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

整数計画法によるベイジアンネットワーク分類器の学習

2023 年 9 月 18 日

情報数理工学プログラム

学籍番号 2010066

稲村 健太郎

指導教員 植野 真臣

令和 5 年度 情報数理工学プログラム卒業論文概要

令和 2 年度　入学	学籍番号　2010066
指導教員　植野 真臣	氏名　稲村 健太郎
題目　整数計画法によるベイジアンネットワーク分類器の学習	

概要

ベイジアンネットワーク分類器は高い予測精度を持つことから, これまで様々な目的で応用されてきた.

本論では, 整数計画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的に一致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.

整数計画法は, 最大親変数数の制限により, 空間計算量を減じることができるため, 従来手法においてメモリ不足により打ち切られる構造学習を最後まで行うことができる. しかし, これまでに提案されている整数計画法の定式化は NCP を最小にして真の分類確率に漸近的に一致するベイジアンネットワーク分類器の構造学習には適さない.

そこで本論では, (1)NCP を最小にして真の分類確率に漸近収束する構造を学習するための目的関数と, (2) 目的変数が親変数を持たないベイジアンネットワーク分類器を学習するための制約を導入した.

提案手法は, 構造学習中に使用されるメモリ使用量が少ないため, 従来手法においてメモリ不足により打ち切られる構造学習を最後まで行うことができる. 複数のベンチマークを用いた実験により, 従来手法では 30 変数程度の構造学習で打ち切りが生じるが, 提案手法では 58 変数の構造学習を最後まで学習できることを示す. また, 従来手法で時間制限やメモリ不足により構造学習を途中で打ち切った 30 変数以上の構造について, 提案手法により最後まで構造学習を行い, 分類精度を改善できることを示す.

1　まえがき

ベイジアンネットワーク分類器 (Bayesian Network Classifier: BNC) は高い予測精度を持つことから, これまで様々な目的で応用されてきた [1, 2, 3].

現在, 最も分類精度が高い BNC として, 菅原ら [4] は, 目的変数が親変数を持たない制約の下で, 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) を最小にする 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) 構造学習では, 全ての構造の中で, 周辺尤度 (Marginal Likelihood: ML) を最大にする構造を推定し [5, 6, 7, 8, 9, 10, 11, 12, 13, 14], 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 [4] 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. 第一ステップは次の手順で行われる. 手順 (i) では, 各変数とその考えられるすべての親変数集合のみから成るネットワークの ML を求める. 手順 (ii) では, 目的変数から始まる変数順序に従う各変数の ML 最大となる親変数集合を決定する. このステップは変数数を n とすると時間計算量, 空間計算量ともに $\mathcal{O}(2^n)$ で計算され, 最大親変数数を d に制限した場合, 時間計算量, 空間計算量ともに $\mathcal{O}(\sum_{i=0}^{d} \binom{n-1}{i})$ に減じることができる. 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. このステップは $\mathcal{O}(n2^n)$ の時間計算量, $\mathcal{O}(2^n)$ の空間計算量で計算される. 第二ステップの探索は探索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. しかし, 彼らのアルゴリズムは第二ステップで膨大な計算時間がかかることから 20 変数程度の構造学習が限界であった.

そこで, 加藤ら [15] 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た. 菅原ら [4] が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. そこで加藤ら [15] 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した. 加藤ら [15] の手法は菅原らの手法 [4] と時間計算量, および空間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. 加藤ら [15] の手法は菅原の手法 [4] よりも計算時間を削減し, 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できる．彼らは, 最大親変数数に制限を設けて 60 変数程度の構造学習の実現を報告している．しかし, 彼らの手法では, 第一ステップの手順 (ii) と第二ステップの探索を別々に行っており, 効率が悪い． また, 彼らの手法は最大親変数数を制限しているのにも関わらず, 実際は 30 変数程度でメモリ制限により学習が打ち切られている．学習が途中で打ち切られた場合, 学習した構造の分類精度が低下する可能性がある．

そこで, 本論では, これらの問題を緩和する新たなアルゴリズムを提案する．加藤ら [15] の手法は最大親変数数の制限によらず $\mathcal{O}(2^n)$ の空間計算量が必要であり, 改良が必要である．本論では, 最大親変数数の制限によって空間計算量を小さくできる整数計画法を用いた新しいアルゴリズムを提案する．

これまでに, Cussens ら [14] は整数計画法を用いた BN 構造学習を行う手法を提案している．彼らは, 整数計画問題の目的関数として ML を用いており, これを最大化している． 一方, 本論では, 真の同時確率を表現可能な構造の中で NCP 最小の構造を探索する目的関数として, ML から NCP に関するペナルティ項を引いた式を提案する．そして, NCP に関するペナルティ項には適切なハイパーパラメータを掛け, 目的関数を最大化することで, 真の同時確率を表現可能な構造の中で NCP 最小の構造を探索する．この目的関数により, 加藤ら [15] の手法では別々に行っている, 第一ステップの手順 (ii), および, 第二ステップの探索を同時に行うことができる． また Cussens ら [14] は, 整数計画問題の制約として, (1) 各変数の親変数集合がただ一つである, (2) 構造に循環を含まない, の 2 つを設定している．本論では, 整数計画問題の制約として, (1)(2) に, (3) 目的変数が親変数を持たない, を加えて学習する．

最大親変数数を d に制限したとき, 提案手法の空間計算量は $\mathcal{O}(n\sum_{i=0}^{d}\binom{n-1}{i})$ となる．たとえば, 最大親変数数を 3 に制限したとき, 加藤ら [15] の手法では, 空間計算量は $\mathcal{O}(2^n)$ であるが, 提案手法の空間計算量は $\mathcal{O}(n^4)$ となる．これにより, 提案手法は従来手法に比べて, より大規模な構造学習をメモリ制限により途中で打ち切られることなく行うことができる．

本論では, 複数のベンチマークによる比較実験を行い, 加藤ら [15] では 30 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できることを示す．また, 加藤ら [15] がメモリ不足により構造学習を途中で打ち切った 30 変数以上の構造について, 提案手法により最後まで構造学習を行い, 分類精度を改善できることを示す．

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 (Bayesian Network Classifier: BNC) はベイジアンネットワーク (Bayesian Network: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 (Dyrected 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, 16, 17, 18, 19, 20]. 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持たないとする.

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持つとする. 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. さらに, X_0 を目的変数, $X_1, ..., X_n$ 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 G における X_i の親変数集合とする. また, 説明変数のデータ $\mathbf{x} = \langle x_1, \ldots, x_n \rangle$ が得られたとき, 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, $\mathbf{Pa}(X_i, G)$ が j 番目のパターンを取ることを $\mathbf{Pa}(X_i, G) = j$ と書き, θ_{ijk} を $\mathbf{Pa}(X_i, G) = j$ となるときに $X_i = k$ となる条件付確率パラメータとする. ここで, 条件付確率パラメータ集合 Θ は θ_{ijk} を用いて $\Theta = \bigcup_{i=0}^{n} \bigcup_{j=1}^{q_i} \bigcup_{k=1}^{r_i} \{\theta_{ijk}\}$ により定義される. 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} の推定値には, 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時の 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 [21].

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

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る. 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 $X \rightarrow Z \leftarrow Y$ 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 分離は次のように定義される.

(定義 2.1)G において $X, Y \in \mathbf{V}$, $\mathbf{Z} \subseteq \mathbf{V} \setminus \{X, Y\}$ として, $\mathbf{Z}$ 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.

(条件 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.

(条件 2)p における合流点 $Z \in \mathbf{Z}$ 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 と Y が $\mathbf{Z}$ を所与として条件付き独立であることを $I(X, Y \mid \mathbf{Z})_M$ と書く. 次に, I-map を以下のように定義する [22].

(定義 2.2) BN 構造 G が次式を満たすとき, 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 ならば, 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 [23].

2.2 目的変数パラメータ数を最小にする BNC

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あり, 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 [4] は, 分類確率のみを効率的に推定する BNC として, 次に定義す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, $NCP_i(\mathbf{Pa}(X_i, G))$ は次のように計算される.

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

菅原ら [4] の手法を説明するため, 変数順序および周辺尤度を定義する. $\mathbf{V}$ 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 $\forall i \in \{1,\ldots,n\}, \mathbf{Pa}(X_{\sigma_i}, G) \subseteq \bigcup_{j=1}^{i-1}\{X_{\sigma_j}\}$ が成立するとき, σ を G の変数順序と定義する. また, G の周辺尤度 $P(D \mid G)$ (以降, ML) は条件付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, 次のような閉形式で表される [24].

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

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 [21, 24]. 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, logBDeu 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, $Score_i(\mathbf{Pa}(X_i, G))$ はローカルスコアと呼ばれ, X_i と $\mathbf{Pa}(X_i, G)$ のみに依存する関数である. ローカルスコア $Score_i(\mathbf{Pa}(X_i, G))$ は以下のように表される [24].

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

菅原ら [4] は変数集合 $\mathbf{V}$ からなる全ての変数順序集合を $\sigma(\mathbf{V})$ としたとき, 次の定理を証明した.

(定理 2.1)$\forall\sigma \in \sigma(\mathbf{V})$ 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, 彼らの手法は次の二つのステップにより構成される. 第一ステップでは, 目的変数から始まるすべ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. 第一ステップを説明するために必要な記号を以下で定義する. 目的変数 X_0 から始まる変数順序の集合を $\sigma_0(\mathbf{V})$, 変数順序 σ において変数 X に先行する変数の集合を $\mathbf{Pre}(X, \sigma)$ とする. また, 変数 X_i と変数集合 $\mathbf{Z} \subseteq \mathbf{V}\backslash\{X_i\}$ について, X_i の最適親変数集合を以下で定義する.

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

第一ステップは次の手順で行われる. 手順 (i) では, $\forall X_i \in \mathbf{V}, \mathbf{W} \subseteq \mathbf{Z}$ について, $Score_i(\mathbf{W})$ を求める. 手順 (ii) では, $\forall X_i \in \mathbf{V}, \sigma \in \sigma_0(\mathbf{V})$ について, $g_i^*(\mathbf{Pre}(X_i, \sigma))$ を求める. このステップは

変数数を n とすると, 時間計算量, 空間計算量ともに $\mathcal{O}(2^n)$ で計算される [8]. そして, 最大親変数数を d に制限した場合, 時間計算量, 空間計算量ともに $\mathcal{O}(\sum_{i=0}^{d} \binom{n-1}{i})$ に減少する. 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. このステップは $\mathcal{O}(n2^n)$ の時間計算量, $\mathcal{O}(2^n)$ の空間計算量で計算される. 第二ステップの探索は探索グラフの最短パス探索問題として定式化される. 菅原ら [4] は, 幅優先探索により最短パスを探索した.

菅原ら [4] が用いている幅優先探索は, 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. そこで加藤ら [15] 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提案した. 加藤ら [15] の手法は, 菅原ら [4] の手法と時間計算量, および空間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. 加藤ら [15] の手法は菅原の手法 [4] よりも計算時間を削減し, 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できる.

3 整数計画法による BNC 学習法の提案

加藤ら [15] の手法では, 第一ステップの手順 (ii) と第二ステップの探索を別々に行っており, 効率が悪い. また, 彼らの手法では, 変数数が大きくなると, 第一ステップの探索を効率的に行うために, 最大親変数数に制限を設ける必要がある. しかし, 彼らの手法では, 最大親変数数に制限を設けても, 第二ステップの探索において保持する探索グラフのノード数は変化しないため, $\mathcal{O}(2^n)$ の空間計算量が必要である. したがって, 彼らの手法は最大親変数数に制限を設けても 30 変数程度でメモリオーバにより学習が打ち切られてしまう. 学習が途中で打ち切られた場合, 学習された構造の分類精度が低下する可能性がある. そこで, 本論では, 最大親変数数を d に制限にしたとき, $\mathcal{O}(n\sum_{i=0}^{d} \binom{n-1}{i})$ に空間計算量を減らすことのできる, 整数計画法を用いた新しいアルゴリズムを提案する.

以下では整数計画問題の定式化を考える. まず, 目的関数についての詳細を述べ, その後, 制約の詳細を述べる.

3.1 目的関数

最初に, 整数計画問題の目的関数を考える. Cussens ら [14] は, 整数計画法を用いて BN 構造学習を行った. 彼らは, 整数計画問題の定式化のために, 各変数 $X \in \mathbf{V}$, および親変数集合 $\mathbf{W}$ に対

して, 二値変数 $I(\mathbf{W} \to X)$ (family variable) を次のように定義した.

$$I(\mathbf{W} \to X) = \begin{cases} 1 & \text{BNC において} \\ & \mathbf{W} \text{ が } X \text{ の親であるとき} \\ 0 & otherwise \end{cases}$$

BNC は, family variable により, 01 に符号化することができる. 図 1 は変数集合 $\mathbf{V} = \{X_0, X_1, X_2\}$ の各変数をノードとして持ち, 目的変数を X_0 とする BNC の一例を表し, 表 1 は図 1 で表されている BNC の family variable での符号化を表している.

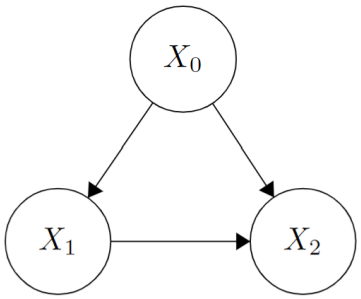

図 1　3 変数の BNC の例

表 1　図 1 の BNC の family variable での符号化

$I(\{\} \to X_0)$	$I(\{X_1\} \to X_0)$	$I(\{X_2\} \to X_0)$	$I(\{X_1, X_2\} \to X_0)$
1	0	0	0
$I(\{\} \to X_1)$	$I(\{X_0\} \to X_1)$	$I(\{X_2\} \to X_1)$	$I(\{X_0, X_2\} \to X_1)$
0	1	0	0
$I(\{\} \to X_2)$	$I(\{X_0\} \to X_2)$	$I(\{X_1\} \to X_2)$	$I(\{X_0, X_1\} \to X_2)$
0	0	0	1

Cussens ら [14] は, この family variable を用いて, BN 構造学習のために次の目的関数を提案した.

$$\text{(最大化)} \sum_{X,\mathbf{W}} Score_X(\mathbf{W}) I(\mathbf{W} \to X) \tag{9}$$

ここで, $Score_X(\mathbf{W})$ はローカルスコアを表す. 式 (9) はすべての変数のローカルスコアの和である.

しかし, この目的関数は, ML を最大にする BN の構造学習を行うために提案されたものであり, NCP を最小にして真の分類確率に漸近的に一致する BNC の構造学習には適さない. そこで, 本論では, 以下の目的関数を提案する.

$$\text{(最大化)} \sum_{X,\mathbf{W}} Score_X(\mathbf{W}) I(\mathbf{W} \to X) - \gamma \times \sum_{X,\mathbf{W}} NCP_X(\mathbf{W}) I(\mathbf{W} \to X) \tag{10}$$

式 (10) の第一項はすべての変数のローカルスコアの和であり, 第二項は NCP の大きさを反映するペナルティ項である. ここで, ローカルスコア $Score_X(\mathbf{W})$ には BDeu を用いている. 目的関数のペナルティ項の係数 γ に適切な値を定め, 目的関数である式 (10) を最大化することで, 目的関数の第一項で表されている BDeu スコアをできるだけ大きくしつつ, ペナルティ項となっている第二項の NCP 項をできるだけ小さくする. この目的関数により, 加藤ら [15] の手法では別々に行っている, 第一ステップの手順 (ii), および, 第二ステップの探索を同時に行うことができる.

3.2 制約

次に, 整数計画問題の制約を考える. Cussens ら [14] は, 整数計画問題の制約として, (1) 各変数 $X \in \mathbf{V}$ の親変数集合 $\mathbf{W}$ がただ一つである, (2) 構造に循環を含まない, の 2 つを定式化した.

彼らは, family variable を用いて, 制約 (1) を次の通り定式化した.

$$\forall X : \sum_{\mathbf{W}} I(\mathbf{W} \to X) = 1 \tag{11}$$

例えば, 図 1 の BNC は, 各変数の親変数集合が 1 つである. 表 1 を見ると, 制約 (1) が成立していることが分かる.

次に, 彼らは制約 (2) を次の通り定式化した.

$$\forall C \subseteq \mathbf{V} : \sum_{X \in C} \sum_{\mathbf{W}:\mathbf{W} \cap C = \emptyset} I(\mathbf{W} \to X) \geq 1 \tag{12}$$

ここで, C は要素数が 2 以上の $\mathbf{V}$ の部分集合であり, クラスタと呼ぶ. クラスタ C 中のノードが循環を形成しているとすると, 左辺が 0 になり, 制約 (2) に違反する. 図 2 は, いずれも変数集合 $\mathbf{V} = \{X_0, X_1, X_2, X_3\}$ の各変数をノードとして持ち, 目的変数を X_0 とする BNC の例を表す. 図 2 の左側の BNC は循環を含まず, 右側の BNC は変数 X_1, X_2, X_3 で循環を形成している. ここで, BNC は構造に循環を含まないが, 制約 (2) を説明するため, 構造に循環を含む例を示してい

る．また，$|C| \geq 2$ である各変数のクラスター C について, $F(C)$ を, C 内の変数のうち, 親変数と C の共通集合が空集合である変数の数とする．すなわち,

$$F(C) = \sum_{X \in C} \sum_{\mathbf{W}: \mathbf{W} \cap C = \emptyset} I(\mathbf{W} \to X) \tag{13}$$

である．以下, 例えば $\{X_0, X_1\}$ を X_0X_1 と略す．左側の BNC では $F(X_0X_1) = 1$, $F(X_0X_2) = 1$, $F(X_0X_3) = 2$, $F(X_1X_2) = 1$, $F(X_1X_3) = 2$, $F(X_2X_3) = 1$, $F(X_0X_1X_2) = 1$, $F(X_0X_1X_3) = 2$, $F(X_0X_2X_3) = 1$, $F(X_1X_2X_3) = 1$, $F(X_0X_1X_2X_3) = 1$ となり, 制約 (2) に違反しない．一方で, 右側の BNC では, $F(X_0X_1) = 1$, $F(X_0X_2) = 2$, $F(X_0X_3) = 2$, $F(X_1X_2) = 1$, $F(X_1X_3) = 1$, $F(X_2X_3) = 1$, $F(X_0X_1X_2) = 1$, $F(X_0X_1X_3) = 2$, $F(X_0X_2X_3) = 2$, $F(X_1X_2X_3) = 0$, $F(X_0X_1X_2X_3) = 1$ となる．右側の BNC では $C = \{X_1, X_2, X_3\}$ のとき, $F(X_1X_2X_3) = 0$ となるので, 制約 (2) に違反する．

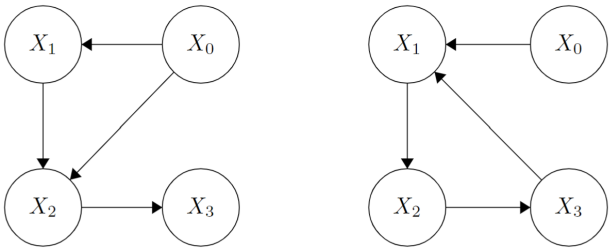

図 2　循環を含まない BNC の例 (左) と循環を含む BNC の例 (右)

また, 彼らは family variable $I(\mathbf{W} \to X)$ の 01 制約を次の通り定式化した．

$$\forall X, \mathbf{W} : I(\mathbf{W} \to X) \in \{0, 1\} \tag{14}$$

一方で, 提案手法では, 目的変数が親変数を持たない BNC 構造学習を行うための定式化を行う必要がある．そこで, 制約 (3) は, family variable を用いて, 次の通り定式化する．

$$I(\emptyset \to X_0) = 1 \tag{15}$$

制約 (1)〜(3), および, family variable の 01 制約は, 学習する構造が, 目的変数が親変数を持たない BNC であることを保証する．提案手法を用いることにより, 従来手法では最後まで構造学習できない規模の NCP 最小 I-map の学習を実現することができる．

提案する整数計画法の定式化は以下のようになる.

$$\text{(最大化)} \sum_{X,\mathbf{W}} Score_X(\mathbf{W}) I(\mathbf{W} \to X) - \gamma \times \sum_{X,\mathbf{W}} NCP_X(\mathbf{W}) I(\mathbf{W} \to X) \tag{16}$$

$$\text{(条件 1)} \ \forall X : \sum_{\mathbf{W}} I(\mathbf{W} \to X) = 1 \tag{17}$$

$$\text{(条件 2)} \ \forall C \subseteq \mathbf{V} : \sum_{X \in C} \sum_{\mathbf{W}:\mathbf{W} \cap C = \emptyset} I(\mathbf{W} \to X) \geq 1 \tag{18}$$

$$\text{(条件 3)} \ I(\emptyset \to X_0) = 1 \tag{19}$$

$$\text{(条件 4)} \ \forall X, \mathbf{W} : I(\mathbf{W} \to X) \in \{0, 1\} \tag{20}$$

各変数 $X \in \mathbf{V}$ の最大親変数数を d に制限したとき, X の親変数集合として X 以外の $n-1$ 個の変数から 0〜d 個選ぶので, X が子供となる family variable は $\sum_{i=0}^{d} \binom{n-1}{i}$ 個となる. よって, これらの制約および目的関数は, $n \sum_{i=0}^{d} \binom{n-1}{i}$ 個の family variable で表現される. この整数計画法における空間計算量は, family variable の数に比例する [25]. したがって, 提案手法の空間計算量は $\mathcal{O}(n \sum_{i=0}^{d} \binom{n-1}{i})$ となる. たとえば, 最大親変数数を 3 に制限したとき, 加藤ら [15] の手法では空間計算量は $\mathcal{O}(2^n)$ であるが, 提案手法の空間計算量は $\mathcal{O}(n^4)$ となる. したがって, 提案手法は最大親変数数を制限したとき, 加藤ら [15] の手法に比べてより大規模な構造学習を実現する.

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 小規模データセットを用いた評価実験

まず, 小規模データセットを用いて以下の 3 種類の手法の分類精度を比較する.

- Naive Bayes (Friedman ら [16])

- 幅優先探索 (菅原ら [4]): 幅優先探索を用いて, NCP 最小 I-map を探索する手法
- 深さ優先分枝限定法 (加藤ら [15]): 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 最小 I-map を探索する手法
- 整数計画法: 整数計画法により NCP 最小 I-map を探索する手法

本論では, 整数計画法を提案手法とする. ここで, 提案手法のペナルティ項に係るハイパーパラメータ γ については, 提案手法の ESS を 1 で固定した [26, 27] ときの実データを用いた評価実験において, $\{0.01, 0.02, 0.03, 0.04, 0.05, 0.06, 0.07, 0.08, 0.09, 0.1\}$ それぞれについて, 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, その平均値が最も高かった $\gamma = 0.05$ を用いた. 幅優先探索, 深さ優先分枝限定法については C++ で実装し, 整数計画法については Cussens ら [14] の Gobnilp を用いた. また, 3.7GHz の 20 コアプロセッサと 132GB 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 [28] 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 [4] 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. 幅優先探索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. 提案手法の ESS については, 3 番, 4 番, 6 番, 8 番, 13 番, 14 番, 20 番のデータセットについては 10 分割交差検証を用いて $\{1, 10, 100\}$ から定め, それ以外のデータセットについては 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 2 に示した. 表 2 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. また, 表中では, 幅優先探索, 深さ優先分枝限定法, 整数計画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 2　各手法の分類精度.

No.	Dataset	Variables	Sample size	SPP	Naive-Bayes	幅優先探索	深さ優先分枝限定法	整数計画法 $\gamma = 0.05$
1	Lymphography	19	148	1.63×10^{-7}	0.8378	0.7635(6.8)	0.7838(7.0)	0.7905(6.8)
2	Breast Cancer Wisconsin	10	683	3.42×10^{-7}	0.9751	0.9737(8.7)	0.9737(8.7)	0.9737(8.5)
3	Hepatitis	20	80	7.63×10^{-5}	0.8500	0.8250(2.1)	0.8000(2.0)	0.8250(4.2)
4	Zoo	17	101	1.03×10^{-4}	0.9802	0.9505(7.1)	0.9208(5.6)	0.9703(5.0)
5	Australian	15	690	2.97×10^{-4}	0.8290	0.8493(4.3)	0.8449(3.8)	0.8580(5.0)
6	Vehicle	19	846	8.07×10^{-4}	0.4314	0.6050(10.3)	0.5843(10.5)	0.5946(7.9)
7	Breast Cancer	10	277	8.33×10^{-4}	0.7364	0.7076(3.0)	0.7365(2.9)	0.7184(2.0)
8	Image Segmentation	19	2310	1.26×10^{-3}	0.7290	0.8264(14.9)	0.8320(15.0)	0.8264(13.0)
9	Congressional Voting Records	17	232	1.77×10^{-3}	0.9095	0.9698(2.9)	0.9655(2.9)	0.9612(4.7)
10	Heart	14	270	2.44×10^{-3}	0.8259	0.8222(5.3)	0.8333(5.8)	0.8222(6.5)
11	Solar Flare	11	1389	3.72×10^{-3}	0.7804	0.8431(0.0)	0.8409(0.4)	0.8398(2.2)
12	Wine	14	178	7.24×10^{-3}	0.9270	0.9494(6.8)	0.9494(6.8)	0.9494(6.4)
13	Letter	17	20000	1.17×10^{-2}	0.4466	0.6290(15.9)	0.6303(16.0)	0.6237(14.0)
14	Pendigits	17	10992	1.68×10^{-2}	0.8032	0.9368(16.0)	0.9373(16.0)	0.9314(16.0)
15	Contraceptive Method Choice	10	1473	5.99×10^{-2}	0.4671	0.4616(2.7)	0.4396(2.5)	0.4742(3.1)
16	Glass	10	214	6.97×10^{-2}	0.5514	0.5794(6.4)	0.6036(7.2)	0.6122(6.3)
17	Hayes-Roth	5	132	2.29×10^{-1}	0.8333	0.8333(3.0)	0.8333(3.0)	0.8333(3.0)
18	Balance Scale	5	625	3.33×10^{-1}	0.9152	0.9152(4.0)	0.9152(4.0)	0.9152(4.0)
19	Lenses	5	24	3.33×10^{-1}	0.7083	0.8750(2.0)	0.8750(2.0)	0.8750(2.0)
20	EEG	15	14980	4.57×10^{-1}	0.5778	0.7155(12.3)	0.7135(12.4)	0.7115(11.8)
21	LED7	8	3200	2.50×10^{0}	0.7294	0.7316(7.0)	0.7325(7.0)	0.7275(7.0)
22	Iris	5	150	3.13×10^{0}	0.7133	0.8200(3.1)	0.8200(3.1)	0.8133(3.4)
23	HTRU2	9	17898	3.50×10^{1}	0.8966	0.9140(7.6)	0.9140(7.5)	0.9141(7.2)
24	Banknote authentication	5	1372	4.29×10^{1}	0.8433	0.8819(2.0)	0.8819(2.0)	0.8812(2.0)
	average				0.7624	0.8070(6.4)	0.8067(6.4)	0.8101(6.3)

表 2 より, 整数計画法の平均分類精度は探索グラフを用いた最短パス探索を行う幅優先探索や深さ優先分枝限定法の平均分類精度とほぼ同等であることが確認できる. この結果は, 提案する整数計画法の最適化が最短パス探索により, 厳密に NCP 最小 I-map を学習する場合に比べて分類精度を低下させないことを示している. ただ, 9, 10, 13, 20 番のデータセットでは, 整数計画法の分類精度が幅優先探索および深さ優先分枝限定法の分類精度を下回っている. このうち, 13, 20 番について, 整数計画法は幅優先探索や深さ優先探索に比べて目的変数の子変数を過小に学習していることが分かる. これらのデータセットでは, 整数計画法における NCP に関するペナルティ項の係数 γ の値が適切な値より大きく, NCP に関するペナルティが大きくなりすぎたと考えられる. その結果, 分類上重要な変数が削除されることで, 分類精度が低下したと考えられる. また, 9, 10 番について, 整数計画法は幅優先探索や深さ優先探索に比べて目的変数の子変数を過大に学習していることが分かる. これらのデータセットでは, 整数計画法における NCP に関するペナルティ項

の係数 γ の値が適切な値より小さく, NCP に関するペナルティが小さくなりすぎたと考えられる. その結果, 分類に関係ない変数が追加され, 分類精度が低下したと考えられる.

次に, 整数計画法の NCP と幅優先探索および深さ優先分枝限定法の NCP を比較するために, 各データセットに対して NCP を推定した. その結果を表 3 に示す.

表 3　各手法によって学習された構造の平均 NCP.

No.	Variables	Sample size	SPP	幅優先探索	深さ優先分枝限定法	整数計画法 $\gamma = 0.05$
1	19	148	1.63×10^{-7}	104	120	123
2	10	683	3.42×10^{-7}	159	158	152
3	20	80	7.63×10^{-5}	10	9	27
4	17	101	1.03×10^{-4}	508	131	110
5	15	690	2.97×10^{-4}	64	60	84
6	19	846	8.07×10^{-4}	1377	1380	123
7	10	277	8.33×10^{-4}	62	37	12
8	19	2310	1.26×10^{-3}	4324	5551	749
9	17	232	1.77×10^{-3}	10	9	45
10	14	270	2.44×10^{-3}	18	22	28
11	11	1389	3.72×10^{-3}	8	12	417
12	14	178	7.24×10^{-3}	28	28	28
13	17	20000	1.17×10^{-2}	12336	12339	7045
14	17	10992	1.68×10^{-2}	9175	9886	3327
15	10	1473	5.99×10^{-2}	37	28	52
16	10	214	6.97×10^{-2}	483	655	266
17	5	132	2.29×10^{-1}	29	29	29
18	5	625	3.33×10^{-1}	50	50	50
19	5	24	3.33×10^{-1}	8	8	8
20	15	14980	4.57×10^{-1}	1849	1849	881
21	8	3200	2.50×10^{0}	94	98	206
22	5	150	3.13×10^{0}	19	19	25
23	9	17898	3.50×10^{1}	198	176	93
24	5	1372	4.29×10^{1}	15	15	25

9, 10 番のデータセットでは, 整数計画法は既存手法よりも平均 NCP 数が大きくなっていることが分かる. この結果から, これらのデータセットでは, NCP に関するペナルティが小さくなり過ぎていることが分かる. また, 13, 20 番のデータセットでは, 既存手法と比較して平均 NCP 数が小さくなっていることが分かる. この結果から, これらのデータセットでは, NCP に関するペナルティが大きくなり過ぎていることが分かる.

4.2 大規模データセットを用いた評価実験

この実験では, 整数計画法を用いることで従来手法より大規模な構造学習が可能になることを示す.

本節では, Naive Bayes, 幅優先探索, 深さ優先分枝限定法, 整数計画法の分類精度を比較する. 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 [26, 27], 最大親変数を 3 に制限した. また, 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. その他の条件は前節と同様である.

表 4　各手法の分類精度.

No.	Dataset	Variables	Sample size	Naive-Bayes	幅優先探索	深さ優先分枝限定法	整数計画法 $\gamma = 0.05$
25	Phishing	31	11055	0.9276	TO	0.9337*	0.9433
26	Postures	31	23906	0.8290	TO	0.8727*	0.8760
27	connect-4	43	67557	0.7058	TO	0.6751*	0.7138
28	PAMAP2	53	174915	0.6862	TO	0.8266*	0.8430
29	spam	58	4601	0.8794	TO	0.9063*	0.9107

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. また, 表 4 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.

表 4 より, 整数計画法は全てのデータセットで最後まで構造学習を行うことができたことが分かる. 一方, 幅優先探索は全てのデータセットで制限時間内に学習を終えることができなかった. また, 深さ優先分枝限定法は全てのデータセットでメモリオーバにより学習が打ち切られた. さらに, 整数計画法の分類精度は全てのデータセットで Naive Bayes および深さ優先分枝限定法の分類精度を上回っていることが確認できる.

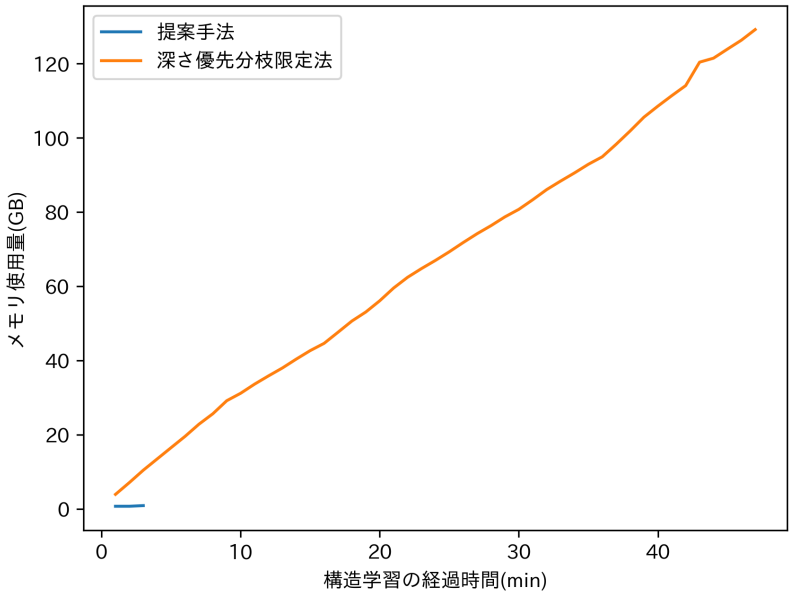

図 3　25 番の構造学習中のメモリ使用量

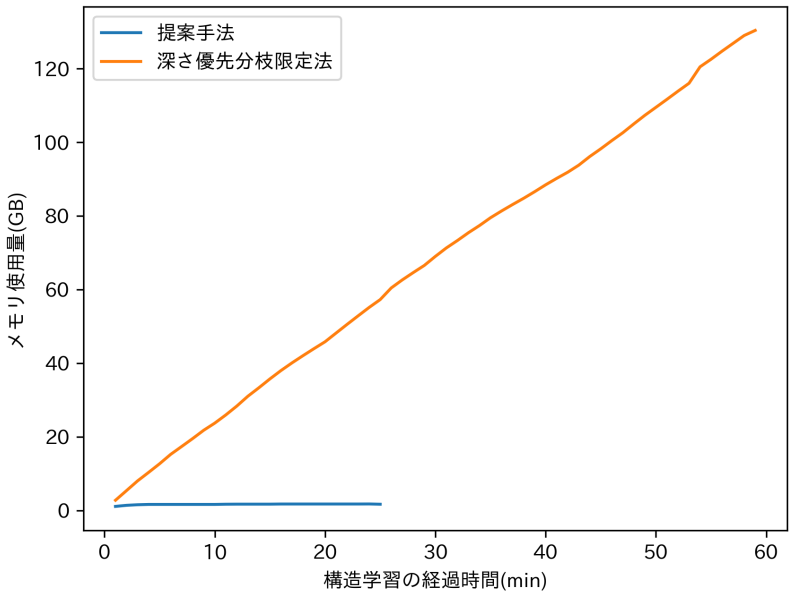

図 4　26 番の構造学習中のメモリ使用量

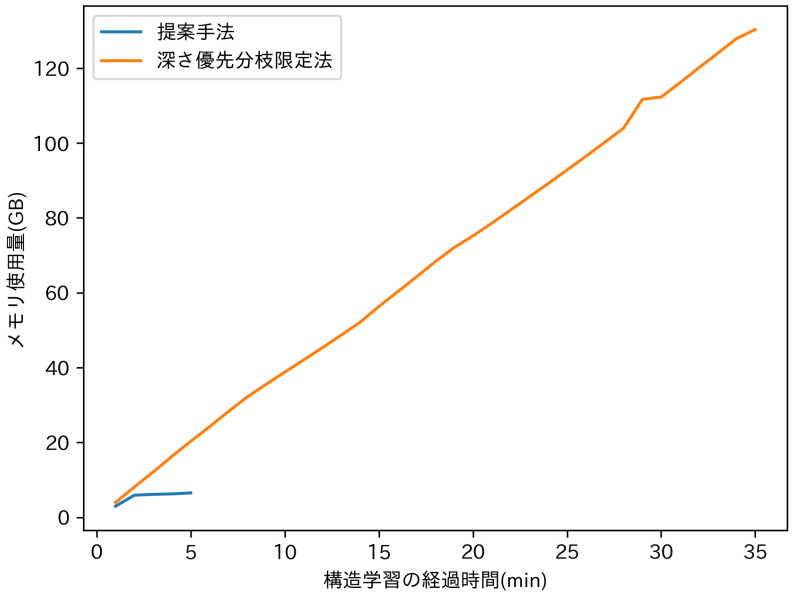

図 5　27 番の構造学習中のメモリ使用量

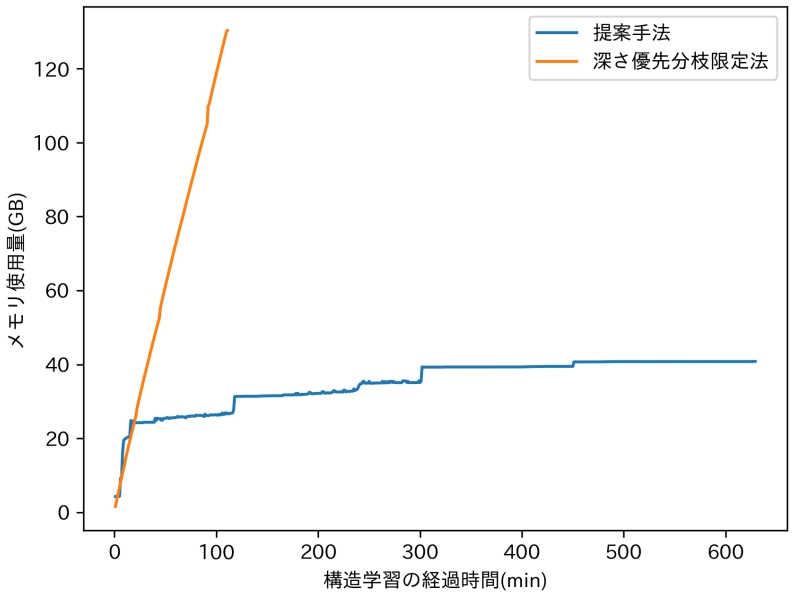

図 6　28 番の構造学習中のメモリ使用量

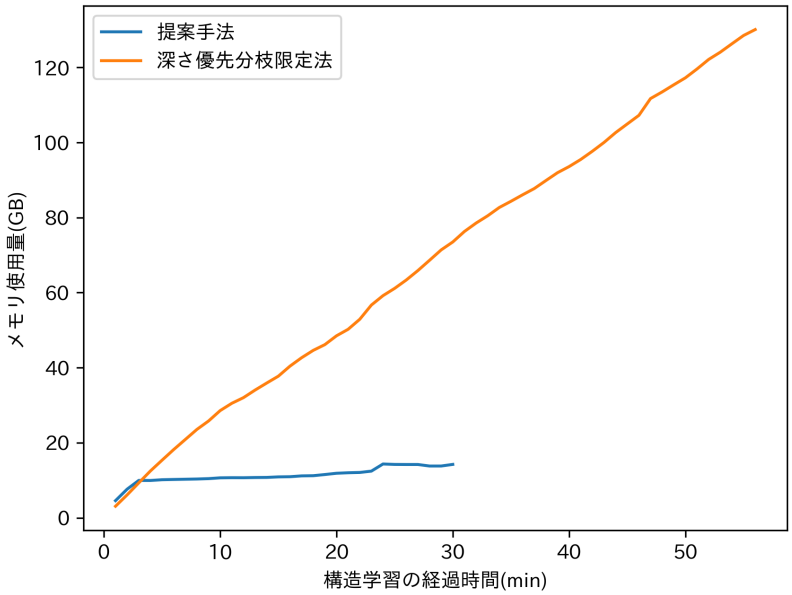

図 7　29 番の構造学習中のメモリ使用量

次に, 整数計画法が深さ優先分枝限定法に比べてどれだけメモリ使用量を削減するのかを確認するために, 各データセットに対して 整数計画法と深さ優先分枝限定法のメモリ使用量の時間変化を測定した. その結果を図 3〜7 に示す. 図 3〜7 より, 全てのデータセットで整数計画法は深さ優先分枝限定法と比較して, 構造学習中のメモリ使用量が少ないことが分かる. また, 28 番のデータセットを除いた全てのデータセットで整数計画法は, 深さ優先分枝限定法で構造学習が打ち切られるよりも早く構造学習を終えることができた.

以上の結果より, 提案手法は従来の幅優先探索では実現できなかった規模の構造学習を実現でき, Naive Bayes や深さ優先分枝限定法で学習した構造よりも高い分類精度をもつ構造を学習できることが示された.

5　むすび

本論では, 整数計画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)NCP を最小にして真の分類確率に漸近収束する構造を学習するための目的関数と, (2) 目的変数が親変数を持たないベイジアンネットワーク分類器を学習するための制約を導入した.

従来手法である加藤ら [15] の手法では, 最大親変数数の制限によらず, $\mathcal{O}(2^n)$ の空間計算量が必要であった. 一方, 提案手法は, 最大親変数数の制限により, 空間計算量を減じることができる. 具体的には, 提案手法は, 最大親変数数を d に制限したとき, $\mathcal{O}(n\sum_{i=0}^{d}\binom{n-1}{i})$ の空間計算量で計算できる. 提案手法は, 構造学習中に使用されるメモリ使用量が少ないため, 従来手法においてメモリオーバにより打ち切られる構造学習を最後まで行うことができる.

複数のベンチマークを用いた実験により, 加藤ら [15] の手法では 30 変数程度の構造学習で打ち切りが生じたが, 提案手法では 58 変数の構造学習を最後まで学習できることを示した. また, 加藤ら [15] の手法でメモリ不足により構造学習を途中で打ち切った 30 変数以上の構造について, 提案手法により最後まで構造学習を行い, 分類精度を改善できることを示した.

今後の課題として, より大規模な変数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ことが挙げられる.

参考文献

[1] P.A. Aguilera, A. Fernández, R. Fernández, R. Rumí, and A. Salmerón. Bayesian networks in environmental modelling. *Environmental Modelling & Software*, 26(12):1376–1388, 2011.

[2] Bruce G. Marcot and Trent D. Penman. Advances in bayesian network modelling: Integration of modelling technologies. *Environmental Modelling & Software*, 111:386–393, 2019.

[3] Bart Baesens, Geert Verstraeten, Dirk Van den Poel, Michael Egmont-Petersen, Patrick Van Kenhove, and Jan Vanthienen. Bayesian network classifiers for identifying the slope of the customer lifecycle of long-life customers. *European Journal of Operational Research*, 156(2):508–523, 2004.

[4] 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.

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

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

[7] Ajit Singh and Andrew Moore. Finding optimal Bayesian networks by dynamic pro-

gramming. Technical report, Carnegie Mellon University, pp.1–16, June 2005.

[8] 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.

[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] 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.

[11] 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.

[12] Joe Suzuki and Jun Kawahara. Branch and bound for regular bayesian network structure learing. In *UAI*, pages 212–221, 2017.

[13] Joe Suzuki. A theoretical analysis of the BDeu scores in bayesian network structure learning. *Behaviormetrika*, 44:97–116, 2017.

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

[15] 加藤弘也. 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. 電気通信大学 卒業論文, 2023.

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

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

[18] 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.

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

[20] Shouta Sugahara, Wakaba Kishida, Koya Kato, and Maomi Ueno. Recursive auton-

omy identification-based learning of augmented naive bayes classifiers. In *International Conference on Probabilistic Graphical Models*, pages 265–276. PMLR, 2022.

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

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

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

[24] 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.

[25] 今井正治, 吉田雄二, and 福村晃夫. 分枝限定アルゴリズムの並列化とその評価. 電子情報通信学会論文誌 *D*, 62(6):403–410, 1979.

[26] 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.

[27] Maomi Ueno. Robust learning Bayesian networks for prior belief. In *Proceedings of Uncertainty in Artificial Intelligence (UAI)*, pages 689–707. AUAI Press, 2011.

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

Abstract Sugahara and Ueno(2022) proposed a Bayesian Network Classifier(BNC) with the minimum number of the class variable parameters (NCP) and reported that it had higher classification accuracy than any other BNCs did. However, their method using a breadth-first search cannot learn BNCs with more than 20 variables.

Kato, Sugahara, and Ueno(2023) proposed a new algorithm using a depth-first search with a pruning. Although the computational complexity of their method is the same as Sugahara and Ueno(2022), the search space is pruned and reduced. They showed that their method reduced the runtime of Sugahara and Ueno(2022) and found the best solution so far even the search stopped early due to lack of memory. They reported that their method could learn a BNC with about 60 variables which Sugahara and Ueno(2022) could not learn. However, even though their method limits the maximum number of parent variables of each variable, learning of BNCs with about 30 variables stops early due to lack of memory. Therefore, the classification accuracy that the structure has may be reduced.

Kato, Sugahara, and Ueno(2023) has space computational complexity of $\mathcal{O}(2^n)$. In this paper, we propose a new algorithm that can reduce space computational complexity using integer programming by limiting the maximum number of parent variables of each variable. We consider the formulation of the integer programming problem that searchs for the structures with the smallest NCP among structures that can express true joint probability distribution. When the number of parent variables of each variable is less than d, the number of variables of this integer programming problem is $n\sum_{i=0}^{d}\binom{n-1}{i}$. Since space computational complexity of the integer programming problem depends on the number of variables, space computational complexity of the proposed method is $\mathcal{O}(n\sum_{i=0}^{d}\binom{n-1}{i})$. Therefore, the proposed method can learn BNCs, that Kato, Sugahara, and Ueno(2023) stops early due to lack of memory, until the end.

In this paper, we conduct comparative experiments using multiple benchmarks and show that the proposed method can improve the classification accuracy by learning the large-scale structures, which Kato, Sugahara, and Ueno(2023) stopped early due to lack of memory, until the end.