Vol.2018-MPS-121 No.17
2018/12/18

Canonical Correlation Forestsにおける スパースカテゴリ行列を考量した分類法に関する一考察

中野 修平 [1,a)]　三川 健太[2]　後藤 正幸[1]

概要：自動分類問題とはカテゴリが既知の学習データから分類規則を学習し，新規の入力データのカテゴリを推定する問題である．自動分類問題へのアプローチの一つとして，決定木のアンサンブルにより高精度な分類を可能とする Random Forest がある．本研究では Random Forest を改良した Canonical Correlation Forests (CCFs) に着目する．CCFs は各ノードで正準相関分析を行い，正準変数を得ることで，柔軟な識別境界を実現し Random Forest より高度な予測を可能とする．しかし，CCFs は one-hot カテゴリ行列がスパースな場合，分類に適した正準変数を得られない場合がある．そこで，本研究ではノードごとで分類に適したカテゴリ表現の最適化法を提案し，その有効性についてベンチマークデータを用い検証する．

キーワード：機械学習，canonical correlation forest，決定木，アンサンブル法

1. はじめに

カテゴリが既知の学習データから分類規則を学習することで，新たな入力データのカテゴリを予測する自動分類手法は広い適用範囲を持つことから，盛んに研究が行われている．自動分類問題への有効なアプローチの一つに複数の弱学習器を学習し，それらの多数決により最終的な分類先を決定するアンサンブル手法が知られている．データから一つの分類器を学習させる一般的なアプローチとは異なり，アンサンブル手法は複数の分類器を学習させ，その組み合わせによって分類精度向上を目指す. 機械学習の分野において，Random Forests (RFs) [1] のような決定木をアンサンブルする手法は自動分類に対して効果的であることが知られている．また，RF をさらに発展させた手法として Canonical Correlation Forests (CCFs) [2] がある．CCFs は 従来の座標軸に対して平行な境界面を作る決定木とは異なり，座標軸にとらわられない境界面を作成することを目指した Oblique Decision Tree (ODT) [3] の考えに基づいた手法である．一般的な RFs とは異なり，CCFs は正準変数上で分割することにより，各葉ノードにおいてより柔軟な境界面を構築することが可能である. さらに，CCFs は超平面の作成に正準相関分析 [5] を用いることによってカテゴリ情報を考慮した超平面を構築することができることが知られている．

CCFs における個々の決定木 (以下，CC-Tree) は説明変数行列とカテゴリ行列から計算された正準変数で構成される超平面上で分岐点を探索する．その結果，CCFs はより柔軟な境界面を作成できる．各ノードごとに一変数に着目して分岐点を探索する一般的な決定木とは異なり，CCFs は各ノードごとに合成変数に着目して分岐点を探索するので CCFs の境界面は座標軸に平行という制約がない．しかしながら，学習データ数が少ない場合，CCFs は各ノードごとに分類に適切な正準変数を得ることができない．これは，CCFs が各ノードにおいて全てのカテゴリを等価に扱うため，カテゴリ数に対してデータの数が十分ではなく過学習しやすくなるためである．

そこで本研究では，ノード内にカテゴリ数に対してデータ数が十分でない場合，過学習をすることを避けるため，CCA を枠組みを考慮した 2 つの行列間の線形写像後の相関が最大となるようなカテゴリ行列を最適化するアルゴリズムを提案する．また，評価実験により提案手法の有効性を示す．

2. 準備

2.1 決定木のアンサンブル法

決定木は枝と葉からなる木構造をした分類モデルである．一般的な決定木のアルゴリズムの共通の考えは，1 つ

[1] 早稲田大学
Waseda University. , Shinjyuku, Tokyo 169–0072, Japan
[2] 湘南工科大学
Shonan Institute of Technology, 1-1-25, Tsujidounishikaigan, Fu-jisawa city, 169-8555, Japan.
[a)] naknao-aioi@suou.waseda.jp

の変数に着目して分割点を探索し，ジニ係数やエントロピーなどある基準を用いて最も有効な分割点を決めることである．決定木の代表的なアルゴリズムとして CART [6] と C4.5 [7] が知られている．どちらのアルゴリズムも分岐点を求めた後，学習データを子ノードへ分岐させる．この処理はノードを分割しても情報利得が得られないか，あるいはノードが同じカテゴリデータの集合になるか，終了条件を満たすまで再帰的に繰り返される．これらの木を組み合わせ，森として扱うことで分類精度が向上することが知られている．

分類精度の向上を目的としたアンサンブル手法の研究が発展した一方で，決定木自体の分類精度向上を目的とした研究も進められていた．その中の一つが，従来の決定木の拡張である Oblique Decision Tree である．Oblique Decision Tree は，合成変数を用いることで，座標軸にとらわれない境界面を構築する手法である．Rotation Forest [8] は Principal Component Analysis (PCA) を説明変数に適用し作成した合成変数を用いることで高い分類精度を達成することが可能となる．Rotation Forest が説明変数のみに着目して合成変数を得ている一方で，CCFs は説明変数とカテゴリ行列を考慮しており分類に適した合成変数を得られることが知られている．

2.2 ノーテーション

本研究で扱う自動分類問題とは，D 次元特徴空間上の特徴ベクトル $\boldsymbol{x} \in \mathbb{R}^D$ から K 個の離散カテゴリ集合 $\mathcal{C} = \{c_1, \dots c_K\}$ への写像を得ることである．そのために，カテゴリが既知である N 個の学習データ $\{(\boldsymbol{x}_n, y_n)\}_{n=1}^N$ を用いて，この写像 (分類器) を学習することを考える．この学習によって得られた分類器を用いて，カテゴリが未知の新規入力データ $\boldsymbol{x}$ の所属カテゴリを推定することができる．ここで，$\boldsymbol{x}_n \in \mathbb{R}^D$ は n 番目の学習データの D 次元の特徴ベクトル，$y_n \in \mathcal{C}$ は n 番目の学習データの $\boldsymbol{x}_n$ の所属カテゴリ，$y_n \in \mathcal{C}$ は n 番目の学習データの $\boldsymbol{x}_n$ の所属カテゴリである．ここで，$\boldsymbol{X} = (\boldsymbol{x}_1, \boldsymbol{x}_2 \dots \boldsymbol{x}_N)^{\mathrm{T}}$ をカテゴリが既知である特徴量行列，$\boldsymbol{y} = (y_1, y_2 \dots y_N)^{\mathrm{T}}$ をカテゴリベクトル，$\mathcal{T} = \{t_i\}_{i=1}^L$ を L 個の木からなる木集合と定義しておく．

$t_i = (\boldsymbol{\Psi}_i, \boldsymbol{\Theta}_i)$ は中間ノード集合 $\boldsymbol{\Psi} = \{\psi_j\}_{j \in \mathcal{J} \setminus \partial\mathcal{J}}$，葉ノード集合 $\boldsymbol{\Theta} = \{\theta_j\}_{j \in \mathcal{J}}$ からなる．ただし，$\mathcal{J}$ はノード番号集合，$\partial\mathcal{J} \subseteq \mathcal{J}$ は葉ノードの部分集合，$\setminus$ は差集合であるとする．また，各中間ノードは $\psi_j = \{\chi_{j1}, \chi_{j2}, \boldsymbol{\phi}_j, s_j\}$ によって定義される．ただし $\{\chi_{j1}, \chi_{j2}\} \subseteq \mathcal{J} \setminus j$ はノード j からの二つの子ノードであり $\boldsymbol{\phi}_j$ はノードにおける特徴ベクトルへの重みベクトル，s_j はノード j における写像空間 $\boldsymbol{X}^{\mathrm{T}}\boldsymbol{\phi}_j$ における分割点であるとする．また，$B(j, t_i)$ は木におけるノード j の部分特徴空間であることを示す．例えば $B(0, t_i)$ は木におけるルートノード，$B(j, t_i)$ は木おけるノードを意味する．したがって，$B(j, t_i) = B(\chi_{j1}, t_i) \cup B(\chi_{j2}, t_i)$ の関係が成り立つ．$B(j, t_i)$ の二つの子ノード $B(\chi_{j1}, t_i)$，$B(\chi_{j2}, t_i)$ は以下の式 (1) – (2) に従う．ここで，$\boldsymbol{z} \in \mathbb{R}^D$ はノードに所属するデータにおける任意の特徴ベクトルであるとする．

$$B(\chi_{j1}, t_i) = B(j, t_i) \cap \left\{\boldsymbol{z}^{\mathrm{T}}\boldsymbol{\phi}_j \leq s_j\right\} \tag{1}$$

$$B(\chi_{j2}, t_i) = B(j, t_i) \cap \left\{\boldsymbol{z}^{\mathrm{T}}\boldsymbol{\phi}_j > s_j\right\} \tag{2}$$

2.3 Canonical Correlation Analysis

Canonical Correlation Analysis (CCA) は二つの行列の線形写像後の相関を最大にするベクトルを得る手法である．この手法は二つの任意の行列 $\boldsymbol{W} \in \mathbb{R}^{N \times D}$，$\boldsymbol{V} \in \mathbb{R}^{N \times K}$ の関係を分析するために用いられる．ここで N はデータ数，D は次元数，K はカテゴリ数であることを示す．また，$\boldsymbol{V}$ は各行に所属するカテゴリに対して 1，それ以外は 0 をとる 1-of-K 表現を持つ行列とする．いま，説明変数行列 $\boldsymbol{W}$，カテゴリ行列 $\boldsymbol{V}$，またそれぞれの行列に対する重みベクトル $\boldsymbol{a} \in \mathbb{R}^D$，$\boldsymbol{b} \in \mathbb{R}^K$ を考える．したがって，CCA の主な目的は $\boldsymbol{W}\boldsymbol{a}$ と $\boldsymbol{V}\boldsymbol{b}$ 間の相関を最大にする二つの任意のベクトル $\boldsymbol{a}$，$\boldsymbol{b}$ を抽出することである．ここで，$corr()$ は二つのベクトル間の相関を計算する関数であるとすると，CCA は以下の最適化問題を解くことにより導出される．

$$\underset{\boldsymbol{a},\boldsymbol{b}}{\text{argmax}} \quad corr(\boldsymbol{X}\boldsymbol{a}, \boldsymbol{Y}\boldsymbol{b}) \tag{3}$$

$$\text{subject to} \quad ||\boldsymbol{a}||_2 = 1 \tag{4}$$

$$||\boldsymbol{b}||_2 = 1 \tag{5}$$

3. Canonical Correlation Forests

3.1 概要

RF や多くの決定木のアンサンブル法のように，CCFs の木は独立して学習される．学習アルゴリズムはすべてのデータが所属するルートノードから始まる．式 (6) – (7) で定義されるように，各ノード $B(j, t_i)$ で分割点 s_j を決定し，その閾値に対する大小で子ノードへのデータを所属させる．この処理をすべてのノードが終了条件を満たすか，ノードに所属するデータが一つのカテゴリになるまで再帰的に繰り返す．RF のような一般的な決定木アルゴリズムとの違いは CCFs は各ノード j において特徴ベクトルとカテゴリを考慮した写像空間 $\boldsymbol{X}\boldsymbol{a}_j$ で分割点 s_j を探索することである．ここで $\boldsymbol{X} \in \mathbb{R}^{N \times D}$ は特徴行列，$\boldsymbol{Y} \in \mathbb{R}^{N \times K}$ はカテゴリベクトルに対して 1-of-K 表現を適用したカテゴリ行列，$\boldsymbol{a}_j$ はノード j において式 (3) – (5) を解いて得られる D 次元のベクトルとする．

$$B(\chi_{j1}, t_i) = B(j, t_i) \cap \left\{\boldsymbol{z}^{\mathrm{T}}\boldsymbol{a}_j \leq s_j\right\} \tag{6}$$

$$B(\chi_{j2}, t_i) = B(j, t_i) \cap \left\{\boldsymbol{z}^{\mathrm{T}}\boldsymbol{a}_j > s_j\right\} \tag{7}$$

3.2 CCFs のアルゴリズム

CCFs の学習アルゴリズムでは一般的な RF と同じくデータセット $(\boldsymbol{X},\boldsymbol{Y})$ からブートストラップサンプルされたデータセットを対象に個々の木が並列に学習を行う．CCFs のアルゴリズムでは各ノードごとに CCA を行い，重みベクトルを導出する．次に，合成変数上で，最適な分割点を探索する．その決定した分割点に基づく分割境界は正準変数上では座標軸に対して平行ではあるが，元の特徴量空間では座標軸に対して平行ではなくなる．これらの合成変数上で分割境界を決定し，さらにそれらの分割境界をアンサンブルすることで CCFs は全体の分割境界の決定を行う．その結果，CCFs は柔軟な分割境界を構築することが可能である．CCFs の学習アルゴリズムを以下に示す．

Step1) $l=0$ とする．
Step2) $j=0$ とする．
Step3) $(\boldsymbol{X},\boldsymbol{Y})$ からブートストラップサンプルを行いデータセット $B(\chi_j,t_l)$ を作成する．
Step4) $B(\chi_j,t_l)$ から D 個の変数からランダムに d 個の変数をランダムに選択する．(ただし $d<D$)
Step5) CCA を行い $\boldsymbol{a}_j$ を求める．
Step6) ジニ係数を基準に最適な分割点 s_j を探索する．
Step7) 式 (6) – (7) を用いて二つの子ノードを計算する．
Step8) 式 (8) – (9) に従い，$B(\chi_{j+1},t_i), B(\chi_{j+2},t_i)$ を更新する．

$$B(\chi_{j+1},t_i) \leftarrow B(\chi_{j1},t_i) \tag{8}$$
$$B(\chi_{j+2},t_i) \leftarrow B(\chi_{j2},t_i) \tag{9}$$

Step9) $j=j+1$ とし，終了条件を満たすまで Step4) から Step8) を繰り返す．
Step10) $l=l+1$ とし，Step2) から Step9) を L 回繰り返す．

4. 提案手法

4.1 概要

CCFs は新しい決定木のアンサンブル手法である. また，CCFs の中の木モデルである CC-Tree は CCA を用いることにより，正準変数上で分割点を探索するモデルである．各ノードで正準変数を求め，それに対する分割境界を決定し，アンサンブルを行うことで，CCFs は柔軟な境界面を構築することができ，高い分類精度が得られることが知られている．しかしながら，学習データ数が少ない場合，CCFs は各ノードごとに分類に適切な正準変数を得ることができない．これは，決定木のような木構造を用いた学習アルゴリズムはノードの深さが大きくなるごとにノードに所属するデータ数が減少していくが，CCFs は各ノードで全てのカテゴリを等価に扱うため，カテゴリ数に対してデータ数が十分でなくなってしまい，式 (3) – (5) によって推定される $\boldsymbol{b}$ が過学習をしやすくなるためである．すなわち，求めるパラメータ数に対してデータ数が不足しやすくなる．また，CCA を用いた写像後の空間では，同じカテゴリのデータは近くに配置されるように写像される．しかし，カテゴリ数が多くなると写像された空間が複雑となり，ある正準変数における閾値のみで分類を可能とするような問題ではなくなる可能性がある．

CCFs が多カテゴリ問題に対して脆弱性を持つ一つの理由は CCA は写像する際に $\boldsymbol{Yb}$ の計算を用いられていることが挙げられる．$\boldsymbol{Y}$ はカテゴリ行列であるため，各行に対してほとんどが 0 の値をもつスパースな行列になっている．そのような行列に対して，$\boldsymbol{b}$ の次元数が大きい場合に式 (3) – (5) の計算を行うと $\boldsymbol{b}$ の推定が過学習する可能性が挙げられる．ここで，本研究ではパラメータ数削減と適切な分割点を得るため CCA を行なった後，カテゴリ行列 $\boldsymbol{Y}$ に対して最適化を行うことを考える (式 (10) – (12))．しかし，直接カテゴリ行列 $\boldsymbol{Y}$ の最適化を行うと $\boldsymbol{Y}$ の全ての要素に対して $\{0,1\}$ を与えるような組み合わせ問題となり $O(N2^K)$ の計算量が必要となる．計算量がデータ数 N に依存する形となり，現実的に実行不可能となるため，計算量削減のため本研究では $\boldsymbol{Y}$ に対する最適化をカテゴリの組み合わせ問題として扱う．以上の議論より，カテゴリ数の多い分類問題を対象に，新しい CCFs のアルゴリズムを提案する．このため，Exhaustive-Correcting Output Code (以下，ECOC) [9] の考えを CCFs へ導入する．ECOC とは 2 値分類器を多値分類問題へと拡張させるためカテゴリの組み合わせ示す符号表である．ECOC に基づくカテゴリの組み合わせに従い，最も式 (10) の値が高くなるような $\boldsymbol{Y}$ を導出する．そうすることで，CCA は分割に適した写像を行うことができる．ここで，$\boldsymbol{X}^{(j)}$，$\boldsymbol{Y}^{(j)}$，$\boldsymbol{a}^{(j)}$，$\boldsymbol{b}^{(j)}$ はそれぞれノード j に対する特徴行列，カテゴリ行列，$\boldsymbol{X}^{(j)}$ に対する重みベクトル，$\boldsymbol{Y}^{(j)}$ に対する重みベクトルし，$\boldsymbol{y}_i$ は i 番目のデータの 1-of-K 表現したカテゴリベクトルである，y_{im} は $\boldsymbol{y}_i$ の m 番目の要素を示す．

$$\underset{\boldsymbol{Y}'^{(j)}}{\text{argmax}} \quad corr(\boldsymbol{X}^{(j)}\boldsymbol{a}^{(j)}, \boldsymbol{Y}^{(j)}\boldsymbol{b}^{(j)}) \tag{10}$$
$$\text{subject to} \quad y_{im} \in \{0,1\} \tag{11}$$
$$||\boldsymbol{y}_i||_2 = 1 \tag{12}$$

4.2 Exhaustive-Correcting Output Code

ECOC は符号表を用いて多値分類問題を行う手法であり，2 値分類器を多分類問題へと拡張させるために提案された．表 1 のように各要素は {0,1} で構成されており，各要素分類器は与えられた 1 と 0 のカテゴリを判別する 2 値分類問題として学習を行う．また，Exhaustive 符号はカテゴリ数 K に対して $2^{K-1}-1$ 個の考えられる全ての 2 値分

類に対する判別器構成となっている．新たなデータを分類する場合，各 2 値分類器の出力結果と符号表とのハミング距離が最も近いカテゴリに分類する手法である．

ここで式 (10) – (12) のような組み合わせ最適を各ノードに対して適用すると計算量は $O(N2^K)$ となる．そこで，本研究では式 (10) – (12) のカテゴリ組み合わせ問題を Exhaustive 符号を用いることで計算量を $O(2^{K-1}-1)$ まで削減を試みる．各ノード間で最適な Exhaustive 符号のカテゴリの組み合わせを探索的に行うことにより，相関が最も高くなるようなカテゴリ行列 $\boldsymbol{Y}'^{(j)}$ を求めることで $\boldsymbol{Y}$ の最適化を行う．

表 1　Exhaustive 符号法（$K=4$）
Table 1　Code word table of Exhaustive method ($K=4$).

C_1	1	1	1	1	1	1	1
C_2	0	0	0	0	1	1	1
C_3	0	0	1	1	0	0	1
C_4	0	0	1	1	0	0	1
C_5	0	1	0	1	0	1	0

4.3　提案アルゴリズム

提案アルゴリズムはノードごとに CCA を行い特徴量行列への重みベクトル $\boldsymbol{a}$，カテゴリ行列への重みベクトル $\boldsymbol{b}$ を導出した後，$\boldsymbol{a}$,$\boldsymbol{b}$ が与えれたものとして最適なカテゴリ行列 $\boldsymbol{Y}$ の最適化を行う．各ノードで CCA を適用した後，ECOC に基づくカテゴリの組み合わせに従い，カテゴリ行列 $\boldsymbol{Y}$ を変換させ，式 (10) で定義される相関が最も高くなったカテゴリの組み合わせを式 (10) – (12) の解とする．その後，CCA によって得られた正準変数と最適化されたカテゴリ行列 $\boldsymbol{Y}$ を用いて，ジニ係数を基準に最適な分割点を探索する．分割点の決定後，式 (6) – (7) に従いデータを子ノードへの所属させる．ただし，一度同じカテゴリとして統合されたカテゴリは子ノードでは再び別カテゴリとされ，再度 Exhaustive 符号に基づき最適なカテゴリごとの組み合わせを求める．これらの処理を終了条件を満たすか，ノードが同じカテゴリデータの集合になるか，終了条件を満たすまで各子ノードにおいて再帰的に繰り返される．

Step1)　$l=0$ とする．
Step2)　$j=0$ とする．
Step3)　$(\boldsymbol{X},\boldsymbol{Y})$ からブートストラップサンプルを行いデータセット $B(\chi_j,t_l)$ を作成する．
Step4)　$B(\chi_j,t_l)$ から D 個の変数からランダムに d 個の変数をランダムに選択する．(ただし $d<D$)
Step5)　CCA を行い $\boldsymbol{a}_j$ を求める．
Step6)　Exhaustive 符号に基づき最適な $\boldsymbol{Y}$ を求める．
Step7)　ジニ係数を基準に最適な分割点を探索する．
Step8)　式 (6) – (7) を用いて二つの子ノードを計算する．
Step9)　式 (8) – (9) に従い，$B(\chi_{j+1},t_i), B(\chi_{j+2},t_i)$ を更新する．
Step10)　$j=j+1$ とし，終了条件を満たすまで Step4) から Step9) を繰り返す．
Step11)　$l=l+1$ とし，Step2) から Step10) を L 回繰り返す．

5.　評価実験

5.1　UCI データセットにおける実験条件

提案手法の有効性を示すため，表 2 で表される UCI データセット (balance scale, nursery, optDigitsHandwritten) [10] を用いて分類実験を行なった．比較手法として RF，Rotation Forest，CCFs を用いた，ならびにアンサンブルを行わない Rotation Forest の木単体モデル (以下，PC-Tree) と Canonical Correlation Forests の木単体モデル (以下，CC-Tree) を用いた．評価指標は以下の式で定義される分類誤り率を用いた．

$$\text{分類誤り率} = 1 - \frac{\text{正しく分類したテストデータ数}}{\text{テストデータ数}} \tag{13}$$

実験条件として，学習データとテストデータの比を 4:1 とする．加えて，本研究では全ての学習データをパラメーター推定に用いず，学習データの f%のみをパラメータ推定に使用する．また，実験回数は 5-fold-cross-validation を 10 回行った．加えて，各パラメータは，木ごとの最大深さを 100，ノードの最小データ数を 20 とし，木の数は 50 から 300 までの 50 ずつ増加させるものとし，学習データ数の変化と分類精度を検証するためパラメータ f は 20 と 40 で行った．これらの条件で，10 回行いその平均の結果が最良のパラメータを実験結果とした．

表 2　UCI データセットの概要
Table 2　About the UCI Dataset.

データセット名	次元数	カテゴリ数	データ数
balance scale	4	3	625
nursery	8	5	12960
optDigitsHandwritten	64	10	5620

5.2　UCI データセットにおける実験結果

UCI のデータセットの実験結果を表 3,4 に示す．表 3 は各手法におけるアンサンブルしていない木単体の分類性能を示している．また，表 (4) はアンサンブルしたものの分類性能を示している．表 (3) より，balance scale では提案手法，nursery では CC-Tree，optDigitsHandwritten では Decision Tree が最も高い精度となった．各データで

20%と 40%とも同じ手法が高い精度を得ているため，データの構造に依存していると考えられる．また，少ないデータ数で学習した場合，特徴行列からの情報を重視するため PC-Tree は低い分類精度だったと考えられる．

表 (4) より，balance scale (40%)，nursery (20%)，(40%)，optDigitsHandwritten (20%)，(40%) で提案手法が高い分類性能を得ていることがわかる．これは，Random Forest，Rotation Forest，CCFs は全カテゴリ等価に扱い分類するように木が学習する一方で，提案手法は各ノードでカテゴリを統合し学習するため，各ノードごとで得られる分割境界が大きく異なり，結果としてアンサンブルの効果が高まったためと考えられる．

表 3　UCI データセットの実験結果 (Tree)
Table 3　The result for best TME for all of tree method in UCI Datasets.

データセット名	Decision Tree	PC Tree	CC Tree	提案 (Tree)
balance scale (20%)	0.672	0.649	0.742	**0.728**
balance scale (40%)	0.750	0776	0.760	**0.786**
nursery (20%)	0.867	0.771	**0.892**	0.858
nursery (40%)	0.877	0.818	**0.916**	0.890
optDigits Handwritten (20%)	**0.700**	0.682	0.580	**0.700**
optDigits Handwritten (40%)	**0.771**	0.762	0.667	0.756

表 4　UCI データセットの実験結果 (アンサンブル)
Table 4　The result for best TME for all of tree method in UCI Datasets.

データセット名	Random Forest	Rotation Forest	CCFs	提案 (Forest)
balance scale (20%)	0.821	0.856	**0.887**	0.883
balance scale (40%)	0.843	0.876	0.902	**0.912**
nursery (20%)	0.932	0.901	0.940	**0.960**
nursery (40%)	0.956	0.934	0.980	**0.982**
optDigits Handwritten (20%)	0.952	0.952	0.964	**0.970**
optDigits Handwritten (40%)	0.967	0.970	0.978	**0.980**

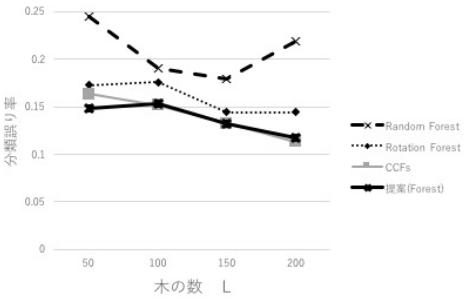

図 1　balance scale の分類性能と木の数 (20%)
Fig. 1　The result of balance scale (20%) by changing the number of Trees.

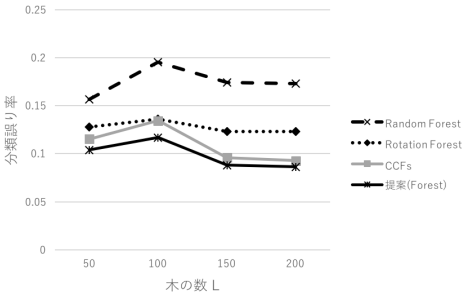

図 2　balance scale の分類性能と木の数 (40%)
Fig. 2　The result of balance scale (40%) by changing the number of Trees.

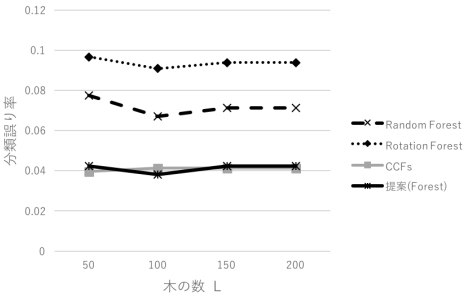

図 3　nursery の分類性能と木の数 (20%)
Fig. 3　The result of nursery (20%) by changing the number of Trees.

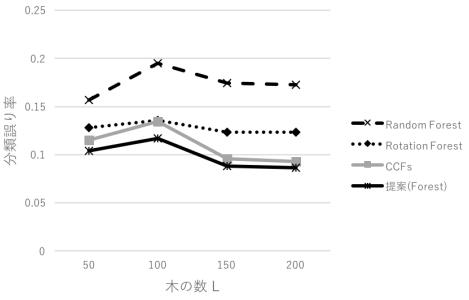

図 4　nursery の分類性能と木の数 (40%)
Fig. 4　The result of nursery (40%) by changing the number of Trees.

6. 考察

Rotation Forest, CCFs, 提案手法は，各ノードごとで合成変数を得て分類したことにより，アンサンブルの効果が高まり，Random Forest に比べて高い分類性能を得られたと考えられる．また，提案手法はカテゴリ行列に対する重み行列を 2 次元にしていることと等価であり，推定するパラメータを減らすことでデータ数が少ない時に，適切な重み行列を得られた．その結果，$\boldsymbol{b}$ が過学習せず適切な分割境界が得られ分類性能が向上したと考えられる．また，図 1 -図 4 が示すようにアンサンブル数が 150 を超えてからは分類性能に大きな変化が得れていない．これは，データの構造に対して過剰なアンサンブル数をとったためであると考えられる．そのため，適切なアンサンブル数の決定方法の検討も必要である．最後に，Exhaustive 符号を用いたカテゴリの組み合わせ最適化は，データ数が少ない時に有効であり，アンサンブルすることでより高い分類性能が得られると言える．一般に，説明変数の数 (特徴空間の次元) が増えると必要となるサンプル数も増えるので，そのような場合に，提案法が有効となる可能性もある．

7. まとめと今後の課題

本研究では，カテゴリ数に対しデータ数が少ない多値分類問題を対象とし，各ノードごとで ECOC の考えに基づいたカテゴリ行列の最適化学習アルゴリズムの提案を行った．実験結果より，学習に扱えるデータ数が少ない場合，提案手法のアンサンブル法を用いることにより高い分類性能が維持できることを示した．今後の課題としては，CCA を行う試行回数の削減とカテゴリ行列を直接最適化する手法の提案などがあげられる．

参考文献

[1] Breiman, L., *Random forests*, Machine learning, 45(1):5-32,(2001)
[2] Rainforth, T. and Wood, F., *Canonical Correlation Forests*, 入手先 ⟨https://arxiv.org/pdf/1507.05444.pdf ⟩ (参照 2018-10-05).
[3] Breiman, L., Friedman, J.H., Olshen, R.A., and Stone, C.J., *Programs for Machine Learning*, Biometrika, 28, 253-240 (1993).
[4] Gama.J., *Functional trees.*, Machine Learning, 55.3 :219-250, (2004)
[5] Hotelling, H., *Relations between two sets of variates*, Morgan Kaufmann, Machine Learning, 28, 321-377 (1936).
[6] Breiman, L., Friedman, J.H., Olshen, R.A., and Stone, C.J., *Classification and Regression Trees*, CRC press (1984).
[7] Quinlan, R. J., *C4. 5: programs for machine learning*, Elsevier (2014.).
[8] Rodriguez, J.J., Kuncheva, L.I., and Alonso, C.J. , *Rotation forest: A new classifier ensemble method. Pattern Analysis and Machine Intelligence*, IEEE Transactions on, 28,1619-1630 (2006).
[9] Dietterich, T.G. , Bakiri, G., *Solving Multiclass Learning Problems via Error-Correcting Output Codes*, Journal of Artificial Intelligence Research, 2, 263-286 (1995).
[10] Dua,D.,Taniskidou.E., UCI Machine Learning Repository 入手先 ⟨http://archive.ics.uci.edu/ml⟩. Irvine, CA: University of California, School of Information and Computer Science. (2017).