論　文

深さ優先分枝限定法による目的変数パラメータ数を最小化するベイジアンネットワーク分類器学習

加藤　弘也[†a)]　　菅原　聖太[†b)]　　植野　真臣[†c)]

Learning Bayesian Network Classifiers to Minimize the Number of Class Variable Parameters by Depth-First Branch and Bound Algorithm

Koya KATO[†a)], Shouta SUGAHARA[†b)], and Maomi UENO[†c)]

あらまし　ベイジアンネットワーク分類器 (Bayesian Network Classifier: BNC) は高い分類精度をもつことが知られている．近年，最も分類精度の高い BNC として，I-map の中で目的変数パラメータ数 (Number of Class variable Parameters: NCP) を最小にして，分類確率の推定のみを最適化する BNC 学習法が提案されている．この BNC 学習では幅優先探索を用いたアルゴリズムが提案されているが，20 変数程度の構造学習が限界である．より大規模な構造を学習するために，本論文では，枝刈りにより構造学習を効率的に行うことができる深さ優先分枝限定法を用いた新たなアルゴリズムを提案する．具体的には，(1) Naive Bayes の NCP がその下限値であることを証明し，(2) 深さ優先分枝限定法のための NPC リバースオーダグラフを提案して，(1) で示した下限値を用いた枝刈りを導入する．提案手法は以下の利点がある．(1) 従来手法よりも計算時間を大幅に削減する．(2) 実行途中にメモリ等のリソースが不足してもそれまでの最適な構造を得ることが可能である．レポジトリデータセットを用いた実験により，提案アルゴリズムは従来手法では学習できない 58 変数の構造学習を実現することを示す．

キーワード　ベイジアンネットワーク，分類器，機械学習，確率的グラフィカルモデル

1. まえがき

確率アプローチに基づく分類器は，漸近的に真の分類確率を推定可能であるだけでなく，高い解釈性をもつことから様々な目的で応用されてきた [1]〜[3]．このような分類器の一つとして，ベイジアンネットワーク分類器 (Bayesian Network Classifier: BNC) が知られている [4]〜[9].

現在，最も分類精度の高い BNC として，菅原ら [9] は，目的変数が親変数をもたない制約の下で，真の同時確率を表現可能な構造のうち，目的変数に影響するパラメータ数 (Number of Class variable Parameters: NCP) のみ最小にする BNC 学習法を提案している．

一方，従来のベイジアンネットワーク (Bayesian Network: BN) 構造学習は，全ての構造の中で，周辺ゆう度 (Marginal Likelihood: ML) を最大にする構造を推定し [10]〜[19] , 分類に影響しない変数を含めた同時確率分布を推定する．菅原ら [9] の手法は分類に影響する目的変数に関わるパラメータ数のみを最小化し分類確率の推定のみを最適化しようとするもので，新たな学習アルゴリズムを必要とした．そこで，彼らは，以下の二つのステップからなる新しいアルゴリズムを提案している．第一ステップでは，目的変数から始まる全ての変数順序について，ML を最大化する構造をそれぞれ求める．このステップは変数数を n とすると $O(2^n)$ の計算量で計算される．第二ステップでは，第一ステップで求めた構造のうち NCP を最小にする構造を探索する．このステップは $O(n2^n)$ の計算量で計算される．第二ステップの探索はグラフの最短パス探索問題として定式化され，彼らのアルゴリズムは幅優先探索により最短パスを探索するが，以下の問題がある．(1) 変数数の増加に従い膨大な計算時間が必要となり，20 変数程度の構造学習が限界である．(2) 探索終了まで解となる構造を得ることができないため，時

† 電気通信大学大学院情報理工学研究科，調布市
Graduate school of Informatics and Engineering, The University of Electro-Communications, 1–5–1 Chofugaoka, Chofu-shi, 182–8585 Japan
a) E-mail: kato@ai.lab.uec.ac.jp
b) E-mail: sugahara@ai.lab.uec.ac.jp
c) E-mail: ueno@ai.is.uec.ac.jp
DOI:10.14923/transinfj.2023JDP7033

間制限やメモリ不足によりそれまでの探索結果が失われ一つも構造を得ることができない場合がある．

本論文では，以上の問題を緩和するために，新しいアルゴリズムを提案する．菅原ら [9] のアルゴリズムの第二ステップは，計算量が第一ステップに比べ大きく，大規模な構造学習のために改良する必要がある．そこで，本論文では，第二ステップにおける探索を枝刈りにより効率化することを考える．菅原ら [9] の用いている幅優先探索は逐次的に最適な構造を更新できないため，枝刈りを適用しても，その効果が制限的である．本論文では，幅優先探索ではなく，逐次的に最適な構造を更新する深さ優先探索に枝刈りを適用する新しい探索アルゴリズムを提案する．提案手法は菅原らの手法 [9] と計算量のオーダは変わらないが，幅優先探索から深さ優先探索に変更することで，探索が進むにつれて枝刈りの回数が増加し，探索空間の削減が加速する．枝刈りには最短パス探索におけるノードから終点までの NCP の下限値が必要になる．本研究では，目的変数の子変数を説明変数とする Naive Bayes の NCP がその下限値を与えることを証明する．この定理に基づき，提案手法は目的変数に関係する変数を Bayes factor による変数選択で求め，それらの変数で構成する Naive Bayes の NCP を探索における下限値とする．このアルゴリズムの利点として，以下が挙げられる．(1) 深さ優先分枝限定法における枝刈りを行うことで，計算量のオーダは変わらないが従来手法より計算時間の削減が可能になる．(2) 実行途中にメモリ等のリソースが不足してもそれまでの最適な構造を得ることが可能になる．

複数のベンチマークによる比較実験で，提案手法が従来手法と同等の精度を保ったまま，計算時間を削減することを示す．また，探索の途中でもその時点までの最適な構造が得られることの有効性を示す．更に，従来手法では 20 変数程度の構造学習が限界であったが，提案手法では 58 変数の構造学習を実現することを示す．

これまでにも枝刈りを行うことで BN 構造学習を効率的に行う手法が提案されている [14]〜[17] が，ML 最大化のために提案されたものであり，探索空間及び目的とする構造が本問題とは異なる．

2. 目的変数パラメータ数最小化による BNC 学習

2.1 ベイジアンネットワーク分類器

ベイジアンネットワーク分類器 (以降，BNC) は，ベイジアンネットワーク (以降，BN) の一つであり，確率変数をノードとし，ノード間の依存関係をエッジで表す非循環有向グラフ (Directed Acyclic Graph: DAG) G と，各ノードの親ノード集合を所与とした条件付き確率パラメータ集合 Θ で表現される [4]〜[9]．BNC では，G において一つのノードを目的変数，その他のノードを説明変数として扱い，目的変数は親変数をもたないとする．

今，G は離散確率変数集合 $\mathbf{V} = \{X_0, X_1, \ldots, X_n\}$ の各変数をノードとしてもつとする．また，各変数 X_i は r_i 個の状態集合 $\{1, \ldots, r_i\}$ から一つの値を取るとし，X_i が状態 k を取るとき，$X_i = k$ と書く．更に，X_0 を目的変数，$X_1, \ldots, X_n$ を説明変数とする．このとき，BNC では，同時確率分布を条件付き確率の積に分解して以下のように表せる．

$$P(X_0, X_1, \ldots, X_n \mid G) = \prod_{i=0}^{n} P(X_i \mid \mathbf{Pa}(X_i, G), G). \tag{1}$$

ここで，$\mathbf{Pa}(X_i, G)$ は X_i の G における親変数集合を表す．また，説明変数のデータ $\mathbf{x} = \langle x_1, \ldots, x_n \rangle$ を得たとき，目的変数の推定値 $\hat{c}$ は次式で表される．

$$\hat{c} = \underset{c \in \{1, \ldots, r_0\}}{\arg\max} \; P(c \mid \mathbf{x}, G, \Theta). \tag{2}$$

$\hat{c}$ を得るためには，条件付き確率パラメータ集合 Θ をデータから推定する必要がある．今，θ_{ijk} を $\mathbf{Pa}(X_i, G)$ が j 番目のパターンをとったとき ($\mathbf{Pa}(X_i, G) = j$ と書く) に，$X_i = k$ となる条件付き確率パラメータとする．ここで，条件付き確率パラメータ集合 Θ は θ_{ijk} を用いて $\Theta = \bigcup_{i=0}^{n} \bigcup_{j=1}^{q_i} \bigcup_{k=1}^{r_i} \{\theta_{ijk}\}$ で定義される．ここで，q_i は $\mathbf{Pa}(X_i, G)$ の取りうるパターン数であり，$q_i = \prod_{l: X_l \in \mathbf{Pa}(X_i, G)} r_l$ で定義される．θ_{ijk} の推定法には，その期待値である Expected A Posteriori (EAP) が最も良く用いられる．サンプルが N 個あるデータが得られたときの EAP は条件付き確率パラメータの事前分布にディリクレ分布を仮定すると以下で表される [20].

$$\hat{\theta}_{ijk} = \frac{\alpha_{ijk} + N_{ijk}}{\alpha_{ij} + N_{ij}}. \tag{3}$$

ここで，N_{ijk} は $X_i = k$ かつ $\mathbf{Pa}(X_i, G) = j$ となる頻度を表す．また，α_{ijk} はディリクレ事前分布のハイパーパラメータを表し，$N_{ij} = \sum_{k=1}^{r_i} N_{ijk}$, $\alpha_{ij} = \sum_{k=1}^{r_i} \alpha_{ijk}$

である．

BNC の条件付き確率パラメータを推定するためには，構造 G をデータから推定する必要があり，この問題を BNC の構造学習という．BNC の構造学習では，一般に，BN 同様 I-map の中で最適な構造を探索する．I-map を定義するために，まず，d 分離の定義を行う．BNC は，G 上で確率分布の条件付き独立性を d 分離により表現する．G における道 p 上の三変数 X,Y,Z が，$X \to Z \leftarrow Y$ と結合するとき，Z を p における合流点と呼ぶ．このとき，d 分離は以下のように定義される [21].

［定義 2.1］ G において $X,Y \in \mathbf{V}, \mathbf{Z} \subseteq \mathbf{V}\backslash\{X,Y\}$ としたとき，$\mathbf{Z}$ が X と Y を結ぶ任意の道 p について以下のいずれかの条件を満たすとき，G において X,Y は $\mathbf{Z}$ によって d 分離されるという．

（1） p における合流点ではない変数 $Z \in \mathbf{Z}$ が p 上に存在する．

（2） p における合流点 Z が p 上に存在し，Z とその子孫は $\mathbf{Z}$ に属さない．

この関係を $I(X,Y \mid \mathbf{Z})_G$ と書く．一方，真の同時確率分布において X,Y が $\mathbf{Z}$ を所与として条件付き独立であるとき，$I(X,Y \mid \mathbf{Z})_M$ と書く．ここで，I-map は以下で定義される [21].

［定義 2.2］ G が以下を満たすとき，G をインディペンデントマップ (Independent map: I-map) という．

$$\forall X,Y \in \mathbf{V}, \forall \mathbf{Z} \subseteq \mathbf{V}\backslash\{X,Y\}, \quad I(X,Y \mid \mathbf{Z})_G \Rightarrow I(X,Y \mid \mathbf{Z})_M. \tag{4}$$

G が I-map であるとき，その構造が表現する同時確率分布は漸近的に真の同時確率分布に収束する [22].

2.2 目的変数パラメータ数を最小にする BNC

2.2.1 手法の概要

前節で紹介した BNC は，全変数の同時確率分布を推定するもので，分類に関係のない変数についても最適化しており，分類確率の推定効率が悪い．そこで，菅原ら [9] は，分類確率のみを効率的に推定する BNC として，以下で定義される目的変数パラメータ数 (以降，NCP) を最小にする I-map の学習法を提案した．

$$NCP(G) = \sum_{i=0}^{n} NCP_i(\mathbf{Pa}(X_i,G)). \tag{5}$$

ここで，$i = 0 \vee X_0 \in \mathbf{Pa}(X_i,G)$ のとき $NCP_i(\mathbf{Pa}(X_i,G)) = (r_i - 1)q_i$ であり，それ以外のとき $NCP_i(\mathbf{Pa}(X_i,G)) = 0$ である．

菅原ら [9] の手法を説明するため，変数順序及び周辺ゆう度を以下で定義する．$\mathbf{V}$ の各変数を要素とするベクトル σ に対し，σ の i 番目の要素 X_{σ^i} について $\forall i \in \{1,\ldots,n\}, \mathbf{Pa}(X_{\sigma^i},G) \subseteq \bigcup_{j=1}^{i-1}\{X_{\sigma^j}\}$ が成り立つとき，σ を G の変数順序という．また，G の周辺ゆう度 $P(D \mid G)$ は条件付き確率パラメータの事前分布をディリクレ分布であると仮定すると，次のように閉形式で表される [23].

$$P(D \mid G) = \prod_{i=0}^{n}\prod_{j=1}^{q_i}\frac{\Gamma(\alpha_{ij})}{\Gamma(\alpha_{ij}+N_{ij})}\prod_{k=1}^{r_i}\frac{\Gamma(\alpha_{ijk}+N_{ijk})}{\Gamma(\alpha_{ijk})}. \tag{6}$$

近年では，$\alpha_{ijk} = \alpha/(r_i q_i)$ とした，Bayesian Dirichlet equivalent uniform (BDeu) が一般的に用いられる [20], [23]．ここで，α は Equivalent Sample Size(ESS) と呼ばれる事前知識の重みを示す擬似サンプルのサイズである．また，$\log BDeu$ は次の性質を満たす．

$$\log BDeu(G) = \sum_{i=0}^{n} Score_i(\mathbf{Pa}(X_i,G)). \tag{7}$$

ここで，$Score_i(\mathbf{Pa}(X_i,G))$ は X_i と $\mathbf{Pa}(X_i,G)$ のみに依存する関数であり，ローカルスコアと呼ばれる．ローカルスコア $Score_i(\mathbf{Pa}(X_i,G))$ は以下のように表せる [23].

$$Score_i(\mathbf{Pa}(X_i,G)) = \sum_{j=1}^{q_i}\left(\log\frac{\Gamma(\alpha_{ij})}{\Gamma(\alpha_{ij}+N_{ij})} + \sum_{k=1}^{r_i}\log\frac{\Gamma(\alpha_{ijk}+N_{ijk})}{\Gamma(\alpha_{ijk})}\right). \tag{8}$$

菅原ら [9] は変数集合 $\mathbf{V}$ からなる全ての変数順序集合を $\boldsymbol{\sigma}(\mathbf{V})$ としたとき，次の定理を証明した．

［定理 2.1］ $\forall\sigma \in \boldsymbol{\sigma}(\mathbf{V})$ について，σ を所与として BDeu を最大化する構造は，σ に従う I-map の中で NCP が最小の構造に漸近的に一致する．

この定理に基づき，彼らの手法は以下の二つのステップから構成される．第一ステップでは，目的変数から始まる全ての変数順序について，BDeu を最大化する構造をそれぞれ求める．第二ステップでは，第一ステップで得られた構造の中で NCP 最小の構造を探索する．

2.2.2 学習アルゴリズム

菅原ら [9] の学習アルゴリズムの具体的な説明のた

め，用語の定義や表記を以下で定める．変数順序 σ において変数 X に先行する変数の集合を $\mathbf{Pre}_X^\sigma$ で表す．また，変数 X_i について，候補親変数集合を $\mathbf{Z}_i \subseteq \mathbf{V}\backslash\{X_i\}$ としたときの最適親変数集合 $g_i^*(\mathbf{Z}_i)$ を以下で定義する．

$$g_i^*(\mathbf{Z}_i) = \arg\max_{\mathbf{W}\subseteq\mathbf{Z}_i} Score_i(\mathbf{W}). \tag{9}$$

$X_0 \in \mathbf{Z}$ となる変数集合 $\mathbf{Z} \subseteq \mathbf{V}$ に対する変数順序 $\sigma_\mathbf{Z} \in \boldsymbol{\sigma}(\mathbf{Z})$ のうち目的変数が先頭になるもの，つまり $X_{\sigma_\mathbf{Z}^1} = X_0$ となるものの集合を $\boldsymbol{\sigma_0}(\mathbf{Z})$ で表す．また，変数順序 $\sigma_\mathbf{Z} \in \boldsymbol{\sigma}(\mathbf{Z})$ に従う変数集合 $\mathbf{Z}$ からなる構造の中で BDeu を最大にする構造を $G^*_{\sigma_\mathbf{Z}}(\mathbf{Z})$ で表す．定理 2.1 より $G^*_{\sigma_\mathbf{Z}}(\mathbf{Z})$ は $\sigma_\mathbf{Z}$ に従う I-map の中で NCP が最小の構造に一致する．また，$G^*(\mathbf{Z})$ を変数集合 $\mathbf{Z}$ からなる I-map の中で NCP が最小の構造とする．$G^*(\mathbf{Z})$ は $\forall\sigma_0 \in \boldsymbol{\sigma_0}(\mathbf{Z})$ に対して，$NCP(G^*_{\sigma^*}(\mathbf{Z})) \leq NCP(G^*_{\sigma_0}(\mathbf{Z}))$ となる $\sigma^* \in \boldsymbol{\sigma_0}(\mathbf{Z})$ についての $G^*_{\sigma^*}(\mathbf{Z})$ と定義される．また，ある変数が子変数をもたないとき，その変数を sink と呼ぶ．

第一ステップでは，$\forall\sigma_0 \in \boldsymbol{\sigma_0}(\mathbf{V})$ について σ_0 に従う I-map の中で NCP が最小の構造 $G^*_{\sigma_0}(\mathbf{V})$ を求める．$\mathbf{V}$ の変数数を $n+1$ としたとき，$G^*_{\sigma_0}(\mathbf{V})$ は各変数とその親変数集合を対とする $n+1$ 組の対 $(X_0, \emptyset), (X_1, g_1^*(\mathbf{Pre}_{X_1}^{\sigma_0})), \ldots, (X_n, g_n^*(\mathbf{Pre}_{X_n}^{\sigma_0}))$ で表される．したがって，$G^*_{\sigma_0}(\mathbf{V})$ を得るには $\forall\sigma_0 \in \boldsymbol{\sigma_0}(\mathbf{V}), \forall i \in \{1,\ldots,n\}$ について $g_i^*(\mathbf{Pre}_{X_i}^{\sigma_0})$ を求めれば良い．しかし，異なる二つの変数順序 σ と σ' について，$\mathbf{Pre}_{X_i}^\sigma = \mathbf{Pre}_{X_i}^{\sigma'}$ のとき，$g_i^*(\mathbf{Pre}_{X_i}^\sigma) = g_i^*(\mathbf{Pre}_{X_i}^{\sigma'})$ となる．この重複探索を避けるためには，$\forall i \in \{1,\ldots,n\}, \forall \mathbf{Z}_i \subseteq \mathbf{V}\backslash\{X_i\}$ に対する $g_i^*(\mathbf{Z}_i)$ の計算のみを行えば良い [9]．$g_i^*(\mathbf{Z}_i)$ は，式 (8), (9) により計算される．このステップは，並列化を行うことで，$O(2^n)$ の計算量で計算される [13]．

第二ステップでは，第一ステップで得られた $\boldsymbol{\sigma_0}(\mathbf{V})$ の各変数順序に従う NCP 最小の I-map の中から NCP が最小の構造を探索する．この構造は，変数集合 $\mathbf{V}$ からなる I-map の中で NCP が最小の構造 $G^*(\mathbf{V})$ に一致する．菅原ら [9] は，この探索を Silander ら [13] の動的計画法を用いて行った．

3. 深さ優先分枝限定法による BNC 学習法の提案

本論文では，菅原 [9] らのアルゴリズムの第二ステップにおける探索を効率化する新たなアルゴリズムを提案する．彼らの手法は第二ステップにおける探索にグラフを用いていないため，枝刈りを適用することができない．そこで，本論文では，第二ステップにおける探索をグラフを用いた最短パス探索問題として定式化する．

枝刈りはその回数が増加するほど探索空間をより削減する．枝刈りの回数は，最適な構造を逐次的に更新することにより効率的に増加する．そこで，本論文では，最適構造の更新が可能な深さ優先探索に枝刈りを適用する．深さ優先探索は，枝刈りの回数を増加させるだけでなく，時間制限やメモリ等のリソースが不足した場合でもそれまでの最適な構造を得ることができる．以下では，このアルゴリズムを深さ優先分枝限定法と呼ぶ．本章では，まず，第二ステップにおける探索を最短パス探索問題として定式化し，その後，深さ優先分枝限定法の詳細を述べる．

3.1 NPC リバースオーダグラフ

本節では，枝刈りを導入するために菅原ら [9] の手法の第二ステップを NPC リバースオーダグラフを用いた最短パス探索問題として定式化する．NPC リバースオーダグラフは，$X_0 \in \mathbf{U}$ となる任意の変数集合 $\mathbf{U} \subseteq \mathbf{V}$ に対応するノード集合と，ノード集合の各ノード $\mathbf{U}$ からノード $\mathbf{U}\backslash\{X_i\}$ に引かれたエッジ集合からなる有向グラフである．4 変数に対する NPC リバースオーダグラフ例を図 1 に示す．ここで，ノード $\mathbf{U}$ からノード $\mathbf{U}\backslash\{X_i\}$ へのエッジは変数集合 $\mathbf{U}$ からなる構造の sink が X_i であり，X_i の親変数集合が $g_i^*(\mathbf{U}\backslash\{X_i\})$ であることを意味する．例えば，エッジ $\{X_0, X_1, X_2, X_3\} \to \{X_0, X_1, X_2\}$ は $\{X_0, X_1, X_2, X_3\}$ からなる構造の sink が X_3 であり，X_3 の親変数集合が $g_3^*(\{X_0, X_1, X_2\})$ であることを意味する．そして，ノード $\mathbf{U}$ からノード $\mathbf{U}\backslash\{X_i\}$ へのエッジがもつ NCP をコストとして $NCP_i(g_i^*(\mathbf{U}\backslash\{X_i\}))$ と定義する．これは，X_i の親変数集合が $g_i^*(\mathbf{U}\backslash\{X_i\})$ であるときの NCP_i に対応する．NPC リバースオーダ

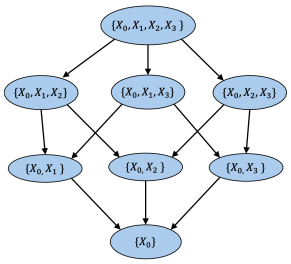


図 1　4 変数の NPC リバースオーダグラフ
Fig. 1　A NPC reverse order graph of four variables.

グラフの始点 $\mathbf{V}$ から終点 $\{X_0\}$ へのノードの連なりであるパスは $\boldsymbol{\sigma_0}(\mathbf{V})$ の各変数順序に対応し，変数順序 $\sigma_0 \in \boldsymbol{\sigma_0}(\mathbf{V})$ に対応するパス p を始点から終点へ辿ることで，変数数を $n+1$ としたとき，sink とその親変数集合の n 組の対 $(X_1, g_1^*(\mathbf{Pre}_{X_1}^{\sigma_0})), \cdots, (X_n, g_n^*(\mathbf{Pre}_{X_n}^{\sigma_0}))$, すなわち $G^*_{\sigma_0}(\mathbf{V})$ が得られる．また，パスのコストをそのパスを始点から終点まで辿ったときのコストの総和に X_0 の NCP_0 である $r_0 - 1$ を加えたものと定義すると，変数順序 σ_0 に対応するパス p のコスト $c(p)$ は

$$\begin{aligned} c(p) &= \sum_{i=1}^{n} NCP_i(g_i^*(\mathbf{Pre}_{X_i}^{\sigma_0})) + r_0 - 1 \\ &= NCP(G^*_{\sigma_0}(\mathbf{V})) \end{aligned} \tag{10}$$

となり，σ_0 に従う I-map の中で NCP が最小の構造 $G^*_{\sigma}(\mathbf{V})$ の NCP, $NCP(G^*_{\sigma}(\mathbf{V}))$ に一致する．最短パス p^* を任意のパス p に対し，$c(p^*) \leq c(p)$ が成り立つパスと定義し，最短パス p^* に対応する変数順序を σ_0^* とすると，最短パスのコスト $c(p^*)$ は

$$\begin{aligned} c(p^*) &= \sum_{i=1}^{n} NCP_i(g_i^*(\mathbf{Pre}_{X_i}^{\sigma^*})) + r_0 - 1 \\ &= NCP(G^*_{\sigma^*}(\mathbf{V})) \\ &\leq NCP(G^*_{\sigma_0}(\mathbf{V})) \end{aligned} \tag{11}$$

と表せる．最短パスの探索は，第一ステップで求めた構造の中で NCP が最小になる構造の変数順序を探索することと等しい．最短パス p^* により，I-map の中で NCP が最小の構造 $G^*(\mathbf{V})$ が得られる．

3.2 深さ優先分枝限定法を用いた最短パス探索

菅原ら [9] の第二ステップの手法は枝刈りによる効率化を行わないので探索グラフを用いていないが，NPC リバースオーダグラフの最短パスを幅優先探索する手法に一致する．図 2 は 4 変数の NPC リバースオーダグラフに対する幅優先探索の動作例を表している．幅優先探索は NPC リバースオーダグラフの始点 $\mathbf{V}$ から探索を開始し，幅優先的にノードの展開を行う．

例えば，図 2 では，まず $\{X_0, X_1, X_2, X_3\}$ を展開し，$\{X_0, X_1, X_2\}, \{X_0, X_1, X_3\}, \{X_0, X_2, X_3\}$ の順で探索する．次に，$\{X_0, X_1, X_2\}$ を展開し，$\{X_0, X_1\}, \{X_0, X_2\}$ の順で探索する．また，$\{X_0, X_1, X_2\}$ の展開が終了すると，$\{X_0, X_1, X_3\}$ の展開が行われ，以後同様に，全てのノードの展開を行う．幅優先探索は変数数を $n+1$ としたとき，$O(n2^n)$ の計算量を必要とする．幅優先探索は変数数の増加に伴い膨大な計算時間が必要になるため，

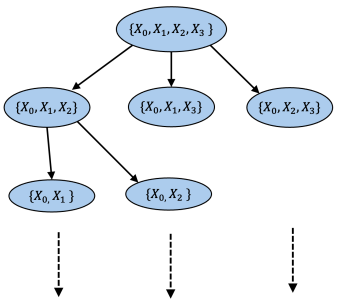


図 2　幅優先探索の動作例
Fig. 2　A running example of the breadth-first search.

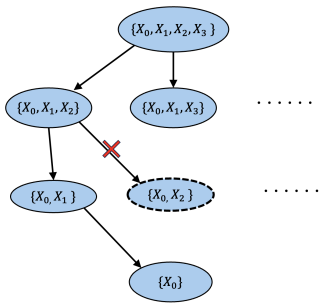


図 3　深さ優先分枝限定法の動作例
Fig. 3　A running example of the depth-first branch and bound algorithm.

20 変数程度の構造学習が限界である．

本論文では，NPC リバースオーダグラフを用いた探索を枝刈りにより効率化することを考える．幅優先探索は，最適構造を逐次的に更新できないため，枝刈りを適用しても，その効果が限定的である．本論文では，最適構造の更新が可能な深さ優先探索に枝刈りを適用する．枝刈りは $f(\mathbf{U})$ コストを利用して行われる．$f(\mathbf{U})$ コストは，NPC リバースオーダグラフにおける始点からノード $\mathbf{U}$ までのパスのコスト ($g(\mathbf{U})$ コスト) と，ノード $\mathbf{U}$ から終点までのパスのコストの下限値 ($h(\mathbf{U})$ コスト) の和として定義される．$f(\mathbf{U})$ コストがこれまでに発見されている最適な構造のコストを上回るとき，ノード $\mathbf{U}$ を通るいずれのパスも最適ではないとわかるため，ノード $\mathbf{U}$ は枝刈りされる．

図 3 は深さ優先分枝限定法の動作例を表している．深さ優先分枝限定法は NPC リバースオーダグラフの始点 $\mathbf{V}$ から探索を開始し，深さ優先的にノードの展開を行う．

例えば，図 3 では，まず $\{X_0, X_1, X_2, X_3\}$ を展開し，$\{X_0, X_1, X_2\}$ を探索する．次に，$\{X_0, X_1, X_2\}$ を展開し，$\{X_0, X_1\}$ を探索する．そして，$\{X_0, X_1\}$ を展開し，終

点である $\{X_0\}$ を探索する．この時点で，変数順序 (X_0, X_1, X_2, X_3) に従う I-map の中で NCP が最小の構造を得られる．そして，この構造の NCP をこれまでの最適構造のコストとする．この更新は，逐次的に最適構造を更新する深さ優先探索特有のプロセスであり，このコストは探索が進むにつれて小さい値に更新される．この更新が，枝刈りの回数を増加させる．また，$\{X_0, X_1\}$ の展開が終了すると，次に $\{X_0, X_1, X_2\}$ から $\{X_0, X_2\}$ への展開を行う．ここで，$f(\{X_0, X_2\})$ がこれまでの最適構造のコストを上回っている場合，$\{X_0, X_1, X_2\}$ から $\{X_0, X_2\}$ への展開は行われず，枝刈りされる．その後，$\{X_0, X_1, X_2, X_3\}$ から $\{X_0, X_1, X_3\}$ への展開が行われ，以後同様に，枝刈りを行いながら探索を行う．深さ優先分枝限定法は，幅優先探索と計算量のオーダは変わらないが，枝刈りにより効率的な探索を実現する．

この深さ優先分枝限定法の詳細を Algorithm 1 に示す．Algorithm 1 では，データ $\mathbf{D}$ を入力として，関数 EXPAND を再帰的に実行することで，NPC リバースオーダグラフの探索を行う．また，関数 EXPAND 内の $h_{exact}(\mathbf{U})$ は，NPC リバースオーダグラフのノード $\mathbf{U}$ から終点までのパスのコストを表し，*optimal* は，それまでに得られている最適なパスのコストを表す．関数 EXPAND の概略は次のとおりである．NPC リバースオーダグラフにおけるノード $\mathbf{U}$ を入力として受け取り，ノード $\mathbf{U}\backslash\{X_i\}$ の探索を開始する (20 行目から 36 行目)．(1) ノード $\mathbf{U}\backslash\{X_i\}$ における $g(\mathbf{U}\backslash\{X_i\})$ の計算を行う (21 行目から 25 行目)．(2) ノード $\mathbf{U}\backslash\{X_i\}$ における $h(\mathbf{U}\backslash\{X_i\})$ の計算を行い，(1) で求めた $g(\mathbf{U}\backslash\{X_i\})$ との和をとることで，$f(\mathbf{U}\backslash\{X_i\})$ を計算する (29 行目)．(3) それまでに得られている最適なコスト *optimal* が (2) で計算した $f(\mathbf{U}\backslash\{X_i\})$ を下回る場合，ノード $\mathbf{U}\backslash\{X_i\}$ を通るいずれのパスも最適ではないとわかるため，関数 EXPAND は呼び出さず，$\mathbf{U}\backslash\{X_i\}$ は枝刈りされる．また，*optimal* が $f(\mathbf{U}\backslash\{X_i\})$ を上回る場合，ノード $\mathbf{U}\backslash\{X_i\}$ を入力にして，関数 EXPAND を再帰的に呼び出す (30 行目から 32 行目)．(4) $h_{exact}(\mathbf{U})$ の更新，*optimal* の更新を行う (33 行目から 42 行目)．(4) の *optimal* の値は探索が進むにつれて小さい値に更新される．次節では，(2) の計算で必要になる $h(\mathbf{U})$ の推定法を提案する．

3.3 Naive Bayes を用いた下限

$h(\mathbf{U})$ コストの計算には，ヒューリスティック関数を用いる．本論文では，以下の定理 3.1 を証明し，新しいヒューリスティック関数を提案する．

Algorithm 1 The depth first branch and bound algorithm

```
1:  function MAIN(D)
    D: データ
2:      第一ステップの探索として ∀i ∈ {1,...,n}, ∀Z_i ⊆ V\{X_i} に対
    する g*_i(Z_i) の計算を行う
3:      for U ∈ 2^{V\{X_0}} do
4:          h_exact(U ∪ {X_0}) ← ∞
5:          g(U ∪ {X_0}) ← ∞
6:      end for
7:      optimal ← ∞
8:      Repair ← (V, 0)
9:      while |Repair| > 0 do
10:         for (S, g) ∈ Repair do
11:             if g(S) > g then
12:                 g(S) ← g
13:                 Repair' ← EXPAND(S, Repair)
14:             end if
15:         end for
16:         Repair ← Repair'
17:     end while
18: end function

19: function EXPAND(U, Repair)
20:     for X_i ∈ U\{X_0} do
21:         g ← g(U) + NCP_i(g*_i(U\{X_i}))
22:         duplicate ← exists(g(U\{X_i}))
23:         if g < g(U\{X_i}) then
24:             g(U\{X_i}) ← g
25:         end if
26:         if duplicate and g < g(U\{X_i}) then
27:             Repair ← Repair ∪ (U, g)
28:         end if
29:         f ← h(U\{X_i}) + g(U\{X_i})
30:         if !duplicate and f < optimal then
31:             EXPAND(U\{X_i}, Repair)
32:         end if
33:         if h_exact(U) > NCP_i(g*_i(U\{X_i})) + h_exact(U\{X_i}) then
34:             h_exact(U) ← NCP_i(g*_i(U\{X_i})) + h_exact(U\{X_i})
35:         end if
36:     end for
37:     if U == {X_0} then
38:         h_exact(U) ← 0
39:     end if
40:     if optimal > h_exact(U) + g(U) then
41:         optimal ← h_exact(U) + g(U)
42:     end if
43:     return Repair
44: end function
```

［定理 3.1］ 任意の変数集合 $\mathbf{V}$ に対して，I-map の中で NCP を最小にする BNC を $G^*(\mathbf{V})$ とし，$G^*(\mathbf{V})$ における目的変数の子変数集合 $\mathbf{V}_c$ を説明変数とする Naive Bayes を $G^{NB}(\mathbf{V}_c)$ とすると

$$NCP(G^{NB}(\mathbf{V}_c)) \leq NCP(G^*(\mathbf{V})) \tag{12}$$

が成り立つ．ただし，Naive Bayes は，全説明変数が目的変数のみを親にもつと仮定する BNC である．

証明は付録に記した．

定理 3.1 より，$G^*(\mathbf{V})$ の目的変数の子変数集合 $\mathbf{V}_c$ を得ることで，$G^*(\mathbf{V})$ の NCP の下限値を推定できる．したがって，本論文では，NPC リバースオーダグラフにおけるノード $\mathbf{U}$ から終点までの下限を計算するために，次式で表される新しいヒューリスティック関数を提案する．

$$h^*(\mathbf{U}) = \sum_{X_i \in (\mathbf{U} \cap \mathbf{V}_c)} NCP_i(X_0). \tag{13}$$

提案するヒューリスティック関数は，目的変数の子変数集合 $\mathbf{V}_c$ が与えられたとき，無矛盾性という性質をもつ．無矛盾性をもつヒューリスティック関数は，最短パスの探索を保証する [24].

［定義 3.1］ 探索グラフの任意のノード $\mathbf{U}$ と $\mathbf{U}$ からエッジが引かれている任意のノード $\mathbf{R}$ について，ヒューリスティック関数 $h(\mathbf{U})$ が以下の不等式を満たすとき，$h(\mathbf{U})$ は無矛盾であるという．

$$h(\mathbf{U}) \leq h(\mathbf{R}) + c(\mathbf{U}, \mathbf{R}). \tag{14}$$

ここで，$c(\mathbf{U}, \mathbf{R})$ はエッジ $\mathbf{U} \rightarrow \mathbf{R}$ のコストである．

［定理 3.2］ h^* は無矛盾である．

証明は付録に記した．

しかし，提案するヒューリスティック関数における $\mathbf{V}_c$ を事前に知ることはできない．そこで，提案手法では，Bayes factor による変数選択により $\mathbf{V}_c$ を推定する．

Sugahara ら [7] は，Bayes factor により目的変数の子変数を選択する手法を提案した．Bayes factor は，目的変数 X_0 と説明変数 X_i 間について従属なモデル g_1 と独立なモデル g_2 の BDeu の比を求めることで厳密なモデル選択を行う手法である．このときの Bayes factor $BF(X_0, X_i)$ は，

$$BF(X_0, X_i) = \frac{BDeu(g_1)}{BDeu(g_2)} \tag{15}$$

で表される．Sugahara ら [7] の手法では，式 (15) の対数 Bayes factor がしきい値より小さい場合に，X_0 と X_i は独立であると判定する．BDeu に基づく Bayes factor は漸近的に真の目的変数の子変数集合を推定できる [7], [25], [26].

提案手法では，深さ優先分枝限定法を行う前に目的変数 X_0 と $\forall X_i \in \mathbf{V} \backslash \{X_0\}$ について Sugahara ら [7] で提案された Bayes factor を用いた変数選択を行う．そして，選択された変数集合を $\mathbf{V}_c$ として用いる．

深さ優先分枝限定法の効率は，下限推定の際の計算量に依存し，その計算量が大きいとき，枝刈りを行わない場合よりも悪化することがある．提案手法において，ヒューリスティック関数自身の計算は変数数を $n+1$ としたとき，$O(n)$ で行われる．また，提案手法は深さ優先分枝限定法の前に Bayes factor を用いた変数選択による $\mathbf{V}_c$ の推定を必要とする．提案手法は $\mathbf{V}_c$ を $O(n)$ で推定可能であり，下限推定の際の計算量は最短パス探索に必要な計算量に比べ十分に小さい．以上から，提案手法は，幅優先探索で全てのノードとエッジを探索する場合に比べ枝刈りによる計算時間の改善が期待される．

これまでに，枝刈りにより BN 構造学習を効率化するアルゴリズムが提案されている [14]〜[17]．これらのアルゴリズムに提案アルゴリズムも大きくインスパイアされているが，本問題とは探索空間及び目的が異なることに留意いただきたい．

4. 評価実験

この章では提案手法の利点を示すための実験を行う．

4.1 実データを用いた分類精度比較

まず，実データを用いて以下の 6 種類の手法の分類精度を比較する．

- Naive Bayes (Friedman ら [4])
- GBN-BDeu: 全ての構造の中で BDeu が最大となる BNC
- ANB-BDeu (菅原ら [5]〜[7]): BDeu が最大となる Augmented Naive Bayes
- 幅優先探索 (菅原ら [9]): 幅優先探索を用いて学習した，NCP 最小 I-map となる BNC
- 幅優先分枝限定法 : Naive Bayes による下限値を用いた幅優先分枝限定法により学習した，NCP 最小 I-map となる BNC
- 深さ優先分枝限定法 (提案手法) : Naive Bayes による下限値を用いた深さ優先分枝限定法により学習した，NCP 最小 I-map となる BNC

Naive Bayes, GBN-BDeu, ANB-BDeu については，Java で実装し，幅優先探索，幅優先分枝限定法，深さ優先分枝限定法については C++ で実装した．また，3.2 GHz の 16 コアプロセッサと 128 GB のメモリを搭載した PC で実験を行った．この実験では，UCI レポジトリデータベース [27] に登録されている 21 個のベンチマークデータセットを用いた．菅原ら [9] と同様

表 1　各手法の分類精度.
Table 1　Classification accuracies of the respective methods.

No.	Dataset	Variables	Sample size	SPP	Naive Bayes	GBN-BDeu	ANB-BDeu	幅優先探索	幅優先分枝限定法	深さ優先分枝限定法
1	Lenses	5	24	3.33×10^{-1}	0.7500	0.8333	0.7500	0.8750 (2.0)	0.8750 (2.0)	0.8750 (2.0)
2	Hayes-Roth	5	132	2.29×10^{-1}	0.8182	0.6212	0.7879	0.8333 (3.0)	0.8333 (3.0)	0.8333 (3.0)
3	Iris	5	150	3.13×10^{0}	0.7133	0.8267	0.8156	0.8200 (3.1)	0.8200 (3.9)	0.8200 (3.1)
4	Balance Scale	5	625	3.33×10^{-1}	0.9152	0.9152	0.9152	0.9152 (4.0)	0.9152 (4.0)	0.9152 (4.0)
5	Banknote authentication	5	1372	4.29×10^{1}	0.8433	0.8812	0.8812	0.8819 (2.0)	0.8819 (2.0)	0.8819 (2.0)
6	LED7	8	3200	2.50×10^{0}	0.7294	0.7303	0.7303	0.7316 (7.0)	0.7309 (7.0)	0.7325 (7.0)
7	HTRU2	9	17898	3.50×10^{1}	0.8966	0.9141	0.9141	0.9140 (7.6)	0.9141 (7.7)	0.9140 (7.5)
8	Breast Cancer	10	277	8.33×10^{-4}	0.7401	0.7256	0.6968	0.7401 (3.0)	0.7509 (2.2)	0.7401 (2.6)
9	Breast Cancer Wisconsin	10	683	3.42×10^{-7}	0.9751	0.9751	0.9751	0.9751 (8.7)	0.9751 (8.7)	0.9751 (8.7)
10	Solar Flare	11	1389	3.72×10^{-3}	0.7811	0.8431	0.8215	0.8431 (0.0)	0.8431 (0.0)	0.8431 (0.0)
11	Wine	14	178	7.24×10^{-3}	0.9270	0.9270	0.9270	0.9494 (6.8)	0.9438 (10.8)	0.9494 (6.8)
12	Heart	14	270	2.44×10^{-3}	0.8259	0.8370	0.8037	0.8407 (5.3)	0.8074 (7.9)	0.8407 (5.4)
13	Australian	15	690	2.97×10^{-4}	0.8290	0.8507	0.8203	0.8493 (4.3)	0.8507 (4.3)	0.8464 (4.0)
14	EEG	15	14980	4.57×10^{-1}	0.5778	0.7246	0.7212	0.7155 (12.3)	0.7155 (12.3)	0.7135 (12.4)
15	Zoo	17	101	1.03×10^{-4}	0.9802	0.9228	0.9406	0.9505 (7.1)	0.9505 (6.9)	0.9406 (6.3)
16	Congressional Voting Records	17	232	1.77×10^{-3}	0.9095	0.9655	0.9483	0.9655 (2.9)	0.9569 (3.6)	0.9698 (2.9)
17	Pendigits	17	10992	1.68×10^{-2}	0.8032	0.9342	0.9332	0.9349 (16.0)	0.9349 (16.0)	0.9344 (16.0)
18	Letter	17	20000	1.17×10^{-2}	0.4466	0.6434	0.6434	0.6309 (15.9)	0.6309 (15.9)	0.6297 (15.9)
19	Lymphography	19	148	1.63×10^{-7}	0.8446	0.7500	0.8108	0.8041 (6.8)	0.8108 (7.5)	0.7905 (7.0)
20	Image Segmentation	19	2310	1.26×10^{-3}	0.7290	0.8255	0.8273	0.8208 (14.9)	0.8212 (14.9)	0.8277 (15.0)
21	Hepatitis	20	80	7.63×10^{-5}	0.8500	0.6125	0.5750	0.7875 (2.1)	0.8000 (2.5)	0.8000 (2.6)
	average				0.8041	0.8219	0.8209	0.8466 (6.4)	0.8458 (6.8)	0.8463 (6.4)

に，各データセットに含まれる連続量はいずれも中央値を境に 2 値に離散化し，欠損値を含むサンプルはデータセットから除去した．いずれの手法においても，構造学習後の BNC の条件付き確率パラメータは全て EAP で推定した．GBN-BDeu, ANB-BDeu, 幅優先探索，幅優先分枝限定法と深さ優先分枝限定法の ESS については，10 分割交差検証を用いて $\{1, 10, 100, 1000\}$ から定めた．また，幅優先分枝限定法と深さ優先分枝限定法における変数選択の際の対数 Bayes factor のしきい値は 0 とした．

各手法，各データセットに対して，10 分割交差検証によるテストデータの平均一致率を求め，分類精度として表 1 に示した．表 1 における SPP(Sample Per Pattern: SPP) は全変数がとりうる値のパターン数でサンプルサイズを割ったものを表す．また，表中では，幅優先探索，幅優先分枝限定法，深さ優先分枝限定法によって学習された構造の目的変数の子変数数を括弧書きで示している．

表 1 より，深さ優先分枝限定法の平均分類精度は枝刈りを行わない幅優先探索の平均分類精度にわずかに劣るが，ほぼ同等であることが確認できる．この結果は，深さ優先分枝限定法の枝刈りがそれを用いない場合に比べて分類精度をそれほど低下させないことを示している．

表 1 より，深さ優先分枝限定法の平均分類精度は Naive Bayes の平均分類精度を上回っていることが確認できる．Naive Bayes は全説明変数が目的変数のみを親にもつため，表現力が低い [28]．ただ，21 番のデータセットでは，深さ優先分枝限定法の分類精度が Naive Bayes の分類精度を下回っている．これらのデータセットでは，SPP が小さいため，深さ優先分枝限定法は目的変数の子変数を過小に学習している．その結果，深さ優先分枝限定法が学習した構造は分類上重要な変数が削除されてしまったと考えられる．

更に，表 1 より，深さ優先分枝限定法の平均分類精度は GBN-BDeu, ANB-BDeu の平均分類精度を上回っていることも確認できる．GBN-BDeu, ANB-BDeu は，NCP の最小化を保証しない [9]．したがって，深さ優先分枝限定法は GBN-BDeu, ANB-BDeu に比べて NCP の小さな構造を学習し，結果として高い分類精度を示したと考えられる．

次に，(1) 枝刈りがそれを用いない場合に比べて NCP を大きくさせないこと，(2) 深さ優先分枝限定法が GBN-BDeu, ANB-BDeu に比べて小さい NCP をもつことを確認するために，各データセットに対して NCP を推定した．その結果を表 2 に示す．

表 2 より，深さ優先分枝限定法の平均 NCP は枝刈りを用いない幅優先探索の平均 NCP より少し大きい

表 2　各手法によって学習された構造の NCP.
Table 2　NCPs of the learned structures of the respective methods.

No.	Variables	Sample size	SPP	GBN-BDeu	ANB-BDeu	幅優先探索	幅優先分枝限定法	深さ優先分枝限定法
1	5	24	3.33×10^{-1}	8	18	8	8	8
2	5	132	2.29×10^{-1}	176	40	29	29	29
3	5	150	3.13×10^{0}	18	28	19	25	19
4	5	625	3.33×10^{-1}	48	50	50	50	50
5	5	1372	4.29×10^{1}	25	31	15	15	15
6	8	3200	2.50×10^{0}	186	187	94	93	98
7	9	17898	3.50×10^{1}	69	77	198	118	176
8	10	277	8.33×10^{-4}	20	105	35	22	30
9	10	683	3.42×10^{-7}	150	179	161	161	161
10	11	1389	3.72×10^{-3}	19	1570	8	8	8
11	14	178	7.24×10^{-3}	36	59	28	49	28
12	14	270	2.44×10^{-3}	32	58	19	36	19
13	15	690	2.97×10^{-4}	65	447	64	69	60
14	15	14980	4.57×10^{-1}	2850	2703	1849	1849	1849
15	17	101	1.03×10^{-4}	4455	816	508	497	290
16	17	232	1.77×10^{-3}	24	121	10	20	9
17	17	10992	1.68×10^{-2}	11042	11215	9175	9175	9887
18	17	20000	1.17×10^{-2}	18360	18386	12354	12354	12354
19	19	148	1.63×10^{-7}	216535	219126	104	133	146
20	19	2310	1.26×10^{-3}	3840	2464	4324	4324	5551
21	20	80	7.63×10^{-5}	2011	5880	10	11	13
average				12379	12550	1384	1383	1467

表 3　各手法の構造探索における平均計算時間.
Table 3　Average runtimes of the respective methods.

No.	Variables	Sample size	SPP	幅優先探索	幅優先分枝限定法	深さ優先分枝限定法
1	5	24	3.33×10^{-1}	0.0131	0.0191	0.0100
2	5	132	2.29×10^{-1}	0.0170	0.0293	0.0149
3	5	150	3.13×10^{0}	0.0181	0.0285	0.0134
4	5	625	3.33×10^{-1}	0.0134	0.0274	0.0170
5	5	1372	4.29×10^{1}	0.0188	0.0376	0.0268
6	8	3200	2.50×10^{0}	0.1214	0.1903	0.1371
7	9	17898	3.50×10^{1}	0.2785	0.4766	0.3636
8	10	277	8.33×10^{-4}	0.3236	0.2839	0.1719
9	10	683	3.42×10^{-7}	0.3175	0.3481	0.1221
10	11	1389	3.72×10^{-3}	0.8851	0.6323	0.4564
11	14	178	7.24×10^{-3}	16.0481	16.7150	4.6310
12	14	270	2.44×10^{-3}	9.9224	9.9403	3.8845
13	15	690	2.97×10^{-4}	18.3376	17.4498	8.1574
14	15	14980	4.57×10^{-1}	166.2700	171.7628	24.3754
15	17	101	1.03×10^{-4}	459.3530	467.0208	21.2139
16	17	232	1.77×10^{-3}	427.0858	421.7075	34.8079
17	17	10992	1.68×10^{-2}	744.8170	769.2896	145.3891
18	17	20000	1.17×10^{-2}	530.7353	535.9485	99.6420
19	19	148	1.63×10^{-7}	555.6909	537.8737	97.3051
20	19	2310	1.26×10^{-3}	5588.0012	5655.7484	261.5876
21	20	80	7.63×10^{-5}	10044.8238	6946.1992	250.7541
geometric mean				5.5101	6.3813	1.7354

が，ほぼ同等であることが確認できる．ただ，19 番のデータセットでは，深さ優先分枝限定法の NCP は枝刈りを用いない幅優先探索の NCP の約 1.4 倍になっている．このデータセットでは，深さ優先分枝限定法は Bayes factor における変数選択において，目的変数の子変数集合を過大に推定し，最短パスも枝刈りした可能性がある．

表 2 より，深さ優先分枝限定法の平均 NCP は GBN-BDeu, ANB-BDeu の平均 NCP を大幅に下回っていることも確認できた．

次に，枝刈りが計算時間を削減するのかを確認するために，幅優先探索，幅優先分枝限定法，深さ優先分枝限定法の平均実行時間を測定し，結果を表 3 に示した．幅優先分枝限定法，深さ優先分枝限定法の平均実行時間については，Bayes factor による変数選択が必要になるため，その計算時間を含んでいる．

表 3 より，幅優先探索と深さ優先分枝限定法の計算時間を比較すると，4,5,6,7 番を除く全てのデータセットにおいて深さ優先分枝限定法は幅優先探索よりも計算時間を削減した．4,5,6,7 番において，深さ優先分枝限定法の計算時間が幅優先探索の計算時間より大きいのは，これらのデータセットはサンプルサイズが大きく，Bayes factor による変数選択の計算時間が枝刈りにより削減した計算時間を上回っているためであると考えられる．ただ，14, 17, 18 番目のようにサンプルサイズは大きいが，変数数が多いデータセットでは，深さ優先分枝限定法の計算時間の方が幅優先探索の計算時間よりも短い．これは，変数数の増加に伴い枝刈りの回数が増加し，削減できる計算時間が大きくなるためであると考えられる．

更に，表 3 より，14 変数以上のデータセットにおいて，深さ優先分枝限定法の計算時間は幅優先分枝限定法の計算時間を大幅に下回っている．これは，深さ優先分枝限定法の枝刈り回数が幅優先分枝限定法の枝刈り回数に比べて大幅に多いためであると考えられる．

次に，深さ優先分枝限定法が幅優先分枝限定法に比べて枝刈りの回数を増加させたことを確認するために，それぞれの枝刈り回数を測定し，結果を表 4 に示した．

表 4 より，深さ優先分枝限定法の平均枝刈り回数は幅優先分枝限定法の平均枝刈り回数より多いことが確認できる．更に，14 変数以上の全てのデータセットで深さ優先分枝限定法の枝刈り回数は幅優先分枝限定法の枝刈り回数より多いことがわかる．この結果は，幅優先探索から深さ優先探索に変更することで，変数数が増えると枝刈りの回数が増加することを示している．

4.2　大規模データセットを用いた分類精度比較

本節では，菅原ら [9] のアルゴリズムでは学習できない 20 変数以上の BNC 学習を評価する．

前節の実験で用いたデータセットに比べ大規模な 31 ~ 58 変数の 5 種類のベンチマークデータセットを用いて実験を行った．この実験では，ESS の値を 1 に固定し [29], [30]，最大親変数を 4 に制限した．また，構造学習は，6 時間の制限時間を設け，超過する場合は学習を打ち切った．その他の条件は前節と同様であり，

Naive Bayes, GBN-BDeu, ANB-BDeu, 幅優先探索，幅優先分枝限定法と深さ優先分枝限定法の分類精度を比較した．

各手法の分類精度を表 5 に示す．表 5 における “TO” は制限時間内に学習できなかったことを表す．また，表 5 の深さ優先分枝限定法の結果において “*” は，メモリオーバによって打ち切りが発生し，それまでに得た最も NCP の小さい構造を用いたときの分類精度に付与されている．そして，表中では，深さ優先分枝限定法によって学習された構造の目的変数の子変数数を括弧書きで示している．

表 5 より，深さ優先分枝限定法は全てのデータセットで学習構造を得ることができた．一方，GBN-BDeu, ANB-BDeu, 幅優先探索，幅優先分枝限定法は全てのデータセットで学習が打ち切られ，構造を得ることができなかった．また，深さ優先分枝限定法の分類精度は，3 番を除く全てのデータセットで Naive Bayes の分類精度を上回っていることが確認できる．3 番のデータセットでは，SPP が非常に小さいため，深さ優先分枝限定法は目的変数の子変数を過小に学習している．その結果，分類上重要な変数を削除してしまったと考えられる．

表 4　幅優先分枝限定法と深さ優先分枝限定法の構造探索における平均枝刈り回数．
Table 4　The number of the pruning of the breadth-first branch and bound algorithm and the depth-first branch and bound algorithm.

No.	Variables	Sample size	SPP	幅優先分枝限定法	深さ優先分枝限定法
1	5	24	3.33×10^{-1}	8.2	3.9
2	5	132	2.29×10^{-1}	7.8	3.3
3	5	150	3.13×10^{0}	6.6	3.2
4	5	625	3.33×10^{-1}	12.0	4.0
5	5	1372	4.29×10^{1}	2.9	4.0
6	8	3200	2.50×10^{0}	100.0	27.6
7	9	17898	3.50×10^{1}	7.8	58.7
8	10	277	8.33×10^{-4}	130.9	176.1
9	10	683	3.42×10^{-7}	310.3	98.0
10	11	1389	3.72×10^{-3}	152.5	326.2
11	14	178	7.24×10^{-3}	3045.8	5945.4
12	14	270	2.44×10^{-3}	3917.8	6275.3
13	15	690	2.97×10^{-4}	5438.1	22030.4
14	15	14980	4.57×10^{-1}	46.4	42485.8
15	17	101	1.03×10^{-4}	3673.8	29600.5
16	17	232	1.77×10^{-3}	17554.6	22001.9
17	17	10992	1.68×10^{-2}	0.0	189638.4
18	17	20000	1.17×10^{-2}	0.0	154339.2
19	19	148	1.63×10^{-7}	63569.0	189638.4
20	19	2310	1.26×10^{-3}	68.4	154339.2
21	20	80	7.63×10^{-5}	14612.1	386621.3
average				5365.0	44368.1

5. むすび

本論文では，深さ優先分枝限定法による NCP を最小にして真の分類確率に漸近収束するベイジアンネットワーク分類器の構造学習法を提案した．具体的には，(1) Naive Bayes の NCP がその下限値であることを証明し，(2) 深さ優先分枝限定法のための NPC リバースオーダグラフを提案して，(1) で示した下限値を用いた分枝限定法を導入した．

提案手法は以下の利点をもつ．(1) 従来の幅優先探索を用いた手法よりも計算時間を大幅に削減する．(2) 深さ優先分枝限定法を用いることで，実行途中にメモリ等のリソースが不足してもそれまでの最適な構造を得ることが可能である．

ベンチマークネットワークによる実験により，提案手法は従来の幅優先探索を用いたアルゴリズムと同等の精度を保ったまま計算時間を削減できることを示した．また，従来手法では 20 変数程度の構造学習が限界であったが，提案手法は 58 変数の構造学習を実現した．

今後の課題として，より大規模なノード数をもつベンチマークを用いて実験を行い，提案手法の有効性を示す．

謝辞　本研究は JSPS 科研費 JP19H05663 と JP22K19825 の助成を受けた．

表 5　大規模データセットにおける各手法の分類精度
Table 5　Classification accuracies of the respective methods for large datasets.

No.	Dataset	Variables	Sample size	SPP	Naive Bayes	GBN-BDeu	ANB-BDeu	幅優先探索	幅優先分枝限定法	深さ優先分枝限定法
1	wdbc	31	569	2.65×10^{-7}	0.9139	TO	TO	TO	TO	0.9350 (5.8)
2	kr-vs-kp	37	3196	1.55×10^{-8}	0.6640	TO	TO	TO	TO	0.9061* (4.4)
3	biodeg	42	1055	4.94×10^{-14}	0.7877	TO	TO	TO	TO	0.7735 (3.2)
4	Flowmeters_D	44	180	5.12×10^{-12}	0.8333	TO	TO	TO	TO	0.8444 (7.3)
5	spam	58	4601	1.60×10^{-14}	0.8794	TO	TO	TO	TO	0.9109* (16.5)
	average				0.8156	-	-	-	-	0.8740 (7.4)

文　献

[1] P. Domingos and M. Pazzani, “On the optimality of the simple bayesian classifier under zero-one loss,” Machine Learning, vol.29, pp.103–130, 1997.

[2] Q. Wang, G.M. Garrity, J.M. Tiedje, and J.R. Cole, “Naive bayesian classifier for rapid assignment of rrna sequences into the new bacterial taxonomy,” Applied and Environmental Microbiology, vol.73, no.16, pp.5261–5267, 2007.

[3] Y. Choi, G. Farnadi, B. Babaki, and G. Van denBroeck, “Learning fair naive bayes classifiers by discovering and eliminating discrimination patterns,” Proc. AAAI Conf. Artificial Intelligence, vol.34, pp.10077–10084, 2020.

[4] N. Friedman, D. Geiger, and M. Goldszmidt, “Bayesian network classifiers,” Machine Learning, vol.29, no.2-3, pp.131–163, 1997.

[5] S. Sugahara, M. Uto, and M. Ueno, “Exact learning augmented naive Bayes classifier,” Int. Conf. Probabilistic Graphical Models, pp.439–450, 2018.

[6] 菅原聖太，植野真臣，“Augmented naive bayes 制約をもつベイジアンネットワーク分類器の厳密学習，” 信学論（D），vol.J103-D, no.4, pp.301–313, April 2020.

[7] S. Sugahara and M. Ueno, “Exact learning augmented naive bayes classifier,” Entropy, vol.23, no.12, p.1703, 2021.

[8] S. Sugahara, W. Kishida, K. Kato, and M. Ueno, “Recursive autonomy identification-based learning of augmented naive bayes classifiers,” Int. Conf. Probabilistic Graphical ModelsPMLR, pp.265–276, 2022.

[9] 菅原聖太，植野真臣，“分類影響パラメータ数を最小化するベイジアンネットワーク分類器学習，” 信学論（D），vol.J105-D, no.11, pp.679–690, Nov. 2022.

[10] R.G. Cowell, “Efficient maximum likelihood pedigree reconstruction,” Theoretical Population Biology, vol.76, no.4, pp.285–291, Dec. 2009.

[11] M. Koivisto and K. Sood, “Exact Bayesian structure discovery in Bayesian networks,” J. Machine Learning Research, vol.5, pp.549–573, Dec. 2004.

[12] A. Singh and A. Moore, “Finding optimal Bayesian networks by dynamic programming,” Technical Report, Carnegie Mellon University, pp.1–16, June 2005.

[13] T. Silander and P. Myllymaki, “A simple approach for finding the globally optimal Bayesian network structure,” Proc. Uncertainty in Artificial Intelligence (UAI), pp.445–452, AUAI Press, 2006.

[14] B.M. Malone, C. Yuan, E.A. Hansen, and S. Bridges, “Improving the Scalability of Optimal Bayesian Network Learning with External-Memory Frontier Breadth-First Branch and Bound Search,” Proc. Uncertainty in Artificial Intelligence, pp.479–488, 2011.

[15] C. Yuan, B. Malone, and W. Xiaojian, “Learning optimal Bayesian networks using A* search,” Int. Joint Conf. Artificial Intelligence (IJCAI), pp.2186–2191, 2011.

[16] B. Malone and C. Yuan, “A depth-first branch and bound algorithm for learning optimal bayesian networks,” Graph Structures for Knowledge Representation and Reasoning, pp.111–122, Springer, 2014.

[17] J. Suzuki and J. Kawahara, “Branch and bound for regular bayesian network structure learing,” UAI, pp.212–221, 2017.

[18] J. Suzuki, “A theoretical analysis of the BDeu scores in bayesian network structure learning,” Behaviormetrika, vol.44, pp.97–116, 2017.

[19] J. Cussens, “Bayesian network learning with cutting planes,” Proc. Uncertainty in Artificial Intelligence (UAI), pp.153–160, AUAI Press, 2011.

[20] W. Buntine, “Theory refinement on Bayesian networks,” Proc. Uncertainty in Artificial Intelligence (UAI), pp.52–60, 1991.

[21] J. Pearl, Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference, Morgan-Kaufmann, 1988.

[22] D. Koller and N. Friedman, Probabilistic Graphical Models: Principles and Techniques, MIT Press, 2009.

[23] D. Heckerman, D. Geiger, and D.M. Chickering, “Learning Bayesian networks: The combination of knowledge and statistical data,” Machine Learning, vol.20, no.3, pp.197–243, 1995.

[24] N.J. Nilsson, Principles of artificial intelligence, Springer Science & Business Media, 1982.

[25] 名取和樹，宇都雅輝，植野真臣，“Bayes 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，” 信学論（D），vol.J101-D, no.5, pp.754–768, May 2018.

[26] K. Natori, M. Uto, and M. Ueno, “Consistent learning Bayesian networks with thousands of variables,” Advanced Methodologies for Bayesian Networks (Proc. of Machine Learning Research), vol.73, pp.57–68, 2017.

[27] M. Lichman, “UCI machine learning repository,” 2013. http://archive.ics.uci.edu/ml

[28] C.X. Ling and H. Zhang, “The representational power of discrete Bayesian networks,” J. Machine Learning Research, vol.3, pp.709–721, 2003.

[29] M. Ueno, “Learning networks determined by the ratio of prior and data,” Proc. 26th Conf. Uncertainty in Artificial Intelligence, p.598–605, AUAI Press, Arlington, Virginia, USA, 2010.

[30] M. Ueno, “Robust learning Bayesian networks for prior belief,” Proc. Uncertainty in Artificial Intelligence (UAI), pp.689–707, AUAI Press, 2011.

付　録

1. 定理 3.1 の証明

［証明 1］（定理 3.1） 任意の変数集合 $\mathbf{V}$ を考える．$\mathbf{V}$ に対する I-map の中で NCP を最小にする BNC, $G^*(\mathbf{V})$ について，$G^*(\mathbf{V})$ における目的変数の子変数集合 $\mathbf{V}_c$ を説明変数とする Naive Bayes を $G^{NB}(\mathbf{V}_c)$ とする．このとき，Naive Bayes の説明変数の親変数は目的変数のみであるため，$G^{NB}(\mathbf{V}_c)$ における $X_i \in \mathbf{V}_c$ についての NCP_i は $NCP_i(\{X_0\})$ で表せる．したがって，$NCP(G^{NB}(\mathbf{V}_c))$ は以下のように表せる．

$$NCP(G^{NB}(\mathbf{V}_c)) = \sum_{X_i \in \mathbf{V}_c} NCP_i(\{X_0\}) + r_0 - 1. \tag{A·1}$$

また，$X_i \in \mathbf{V}_c$ としたとき，$G^{NB}(\mathbf{V}_c)$ における X_i に

ついての NCP$_i$, $NCP_i(\{X_0\})$ と，$G^*(\mathbf{V})$ における X_i についての NCP$_i$, $NCP_i(\mathbf{Pa}(X_i, G^*(\mathbf{V})))$ に関して，

$$NCP_i(\{X_0\}) = (r_i - 1)r_0, \tag{A·2}$$

$$NCP_i(\mathbf{Pa}(X_i, G^*(\mathbf{V}))) = (r_i - 1)q_i^* \tag{A·3}$$

が成り立つ．ここで，q_i^* は $\mathbf{Pa}(X_i, G^*(\mathbf{V}))$ のとり得るパターン数を表す．また，$X_i \in \mathbf{V}_c$ より，$X_0 \in \mathbf{Pa}(X_i, G^*(\mathbf{V}))$ となるため，$r_0 \leq q_i^*$ となる．したがって，以下が成り立つ．

$$NCP_i(\{X_0\}) \leq NCP_i(\mathbf{Pa}(X_i, G^*(\mathbf{V}))) \tag{A·4}$$

以上より，

$$\begin{aligned} &NCP(G^{NB}(\mathbf{V}_c)) \\ &= \sum_{X_i \in \mathbf{V}_c} NCP_i(\{X_0\}) + r_0 - 1 \\ &\leq \sum_{X_i \in \mathbf{V}_c} NCP_i(\mathbf{Pa}(X_i, G^*(\mathbf{V}))) + r_0 - 1 \\ &= NCP(G^*(\mathbf{V})). \end{aligned} \tag{A·5}$$

したがって，任意の変数集合 $\mathbf{V}$ について，$\mathbf{V}$ に対する I-map の中で NCP を最小にする BNC, $G^*(\mathbf{V})$ における目的変数の子変数集合 $\mathbf{V}_c$ を説明変数とする Naive Bayes を $G^{NB}(\mathbf{V}_c)$ とすると，以下が成り立つ．

$$NCP(G^{NB}(\mathbf{V}_c)) \leq NCP(G^*(\mathbf{V})) \tag{A·6}$$

□

2. 定理 3.2 の証明

[証明 2]（定理 3.2） NPC リバースオーダグラフの任意のノード $\mathbf{U}$ と $\mathbf{U}$ からエッジが引かれている任意のノード $\mathbf{R}$ について，$X_j \in \mathbf{U}\backslash\mathbf{R}$ とし，$c(\mathbf{U}, \mathbf{R})$ をノード $\mathbf{U}$ からノード $\mathbf{R}$ へのエッジがもつコストであるとする．ここで，$X_j \notin \mathbf{V}_c$ のとき，

$$\begin{aligned} h^*(\mathbf{U}) &= \sum_{X_i \in (\mathbf{U} \cap \mathbf{V}_c)} NCP_i(X_0) \\ &= \sum_{X_i \in (\mathbf{R} \cap \mathbf{V}_c)} NCP_i(X_0) \\ &\leq \sum_{X_i \in (\mathbf{R} \cap \mathbf{V}_c)} NCP_i(X_0) + NCP_j(g_j^*(\mathbf{U}\backslash\{X_j\})) \\ &= h^*(\mathbf{R}) + c(\mathbf{U}, \mathbf{R}) \end{aligned} \tag{A·7}$$

が成り立つ．また，$X_j \in \mathbf{V}_c$ のとき，$X_0 \in g_j^*(\mathbf{U}\backslash\{X_j\})$ が成り立つ．したがって，次式が成り立つ．

$$\begin{aligned} h^*(\mathbf{U}) &= \sum_{X_i \in (\mathbf{U} \cap \mathbf{V}_c)} NCP_i(X_0) \\ &= \sum_{X_i \in (\mathbf{U} \cap \mathbf{V}_c)\backslash\{X_j\}} NCP_i(X_0) + NCP_j(X_0) \\ &= \sum_{X_i \in (\mathbf{R} \cap \mathbf{V}_c)} NCP_i(X_0) + NCP_j(X_0) \\ &\leq \sum_{X_i \in (\mathbf{R} \cap \mathbf{V}_c)} NCP_i(X_0) + NCP_j(g_j^*(\mathbf{U}\backslash\{X_j\})) \\ &= h^*(\mathbf{R}) + c(\mathbf{U}, \mathbf{R}). \end{aligned} \tag{A·8}$$

以上から，

$$h^*(\mathbf{U}) \leq h^*(\mathbf{R}) + c(\mathbf{U}, \mathbf{R}). \tag{A·9}$$

したがって，h^* は無矛盾である． □

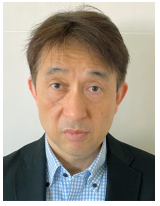
（2023 年 6 月 8 日受付，11 月 10 日早期公開）

加藤　弘也　（学生員）

2023 電気通信大学情報理工学域卒．同年，電気通信大学大学院情報理工学研究科情報・ネットワーク工学専攻博士前期課程入学，現在に至る．

菅原　聖太

2023 電気通信大学大学院情報理工学研究科博士後期課程了．博士（工学）．同年，電気通信大学助教，現在に至る．



植野　真臣　（正員）

1992 神戸大学大学院教育学研究科了，1994 東京工業大学大学院総合理工学研究科了．博士（工学）．東京工業大学，千葉大学，長岡技術科学大学を経て 2006 より電気通信大学助教授，2013 より教授，現在に至る．