電気通信大学情報理工学域
I 類 (情報系) 情報数理工学プログラム卒業論文

深さ優先分枝限定法による
ベイジアンネットワーク分類器学習

2022 年 1 月 31 日

情報数理工学プログラム

学籍番号 1910164

加藤 弘也

指導教員 植野 真臣

令和４年度 情報数理工学プログラム卒業論文概要

平成 31 年度 入学	学籍番号 1910164
指導教員 植野 真臣	氏名 加藤 弘也
題目	深さ優先分枝限定法によるベイジアンネットワーク分類器学習

概要

ベイジアンネットワーク分類器は離散変数を扱う高精度の分類器として知られている.

本論では, 深さ優先分枝限定法による分類に影響する目的変数パラメータ数 (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手法を提案する.

深さ優先分枝限定法は探索の途中であってもその時点までの最適な構造の取得を可能にし, 枝刈りによる効率的な構造の探索を実現する. ただ, 枝刈りで用いるヒューリスティック関数については, これまでに提案されているヒューリスティック関数を用いることができない.

そこで本論では, (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 NCP 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.

提案手法には以下の利点がある. (1) 従来の動的計画法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. 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, 計算時間を削減できることを示す. また, 探索の途中であって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従来手法との比較実験により示す.

1 まえがき

ベイジアンネットワーク (Bayesian network: BN) は, 離散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) で表す確率的グラフィカルモデルである. BN における一つのノードを目的変数とし, その他のノードを説明変数としたベイジアンネットワーク分類器 (Bayesian Network Classifier: BNC) は, 離散変数を扱う分類器として知られている [1].

BN の構造は一般にデータから推定する必要があり, この問題を BN の構造学習と呼ぶ. 構造学習では, すべての構造の候補から最適な学習スコアを持つ構造を探索する厳密解探索アプローチが従来から行われてきた. 本論では候補構造に制約を課さずに学習した BNC を General Bayesian Network (GBN) と呼ぶ. 一般に厳密解探索アプローチでは, グラフが表現する全ての条件付き独立性が真の条件付き独立性に一致する構造 (independence map: I-map) のうち, 全パラメータ数が最小の構造への漸近一致性を有する, 構造の周辺尤度 (Marginal Likelihood: ML) を学習スコアとして用いる. しかし, このアプローチは構造の候補数がノード数に対し指数的に増加する NP 困難問題 [2] であり, 膨大な計算時間およびメモリを必要とする. 効率的に厳密解, すなわち最適な学習スコアをもつ構造を探索するために, 動的計画法 [3, 4, 5, 6, 7], A^* 探索 [8], 幅優先分枝限定法 [9], 深さ優先分枝限定法 [10], 整数計画法 [11] などの従来の人工知能アプローチによる構造学習法が提案されている.

BNC に関して, 菅原ら [12, 13, 14] は, サンプルサイズが小さい場合, ML による GBN の厳密学習の分類精度が低くなり, 最も単純な構造をもつ Naive Bayes よりも低い分類精度を示す場合があることを報告している. 特に, 目的変数の親変数が多く子変数が少ないような構造を学習する場合に分類精度が低くなっていた. その理由は, 目的変数の親変数数が多いと, パラメータ数が指数的に増加するため, 一つのパラメータ学習のためのサンプルサイズが小さくなり, 推定精度が悪くなってしまうからである.

この問題を緩和するため, 菅原ら [12, 13, 14, 15] は, 目的変数がすべての説明変数を子に持ち, 目的変数自身は親を持たない Augmented Naive Bayes (ANB) 構造を制約とした BNC の厳密学習手法を提案した. 彼らの手法は, I-map のうち全パラメータ数が最小の ANB に漸近的に一致する構造を学習できる. また, 彼らの手法で学習した ANB と ML で厳密学習した GBN は漸近的に等しい分類確率をもつことが示されている. しかし, 真のモデルが BN に従っていない場合, ML 最大化は分類に影響する目的変数パラメータ数 (the number of the class variable parameters: NCP) を最小にする保証がない.

この問題を解決するために, 菅原ら [16] は真のモデルが BN に従っているか否かに関わらず, NCP を最小とする I-map に学習構造が漸近的に一致する手法を提案した. 彼らの手法では探索空間を目的変数が親変数を持たないような構造集合 (NPCDAG: no parents class DAG) としてい

る．NPCDAG のように目的変数に親変数がない構造は, GBN より高い分類精度を持つ傾向があることが報告されている [12, 13, 14]．また, BN が表現可能な全ての分類確率は, NPCDAG のみで表現できる [17]．さらに, 菅原らはある変数順序のもとで ML を最大化する構造が, その順序に従う構造の中で NCP 最小の I-map に漸近的に一致することを証明した [16]．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順序について, ML 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うち NCP を最小にする構造を探索する．結果として得られる構造は, 真のモデルが BN に従っていない場合でも, NCP を最小にして真の分類確率に漸近収束する NPCDAG となる．ただ, 彼らは, 第一ステップと第二ステップの探索において動的計画法を用いているため, 変数数の増加に伴い指数的に計算時間が増加してしまう．また, 動的計画法を用いる場合, 探索終了まで構造を得ることができない．したがって, 時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．

この問題を緩和するために, 本論では, 深さ優先分枝限定法による NCP を最小とする I-map に学習構造が漸近的に一致する手法を提案する．提案手法は, 菅原ら [16] の手法の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, 深さ優先分枝限定法を用いた効率的な探索を実現する．深さ優先分枝限定法は, 探索の途中であってもその時点までの最適な構造の取得を可能にし, 枝刈りを行うことで動的計画法よりも効率的な探索を実現する．ただ, 枝刈りで用いる, 最短パス探索におけるあるノードから終点までのコストの下限値については, これまでに提案されているコストの下限値を用いることができない．そこで, 本論では, NPCDAG の中で NCP を最小にする I-map について, その構造の目的変数の子変数を説明変数とする Naive Bayes の NCP がコスト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のある変数を相互情報量による変数選択により求め, それらの変数で構成する Naive Bayes の NCP を探索におけるコストの下限値とする．この手法の利点として, 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, 従来手法である動的計画法より計算時間の削減が可能になる．
2. 従来手法と異なり, 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．

本論では, 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, 計算時間を削減できることを示す．また, 探索の途中であって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従来手法との比較実験により示す．

2　ベイジアンネットワーク

ベイジアンネットワーク (Bayesian network: BN) は, 確率変数をノードとし, ノード間の依存関係を表現した非循環有向グラフと, 各ノードの親ノード集合を所与とした条件付き確率で表現される確率的グラフィカルモデルである. 今, $\mathbf{V} = \{X_0, \ldots, X_i, \ldots, X_n\}$ を離散確率変数集合とする. 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, $X_i = k$ と書く. また, 構造 G における変数 X の親変数集合を $\mathbf{Pa}(X, G)$ と表すと, 同時確率分布 $P(X_0, \ldots, X_n \mid G, \Theta)$ は各変数の条件付き確率パラメータの積に分解して以下のように表現できる.

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

G の各変数を要素とするベクトル σ に対し σ の i 番目の要素を X_{σ^i} で表すと, $\forall i, \mathbf{Pa}(X_{\sigma^i}, G) \subseteq \bigcup_{j=1}^{i-1}\{X_{\sigma^j}\}$ が成り立つとき, σ を G の変数順序という. さらに, θ_{ijk} を $\mathbf{Pa}(X_i, G)$ が j 番目のパターンをとるとき ($\mathbf{Pa}(X_i, G) = j$ と書く) , $X_i = k$ となる条件付き確率 $P(X_i = k \mid \mathbf{Pa}(X_i, G) = j)$ を示すパラメータとする. また, 条件付き確率パラメータ集合 $\Theta_{ij}, \Theta_i, \Theta$ をそれぞれ $\Theta_{ij} = \bigcup_{k=1}^{r_i}\{\theta_{ijk}\}, \Theta_i = \bigcup_{j=1}^{q_i}\{\Theta_{ij}\}, \Theta = \bigcup_{i=0}^{n}\{\Theta_i\}$ で定義する. ここで, $q_i = \prod_{l:X_l \in \mathbf{Pa}(X_i, G)} r_l$ である.

BN の構造は確率分布の条件付き独立性を d 分離によって表す. 構造 $G = (\mathbf{V}, \mathbf{E})$ 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.

定義 2.1. *G において $X, Y \in \boldsymbol{V}, \boldsymbol{Z} \subseteq \boldsymbol{V}\backslash\{X, Y\}$ としたとき, $\boldsymbol{Z}$ 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 $\boldsymbol{Z}$ 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 $Z \in \boldsymbol{Z}$ 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 $\boldsymbol{Z}$ に属さない.*

この関係を $I_G(X, Y \mid \mathbf{Z})$ で表す. そして, 真の同時確率分布において X と Y が $\mathbf{Z}$ を所与として条件付き独立であることを $I_M(X, Y \mid \mathbf{Z})$ で表す. また, I-map を以下で定義する.

定義 2.2. *BN構造 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.*

$$\forall X, Y \in \boldsymbol{V}, \forall \boldsymbol{Z} \subseteq \boldsymbol{V}\backslash\{X, Y\}, \\ I_G(X, Y \mid \boldsymbol{Z}) \Rightarrow I_M(X, Y \mid \boldsymbol{Z}).$$

I-map 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.

今, サンプルが N 個あり, 各サンプルは独立で同一な分布に従うとする. t 番目のサンプルを $\mathbf{d}^t = \langle x_0^t, \ldots, x_n^t \rangle$ と表し, 学習データを $D = \langle \mathbf{d}^1, \ldots, \mathbf{d}^N \rangle$ と表す. D が得られたときの BN$\langle G, \Theta \rangle$ のパラメータ推定量として, θ_{ijk} の期待値である Expected A Posteriori (EAP) が最も良く用いられる. BN の構造 G に対し, 式 (1) のようにパラメータの事前分布にディリクレ分布を仮定すると, 式 (2) の事後分布 $p(\Theta_{ij} \mid D, G)$ が得られる. そして, その事後分布から式 (3) のように EAP を求めることができる.

$$p(\Theta_{ij} \mid G) = \frac{\Gamma(\sum_{k=1}^{r_i} \alpha_{ijk})}{\prod_{k=1}^{r_i} \Gamma(\alpha_{ijk})} \prod_{k=1}^{r_i} \theta_{ijk}^{\alpha_{ijk}-1}. \tag{1}$$

$$p(\Theta_{ij} \mid D, G) = \frac{\Gamma(\sum_{k=1}^{r_i} (\alpha_{ijk} + N_{ijk}))}{\prod_{k=1}^{r_i} \Gamma(\alpha_{ijk} + N_{ijk})} \prod_{k=1}^{r_i} \theta_{ijk}^{\alpha_{ijk}+N_{ijk}-1}. \tag{2}$$

$$\hat{\theta}_{ijk} = \int \theta_{ijk} \cdot p(\Theta_{ij} \mid D, G) d\Theta_{ij} = \frac{\alpha_{ijk} + N_{ijk}}{\alpha_{ij} + N_{ij}}. \tag{3}$$

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.

BN のパラメータを推定するためには, 最適な構造をデータから推定する必要がある. この問題を BN の構造学習と呼ぶ. 構造学習では, すべての構造の候補から最適な学習スコアを持つ構造を探索する厳密解探索アプローチが従来から行われてきた. 一般に学習スコアとして周辺尤度 $P(D \mid G)$ (Marginal Likelihood: ML) が用いられ, 厳密解探索アプローチでは, ML を最大にする構造を最適な構造とする. ML を最大にする構造は漸近的に全パラメータ数が最小の I-map に一致する. この性質をパラメータ数最小 I-map への漸近一致性と呼ぶ. パラメータの事前分布をディリクレ分布と仮定すると, ML は次のように閉形式で表すことができる.

$$P(D \mid G) = \prod_{i=0}^{n} \prod_{j=1}^{q_i} \frac{\Gamma(\alpha_{ij})}{\Gamma(\alpha_{ij} + N_{ij})} \prod_{k=1}^{r_i} \frac{\Gamma(\alpha_{ijk} + N_{ijk})}{\Gamma(\alpha_{ijk})}. \tag{4}$$

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 [18, 19]. 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である.

一方, $\alpha_{ijk} = 1$(事前分布が一様分布) とした学習結果は, Bayesian Information Criterion(BIC)[20], Minimum Description Length(MDL)[21] の結果に漸近的に一致することが知られている.[22].

logBDeu, BIC, MDL は, スコアとして次の性質を満たす.

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

ここで, $Score_i(\mathbf{Pa}(X_i, G))$ は変数 X_i とその親変数集合 $\mathbf{Pa}(X_i, G)$ のみに依存する関数であり, ローカルスコアと呼ぶ. 例えば logBDeu の変数 X_i と親変数集合 $\mathbf{Pa}(X_i, G)$ についてのローカルスコア $Score_i(\mathbf{Pa}(X_i, G))$ は以下のように表せる.

$$\begin{aligned} &Score_i(\mathbf{Pa}(X_i, G)) \\ &= \sum_{j=1}^{q_i} \left(\log \frac{\Gamma(\alpha_{ij})}{\Gamma(\alpha_{ij} + N_{ij})} + \sum_{k=1}^{r_i} \log \frac{\Gamma(\alpha_{ijk} + N_{ijk})}{\Gamma(\alpha_{ijk})} \right). \end{aligned} \tag{6}$$

また, 式 (5) を満たすスコアを分解可能であると言い, 分解可能なスコアを用いた効率的な構造探索手法が提案されている.[6, 8, 9, 11, 23, 24]

3 ベイジアンネットワーク分類器

3.1 ベイジアンネットワーク分類器による分類

ベイジアンネットワーク分類器 (Bayesian Network Classifier: BNC) は, BN における一つのノードを目的変数とし, その他のノードを説明変数とする離散変数を扱う分類器である [1]. 今, $X_1, \ldots, X_n$ を説明変数とし, X_0 を目的変数とした BNC を考える. 説明変数のデータ $\mathbf{x} = \langle x_1, \ldots, x_n \rangle$ が与えられたとき, 目的変数の推定値 $\hat{c}$ は以下のように得られる.

$$\begin{aligned} \hat{c} &= \arg\max_{c \in \{1,\ldots,r_0\}} P(c \mid \mathbf{x}, G, \Theta) \\ &= \arg\max_{c \in \{1,\ldots,r_0\}} \frac{P(c, \mathbf{x} \mid G, \Theta)}{P(\mathbf{x} \mid G, \Theta)} \\ &= \arg\max_{c \in \{1,\ldots,r_0\}} P(c, \mathbf{x} \mid G, \Theta) \\ &= \arg\max_{c \in \{1,\ldots,r_0\}} \prod_{i=0}^{n} \prod_{j=1}^{q_i} \prod_{k=1}^{r_i} (\theta_{ijk})^{I_{ijk}} \\ &= \arg\max_{c \in \{1,\ldots,r_0\}} \left[\prod_{j=1}^{q_0} \prod_{k=1}^{r_0} (\theta_{0jk})^{I_{0jk}} \times \prod_{i:X_i \in \mathbf{Ch}} \prod_{j=1}^{q_i} \prod_{k=1}^{r_i} (\theta_{ijk})^{I_{ijk}} \right]. \end{aligned} \tag{7}$$

ここで, I_{ijk} は変数列 $\langle c, x_1, \ldots, x_n \rangle$ において $X_i = k$ かつ $\mathbf{Pa}(X_i, G) = j$ のときに 1 をとり, それ以外のときは 0 をとる変数である. また, $\mathbf{Ch}$ は目的変数の子変数の集合である. 式 (7) の最右

辺から, 分類に影響する説明変数は, 目的変数の子変数と親変数, および目的変数と子を共有する変数のみであることが分かる. これらの変数集合を目的変数のマルコフブランケットと呼ぶ.

3.2 ベイジアンネットワーク分類器への制約

一般に, BN の構造学習では, とりうる全ての構造を候補構造とする. そのような候補構造に対して学習スコアを最適化して学習される BNC は General Bayesian Network (GBN) と呼ばれる. つまり, 分類器として用いられる, 制約のない一般的な BN を GBN と呼ぶ. ノード数の多いネットワークでは GBN の学習に膨大な時間がかかるため, 候補構造に制約を入れて学習することが多い. 例えば, GBN の下位構造として, 全説明変数が目的変数のみを親に持つと仮定する Naive Bayes[25] などが知られている. Naive Bayes の構造は一意に定まるため, 構造学習の必要はない. また, Naive Bayes を一般化した, より表現力の高い制約として, 目的変数が全説明変数を子にもつという制約のみで説明変数間の関係には制約をおかない Augmented Naive Bayes(ANB)[1] が知られている.

3.3 ベイジアンネットワーク分類器の学習

菅原ら [12, 13, 14] は BDeu による GBN の厳密学習において, サンプルサイズが小さくなると分類精度が低下し, 最も単純な構造をもつ Naive Bayes よりも低い分類精度を示す場合があることを報告している. 特に, 目的変数の親変数数が多く子変数が少ないような構造を学習する場合に分類精度が低くなっていた. その理由は, 目的変数の親変数が多いと, パラメータ数が指数的に増加するため, 一つのパラメータ学習のためのサンプルサイズが小さくなり, 推定精度が悪くなってしまうからである.

この問題を緩和するため, 菅原ら [12, 13, 14, 15] は, ANB 構造を制約とした BNC の厳密学習手法を提案した. 彼らの手法は I-map の中で全パラメータが最小となる ANB を漸近的に学習できる. さらに, 菅原ら [14] は以下の仮定 3.1 と 3.2 の下で, 漸近的に ANB の厳密学習により得られる構造が真の構造と分類等価であることを示した.

定義 3.1. 二つの構造 G, G' が, 全説明変数に対する任意のインスタンス $\boldsymbol{d} = (x_1, \ldots, x_n)$ について $P(X_0 \mid \boldsymbol{d}, G) = P(X_0 \mid \boldsymbol{d}, G')$ となるとき, G と G' は分類等価という.

仮定 3.1. 以下を満たす構造 G^* が存在する.

$$\forall X, Y \in \boldsymbol{V}, \forall \boldsymbol{Z} \subseteq \boldsymbol{V} \backslash \{X, Y\}, \\ I_{G^*}(X, Y \mid \boldsymbol{Z}) \Leftrightarrow I_M(X, Y \mid \boldsymbol{Z}).$$

仮定 3.2. $\forall X \in \boldsymbol{V}$ について, X と X_0 は G^* において隣接する.

定理 3.1. 仮定 *3.1* と *3.2* の下で, *BDeu* を用いて厳密学習された *ANB* は G^* と漸近的に分類等価である.

4 分類に影響する目的変数パラメータ数最小化による BNC 学習

菅原ら [12, 13, 14] の提案手法によって学習される I-map は, 仮定 3.1 を満たさない場合, すなわち真のモデルが BN に従っていない場合, 以下で定義される分類に影響する目的変数パラメータ数 (the number of the class variable parameters: NCP) を最小化する保証がない.

$$NCP(G) = \sum_{i=0}^{n} NCP_i(\mathbf{Pa}(X_i, G)). \tag{8}$$

ここで,

$$NCP_i(\mathbf{Pa}(X_i, G)) = \begin{cases} (r_i - 1)q_i & (i = 0 \vee X_0 \in \mathbf{Pa}(X_i, G)) \\ 0 & (otherwise) \end{cases}$$

である.

この問題を解決するため, 菅原らは, 真のモデルが BN に従っていない場合でも, 学習構造が NCP 最小の I-map に漸近的に一致する手法を提案した [16]. 彼らの手法では目的変数が親変数を持たないような構造集合 (NPCDAG: no parents class DAG) を探索空間としている. BN が表現可能な全ての分類確率は, NPCDAG のみで表現できることが証明されている [17]. また, NPCDAG のように目的変数に親変数がない構造は, GBN よりも高い分類精度をもつ傾向があることが報告されている [12, 13, 14, 15].

また, 菅原ら [16] 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, 次の定理を証明している.

定理 4.1. $\forall \sigma \in \boldsymbol{\sigma}(\boldsymbol{V})$ 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, 目的変数を先頭とする全ての変数順序について, BDeu を最大化する構造を求める. ここで, 各変数順序を所与として得られた構造は, その変数順序における NCP 最小の I-map に漸近的に一致する. したがって, この中で NCP を最小とする構造が求めたい NPCDAG となる. 第二ステップでは, 第一ステップで得られた構造のうち NCP を最小にする構造を探索する.

菅原ら [16] の学習アルゴリズムの説明のため, 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, 変数順序 σ に従う構造の中で BDeu

を最大にする構造を G^*_σ で表す. 変数 X_i について, 候補親変数集合を $\mathbf{Z} \subseteq \mathbf{V}\backslash\{X_i\}$ としたときの最適親変数集合 $g^*_i(\mathbf{Z})$ は以下で定義される.

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

また, 構造 G を各変数の親変数集合からなるベクトル $(\mathbf{Pa}^G_{X_0}, \mathbf{Pa}^G_{X_1}, \ldots, \mathbf{Pa}^G_{X_n})$ で表す. 変数集合 $\mathbf{Z}$ に対する変数順序のうち目的変数が先頭になるもの, つまり $X_{\sigma^1} = X_0$ となるものの集合を $\boldsymbol{\sigma_0}(\mathbf{Z})$ で表す. 変数集合 $\mathbf{Z}$ で構成され, $\boldsymbol{\sigma_0}(\mathbf{Z})$ に属する変数順序に従う全ての構造の中で BDeu を最大化する構造を $G^*(\mathbf{Z})$ で表す. ある変数が子変数を持たないとき, 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 $G^*_{\sigma_0}$ を求める. $G^*_{\sigma_0} = (\emptyset, g^*_1(\mathbf{Pre}^{\sigma_0}_{X_1}), \ldots, g^*_n(\mathbf{Pre}^{\sigma_0}_{X_n}))$ と表せるため, $G^*_{\sigma_0}$ を得るには $\forall\sigma_0 \in\boldsymbol{\sigma_0}(\mathbf{V}), \forall i \in \{1, \ldots, n\}$ について $g^*_i(\mathbf{Pre}^{\sigma_0}_{X_i})$ を求めればよい. しかし, 異なる二つの変数順序 σ と σ' について, $\mathbf{Pre}^{\sigma}_X = \mathbf{Pre}^{\sigma'}_X$ のとき, σ の下での X の最適親変数集合と σ' の下での X の最適親変数集合は等しいため, 重複して探索を行ってしまう. これを避けるためには, $\forall i \in \{1, \ldots, n\}, \forall\mathbf{Z} \subseteq \mathbf{V}\backslash\{X_i\}$ に対する $g^*_i(\mathbf{Z})$ の計算のみを行えば良い. したがって, 彼らはこれらの最適親変数集合の探索に Silander ら [6] によって提案された動的計画法を用いる.

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 を最小にする構造, すなわち $G^*(\mathbf{V})$ を探索する. 菅原らは, この探索も Silander ら [6] の動的計画法を用いている. $G^*(\mathbf{Z})$ において, Sink が X_i のとき, その親変数集合は $g^*_i(\mathbf{Z}\backslash\{X_i\})$ である. 構造 $G^*(\mathbf{Z})$ から変数 X_i と $g^*_i(\mathbf{Z}\backslash\{X_i\})$ から X_i に引かれるエッジを取り除いた構造は $G^*(\mathbf{Z}\backslash\{X_i\})$ となる. したがって, $G^*(\mathbf{Z})$ における Sink $S^*(\mathbf{Z})$ は以下で表せる.

$$S^*(\mathbf{Z}) = \arg\min_{X_i\in\mathbf{Z}} \{NCP_i(g^*_i(\mathbf{Z}\backslash\{X_i\})) + NCP(G^*(\mathbf{Z}\backslash\{X_i\}))\}. \tag{10}$$

$G^*(\mathbf{V})$ に対してこの分解を再帰的に行うことで, Sink とその親変数集合を対とする $n+1$ 組 $(X_0, \emptyset), (X_1, g^*_1), \ldots, (X_n, g^*_n)$ を求めることができる. したがって, $G^*(\mathbf{V}) = (\emptyset, g^*_1, \ldots, g^*_n)$ が得られる.

彼らの手法では目的変数の親変数集合を探索しないため, 彼らの手法の計算時間は GBN の厳密学習の計算時間よりも短い. しかし, 彼らの手法は第一ステップと第二ステップの探索において動的計画法を用いているため, 変数数の増加に伴い指数的に計算時間が増加してしまう. また, 動的計画法を用いる場合, 探索終了まで構造を得ることができない. したがって, 時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.

5　深さ優先分枝限定法による BNC 学習法の提案

これまでに動的計画法より計算時間が改善された厳密解探索アプローチのアルゴリズムが提案されている．本論では, 深さ優先分枝限定法による NCP を最小とする I-map に学習構造が漸近的に一致する手法を提案する．まず, 菅原ら [16] の手法の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, 深さ優先分枝限定法を用いた効率的な探索を提案する．深さ優先分枝限定法は, 探索の途中であってもその時点までの最適な構造の取得を可能にし, 枝刈りを行うことで動的計画法よりも効率的な探索を実現する．ただ, 枝刈りで必要となる, 最短パス探索におけるあるノードから終点までのコストの下限値については, これまでに提案されているコストの下限値を用いることができない．そこで, 本論では, NPCDAG の中で NCP を最小にする I-map について, その構造の目的変数の子変数を説明変数とする Naive Bayes の NCP がコスト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のある変数を相互情報量による変数選択により求め, それらの変数で構成する Naive Bayes の NCP を探索におけるコストの下限値とする．この提案手法は, 従来の動的計画法を用いた手法と同等の精度を保ったまま計算時間を削減できる．

5.1　NPC リバースオーダグラフ

Yuan ら [8] は, ベイジアンネットワークの構造学習をオーダグラフと呼ばれるグラフを用いた最短パス探索問題として定式化した．4 変数に対するオーダグラフは図 1 のように可視化され, 各ノードには変数集合 $\mathbf{V}$ の部分集合 $\mathbf{U} \subseteq \mathbf{V}$ に対応したラベルが付与されている．オーダグラフは, ラベルの変数数が等しいノードを同じ層にまとめた格子構造を持つ．オーダグラフにおける最上層のノードのラベルは空集合であり, このノードを探索の始点とする．また, 最下層のノードは変数集合 $\mathbf{V}$ をラベルに持つノードであり, このノードを探索の終点とする．オーダグラフの始点から終点までのパスは, 変数順序に 1 対 1 で対応する．例えば, 図 1 においてパス $\{\}, \{X_0\}, \{X_0, X_1\}, \{X_0, X_1, X_2\}, \{X_0, X_1, X_2, X_3\}$ は変数順序 X_0, X_1, X_2, X_3 に対応する．

Malone ら [10] は, 深さ優先分枝限定法における最適親変数集合を効率的に探索するために, リバースオーダグラフを提案している． 4 変数に対するリバースオーダグラフは図 2 のように可視化される． リバースオーダグラフでは, 最上層のノードのラベルが変数集合 $\mathbf{V}$ となり, 最下層のノードのラベルが空集合となる． 例えば, 図 2 においてパス $\{X_0, X_1, X_2, X_3\}, \{X_0, X_1, X_2\}, \{X_0, X_1\}, \{X_0\}, \{\}$ は変数順序 X_0, X_1, X_2, X_3 に対応する．しかし, これまでに提案されているオーダグラフやリバースオーダグラフは, 候補構造に制約のない一般的な BN を学習するために提案されたものであり, 目的変数が親変数を持たないような構造の学習には適さない．そこで本論では, 目的変数が親変数を持たないような構造を深さ優先分枝限

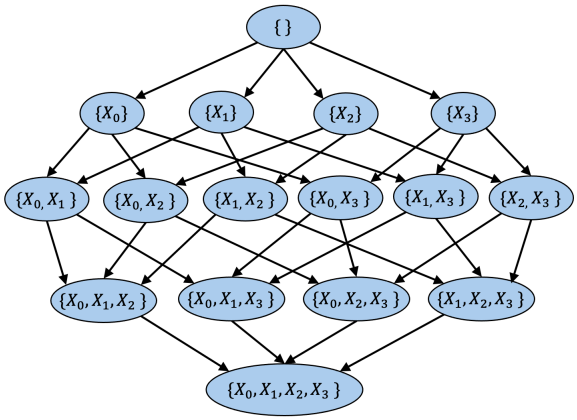

図 1: オーダグラフ

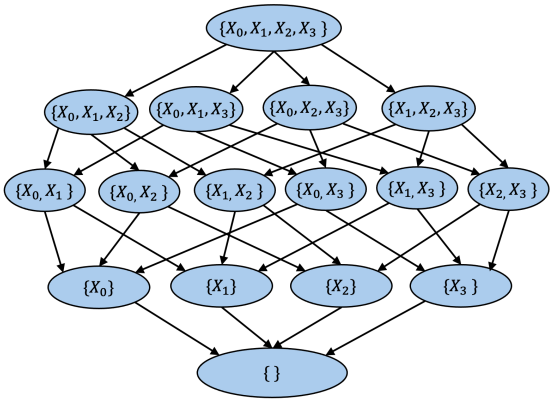

図 2: リバースオーダグラフ

定法で学習するために, 図 3 で表される NPC リバースオーダグラフを提案する. NPC リバースオーダグラフでは, 最上層のノードのラベルが変数集合 $\mathbf{V}$ であり, 最下層のノードのラベルは変数集合 $\{X_0\}$ である. NPC リバースオーダグラフは $\mathbf{V}\backslash\{X_0\}$ からなるリバースオーダグラフのすべてのノードのラベルに X_0 を加えたグラフとなる. このように定めることで, 目的変数が先頭となる変数順序のみを探索可能である. つまり, 目的変数が親変数を持たないような構造の学習が可能になる. 以下では, NPC リバースグラフにおいてラベル $\mathbf{U}$ を持つノードをノード $\mathbf{U}$ と呼称する.

NPC リバースオーダグラフにおけるノード $\mathbf{U}$ からノード $\mathbf{U}\backslash\{X_i\}$ へのエッジは $NCP_i(g_i^*(\mathbf{U}\backslash\{X_i\}))$ をコストとして持つ. NPC リバースオーダグラフの始点から終点までの最短パスを探索することで得られた構造は, NCP を最小とする I-map に漸近的に一致する. すなわち, NPC リバースオーダグラフの最短パス探索は菅原ら [16] の手法の第二ステップの探索に相当する. この NPC リバースオーダグラフの探索を Malone ら [10] の提案した深さ優先分枝限定法で行う. 深さ優先分枝限定法を用いることで, 探索において NPC リバースオーダグラフの終点に達するたびに構造を学習でき, 探索を終了するまで構造を得ることができない問題を解決できる. また, 深さ優先分枝限定法は動的計画法に比べ効率的に最短パス探索が可能である.

深さ優先分枝限定法の効率は枝刈りによって向上する. 枝刈りは $f(\mathbf{U})$ コストを利用して行う. $f(\mathbf{U})$ コストとは, NPC リバースオーダグラフにおける始点からノード $\mathbf{U}$ までのパスの厳密なコスト ($g(\mathbf{U})$ コスト) と, ノード $\mathbf{U}$ から終点までのパスの見積もりコスト ($h(\mathbf{U})$ 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スコアを上回るとき, 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. $h(\mathbf{U})$ コストの計算には, ヒューリスティック関数を用いる. ヒューリスティック関数が無矛盾であるとき, 最短パスの探索が保証される. ただ, これまでに提案されている $h(\mathbf{U})$ コストの計算手法は, ML 最大化のために提案されたものであり, NCP 最小化には適さない.

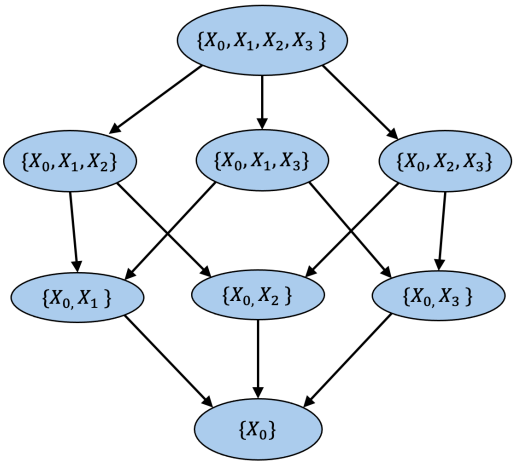

図 3: NPC リバースオーダグラフ

5.2 Naive Bayes を用いた下限

本論では, $h(\mathbf{U})$ コストに関する定理を証明し, 新しい $h(\mathbf{U})$ コストを提案する.

定理 5.1. 任意の変数集合 $\boldsymbol{V}$ に対して, *NPCDAG* の中で *NCP* を最小にする *I-map* を $G^*(\boldsymbol{V})$ とし, $G^*(\boldsymbol{V})$ における目的変数の子変数集合 $\boldsymbol{V}_c$ を説明変数とする *Naive Bayes* を $G^{NB}(\boldsymbol{V}_c)$ とすると

$$NCP(G^{NB}(\boldsymbol{V}_c)) \leq NCP(G^*(\boldsymbol{V}))$$

が成り立つ.

証明は付録に記した.

この定理に基づき, 提案手法は目的変数に関係のある変数を相互情報量による変数選択により求め, それらの変数で構成する Naive Bayes の NCP を探索におけるコストの下限値とする.

定理 5.1 より, 変数集合 $\mathbf{V}$ に対する NPCDAG の中で NCP を最小にする I-map $G^*(\mathbf{V})$ の目的変数の子変数集合 $\mathbf{V}_c$ が与えられれば, 適切な $h(\mathbf{U})$ を推定できる. しかし, $\mathbf{V}_c$ を事前に知ることはできない. そこで, 本論では, 相互情報量による変数選択により $\mathbf{V}_c$ を近似的に求め, $h(\mathbf{U})$ の推定に利用する手法を提案する.

変数選択は, 以下の式 (11) の相互情報量 $\mathrm{MI}(X, Y)$ が, しきい値より小さい場合に, 2 ノード X, Y が独立と判定する.

$$MI(X,Y) = \sum_{x=1}^{r_X} \sum_{y=1}^{r_Y} P(x,y) \log \frac{P(x,y)}{P(x)P(y)}. \tag{11}$$

ここで, $P(x,y)$ は $X=x, Y=y$ となる同時確率, $P(x)$ は $X=x$ となる確率, $P(y)$ は $Y=y$ となる確率を表す. また, r_X, r_Y はそれぞれ X, Y が取りうる状態数を表す.

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 $\forall X \in \mathbf{V}\backslash\{X_0\}$ について相互情報量を用いた変数選択を行い, 目的変数と関係のある変数集合 $\mathbf{M}$ を求める. そして, 目的変数と関係のある変数集合 $\mathbf{M}$ を説明変数とした Naive Bayes の NCP を深さ優先分枝限定法におけるコストの下限として用いる. NPC リバースオーダグラフにおけるノード $\mathbf{U}$ についてこ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.

$$h_1(\mathbf{U}) = \sum_{X_i \in (\mathbf{U} \cap \mathbf{M})} NCP_i(X_0). \tag{12}$$

提案するヒューリスティック関数について, 以下の仮定 5.1 の下で次の定理が成り立つ.

仮定 5.1. 任意の変数集合 $\boldsymbol{V}$ に対して, *NPCDAG* の中で *NCP* を最小にする *I-map* を $G^*(\boldsymbol{V})$ とし, $G^*(\boldsymbol{V})$ における目的変数の子変数集合を $\boldsymbol{V}_c$ としたとき, 相互情報量により選択された変数集合 $\boldsymbol{M}$ について, $\boldsymbol{V}_c = \boldsymbol{M}$ が成り立つ.

定理 5.2. 仮定 *5.1* の下で h_1 は無矛盾である.

証明は付録に記した.

提案手法は深さ優先分枝限定法の前に相互情報量を用いた変数選択を一度だけ行うため, 計算コストが低い. また, 探索の各ステップにかかる h_1 の計算コストも低く, 動的計画法に比べ枝刈りによる計算時間の改善が期待できる. また, 仮定 5.1 の下で無矛盾なヒューリスティック関数を用いるため, 最短パスの探索が保証される.

6 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.

6.1 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 5 種類の手法の分類精度を比較する.

- GBN-BDeu: BDeu を用いて厳密学習した GBN
- Naive Bayes
- ANB-BDeu (菅原ら [12, 13, 14]): BDeu を用いて厳密学習した ANB
- NPCDAG-DP (菅原ら [16]): 動的計画法を用いて, NCP を最小化する構造を探索する手法
- NPCDAG-DFBNB : Naive Bayes による下限値を用いて, 深さ優先分枝限定法により NCP を最小化する構造を探索する手法

本論では, NPCDAG-DFBNB を提案手法とする．GBN-BDeu, Naive Bayes, ANB-BDeu については, Java で実装し, NPCDAG-DP, NPCDAG-DFBNB については C++ で実装した．また, 3.2GHz の 16 コアプロセッサと 128GB のメモリを搭載した PC で実験を行った．UCI レポジトリデータベース [26] に登録されている 24 個のベンチマークデータセットを用いて実験を行った．菅原ら [16] と同様に, 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 のパラメータは全て EAP で推定した．GBN-BDeu, ANB-BDeu, NPCDAG-DP と提案手法の 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．

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 のデータセットは, 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 (sample per pattern: SPP) で昇順に上から並んでいる．表 2 は, GBN-BDeu, ANB-BDeu, NPCDAG-DP, 提案手法の平均 NCP をそれぞれ示している．さらに, 表 3 は, NPCDAG-DP と提案手法の NCP 最小の構造探索における平均実行時間を示している．提案手法の平均実行時間については, 相互情報量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 1 より, 菅原ら [16] の結果と同様に, SPP の大きい 19 番や 20 番のデータセットでは Naive Bayes は NPCDAG-DP と提案手法よりも分類精度が低いことが確認できる．Naive Bayes は親変数数に制限が設けられており, 親変数数の上限が小さいと表現力が低下する [27]．また, SPP の小さな 3 番のデータセットでは, NPCDAG-DP と提案手法は GBN-BDeu と ANB-BDeu よりも著しく高い分類精度を示している．このデータセットでは, NPCDAG-DP と提案手法は, GBN-BDeu と ANB-BDeu より NCP が小さいことが表 2 からわかる．また, NPCDAG-DP と提案手法の各データセットの分類精度の平均値は, GBN-BDeu と ANB-BDeu の各データセットの分類精度の平均値を上回っていることが確認できる．次に, NPCDAG-DP と提案手法を比較すると, NPCDAG-DP と提案手法の各データセットの分類精度の平均値はほぼ同等であることがわかる．以上から, 提案手法である NPCDAG-DFBNB は動的計画法を用いた従来手法と同等の精度を保っていることが示された．

表 2 より, NPCDAG-DP と提案手法の各データセットの NCP の平均値は GBN-BDeu と ANB-BDeu の NCP の平均値を下回っていることが分かる．また, NPCDAG-DP と提案手法の平均 NCP を比較すると, ほとんどのデータセットで同等の値を示している．

表 3 より, NPCDAG-DP と提案手法の計算時間を比較すると, 提案手法は 21 番と 23 番を除く全てのデータセットにおいて NPCDAG-DP よりも計算時間を削減した．21 番と 23 番のデータセットにおいて, 提案手法の計算時間が NPCDAG-DP の計算時間より大きくなっているのは, 21 番と 23 番のデータセットはサンプルサイズが大きく, 相互情報量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, 13, 14, 20 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, 提案手法の計算時間のほうが

表 1: 各手法の分類精度.

No.	Dataset	Variables	Sample size	SPP	Naive-Bayes	GBN-BDeu	ANB-BDeu	NPCDAG-DP	NPCDAG-DFBNB
1	Lymphography	19	148	1.63×10^{-7}	0.8446	0.7500	0.8108	0.7635	0.7838
2	Breast Cancer Wisconsin	10	683	3.42×10^{-7}	0.9751	0.9751	0.9751	0.9737	0.9737
3	Hepatitis	20	80	7.63×10^{-5}	0.8500	0.6125	0.5750	0.8250	0.8000
4	Zoo	17	101	1.03×10^{-4}	0.9802	0.9228	0.9406	0.9505	0.9208
5	Australian	15	690	2.97×10^{-4}	0.8290	0.8507	0.8203	0.8493	0.8449
6	Vehicle	19	846	8.07×10^{-4}	0.4350	0.5898	0.6217	0.6050	0.5843
7	Breast Cancer	10	277	8.33×10^{-4}	0.7401	0.7256	0.6968	0.7076	0.7365
8	Image Segmentation	19	2310	1.26×10^{-3}	0.7290	0.8255	0.8273	0.8264	0.8320
9	Congressional Voting Records	17	232	1.77×10^{-3}	0.9095	0.9655	0.9483	0.9698	0.9655
10	Heart	14	270	2.44×10^{-3}	0.8259	0.8370	0.8037	0.8222	0.8333
11	Solar Flare	11	1389	3.72×10^{-3}	0.7811	0.8431	0.8215	0.8431	0.8409
12	Wine	14	178	7.24×10^{-3}	0.9270	0.9270	0.9270	0.9494	0.9494
13	Letter	17	20000	1.17×10^{-2}	0.4466	0.6434	0.6434	0.6290	0.6303
14	Pendigits	17	10992	1.68×10^{-2}	0.8032	0.9342	0.9332	0.9368	0.9373
15	Contraceptive Method Choice	10	1473	5.99×10^{-2}	0.4671	0.4792	0.4481	0.4616	0.4396
16	Glass	10	214	6.97×10^{-2}	0.5561	0.5888	0.6355	0.5794	0.6036
17	Hayes-Roth	5	132	2.29×10^{-1}	0.8182	0.6212	0.7879	0.8333	0.8333
18	Balance Scale	5	625	3.33×10^{-1}	0.9152	0.9152	0.9152	0.9152	0.9152
19	Lenses	5	24	3.33×10^{-1}	0.7500	0.8333	0.7500	0.8750	0.8750
20	EEG	15	14980	4.57×10^{-1}	0.5778	0.7246	0.7212	0.7155	0.7135
21	LED7	8	3200	2.50×10^{0}	0.7294	0.7303	0.7303	0.7316	0.7325
22	Iris	5	150	3.13×10^{0}	0.7133	0.8267	0.8156	0.8200	0.8200
23	HTRU2	9	17898	3.50×10^{1}	0.8966	0.9141	0.9141	0.9140	0.9140
24	Banknote authentication	5	1372	4.29×10^{1}	0.8433	0.8812	0.8812	0.8819	0.8819
	average				0.7643	0.7882	0.7893	0.8070	0.8080

表 2: 各手法によって学習された構造の平均 NCP.

No.	Variables	Sample size	SPP	GBN-BDeu	ANB-BDeu	NPCDAG-DP	NPCDAG-DFBNB
1	19	148	1.63×10^{-7}	216535	219126	104	120
2	10	683	3.42×10^{-7}	150	179	159	158
3	20	80	7.63×10^{-5}	2011	5880	10	9
4	17	101	1.03×10^{-4}	4455	816	508	131
5	15	690	2.97×10^{-4}	65	447	64	60
6	19	846	8.07×10^{-4}	987	556	1377	1380
7	10	277	8.33×10^{-4}	20	105	62	37
8	19	2310	1.26×10^{-3}	3840	2464	4324	5551
9	17	232	1.77×10^{-3}	24	121	10	9
10	14	270	2.44×10^{-3}	32	58	18	22
11	11	1389	3.72×10^{-3}	19	1570	8	12
12	14	178	7.24×10^{-3}	36	59	28	28
13	17	20000	1.17×10^{-2}	18360	18386	12336	12339
14	17	10992	1.68×10^{-2}	11042	11215	9175	9886
15	10	1473	5.99×10^{-2}	40	196	37	28
16	10	214	6.97×10^{-2}	354	444	483	655
17	5	132	2.29×10^{-1}	176	40	29	29
18	5	625	3.33×10^{-1}	48	50	50	50
19	5	24	3.33×10^{-1}	8	18	8	8
20	15	14980	4.57×10^{-1}	2850	2703	1849	1849
21	8	3200	2.50×10^{0}	186	187	94	98
22	5	150	3.13×10^{0}	18	28	19	19
23	9	17898	3.50×10^{1}	69	77	198	176
24	5	1372	4.29×10^{1}	25	31	15	15
average				10890	11032	1290	1361

NPCDAG-DP の計算時間よりも短いことが分かる．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．以上から, 提案手法が動的計画法を用いた従来手法よりも計算時間を削減できることが示された.

6.2 大規模データセットを用いた NCP 及び分類精度の時間変化

この実験では, 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が得られることの有効性を示す．具体的には提案手法によって得られたある時点までの最適解である暫定解について NCP と分類精度を推定し, 時間経過によって NCP と分類精度がどのように変化するのかを示す．また, 本節では, Naive Bayes, NPCDAG-DP と提案手法の分類精度を比較する．ベンチマークデータセットについては, 前節の実験で用いたデータセットに比べ大規模な 36 変数, サンプルサイズ 3196 の UCI レポジトリデータベースに登録されている kr-vs-kp を用いた．構造学習後の BNC のパラメータは全て EAP で推定し, 学習スコアとしては ESS の設定が必要ない BIC を用いた．構造学習は, Malone ら [9] の制限時間と同様の 24 時間で打ち切った．また, 深さ優先分枝限定法によって得られた暫定解について NCP と分類精度を推定し, NCP と分類精度をプロットした.

表 3: 各手法の NCP 最小の構造探索における
平均計算時間.

No.	Variables	Sample size	SPP	NPCDAG-DP	NPCDAG-DFBNB
1	19	148	1.63×10^{-7}	555.6909	48.2147
2	10	683	3.42×10^{-7}	0.3175	0.0795
3	20	80	7.63×10^{-5}	10044.8238	179.6022
4	17	101	1.03×10^{-4}	459.3530	19.6462
5	15	690	2.97×10^{-4}	18.3376	4.3371
6	19	846	8.07×10^{-4}	6527.5938	193.2480
7	10	277	8.33×10^{-4}	0.3236	0.0959
8	19	2310	1.26×10^{-3}	5588.0012	245.6302
9	17	232	1.77×10^{-3}	427.0858	31.2531
10	14	270	2.44×10^{-3}	9.9224	2.7931
11	11	1389	3.72×10^{-3}	0.8851	0.3960
12	14	178	7.24×10^{-3}	16.0481	3.8875
13	17	20000	1.17×10^{-2}	530.7353	96.0885
14	17	10992	1.68×10^{-2}	744.8170	140.5306
15	10	1473	5.99×10^{-2}	0.3885	0.1506
16	10	214	6.97×10^{-2}	0.4825	0.2501
17	5	132	2.29×10^{-1}	0.0170	0.0084
18	5	625	3.33×10^{-1}	0.0134	0.0121
19	5	24	3.33×10^{-1}	0.0131	0.0067
20	15	14980	4.57×10^{-1}	166.2700	22.7070
21	8	3200	2.50×10^{0}	0.1214	0.1254
22	5	150	3.13×10^{0}	0.0181	0.0154
23	9	17898	3.50×10^{1}	0.2785	0.4542
24	5	1372	4.29×10^{1}	0.0188	0.0180
average				1045.4815	41.2313

図 4 は提案手法における暫定解の NCP の時間変化を示し, 図 5 は提案手法における暫定解の分類精度の時間変化を示している.

図 4, 図 5 では, 536 秒の時点でプロットが終了している. これは, 536 秒の時点でメモリオーバによって構造学習が終了したことを表す. 図 4 からは時間経過に伴い NCP が単調に減少していることが確認できる. また, 探索により得られた最終的な構造の NCP は, 探索の最初に得た構造の NCP の約四分の一になっており, 十分に小さな NCP を持つ構造を学習できていることが分かる.

次に, 図 5 から, 200 秒付近で分類精度が大きく上昇していることが分かる. これは, NCP の減少により一つのパラメータ学習のためのサンプルサイズが大きくなり, 推定精度が上昇したためであると考えられる. 実際, 図 4 より, 200 秒付近で暫定解の NCP が大きく減少していることが分かる.

次に, Naive Bayes, DFBNB-DP, 提案手法の分類精度を比較する. 提案手法の分類精度は, 探索の最後に得た最も NCP の小さな構造の分類精度とする. 各手法の分類精度を表 4 に示す. 表 4 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.

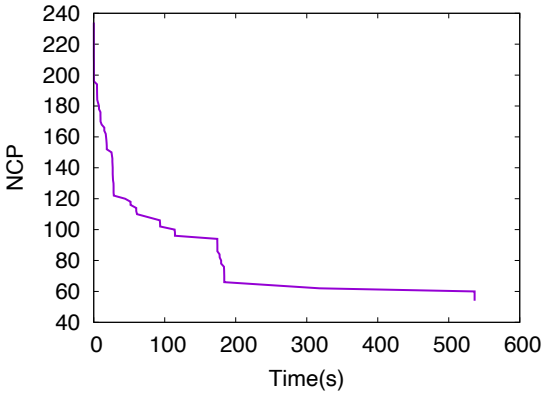

図 4: 暫定解の NCP の時間変化

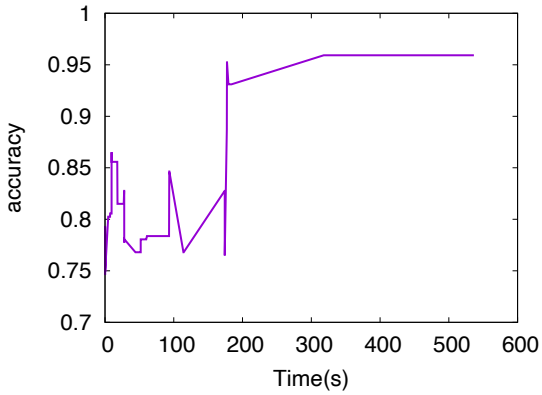

図 5: 暫定解の分類精度の時間変化

表 4: kr-vs-kp における各手法の分類精度

Dataset	Naive Bayes	NPCDAG-DP	NPCDAG-DFBNB
kr-vs-kp	0.7994	TO	0.9593

表 4 より, Naive Bayes と提案手法の分類精度を比較すると, 提案手法の分類精度が Naive Bayes の分類精度を大きく上回っていることが分かる. また, 図 5 より, 探索開始から 180 秒以降に得た全ての構造の分類精度が Naive Bayes の分類精度を上回っていることが確認できる. DFBNB-DP と提案手法を比較すると, DFBNB-DP は 24 時間の制限時間内に構造学習を終了することができず, 構造を一つも得ることができなかった. 以上から, 提案手法は時間制限が設けられている場合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, 従来手法である動的計画法では実現できなかった規模の学習を実現できることが示された.

7 むすび

本論では, 深さ優先分枝限定法による分類に影響する目的変数パラメータ数 (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手法を提案した。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持つ. (1) 従来の動的計画法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, 提案手法は従来の動的計画法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, 深さ優先分枝限定法を用いることで, 従来手法の問題点であった, 探索が終了するまで構造を得ることができない問題を解決し, 36 変数の構造学習ができることを示した.

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.

参考文献

[1] Nir Friedman, Dan Geiger, and Moises Goldszmidt. Bayesian network classifiers. *Machine learning*, 29(2-3):131–163, 1997.

[2] D. M. Chickering. Learning Bayesian networks is NP-Complete. In *Learning from Data: Artificial Intelligence and Statistics*, volume V, pages 121–130. Springer, 1996.

[3] R. G. Cowell. Efficient maximum likelihood pedigree reconstruction. *Theoretical Population Biology*, 76(4):285–291, December 2009.

[4] Mikko Koivisto and Kismat Sood. Exact Bayesian structure discovery in Bayesian networks. *Journal of Machine Learning Research*, 5:549–573, December 2004.

[5] Ajit Singh and Andrew Moore. Finding optimal Bayesian networks by dynamic programming. Technical report, Carnegie Mellon University, pp.1–16, June 2005.

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

[7] B. Malone, C. Yuan, and Eric A. Hansen. Memory-efficient dynamic programming for learning optimal Bayesian networks. In *Proc. of the 25th AAAI Conference*, pages 1057–1062, 2011.

[8] C. Yuan, B. Malone, and W. Xiaojian. Learning optimal Bayesian networks using A* search. In *International Joint Conference on Artificial Intelligence (IJCAI)*, pages 2186–2191, 2011.

[9] Brandon M. Malone, Changhe Yuan, Eric A. Hansen, and Susan Bridges. Improving the Scalability of Optimal Bayesian Network Learning with External-Memory Frontier Breadth-First Branch and Bound Search. In *Proceedings of Uncertainty in Artificial Intelligence*, pages 479–488, 2011.

[10] Brandon Malone and Changhe Yuan. A depth-first branch and bound algorithm for learning optimal bayesian networks. In *Graph Structures for Knowledge Representation and Reasoning*, pages 111–122. Springer, 2014.

[11] J. Cussens. Bayesian network learning with cutting planes. In *Proceedings of Uncertainty*

in Artificial Intelligence (UAI), pages 153–160. AUAI Press, 2011.

[12] Shouta Sugahara, Masaki Uto, and Maomi Ueno. Exact learning augmented naive Bayes classifier. In *International Conference on Probabilistic Graphical Models*, pages 439–450, 2018.

[13] 菅原聖太 and 植野真臣. Augmented naive bayes 制約を持つベイジアンネットワーク分類器の厳密学習. **電子情報通信学会論文誌** *D*, 103:301–313, 2020.

[14] Shouta Sugahara and Maomi Ueno. Exact learning augmented naive bayes classifier. *Entropy*, 23(12):1703, 2021.

[15] Shouta Sugahara, Wakaba Kishida, Koya Kato, and Maomi Ueno. Recursive autonomy identification-based learning of augmented naive bayes classifiers. In *International Conference on Probabilistic Graphical Models*, pages 265–276. PMLR, 2022.

[16] 菅原聖太 and 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. **電子情報通信学会論文誌** *D*, 105(11):679–690, 2022.

[17] Bojan Mihaljević, Concha Bielza, and Pedro Larrañaga. Learning Bayesian network classifiers with completed partially directed acyclic graphs. In *Proceedings of the Ninth International Conference on Probabilistic Graphical Models*, volume 72 of *Proceedings of Machine Learning Research*, pages 272–283, 2018.

[18] W. Buntine. Theory refinement on Bayesian networks. In *Proceedings of Uncertainty in Artificial Intelligence (UAI)*, pages 52–60, 1991.

[19] David Heckerman, Dan Geiger, and David M. Chickering. Learning Bayesian Networks: The Combination of Knowledge and Statistical Data. *Machine Learning*, 20(3):197–243, 1995.

[20] G. Schwarz. Estimating the Dimension of a Model. *Annals of Statistics*, 6(2):461–464, 1978.

[21] J. Rissanen. Modeling by shortest data description. *Automatica*, 14(5):465 – 471, 1978.

[22] Maomi Ueno. Learning Networks Determined by the Ratio of Prior and Data. In *Proceedings of the Twenty-Sixth Conference on Uncertainty in Artificial Intelligence*, page 598–605, Arlington, Virginia, USA, 2010. AUAI Press.

[23] Cassio P. de Campos and Qiang Ji. Efficient Structure Learning of Bayesian Networks Using Constraints. *Journal of Machine Learning Research*, 12(12):663–689, 2011.

[24] Mark Barlett and James Cussens. Advances in Bayesian Network Learning Using Integer Programming. In *Proceedings of the Twenty-Ninth Conference on Uncertainty in Artificial Intelligence*, pages 182–191, 2013.

[25] Marvin Minsky. Steps toward Artificial Intelligence. In *Proceedings of the IRE*, volume 49, pages 8–30, 1961.

[26] M. Lichman. UCI Machine Learning Repository, 2013.

[27] Charles X. Ling and Huajie Zhang. The Representational Power of Discrete Bayesian Networks. *Journal of Machine Learning Research*, 3:709–721, 2003.

付録 A　定理 5.1 の証明

証明

(定理 5.1)**.** 任意の変数集合 $\boldsymbol{V}$ を考える．$\boldsymbol{V}$ に対する *NPCDAG* の中で *NCP* を最小にする *I-map* $G^*(\boldsymbol{V})$ について, $G^*(\boldsymbol{V})$ における目的変数の子変数集合 $\boldsymbol{V}_c$ を説明変数とする *Naive Bayes* を $G^{NB}(\boldsymbol{V}_c)$ 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, $G^{NB}(\boldsymbol{V}_c)$ における $X_i \in \boldsymbol{V}_c$ についての *NCP* は $NCP_i(\{X_0\})$ で表せる．したがって, $NCP(G^{NB}(\boldsymbol{V}_c))$ は以下のように表せる．

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

また, $X_i \in \boldsymbol{V}_c$ としたとき, $G^{NB}(\boldsymbol{V}_c)$ における X_i についての *NCP* $NCP_i(\{X_0\})$ と, $G^*(\boldsymbol{V})$ における X_i についての *NCP* $NCP_i(\boldsymbol{Pa}(X_i, G^*(\boldsymbol{V})))$ に関して,

$$NCP_i(\{X_0\}) = (r_i - 1)r_0,$$
$$NCP_i(\boldsymbol{Pa}(X_i, G^*(\boldsymbol{V}))) = (r_i - 1)q_i$$

が成り立つ．ここで, q_i は $\boldsymbol{Pa}(X_i, G^*(\boldsymbol{V}))$ のとり得るパターン数を表す．また, $X_i \in \boldsymbol{V}_c$ より, $X_0 \in \boldsymbol{Pa}(X_i, G^*(\boldsymbol{V}))$ 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．

$$NCP_i(\{X_0\}) \leq NCP_i(\boldsymbol{Pa}(X_i, G^*(\boldsymbol{V})))$$

以上より,

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

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

$$NCP(G^{NB}(\boldsymbol{V}_c)) \leq NCP(G^*(\boldsymbol{V}))$$

□

付録 B　定理 5.2 の証明

証明

(定理 5.2)**.** 変数集合 $\boldsymbol{V}$ に対して, X_0 と $\forall X \in \boldsymbol{V}\backslash\{X_0\}$ についての相互情報量により選択された変数集合を $\boldsymbol{M}$ とする. このとき, $G^*(\boldsymbol{V})$ における目的変数の子変数集合を $\boldsymbol{V}_c$ としたとき, $\boldsymbol{V}_c = \boldsymbol{M}$ が成り立つと仮定する. NPC リバースオーダグラフの任意のノード $\boldsymbol{U}$ と $\boldsymbol{U}$ からエッジが引かれている任意のノード $\boldsymbol{R}$ について, $X_j \in \boldsymbol{U}\backslash\boldsymbol{R}$ とし, $c(\boldsymbol{U}, \boldsymbol{R})$ をノード $\boldsymbol{U}$ からノード $\boldsymbol{R}$ へのエッジがもつコストであるとする. ここで, $X_j \notin \boldsymbol{V}_c$ のとき,

$$
\begin{aligned}
h_1(\boldsymbol{U}) &= \sum_{X_i \in (\boldsymbol{U} \cap \boldsymbol{M})} NCP_i(X_0) \\
&= \sum_{X_i \in (\boldsymbol{U} \cap \boldsymbol{V}_c)} NCP_i(X_0) \\
&= \sum_{X_i \in (\boldsymbol{R} \cap \boldsymbol{V}_c)} NCP_i(X_0) \\
&\leq \sum_{X_i \in (\boldsymbol{R} \cap \boldsymbol{V}_c)} NCP_i(X_0) + NCP_j(g_j^*(\boldsymbol{U}\backslash\{X_j\})) \\
&= h_1(\boldsymbol{R}) + c(\boldsymbol{U}, \boldsymbol{R})
\end{aligned}
$$

が成り立つ. また, $X_j \in \boldsymbol{V}_c$ のとき,

$$
\begin{aligned}
h_1(\boldsymbol{U}) &= \sum_{X_i \in (\boldsymbol{U} \cap \boldsymbol{M})} NCP_i(X_0) \\
&= \sum_{X_i \in (\boldsymbol{U} \cap \boldsymbol{V}_c)} NCP_i(X_0) \\
&= \sum_{X_i \in (\boldsymbol{U} \cap \boldsymbol{V}_c)\backslash\{X_j\}} NCP_i(X_0) + NCP_j(X_0) \\
&= \sum_{X_i \in (\boldsymbol{R} \cap \boldsymbol{V}_c)} NCP_i(X_0) + NCP_j(X_0) \\
&\leq \sum_{X_i \in (\boldsymbol{R} \cap \boldsymbol{V}_c)} NCP_i(X_0) + NCP_j(g_j^*(\boldsymbol{U}\backslash\{X_j\})) \\
&= h_1(\boldsymbol{R}) + c(\boldsymbol{U}, \boldsymbol{R}).
\end{aligned}
$$

以上から,

$$
h_1(\boldsymbol{U}) \leq h_1(\boldsymbol{R}) + c(\boldsymbol{U}, \boldsymbol{R}).
$$

したがって, h_1 は仮定 *5.1* が成り立つならば無矛盾である. □