRAIアルゴリズムによる 大規模ベイジアンネットワーク構造学習

Learning huge Bayesian network structures by RAI algorithm with transitivity

本田和雅 名取和樹 菅原聖太 磯崎隆司 植野真臣
Kazunori Honda Kazuki Natori Shouta Sugahara Takashi Isozaki Maomi Ueno

電気通信大学大学院情報理工学研究所

Graduate school of Information and Engineering, The University of Electro-Communications

Learning Bayesian networks (BNs) is NP-hard. Recently, we can learn 1000 nodes BNs with consistency by the RAI algorithm using Bayes factor, which is the state-of-the-art learning method. However, it is important to enable learning huge BNs to apply it in practice. This paper proves that conditional independence (CI) of BNs has the transitivity that can infer, from CI between a pair of variables, CI between each of them and another variable, and proposes a constraint-based algorithm, using the RAI algorithm with the transitivity. The experimental results show that the proposed method decreases the number of CI tests and run-time, and can learn huge BNs which prototypical constraint-based algorithms cannot learn.

1. はじめに

ベイジアンネットワークは、確率変数をノードとし、ノード間の依存関係を非循環有向グラフ (Directed Acyclic Graph: DAG) と各ノードの条件付き確率で表現する確率的グラフィカルモデルである。ベイジアンネットワークのネットワーク構造は一般に未知であるため、データから推定する必要がある。この問題をベイジアンネットワーク構造学習と呼ぶ。

ベイジアンネットワーク構造学習手法として、漸近一致性を有する学習スコアが最大となる構造を全ての構造の候補から探索する厳密解探索アプローチが知られている。この手法は構造の探索数がノード数に対し指数的に増加する NP 困難問題 [Chickering 96] である。厳密解を効率的に探索するために、動的計画法 [Silander 06], A^* 探索 [Yuan 11], 整数計画法 [Cussens 11] などの最適化手法による構造学習法が提案されてきたが、未だ 60 変数程度の構造学習が限界である。

一方、因果モデルの分野では、より計算効率の高い手法である制約ベースアプローチが提案されている。制約ベースアプローチは完全無向グラフの各エッジに対する条件付き独立性検定 (Conditional Independence test: CI テスト) で独立と判定されたエッジを削除し、それによって得られた無向グラフをオリエンテーションルール [Pearl 00] で方向付けることで構造を推定する。この手法の代表的なアルゴリズムとして、PC アルゴリズム [Spirtes 00], MMHC アルゴリズム [Tsamardinos 06], RAI アルゴリズム [Yehezkel 09] が知られており、RAI アルゴリズムが最も高速に学習できると報告されている。近年では RAI アルゴリズムの CI テストに Bayes factor を用いることにより、漸近一致性を有しつつ 1000 変数程度の学習が可能となった [Natori 17][名取 18] が、本研究ではより大規模の構造学習を実現できる手法の開発を目的とする。

制約ベースアプローチの時間計算量は学習に要する CI テスト数に依存し、一般に、学習のできる限り早期にエッジを削除するほど CI テスト数を少なくして学習できる。そこで本研究では、まず、ある二変数の条件付き独立性からその各変数と他変数との条件付き独立性の少なくとも一つを保証できる推移

性が成り立つことを示す。そして、ある条件付き独立性を検出したとき、推移性により少なくとも一つが条件付き独立となる二組のエッジを優先して CI テストできるエッジ削除法を提案し、Bayes factor を用いた RAI アルゴリズムに組み込む。これにより、より学習の早期にエッジを削除することで CI テスト数を削減でき、従来手法では実現できない大規模構造学習の実現を期待できる。

本研究は数千ノードのネットワークを用いたシミュレーション実験により従来ではできない大規模構造学習を実現する。

2. ベイジアンネットワーク構造学習

今、 $\mathbf{V} = \{X_1, \dots, X_n\}$ を n 個の離散確率変数集合とし、各変数 X_i は r_i 個の状態集合 $\{1, \dots, r_i\}$ から一つの値 k を取る ($X_i = k$ と書く) とする。また、ベイジアンネットワーク構造 G において、各変数 X_i の親変数集合を Π_i^G とする。このとき、ベイジアンネットワークは非循環有向グラフ (Directed Acyclic Graph: DAG) を仮定することで同時確率分布 $P(X_1, \dots, X_n)$ を以下の条件付き確率の積に厳密分解できる。

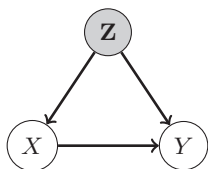
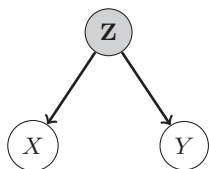
$$P(X_1, \dots, X_n) = \prod_{i=1}^n P(X_i | \Pi_i^G). \quad (1)$$

ベイジアンネットワーク構造学習では漸近一致性を有する周辺尤度スコアを最大にする構造を探索する厳密解探索アプローチが一般的に用いられる。今、観測データを $\mathbf{D} = \{D_1, \dots, D_N\}$ とすると、ネットワーク G における周辺尤度スコア $P(\mathbf{D} | G)$ は以下となる。

$$P(\mathbf{D} | G) = \prod_{i=1}^n \prod_{j=1}^{q_i} \frac{\Gamma(\alpha_{ij})}{\Gamma(\alpha_{ij} + N_{ij})} \prod_{k=1}^{r_i} \frac{\Gamma(\alpha_{ijk} + N_{ijk})}{\Gamma(\alpha_{ijk})}. \quad (2)$$

ここで、 N_{ijk} は変数 X_i の親変数集合 Π_i^G が j ($j = 1, \dots, q_i$) 番目のパターンを取るときの $X_i = k$ となる頻度を表し、 $N_{ij} = \sum_{k=1}^{r_i} N_{ijk}$ である。また、 α_{ijk} はハイパーパラメータを表し、 $\alpha_{ij} = \sum_{k=1}^{r_i} \alpha_{ijk}$ である。近年では、 $\alpha_{ijk} = \alpha / (r_i q_i)$ とした Bayesian Dirichlet equivalent uniform (BDeu) が最も用いられる [Heckerman 95]。ここで、 α は Equivalent Sample

連絡先: 電気通信大学大学院情報理工学研究所,
東京都調布市調布ヶ丘 1-5-1,
{hondak,natori,sugahara,ueno}@ai.lab.uec.ac.jp

図 1: 従属モデル G_1 図 2: 独立モデル G_2

Size (ESS) と呼ばれる事前知識の重みを示す擬似サンプルである。

周辺尤度スコアを最大化する厳密解探索アプローチとして、動的計画法 [Silander 06], A^* 探索 [Yuan 11], 整数計画法 [Cussens 11] などが提案されてきたが、未だ 60 変数程度の構造学習が限界であり、大規模構造を学習できない。

因果モデル分野では、大幅に計算量を削減できる制約ベースアプローチと呼ばれる構造学習法が提案されてきた。このアプローチの基本的なアルゴリズムは以下の通りである。

1. 確率変数をノードとした完全無向グラフを生成する。
2. 1 の完全無向グラフに対し条件付き独立性検定 (Conditional Independence test: CI テスト) を行い、独立と判定されたエッジを削除する。
3. 2 で得られた無向グラフに対してオリエンテーションルール [Pearl 00] を用いて方向付けを行う。

制約ベースアプローチによる学習アルゴリズムとして、PC アルゴリズム [Spirtes 00] や MMHC アルゴリズム [Tsamardinos 06], RAI アルゴリズム [Yehezkel 09] が提案されており、RAI アルゴリズムは最も高速に学習できると報告されている。しかし、これらのアルゴリズムは CI テストに χ^2 検定や G^2 検定、条件付き相互情報量を用いるために漸近一致性を持たない。

Steck らは Bayes factor を用いた CI テストを提案している [Steck 02]。今、二ノード X, Y とそれらの共通親ノード集合 \mathbf{Z} からなる従属モデルを G_1 、独立モデルを G_2 とし、それぞれ図 1, 2 に示す。このときの Bayes factor を $\text{BF}(X, Y | \mathbf{Z})$ とすると、対数 Bayes factor は以下となる。

$$\log \text{BF}(X, Y | \mathbf{Z}) = \log \frac{P(\mathbf{D} | G_1, \alpha)}{P(\mathbf{D} | G_2, \alpha)}. \quad (3)$$

ここで、 $P(\mathbf{D} | G_1, \alpha)$, $P(\mathbf{D} | G_2, \alpha)$ は式 (2) の BDeu を用いる。Bayes factor を用いた CI テストでは対数 Bayes factor が 0 以上か否かで図 1, 2 のどちらを選択するか判定する。この CI テストは漸近的に真の条件付き独立性を判定できる [名取 18]。

3. Bayes factor を用いた RAI アルゴリズム

RAI アルゴリズムは、制約ベースアプローチの中で最も高速に学習できるアルゴリズムである [Yehezkel 09]。RAI アルゴリズムは学習途中にエッジを方向付け、全体のグラフを部分グラフへ分割するため、各ノードの親候補となるノード集合を制約し、高次の CI テストを削減して学習できる。近年では、RAI アルゴリズムの CI テストに Bayes factor を用いることで漸近一致性を有しつつ最も大規模の構造学習を実現している [Natori 17][名取 18]。

Bayes factor を用いた RAI アルゴリズムの詳細を Algorithm 1 に示す。関数 RAI はグラフとデータ、CI テストの次

Algorithm 1 The RAI algorithm using Bayes factor

```

1: function MAIN( $G_{uc}, \mathbf{D}$ )
    $G_{uc} = (\mathbf{V}_{uc}, \mathbf{E}_{uc})$ : 完全無向グラフ
    $\mathbf{D}$ : データ
2:   return RAI(0,  $G_{uc}$ ,  $\phi$ ,  $G_{uc}$ ,  $\mathbf{D}$ )
3: end function

4: function RAI( $n_z, G_s, \mathbf{G}_{ex}, G_{all}, \mathbf{D}$ )
    $n_z$ : CI テストの次数
    $G_s = (\mathbf{V}_s, \mathbf{E}_s)$ : 入力グラフ
    $\mathbf{G}_{ex}$ : 分割されたグラフの集合
    $G_{all} = (\mathbf{V}_{all}, \mathbf{E}_{all})$ : CI テストと方向付けによって得られる出力グラフ
5:   if 全ての  $V \in \mathbf{V}_s$  について  $|\text{Pa}_p(V, G_{all})| < n_z + 1$  then
6:     return  $G_{all}$ 
7:   end if
8:   // CI テストによるエッジの削除
9:   for  $G_{ex} = (\mathbf{V}_{ex}, \mathbf{E}_{ex}) \in \mathbf{G}_{ex}$  do
10:    for  $X \in \mathbf{V}_s, Y \in \mathbf{V}_{ex}$  do
11:      for  $\mathbf{Z} \subseteq \text{Pa}_p(X, G_s) \cup \Pi_X^{G_{ex}} \setminus \{Y\}$  do
12:        if  $|\mathbf{Z}| = n_z$  かつ  $\log \text{BF}(X, Y | \mathbf{Z}) < 0$  then
13:           $\mathbf{E}_{all} \leftarrow \mathbf{E}_{all} \setminus \{E_{XY}\}$   $\triangleright E_{XY}$ :  $XY$  間のエッジ
14:        end if
15:      end for
16:    end for
17:  end for
18:  for  $X \in \mathbf{V}_s, Y \in \mathbf{V}_s$  do
19:    for  $\mathbf{Z} \subseteq \text{Pa}_p(X, G_s) \cup \Pi_X^{G_{ex}} \setminus \{Y\}$  do
20:      if  $|\mathbf{Z}| = n_z$  かつ  $\log \text{BF}(X, Y | \mathbf{Z}) < 0$  then
21:         $\mathbf{E}_{all} \leftarrow \mathbf{E}_{all} \setminus \{E_{XY}\}, \mathbf{E}_s \leftarrow \mathbf{E}_s \setminus \{E_{XY}\}$ 
22:      end if
23:    end for
24:  end for
25:  //  $G_s$  を部分グラフに分割
26:   $\mathbf{V}_c \leftarrow \{V | V \in \mathbf{V}_s, V \text{ は子ノードを持たず, 無向エッジで親に到達不可能}\}$ 
27:   $\mathbf{E}_c \leftarrow \{E_{XY} | X, Y \in \mathbf{V}_c, E_{XY} \text{ は無向エッジ}\}$ 
28:   $G_e \leftarrow G_s \setminus (\mathbf{V}_c, \mathbf{E}_c)$ 
29:   $\mathbf{G}_e \leftarrow G_e$  の個々の連結グラフからなる集合
30:  // 再帰的に関数 RAI を呼び出す
31:  for  $i = 1$  to  $|\mathbf{G}_e|$  do
32:     $G_{all} \leftarrow \text{RAI}(n_z + 1, \mathbf{G}_e[i], \mathbf{G}_{ex}, G_{all}, \mathbf{D})$ 
33:  end for
34:   $\mathbf{G}_{ex} \leftarrow \mathbf{G}_{ex} \cup \mathbf{G}_e$ 
35:  return RAI( $n_z + 1, (\mathbf{V}_c, \mathbf{E}_c), \mathbf{G}_{ex}, G_{all}, \mathbf{D}$ )

```

数 n_z を入力されたとき、まず、各ノード V の子ノードを除いた隣接ノード集合 $\text{Pa}_p(V, G_s)$ の大きさが $n_z + 1$ 未満か判定し、そうであれば学習を終了する (5 から 7 行目)。学習を続ける場合、各エッジに対して Bayes factor を用いた CI テストを行いエッジを削除し (8 から 23 行目)、残ったエッジをオリエンテーションルールで方向付ける (24 行目)。方向付けられたグラフから子ノードを持たないノード集合からなる部分グラフを取り出し、それ以外のグラフを個々の連結な部分グラフに分割する (25 から 28 行目)。以上を各部分グラフに対して再帰的に実行する (29 から 33 行目)。これにより各ノードの親ノード候補を制約することができ、高速に学習できる。

4. 提案手法

制約ベースアプローチの時間計算量は学習に要する CI テスト数に依存する。一般的に、制約ベースアプローチは学習のできる限り早期にエッジを削除するほど CI テスト数を抑えて学習できる。本研究は、まず、ベイジアンネットワークのある二変数の条件付き独立性からその各変数と他変数との条件付き独立性の少なくとも一つを保証できる推移性が成り立つことを示す。そして、推移性を利用することで、ある条件付き独立性を検出したときに少なくとも一つが条件付き独立となる二組のエッジを優先して CI テストする手法を提案し、Bayes factor を用いた RAI アルゴリズムに組み込む。これにより、より学習の早期にエッジを削除できるため、学習に要する CI テスト数を削減し、従来より大規模の構造学習の実現を期待できる。

4.1 推移性

今、ノード X からノード Y への隣接ノードの連なりを道と言ひ、道上で三ノードが $A \rightarrow C \leftarrow B$ と結合するとき、 C を合流結合するノードと言ふ。また、 $X \perp Y \mid \mathbf{Z}$ は X と Y のノード集合 \mathbf{Z} を所与とした条件付き独立性を表すとする。このとき、以下の定理で示す推移性が成り立つ。

定理 1 $G = (\mathbf{V}, \mathbf{E})$ を DAG とし、 $X, Y \in \mathbf{V}$ で、 Y は X の非子孫とする。このとき、 $A \in \mathbf{V} \setminus (\{X, Y\} \cup \Pi_X^G \cup \mathbf{W})$ とすると、以下が成り立つ。

$$X \perp Y \mid \Pi_X^G \Rightarrow X \perp A \mid \Pi_X^G \text{ or } A \perp Y \mid \Pi_X^G. \quad (4)$$

ここで、 \mathbf{W} は X の子孫であり、 X と Y が合流結合するノードとその子孫からなるノード集合を表し、 Π_X^G は X の G における親ノード集合を表す。

証明は省略する。定理 1 より、ある二変数の条件付き独立性からその各変数と他変数との条件付き独立性を少なくとも一つ保証できる。これを制約ベースアプローチに利用することで、あるエッジを削除したとき、その両端と他ノードとのエッジの少なくとも一つの削除が保証される。これにより、より学習の早期にエッジを削除でき、CI テスト数の削減を期待できる。

4.2 提案アルゴリズム

本節では、定理 1 の推移性を用いたエッジ削除法を提案し、それを Bayes factor を用いた RAI アルゴリズムへ組み込む。

定理 1 の推移性を利用するためには推移性を有する候補ノード A の探索空間を同定する必要がある。今、 \mathbf{V} をノード集合とし、 $X, Y \in \mathbf{V}$ 、 Π_X^G を X のグラフ G における親ノード集合とする。また、 \mathbf{W} を X の子孫で、 X と Y が合流結合するノードとその子孫からなるノード集合とする。このとき、候補ノード A の探索空間は $\mathbf{V} \setminus (\{X, Y\} \cup \Pi_X^G \cup \mathbf{W})$ であるため、まずノード集合 \mathbf{W} の要素を列挙しなければならない。 \mathbf{W} の要素を列挙するためには合流結合を同定する必要があるが、これは X, Y 間で X の子孫を持つ全ての道を探さなければならない、計算量が大きい。しかし、推移性を有するノード A の探索空間は式 (4) でさらに制約でき、これにより探索空間を同定する計算量を軽減できる。今、 X, Y のエッジを削除したとき、 X と A か A と Y (もしくは両方) のエッジがすでに削除されていたとする。このとき、 X と A か A と Y (もしくは両方) の条件付き独立性がすでに検出されているため、式 (4) はすでに成り立っており、推移性を利用する必要がない。すなわち、 A の探索空間は X と Y の共通隣接ノード集合 $\text{Adj}(X, G) \cap \text{Adj}(Y, G)$ に含まれなければならない。ここで、 X と Y の共通子ノード集合 $\text{Ch}(X, G) \cap \text{Ch}(Y, G)$ はノード集合 \mathbf{W} に含まれるために A の探索空間から除外する。また、定理 1 より、 Π_X^G も除外する。したがって、 A の探索空間は $\text{Adj}(X, G) \cap \text{Adj}(Y, G) \setminus (\text{Ch}(X, G) \cap \text{Ch}(Y, G) \cup \Pi_X^G)$ となる。この探索空間は条件付き独立となる二ノード X, Y の共通隣接ノードを探索すれば同定できるため、ノード数に対して線形時間で計算できる。

定理 1 の推移性を利用した CI テストとエッジ削除法を Algorithm2 に示す。関数 TRANSITIVE_CUT は学習途中のグラフ G 、 $\log \text{BF}(X, Y \mid \mathbf{Z}) < 0$ となる二ノード X, Y とノード集合 \mathbf{Z} 、データ \mathbf{D} を入力とし、推移性に基づくエッジの削除を行ったあとにグラフを出力する。具体的には、 $\text{Adj}(X, G) \cap \text{Adj}(Y, G) \setminus (\text{Ch}(X, G) \cap \text{Ch}(Y, G) \cup \mathbf{Z})$ を前述の探索空間とし、この探索空間に属するノードを A とする。二つのノード対 X と A 、 A と Y のそれぞれに対して \mathbf{Z} を所

Algorithm 2 Edge cutting with transitivity

```

1: function TRANSITIVE_CUT( $G, X, Y, \mathbf{Z}, \mathbf{D}$ )
    $G = (\mathbf{V}, \mathbf{E})$ : 全体グラフ
    $X, Y, \mathbf{Z}$ :  $\log \text{BF}(X, Y \mid \mathbf{Z}) < 0$  となる二ノード  $X, Y$  とノード集合  $\mathbf{Z}$ 
    $\mathbf{D}$ : データ
2:    $\mathbf{A} \leftarrow \text{Adj}(X, G) \cap \text{Adj}(Y, G) \setminus (\text{Ch}(X, G) \cap \text{Ch}(Y, G) \cup \mathbf{Z})$ 
3:   for  $A \in \mathbf{A}$  do
4:     if  $\log \text{BF}(X, A \mid \mathbf{Z}) < 0$  then
5:        $\mathbf{E} \leftarrow \mathbf{E} \setminus \{E_{XA}\}$  ▷  $E_{XA}$ :  $XA$  間のエッジ
6:     end if
7:     if  $\log \text{BF}(A, Y \mid \mathbf{Z}) < 0$  then
8:        $\mathbf{E} \leftarrow \mathbf{E} \setminus \{E_{AY}\}$ 
9:     end if
10:  end for
11:  return  $G$ 
12: end function

```

与とした Bayes factor を用いる CI テストを行い、独立と判定されたノード対の間のエッジをグラフから削除する。以上を探索空間に属する全てのノードに対して繰り返す。

前述のとおり、推移性を有するノード A の探索空間は条件付き独立となる二ノード X, Y の共通子ノード集合を含まない。すなわち、推移性を利用したエッジ削除法を利用するためには事前にエッジを方向付け、子ノードを同定する必要がある。したがって、無向グラフにおいてエッジを削除する PC アルゴリズムや MMHC アルゴリズムでは推移性を利用できない。一方で、RAI アルゴリズムは学習途中にエッジを方向付けるため、推移性を利用できる。そこで、本研究では推移性を利用したエッジ削除法を RAI アルゴリズムに組み込む。

推移性を用いたエッジ削除法を RAI アルゴリズムに利用するためには関数 TRANSITIVE_CUT を関数 RAI におけるエッジの削除後 (Algorithm1 の 12 行目と 20 行目の直後) に呼び出せば良い。ここで、CI テストの次数が 0 のとき、関数 RAI は、推移性の利用の有無に関わらず、全てのノード対に対して一回ずつの CI テストを行わなければならない。そのため、提案手法は関数 TRANSITIVE_CUT を 0 次の CI テストでは呼び出さず、1 次以降のときにのみ呼び出す。

提案手法は推移性によって少なくとも一つの条件付き独立性が保証される二組のエッジの CI テストを優先的に実施できるため、学習の早期にエッジを削除でき、CI テスト数を削減できる。制約ベースアプローチの時間計算量は CI テスト数に依存するため、提案手法は計算時間も削減できる。また、データ数が十分に大きくないとき、信頼性の低い CI テストを削減するため、従来の制約ベースアプローチ以上の学習精度を保証する。さらに、提案手法は従来では実現してこなかった大規模構造学習の実現を期待できる。

5. 比較実験

本提案手法の有効性を示すため、シミュレーション実験によって従来では最も高速な手法である RAI アルゴリズムと比較する。各手法の CI テストには Bayes factor ($\text{ESS} = 1.0$ [Ueno 10]) を用いる。真のネットワークとして *bnlearn*[Scutari 11] に登録されているベンチマークネットワークの *munin* (ノード数: 1041, エッジ数: 1397) と *BNGenerator**1 を用いてランダムに生成したネットワーク *random* (ノード数: 2000, エッジ数: 2973) を使用する。実験は、(1) 各ネットワークからデータをランダム生成し、(2) 各手法で構造学習することを (3) それぞれ 10 回繰り返すことで行う。ただし、6 時間の制限時間を設け、超過する場合は実験を打ち切った。評価尺度は、学習速度を表す CI テスト数、計算時間と学習精度を表す Structural Hamming

*1 <http://sites.poli.usp.br/pmr/ltd/software/bngenerator/>

表 1: 実験結果

| ネットワーク | munin | | random | |
|---------------------------|-------|-------|--------|--------|
| CI テスト数 ($\times 10^3$) | | | | |
| データ数 | RAI | 提案 | RAI | 提案 |
| 10,000 | 560.4 | 553.4 | 2024.6 | 2022.8 |
| 20,000 | 588.9 | 562.4 | 2025.6 | 2022.8 |
| 50,000 | 653.4 | 575.8 | 2030.0 | 2025.1 |
| 100,000 | 705.1 | 582.1 | 2034.9 | 2027.5 |
| 200,000 | 665.3 | 588.2 | - | 2031.3 |
| 計算時間 (秒) | | | | |
| データ数 | RAI | 提案 | RAI | 提案 |
| 10,000 | 773 | 621 | 4933 | 3189 |
| 20,000 | 1593 | 871 | 4794 | 2945 |
| 50,000 | 2587 | 1457 | 6354 | 3625 |
| 100,000 | 4784 | 2195 | 7761 | 6907 |
| 200,000 | 7256 | 3908 | - | 13172 |
| SHD | | | | |
| データ数 | RAI | 提案 | RAI | 提案 |
| 10,000 | 504.4 | 461.8 | 2736.1 | 2545.7 |
| 20,000 | 504.1 | 430.4 | 2800.1 | 2546.6 |
| 50,000 | 525.9 | 404.0 | 2924.4 | 2569.6 |
| 100,000 | 552.5 | 405.6 | 3037.9 | 2592.6 |
| 200,000 | 540.6 | 404.1 | - | 2627.1 |

Distance (SHD) [Tsamardinos 06] を使用する。SHD は真の構造と推定された構造間の距離を表し、小さいほど学習精度が高いことを示す。

表 1 に各手法の学習結果を示す。表 1 より、提案手法は従来の RAI アルゴリズムより CI テスト数と計算時間を削減して学習した。したがって、提案手法は大規模構造を従来より高速に学習できる。また、提案手法は従来の RAI アルゴリズムより SHD を減少して学習した。大規模構造ではデータがスパースになるために CI テストの信頼性が低下する傾向がある。提案手法は CI テストを削減したことで、信頼性の低い CI テストを減少して学習できたためと考えられる。従来の RAI アルゴリズムはデータ数 200,000 のときに random を制限時間内に学習できなかったが、提案手法は全パターンで学習できた。これより、提案手法は従来手法では実現できない構造学習を実現できる。

6. まとめ

本研究はベイジアンネットワークにおける条件付き独立性の推移性を示し、Bayes factor を用いた RAI アルゴリズムの CI テストとエッジ削除に推移性を利用した学習手法を提案した。これにより、学習に要する CI テスト数を削減し、より短い計算時間で大規模構造学習を実現した。また、信頼性の低い CI テストを減少して学習したために従来手法以上の精度で大規模構造を学習した。さらに、提案手法は従来手法では実現できない構造学習を実現した。

今後の課題として、大規模構造学習の精度向上と、推論精度を最適化する学習やベイジアンネットワーク分類器の厳密学習への拡張が挙げられる。

参考文献

[Chickering 96] Chickering, D. M.: Learning Bayesian Networks is NP-Complete, in *Learning from Data: Artificial*

Intelligence and Statistics, Vol. V, pp. 121–130, Springer (1996)

[Cussens 11] Cussens, J.: Bayesian network learning with cutting planes, in *Proceedings of Uncertainty in Artificial Intelligence (UAI)*, pp. 153–160, AUAI Press (2011)

[Heckerman 95] Heckerman, D., Geiger, D., and Chickering, D.: Learning Bayesian Networks: The Combination of Knowledge and Statistical Data, *Machine Learning*, Vol. 20, pp. 197–243 (1995)

[Natori 17] Natori, K., Uto, M., and Ueno, M.: Consistent learning Bayesian networks with thousands of variables, in *Advanced Methodologies for Bayesian Networks (Proceedings of Machine Learning Research)*, Vol. 73, pp. 57–68 (2017)

[Pearl 00] Pearl, J.: *Causality: Models, Reasoning, and Inference*, Cambridge University Press (2000)

[Scutari 11] Scutari, M.: Learning Bayesian Networks with the bnlearn R Package, *Journal of Statistical Software*, Vol. 35, No. 3, pp. 1–22 (2011)

[Silander 06] Silander, T. and Myllymaki, P.: A simple approach for finding the globally optimal Bayesian network structure, in *Proceedings of Uncertainty in Artificial Intelligence (UAI)*, pp. 445–452, AUAI Press (2006)

[Spirites 00] Spirites, P., Glymour, C., and Scheines, R.: *Causation, Prediction, and Search*, MIT press (2000)

[Steck 02] Steck, H. and Jaakkola, T.: On the Dirichlet Prior and Bayesian Regularization., in *Neural Information Processing Systems (NIPS 2002)*, pp. 697–704, MIT Press (2002)

[Tsamardinos 06] Tsamardinos, I., Brown, L. E., and Aliferis, C. F.: The Max-min Hill-climbing Bayesian Network Structure Learning Algorithm, *Machine Learning*, Vol. 65, No. 1, pp. 31–78 (2006)

[Ueno 10] Ueno, M.: Learning networks determined by the ratio of prior and data, in *Proceedings of Uncertainty in Artificial Intelligence (UAI)*, pp. 598–605, AUAI Press (2010)

[Yehezkel 09] Yehezkel, R. and Lerner, B.: Bayesian Network Structure Learning by Recursive Autonomy Identification, *Journal of Machine Learning Research*, Vol. 10, pp. 1527–1570 (2009)

[Yuan 11] Yuan, C., Malone, B., and Xiaojian, W.: Learning Optimal Bayesian Networks Using A* Search, in *International Joint Conference on Artificial Intelligence (IJCAI)*, pp. 2186–2191 (2011)

[名取 18] 名取和樹, 宇都雅輝, 植野真臣 FBayes factor を用いた RAI アルゴリズムによる大規模ベイジアンネットワーク学習, 電子情報通信学会論文誌 D, Vol. 101, No. 5, pp. 754–768 (2018)