

NLFBDT : SVM を組み込んだ非線形 Fish Bone Decision Tree による識別と特徴の可視化

松尾大典[†] 和田俊和[†]

概要 : 決定木は、複数のクラスラベルが付与されたデータ集合を、同一ラベルの部分集合になるまで再帰的に分割することによって、識別規則を学習する機械学習の手法であり、得られる規則が知識として把握しやすいという特長がある。実ベクトルを対象とした決定木では、データ集合を射影軸に射影して閾値処理を行うことでデータを分割する方法がある。この方法では、射影軸は本質的にある2クラスに着目して決定されるが、射影後の閾値は全クラスラベルを対象とした分割後のクラスラベルの純度によって決定される。つまり、射影軸と閾値を決定する際に参照されるクラスラベルが異なるという問題点がある。また、射影軸の決定に関与しないクラスのデータを過剰に分割してしまうという問題点もある。これら2つの問題点を解消するために、我々は最も識別しやすい2クラスにスポッティングしてデータの分割を繰り返す FishBoneDecisionTree (FBDT) を提案し、その有効性を示した。今回は、線形な分割面だけを用いる FBDT を拡張し、SVM による非線形分割面も取り入れ、識別性能をさらに向上させた NLFBDT : Non-Linear FBDT を提案する。NLFBDT では線形識別によって学習サンプル数を削減した後に SVM を適用するため、学習サンプル数の増加に対してある程度対処することができ、SVM のみを用いた決定木と同等以上の識別率を持つことを実験的に明らかにする。また、応用例として、学習された識別規則を可視化することが可能であることも示す。

キーワード : 決定木, データ分割, SVM

NLFBDT : Non-Linear Fish Bone Decision Tree incorporating SVM for classification and visualization

DAISUKE MATSUO[†] TOSHIKAZU WADA[†]

Abstract. Decision tree is a data structure for learning classification rules by recursively dividing given multi-class dataset. this classification rules can be analysed by simple way. the data division for real vectors is realized by thresholding the scalar values obtained by inner product with certain vectors. certain vector is computed by focusing on class pair. However threshold is decided by focusing on multi class. inconsistency between computing a vector and to decide a threshold is a problem. there is also a problem the division for unfocused data. we proposed the method of building decision tree that focus on class pair for solve the problem and name it Fish Bone Decision Tree (FBDT). FBDT that based on linear dividing. we propose Non-Linear FBDT (NLFBDT) that can Non-Linear dividing by SVM in this paper. NLFBDT can learn large scale dataset because SVM is givened data that is divided by linear dividing. this paper reports performance of NLFBDT in classification. also, we visualize a classification rules that is learned by NLFBDT.

Keywords: decision tree, data partition, SVM

1. はじめに

決定木[1]は、機械学習の一手法であり、学習サンプル集合を再帰的に分割する事によって構築される木構造である。決定木の各ノードでは、サンプルに付けられたクラスラベルの純度ができるだけ浅い段階で高くなるように分割が行われるが、この分割によって決まる識別ルールを木構築後に解析し易いという特長がある。

決定木の構築法としては、CART法[4]やID3[5]等のサンプルの特定の次元に注目した分割を行うものと、ある軸上にサンプルを射影し、その軸上でサンプル集合の分割を行うものが考えられるが、本報告では後者の射影に基づく決定木の構築法について考える。射影に基づく決定木構築の手順は、1)射影軸の決定、と、2)射影したサンプル集合の分

割、の2段階に分けることができる。

射影軸の決定法としては、主成分分析や判別分析を用いる方法が考えられる。主成分分析ではサンプル全体を参照し、分散最大となるベクトルを、判別分析ではクラス間分散に対するクラス内分散を最大にするベクトルを、射影軸として求めることになる。

射影後のサンプル集合の分割法としては、一般にサンプルの全クラスラベルを参照し、分割後の純度が最大になるように閾値を決定する方法が良く用いられる。

クラスラベルの純度が分割によって上昇した度合いは、Gini・ゲインやエントロピー・ゲインで測られるが、本報告ではエントロピー・ゲインを用いることにする。

サンプルデータの分割は、クラスラベルの純度が上昇するように決められるべきであるが、純度を上げる射影軸を

[†] 和歌山大学院システム工学研究科
Wakayama University, Faculty of Systems Engineering

求める方法が存在しないため、実際には判別軸が良く用いられる。

判別軸は、多クラスの場合であっても、ペアワイズなクラス間分散を用いて計算しているため、本質的には特定の2クラスに注目した射影軸を計算していることになる。これに対して、分割に用いられる閾値の決定では、全クラスラベルを参照したエントロピーの計算が行われる。

このように、軸の決定では2クラスに注目し、閾値の決定では全クラスラベルを参照するという齟齬があるため、判別軸とエントロピー・ゲインを用いた決定木構築は矛盾を内包していると言える。

また、ある2つのクラスを区別するためにサンプル集合を分割すると、他のクラスに属するサンプル集合も分断してしまう。これを繰り返すことによって過剰にサンプル集合を分割するため、決定木の汎化性能が退化することが考えられる。

これら、「射影軸と閾値決定法の矛盾」と、「サンプル集合の過剰分割」という2つの問題点の解決のために、我々は Fish Bone Decision Tree (FBDT) [7]を提案した。これは2クラスのみ注目して、射影軸と分割の閾値を決定する一貫した戦略に則った決定木の構築法である。

FBDT は、一般的な決定木と同様に、射影軸と閾値の決定の手順を踏むが、サンプル内で平均ベクトル同士が最も離れている2クラスのペアに注目し、それらを分離するように多段に線形分割を行い、注目した2クラス以外のクラスが分けられたサンプル集合に関しては、分割後にマージを行う。こうすることで木の各ノードでは、射影軸と閾値を一貫して特定の2クラスに注目した決定をすることが可能になる。また、サンプル集合のマージを行うことで過分割による汎化性能の低下を抑止することが期待できる。木の構造としては、束構造を含む2分木が構築され、結果的には図のように注目した2クラスとその他クラスとを分ける3分木のような構造の決定木が構築される。

FBDT と他の決定木構築手法とで、構築された木の識別性能を比較する実験では、比較対象内で FBDT の識別性能が高いことが確認された。

しかし、近年の画像解析の分野では、深層学習を用いた手法が多数提案されており、それらの識別性能と比べると FBDT の識別性能は十分でない。

FBDT は決定木を基本とする手法であるので、木の構築後に、識別ルールの解析が容易であるという特長がある。この特徴は、深層学習による手法と比べて、FBDT が優れている部分と言える。

そこで、本報告では、FBDT と Support Vector Machine (SVM) を統合した手法を考え、識別ルールの解析等の決定木の特長を損なうことなく、FBDT の識別性能の向上を図る。

この提案手法を、Non-Linear FBDT (NLFBDT) と呼び、

以降の章では、NLFBDT の構築方法、他の手法との識別性能の比較、抽出された識別ルールの可視化を行う。

2. 関連研究

本章では、ベクトルデータの特定の次元に着目して決定木を作成する CART 法や C4.5 と、何らかの方法で求めたベクトルに対する射影成分の閾値処理で決定木を構築する手法、および、決定木と同様に識別問題で用いられる SVM について述べる。

2.1 決定木を構築する手法

CART 法や C4.5 では、ベクトルデータ \mathbf{x} の各要素のうちのどの次元 d を用いてデータセットの分割を行えば、エントロピー・ゲインや Gini ・ゲインなど、クラスラベルの純度の指標が最も大きくなるかをテストし、その時の次元 d と閾値 θ を用いてデータ集合の分割を行うというものである。以下にエントロピー・ゲインを用いた場合の次元 d と閾値 θ の求め方を示す。

ベクトルデータ集合を S_n とする。これを $S_n^{left}(d, \theta)$ と $S_n^{right}(d, \theta)$ という直和集合に分割する問題を考える。

$S_n = S_n^{left}(d, \theta) \cup S_n^{right}(d, \theta), S_n^{left}(d, \theta) \cap S_n^{right}(d, \theta) = \emptyset$ である。これらの分割された集合は、それぞれ

$$S_n^{left}(d, \theta) = \{\mathbf{x} \in S_n | x_d \leq \theta\} \quad (1)$$

$$S_n^{right}(d, \theta) = \{\mathbf{x} \in S_n | x_d > \theta\} \quad (2)$$

と表現することが出来る。

また、集合 S_n は、個々のデータに付与されたクラスラベル c に基づいて C 個の部分集合 S_{nc} に分割することが出来る。

$$S_{nc} = \{\mathbf{x} \in S_n | class(\mathbf{x}) = c\}, \quad c = 1, \dots, C$$

このとき、エントロピーは下記のように表すことが出来る。

$$Ent(S_n) = - \sum_{c=1}^C \frac{|S_{nc}|}{|S_n|} \log \frac{|S_{nc}|}{|S_n|} \quad (3)$$

エントロピー・ゲインとは、 S_n に対するエントロピー $Ent(S_n)$ と、分割後の各エントロピー $Ent(S_n^{left}(d, \theta))$ と $Ent(S_n^{right}(d, \theta))$ によって定義される量

$$\begin{aligned} & Egain(d, \theta) \\ &= Ent(S_n) \\ &\quad - \frac{|S_n^{left}(d, \theta)|}{|S_n|} Ent(S_n^{left}(d, \theta)) \\ &\quad - \frac{|S_n^{right}(d, \theta)|}{|S_n|} Ent(S_n^{right}(d, \theta)) \end{aligned} \quad (4)$$

であり、これを最大化する d, θ が S_n に対して最も良い分割パラメータと見なされることが多い。すなわち、

$$(d_n^*, \theta_n^*) = \underset{(d, \theta)}{\operatorname{argmax}} Egain(d, \theta) \quad (5)$$

である。

これに対して、次式のように、ある射影軸 \mathbf{a} に対して射影し、同様にデータを分割する方法[2]もある。

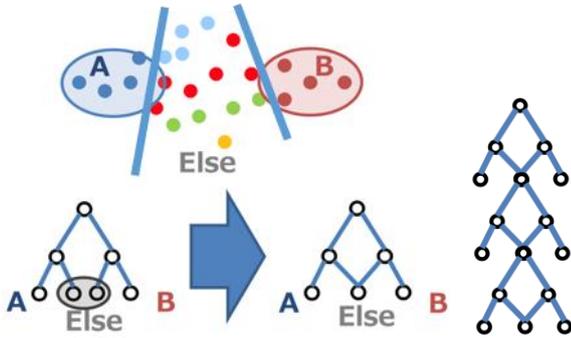


図1 Fish Bone Decision Tree の概要

$$S_n^{left}(\mathbf{a}, \theta) = \{x \in S_n | \mathbf{a} \cdot x \leq \theta\}$$

$$S_n^{right}(\mathbf{a}, \theta) = \{x \in S_n | \mathbf{a} \cdot x > \theta\}.$$

この場合、 \mathbf{a} の解空間は非常に広いため、判別分析を用いる方法やランダムに \mathbf{a} を生成する方法などがある。もちろん、このような方法だけでなく、内積の代わりにカーネルを用いる方法や、距離計算を用いる方法など、ベクトルを対象とした決定木構成法については数多くの可能性があるが、これらについては過去の研究であまり詳しく調べられていない。

2.2 SVM

次に、SVMについて説明する。SVMは2つのクラスラベルを持つサンプル集合の識別においてクラス間のマージン最大となる識別面を求め、識別面から最も近いベクトルをサポートベクター (SV) として学習する。SVMの学習時に与えられた l 個のベクトルを $x_i, 1, \dots, l$ ラベルを $y_i \in \{1, -1\}$ とすると、最適な識別面の学習は、以下の式(6)において

$$\min_{\alpha} f(\alpha) = \frac{1}{2} \mathbf{a}^T Q \alpha - \mathbf{e}^T \alpha \quad (6)$$

$$0 \leq \alpha \leq C, i = 1, \dots, l, \mathbf{y}^T \alpha = 0$$

最適な α を求める問題に帰着する。 C は上限値、 \mathbf{e} は要素全てが1のベクトル、 Q は縦 l 、横 l のグラム行列であり Q の要素 Q_{ij} は $Q_{ij} = y_i y_j K(x_i, x_j)$ と表される。 $K(x_i, x_j)$ は任意のカーネル関数を指す。SVMは非常に強力な識別器と知られているが、式(6)において、 Q がサンプル集合のサイズ l に依存して巨大になるため、計算量の問題から学習が困難になる場合が考えられる。この問題を軽減するために Sequential Minimal Optimization (SMO) [8][9]という手法が一般に用いられる。SMOは、 Q の個々の要素 Q_{ij} に注目し、個別に Kuhn-Tucker 条件のチェックを行うことで、計算量の問題を軽減する。

しかし、SMOを用いたとしても、 Q のサイズに変わりはなく、SVMがサンプルのサイズに依存した計算量を必要とする問題は本質的には解決していない。このことからサンプル集合をあらかじめ分割したのちに、SVMでの学習を行うという工夫が考えられる。

大規模サンプルを対象とした SVM による学習を可能と

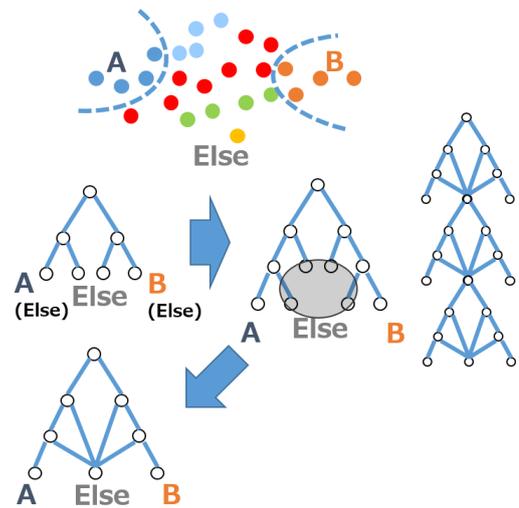


図2 Non-Linear FBBDT の概要

するために、決定木と SVM を組み合わせる DTSVM という手法が提案されている[11]。これは、サンプル集合に対して決定木によるデータ分割を行った後に SVM を実行する事で、大規模サンプルを対象とした学習を可能にしている。

しかし、DTSVM では、決定木の構築時に木の各ノードでサンプルの特定の次元に注目して分割を行っている。こういった単純な分割は、多次元、多クラスラベルのサンプル集合を学習した場合に、汎化性能を損なうことが考えられる。また、そういった複雑なサンプル集合を学習した場合の汎化性能については、先行研究において深く考察されていない。

以上より、サンプルのクラス情報を用いて、射影軸と閾値を決定する木構築と SVM とを統合した手法を研究する意義はあると考えられる。

3. FBBDT

ここでは、提案手法の基礎となる FBBDT について説明する。FBBDT では、内積を用いてデータ集合を分割する。このため、射影軸 \mathbf{a} をどのように求めれば良いかについて検討することが重要になる。

射影軸 \mathbf{a} の決定の際には「データ集合を2つに分割する」という明示的な意図がある。これに対して、分割の善し悪しを判定するエントロピー・ゲインは、全てのクラスのデータがどちらの集合に分かれたのかによって決まる。このため、データを二分しながら全クラスラベルを参照して閾値を決定しなければならないという矛盾が生じる。このため、FBBDT では、以下のようにデータの分割を行う。

1. まず、特定の2クラスに属するデータ S_{nA^*}, S_{nB^*} のみに着目して射影軸 \mathbf{a} を求める。さらに、そのデータのみに着目してエントロピーを計算し、各データの射影成分 $\{\mathbf{a} \cdot x\}$ に対する閾値の決定を行い、データ集合 S_n を S_n^{left} と S_n^{right} に分割する。

2. 次に、これまで無視してきたクラスラベルを持つデータと S_{nA^*}, S_{nB^*} のデータの分離を行うため、 S_n^{left} と S_n^{right} 内で、 S_{nA^*} と $S_n^{left} \setminus S_{nA^*}$ 、および S_{nB^*} と $S_n^{right} \setminus S_{nB^*}$ の分割を行うための射影軸と閾値の学習を行い、4つの集合 $S_n^{left^*}, S_n^{left} \setminus S_n^{left^*}, S_n^{right} \setminus S_n^{right^*}, S_n^{right^*}$ を得る。

FBDT では以下のように特定の2クラスを決定する。

各クラスのデータ集合 S_{nc} について、その重心を以下のよう求める。

$$\bar{x}_{nc} = \frac{1}{|S_{nc}|} \sum_{x \in S_{nc}} x, \quad c = 1, \dots, C \quad (7)$$

これらの間で最も距離の開いているクラスのペア A^*, B^* のみ注目して射影軸と閾値の決定を行う。

$$(A^*, B^*) = \operatorname{argmax}_{A, B} \|\bar{x}_{nA} - \bar{x}_{nB}\| \quad (8)$$

また、射影軸も単純に以下のように求める。

$$\mathbf{a} = (\bar{x}_{nA} - \bar{x}_{nB}) / \|\bar{x}_{nA} - \bar{x}_{nB}\| \quad (9)$$

得られた4つの集合のうち、 $S_n^{left^*}$ は、 S_{nA^*} を切り出すようにして得られた集合、 $S_n^{right^*}$ は、 S_{nB^*} を切り出すようにして得られた集合であり、 $S_n^{left} \setminus S_n^{left^*}$ と $S_n^{right} \setminus S_n^{right^*}$ は副産物であるので、再びマージする。これによって、集合 S_n を $S_n^{left^*}, S_n^{else}, S_n^{right^*}$ の3つに分割するノードが得られる。図1はFBDTによる木構築を模式的に示した図である。

現実のデータセットを対象に木を構築する場合、純粋なクラスラベルの集合として $S_n^{left^*}, S_n^{right^*}$ を得ることは難しく、これらに対して木を再帰的に構築することになる。

我々は、いくつかの手法とFBDTとで識別率を比較し、その中でFBDTの識別能力が最も高いことを確認した。だが、SVMやDNN等の識別能力との比較においては、FBDTの識別能力は十分とは言えない。

識別能力を向上するために、決定木の応用として、フォレスト[2][3]などのアンサンブル学習が考えられる。しかし、フォレストは複数の木の識別の総和から識別結果を得るため、識別結果と識別ルールとの関係を解析することは難しく、決定木の特長を損なってしまう。

そこで、識別ルールの解析を可能としたままFBDTの識別率を向上する方法について考える。

4. NLFBDT

本報告における提案手法であるNLFBDTについて説明をする。

先行研究[2]では、木の構築において非線形な分割面を用いる場合に、高い識別能力を持つことが示されていた。そこで、SVMをFBDTに組み込むことで非線形な分割面を持った決定木の構築方法を考える。

FBDTでは、注目したクラスのペア A^*, B^* に注目しデータ集合 S_n から S_{nA^*}, S_{nB^*} を切り出すために分割した結果 $S_n^{left^*}, S_n^{right^*}$ の集合が得られるが、線形分割では各集合がクラス A^* もしくは B^* のみを含むような分割結果が得られること

は現実的ではない。そこで分割後の $S_n^{left^*}, S_n^{right^*}$ を対象にSVMを実行することで純粋なクラスの集合である S_{nA^*}, S_{nB^*} を得る方法を考える。

1. まず、特定の2クラスに属するデータ S_{nA^*}, S_{nB^*} のみに着目して射影軸 \mathbf{a} を求める。さらに、そのデータのみに着目してエントロピーを計算し、各データの射影成分 $\{\mathbf{a} \cdot \mathbf{x}\}$ に対する閾値の決定を行い、データ集合 S_n を S_n^{left} と S_n^{right} に分割する。
2. 次に、これまで無視してきたクラスラベルを持つデータと S_{nA^*}, S_{nB^*} のデータの分離を行うため、 S_n^{left} と S_n^{right} 内で、 S_{nA^*} と $S_n^{left} \setminus S_{nA^*}$ 、および S_{nB^*} と $S_n^{right} \setminus S_{nB^*}$ の分割を行うための射影軸と閾値の学習を行い、4つの集合 $S_n^{left^*}, S_n^{left} \setminus S_n^{left^*}, S_n^{right} \setminus S_n^{right^*}, S_n^{right^*}$ を得る。
3. $S_n^{left^*}, S_n^{right^*}$ が分割の目的としていた A^*, B^* 以外のクラスラベルを持つデータを含む場合にSVMを実行し、 $S_{nA^*}, S_n^{left^*} \setminus S_{nA^*}, S_n^{right^*} \setminus S_{nB^*}, S_{nB^*}$ の4つの集合を得る。

SVMによる分割を行うと、 $S_n^{left^*}$ から S_{nA^*} を切り出すことで $S_n^{left^*} \setminus S_{nA^*}, S_n^{right^*}$ から S_{nB^*} を切り出すことで $S_n^{right^*} \setminus S_{nB^*}$ が副産物として得られる。これらに対してFBDT同様マージを行うが、直前の線形分割時の処理と合わせると、 $S_n^{left} \setminus S_n^{left^*}, S_n^{right} \setminus S_n^{right^*}, S_n^{left^*} \setminus S_{nA^*}, S_n^{right^*} \setminus S_{nB^*}$ が S_n^{else} にマージされたことになる。図2はNLFBDTによる木構築を模式的に示した図である。

NLFBDTでは、まずFBDTと同様の木構築を行う事でサンプル集合を分割し、分割後のサンプルを対象にSVMを実行することで、線形分割の段階で、どの2クラスに注目し分割を行ったのかを、識別ルールとして確認することが可能であり、また、分割後のサンプルを対象にSVMを実行するため、SVMのスケラビリティに関する問題を解決することができる。

次章で、提案手法の性能を実験的に明らかにするための実験内容と、その結果を示す。

5. 実験

提案手法の性能を確かめるための実験について、その条件と結果について述べる。

最初は識別能力について、次にSVMで学習が困難と考えられる大規模サンプルを対象とした場合の実験を行う。

5.1 識別性能の確認

提案手法と他の手法の識別率を比較する事で、提案手法の識別能力を実験的に明らかにする。

一般的な決定木構築法であるCART法、提案手法と同様にクラス情報を参照するが射影軸を判別分析より求めて単純な2分木を構築する手法、提案手法の基礎であるFBDT、

手法 データ	NL FBDT	FBDT	CART 法	判別分析 軸	LIB SVM
MNIST	96.75	93.44	88.51	93.11	97.91

表 1 MNIST 対象時の各手法の識別率 (%)

同様に SVM を比較対象として、提案手法と同様のデータセットを対象とした場合の識別能力を比較する。

CART 法は Python のライブラリである Scikit-learn のものを使用し、事前に設定するパラメータとしては、分割基準としてエントロピー・ゲインを使用することを選択した。SVM は C++言語のライブラリである LIBSVM[10]を使用し、パラメータとしてはカーネル関数として多項式カーネルを選択し、次数は 3 に設定した。提案手法である NLFBDT でも LIBSVM を使用しているが、パラメータは基本的に単体の LIBSVM と共通とする。ただし、提案手法では、注目するクラスとその他クラスとを分割する、2 クラスの識別問題として LIBSVM を運用するのに対して、LIBSVM 単体では、多クラス識別の問題としての運用をする。

実験には、手書きの 0 から 9 (10 クラス) の数字のデータセット MNIST[6]を用いた。MNIST は、学習データ 6 万件とテストデータ 1 万件とに分かれているので、それぞれをサンプルとテストとして、各手法による学習と識別を行い、識別率の確認を行った。実験の結果を表 1 に示す。

表 1 より、提案手法である NLFBDT は、元となった FBDT よりも高い識別率であることが確認できた。また、CART 法や判別分析軸に基づく木構築など、他の決定木構築法よりも NLFBDT の識別率は高かった。しかし、LIBSVM との比較では、共通したパラメータを使用しているにもかかわらず、NLFBDT の識別能力が劣っていることが見て取れる。

これは、NLFBDT では SVM の実行前に、サンプルを分割するため、SVM で学習するサンプル集合が十分でないことが原因だと考えられる。

そこで、NLFBDT において、SVM 実行前の線形分割で分割面を調整することで、SVM が学習する比重を操作する方法を考える。閾値決定において、エントロピー・ゲインは式(4)のように計算されるが、エントロピーに重みとして係数 w をかけたエントロピー・ゲインを $E_{gain}'(d, \theta)$ として式(10)のように計算する。式(10)を用いることで、本来の識別境界からずれた閾値が決定され、SVM で学習するサンプル集合のサイズを操作することができる。

$$\begin{aligned}
 E_{gain}'(d, \theta) &= Ent(S_n) \\
 &- w \frac{|S_n^{left}(d, \theta)|}{|S_n|} Ent(S_n^{left}(d, \theta)) \\
 &-(1-w) \frac{|S_n^{right}(d, \theta)|}{|S_n|} Ent(S_n^{right}(d, \theta)) \\
 &0 \leq w \leq 1
 \end{aligned} \tag{10}$$

w	0.1	0.2	0.3	0.4	0.5
識別率	97.61	97.67	97.53	97.01	96.75

表 2 MNIST 対象時に NLFBDT で w を変化させた場合の識別率 (%)

w を変化させ複数の NLFBDT の木を構築し、それぞれの識別率を確認した結果を表 2 に示す。

表 2 より、 w を変化させることで、元の NLFBDT と比べて識別率が向上することが確認できた。特に、 $w = 0.2$ で識別率 97.67% と、比較的高い識別率を記録した。0.2 付近の係数を試したところ、 $w = 0.25$ で、識別率 98.13% を記録し、FBDT 及び SVM を上回る識別率であることを確認した。

この結果は、サンプル全体から学習に有効な集合を切り出し、個別に SVM を実行したことに起因すると考えられる。つまり、線形識別と SVM による非線形識別を組み合わせることで、高い識別率を実現できることを示している。

しかし、本報告では特定のデータセットのみでの実験であることから、係数が他のデータセットに対しても最適である保証は無いこと、データセットごとにそれぞれ最適な係数を調べる必要があることが考えられる。係数と識別率との間に単純な相関が無いことに注目できるが、本報告では深く考察しないものとする。

5.2 対大規模サンプル対象時木構築確認

2 章で述べた通り、サイズの大きいサンプルの学習は、SVM では困難と言える。NLFBDT は線形分割後に SVM による学習を行うので、この問題を回避できると考えられる。

本節では、NLFBDT と SVM で大規模なサンプルを学習した場合の、学習の能否と実行時間について述べる。

大規模サンプルを対象に NLFBDT と LIBSVM それぞれを実行し、学習のスケーラビリティについて比較する。学習の対象として、NIST[12]を用いる。NIST は 0 から 9, a から z, A から Z の英数字 62 クラスの手書き画像のデータセットだが、本実験では 0 から 9 の数字画像サンプルのみを抜き出し学習を行った。また、学習には表 3 の計算機を共通に使用した。

項目	内容
CPU	Intel Core i7-4790 CPU 3.60Ghz * 8Core
Mem	23.2GiB
OS	Ubuntu 16.04 LTS
コンパイラ	g++ 5.4.0
OpenCV	OpenCV 3.1.0

表 3 実験に使用した計算機のスペック

学習の結果としては、NLFBDT で 134 時間 43 分で学習を完了した。決定木は、ノード数 505、深さ ~ ~ ~ の木が

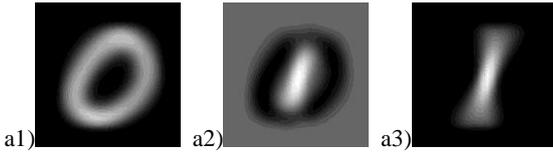


図 3 MNIST を学習した NLFBDT の根ノードの射影軸と計算に用いた平均ベクトル

a1,a3 はクラス 0 と 1 の平均ベクトル, a2 はそれらから求めた射影軸をそれぞれ画像に変換したもの

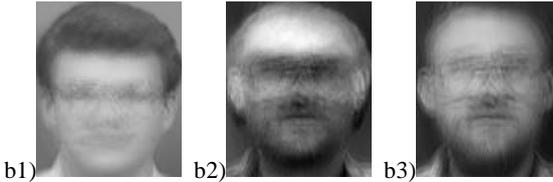


図 4 ORL を学習した NLFBDT の根ノードの射影軸と計算に用いた平均ベクトル

b1,b3 はクラス 6 と 14 の平均ベクトル, b2 はそれらから求めた射影軸をそれぞれ画像に変換したもの



図 5 NLFBDT の学習に用いた ORL 内のクラス 6 と 14 の代表的な画像

c1 はクラス 6, c2 はクラス 14 の画像

構築された。対して、LIBSVM では実行中にプログラムが停止し、学習を完了することができなかった。プログラムの実装上、NLFBDT は SVM の処理を行うために LIBSVM を参照しているため、LIBSVM で固有の不具合が発生していることは考えられない。このことから、NLFBDT は SVM が不得意とする大規模サンプルを学習する場合に有利であり、少なくとも、LIBSVM では学習が完了しなかったデータセットを対象に学習が可能であることが確認できた。

ただし、NLFBDT で分割をコントロールすることは難しく、線形分割後であっても、サンプル集合が大規模で、LIBSVM の実行が困難な場合は十分考えられる。

5.3 実験のまとめ

SVM を組み込みこむことで、NLFBDT は FBDT よりも識別性能が向上したことが確認できた。LIBSVM には識別能力で劣るが、線形分割時に工夫をすることで、NLFBDT の識別性能が勝る場合も確認することができた。また、SVM で学習が困難な大規模データであっても、NLFBDT では学習が可能であることが実験から確認できた。

決定木を基本とする NLFBDT では、木構築時に抽出され

た識別ルールを確認することができる。次章では抽出された識別ルールの可視化の方法と、実際に可視化した結果を示す。

6. 抽出された識別ルールの可視化

決定木の特徴として木構築時に抽出される識別ルールの確認が容易であることが挙げられる。CART 法など、特定の次元に注目する木構築の場合には、木の各ノードで分割基準として注目する次元と閾値を決定するが、この分割基準が CART 法での識別ルールと言える。決定木の根ノードから葉に向かって順に識別ルールを確認する事で、ベクトルにおいて重要な属性を解析することができる。特に真偽などの非計量データを対象とする場合に CART 法による解析は有効と考えられる。

画像などの実ベクトルを対象とする場合には、単一の次元に注目した木構築では、識別ルールの解析は複雑になり、有効であるとは考えられない。提案手法である NLFBDT では、各ノードでクラスごとの平均ベクトルを計算し、平均同士の差ベクトルから射影軸 \mathbf{a} を計算する。画像をラスタスキャンしたデータセットを対象に木構築を行う場合には射影軸をラスタスキャンの入出力を反転し、画像化することで、NLFBDT では視覚的に識別ルールを確認できる。

$$\begin{aligned} \text{img}_{ij} &= \mathbf{x}_{j+i \cdot W} \\ i &= 1, \dots, H, j = 1, \dots, W \end{aligned} \quad (11)$$

img は高さ H 、幅 W の行列

画像をラスタスキャンしたベクトルを集めたデータセットである MNIST, ORL[13] を対象に、NLFBDT で木構築を行った場合に抽出される識別ルールを画像化したものを図 3,4 に示す。MNIST, ORL それぞれで木構築を行った場合の各木での根ノードにおける識別ルールである射影軸と計算のために用いた平均ベクトルを画像化した。射影軸に関してベクトルは長さ 1 に正規化されているのでベクトル \mathbf{x} の要素を $0 \leq x_i \leq 255$ の範囲に正規化した後に表示している。

まず、図 3 の MNIST 対象時に得られた射影軸と平均ベクトルに注目する。MNIST は 0 から 9 の数字の手書き画像をベクトル化したデータセットだが、NLFBDT は根ノードで 0 と 1 のクラスに注目した射影軸の決定と分割をする。クラス 0 と 1 の平均ベクトルは確かに視覚的に各クラスラベルに対応するものと判断できる。また、射影軸を可視化した画像も各平均ベクトルの特長を残してどのような特徴に注目してサンプルの射影と分割を行ったのか視覚的に判断できる。

次に、図 4 の ORL 対象時に得られた射影軸と平均ベクトルに注目する。ORL は 40 人の人物の顔を陰影や角度を様々に変えて撮影した画像をベクトル化したデータセットだが、NLFBDT は根ノードで 6 と 14 のクラスに注目した射影軸の決定と分割をする。クラス 6 と 14 それぞれの人

物の顔画像の一部を図 5 に示す。各クラスの代表的な顔画像から平均ベクトルを可視化したものはクラス 6 と 14 それぞれの人物の平均的なパターンが出力されていることが分かる。また射影軸を可視化した画像を確認すると、クラス 6 の画像の頭髪とクラス 14 の顎髪との間で大きく差が生まれていることが視覚的に判断でき、画像のどのような特徴に注目して分割が行われたのかが確認できる。

7. まとめ

サンプル内の 2 クラスに注目し、射影軸と分割の決定を繰り返すことで決定木を構築する手法である FBDT に、SVM による非線形な分割を取り入れた NLFBDT を提案し、その性能と、特長である識別ルールの可視化について実験を行い、その結果を示した。結果から、FBDT と比べ識別性能における向上が確認され、識別ルールの可視化についても可能であることが確認できた。このことから、決定木としての特長を失わずに性能の向上を実現し、かつ、SVM が大規模なサンプル集合を学習することが困難であるという問題を解決し、両者の長短所を補完し合う手法が構築できた。

今後の課題としては、本報告で対象としたデータセット以外で提案手法により構築した決定木の識別能力の確認や、DTSVM 等の、決定木と SVM を組み合わせた手法と提案手法とを比較することを考えている。

参考文献

- [1] J. R. Quinlan, "Induction of decision trees," *Machine learning* 1.1, pp. 81-106, 1986
- [2] A. Criminisi, J. Shotton, E. Konukoglu, "Decision Forests: A Unified Framework for Classification, Regression, Density Estimation, Manifold Learning and Semi-Supervised Learning," *Foundations and Trends® in Computer Graphics and Vision*: Vol. 7: No 2-3, pp 81-227, 2012
- [3] L. Breiman, "Random forests," *Machine learning*, Vol. 45, no. 1, pp. 5-32, 2001
- [4] L. Breiman, J. H. Friedman, R. A. Olshen, and C. J. Stone, "*Classification and regression trees*," Monterey, CA: Wadsworth & Brooks, 1984
- [5] J. R. Quinlan, "C4.5: Programs for Machine Learning," Morgan Kaufmann Publishers, 1993.
- [6] "UCI Machine Learning Repository", (<http://archive.ics.uci.edu/ml/>)
- [7] 松尾大典; 和田俊和. Fish Bone Decision Tree: 最大距離クラスを分離する決定木構築法とその応用 (パターン認識・メディア理解). *電子情報通信学会技術研究報告= IEICE technical report: 信学技報*, 2017, 117.210: 173-179.
- [8] Platt, John. "Sequential minimal optimization: A fast algorithm for training support vector machines." (1998).
- [9] Fan, Rong-En, Pai-Hsuen Chen, and Chih-Jen Lin. "Working set selection using second order information for training support vector machines." *Journal of machine learning research* 6.Dec (2005): 1889-1918.
- [10] Chang, Chih-Chung, and Chih-Jen Lin. "LIBSVM: a library for support vector machines." *ACM transactions on intelligent systems and technology (TIST)* 2.3 (2011): 27.

- [11] Chang, Fu, et al. "Tree decomposition for large-scale SVM problems." *Journal of Machine Learning Research* 11.Oct (2010): 2935-2972.
- [12] "NIST Special Database 19", (<https://www.nist.gov/srd/nist-special-database-19>)
- [13] Samaria, Ferdinando S., and Andy C. Harter. "Parameterisation of a stochastic model for human face identification." *Applications of Computer Vision, 1994., Proceedings of the Second IEEE Workshop on. IEEE, 1994.*