2003－MPS－46　(5)
2003／9／18

アンサンブル学習の適用による Linear Discriminant Analysis アルゴリズムの改善

野宮 浩揮[1]　　上原邦昭[1]

Linear Discriminant Analysis (LDA) は，2 クラスの分類を行う学習アルゴリズムのひとつである．LDA の欠点として，2 つのクラスの境界が線形関数により近似困難である場合，すなわち境界が非線形である場合に性能が低下することがあげられる．本稿では，LDA を用いて アンサンブル学習を行い，Error-Correcting Output Codes (ECOC) を適用して性能の改善を図る手法を提案する．また，分類精度の低い仮説の影響を低減するため，複数の仮説を統合する際に重み付き投票を導入する．重み付き投票とは，生成された各仮説に，訓練集合に対する分類精度にしたがって重みをつける手法である．さらに，ECOC の適用により，LDA が多クラス問題にも適用可能となることを示す．

Improvement of Linear Discriminant Analysis by Applying the Ensemble Method

Hiroki Nomiya[2]　　KuniakiUehara[2]

Linear Discriminant Analysis (LDA) is one of the learning algorithms for the binary classification problems. One of the drawbacks of LDA is the degradation of its performance when it is difficult to approximate the boundary between two classes with a linear function, if the boundary is non-linear. In this research, we implement the ensemble learning using LDA, and apply Error-Correcting Output Codes (ECOC) to LDA. Further, we introduce the weighted voting method to reduce the influence of poor hypotheses on the classification accuracy. The hypotheses are weighted according to their classification accuracy. Moreover, although LDA was originally designed for applying to only binary classification problems, it can be extended to the multi-class problems by introducing ECOC.

1　LDA

Linear Discriminant Analysis (LDA) [1] は，Fisher により提案された，2 クラスの分類を行う学習アルゴリズムである．LDA は，パラメトリックな学習アルゴリズムであり，訓練事例の写像の統計量に基づいて，線形関数で表される仮説を生成する．LDA の短所として，線形関数による分類を行うことに起因する表現能力の低さがあげられる．表現能力とは，クラス間の境界を仮説が 2 つのクラスの境界面を正確に近似する能力のことである．LDA の仮説は線形関数であるため，クラス間の境界面が線形に近い場合は境界面を精度よく近似することができるが，一般にクラス間の境界面は非線形であるので，線形関数では近似が困難であり，結果として分類精度が低下する．そこで，LDA の表現能力を改善するために，アンサンブル学習法 [2] を導入する．

[1]神戸大学大学院 自然科学研究科
[2]Graduate School of Science and Technology, Kobe University

2 LDA アルゴリズムの改善

2.1 アンサンブル学習

単体では性能の低い学習アルゴリズムであっても，いくつかの仮説を統合すれば性能を改善することができる．単一の学習アルゴリズムによって生成された仮説を組み合わせ，最終的に 1 つの仮説として分類を行う方法をアンサンブル学習という．

LDA は，訓練集合に含まれる事例が多少変化しても，仮説がその影響をほとんど受けない，安定な学習アルゴリズムなので，Boosting [3] のように訓練集合の部分集合を用いて複数の仮説を生成するアルゴリズムでは効果が低い．

LDA のように安定な学習アルゴリズムに対しても効果があり，LDA の表現能力を改善するアルゴリズムとして，Error-Correcting Output Codes (ECOC) [4] がある．ECOC は符号理論の誤り訂正符号に基づいており，各仮説に対してそれぞれのクラスに対応する符号語を割り当てることによって，訓練事例を 2 つのグループに分け，2 クラスからなる集合とみなして個々の仮説を生成する．このため，ECOC では，0, 1 の 2 値の値を要素に持つ行列を用いている．訓練事例が 4 クラスからなる場合の行列を **表 1** に示す．この行列の行数はクラスの数に等しく，列の数は生成される仮説の数に等しい．行列の各行を codeword と呼ぶ．以後，この行列を codeword 行列と呼ぶ．

表 1: codeword 行列

	h_1	h_2	h_3	h_4	h_5	h_6	h_7
class 1	1	1	1	1	1	1	1
class 2	0	0	0	0	1	1	1
class 3	0	0	1	1	0	0	1
class 4	0	1	0	1	0	1	0

Codeword 行列の各列は，任意の 2 列 h_i, h_j $(i \neq j)$ が同一，あるいは 0 と 1 を反転したときに同一にならないように選択される．したがって，クラス数が k のとき，可能な列，すなわち可能な仮説の数は $(2^{k-1}-1)$ 個となる．クラス数が増加すると，可能な列の数は指数関数的に増加するので，クラス数が多い場合は，適切な列を選択することが重要となる．

ECOC では，codeword の値が 0 であるクラスに属する事例を一方のグループ，値が 1 であるクラスに属する事例を他方のグループとし，この 2 つのグループを識別する仮説を生成する．例えば，表 1 について，h_1 はクラス 1 に属する事例とその他の 3 つのクラスに属する事例を識別する仮説である．事例を分類する際には，その事例に対する各仮説の出力と，各クラスの codeword の値を比較し，ハミング距離（h_i の値が相異なるものの数）が最も少ないクラスに分類する．事例を分類するこの操作は，符号理論で誤り訂正を行うことに相当する．

このように，ECOC では，訓練事例は属するクラスにしたがって 2 つのグループに分割されるので，その境界はグループの分け方によってそれぞれ大きく異なったものとなる．したがって，LDA を用いても互いに大きく異なる仮説を生成でき，それらを統合すれば，LDA を単体で実行した場合に比べて，精度の改善が期待できる．

2.2 分類精度による重み付け

ECOC を適用すると，各仮説は codeword 行列の列に基づいて生成される．したがって，列の選択が適切でないと，LDA は分類精度の高い仮説を生成できず，結果として，それらを統合した最終仮説の分類精度が悪くなることがある．**図 1** (a) は LDA が精度の高い仮説を生成できるグループ分けを示しており，図 1 (b) は精度の高い仮説を生成できないグループ分けを示している．色のついていない領域に属する 2 クラスが一方のグループで，色のついている領域に属する 2 クラスが他方のグループである．それぞれ，表 1 の h_5, h_2 に対応している．

図 1 (a) では，グループの境界がほぼ線形なので，正確に近似できるが，図 1 (b) では，線形関数により 2 つのグループを識別すること

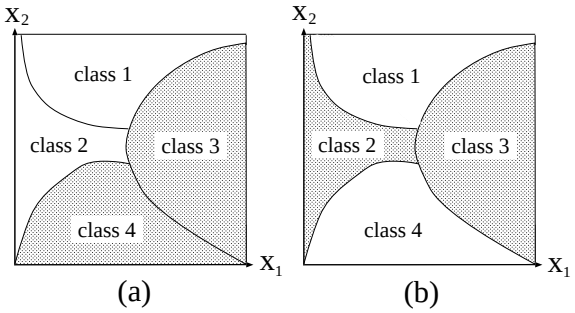

図 1: グループ分けの例

は困難である．したがって，LDA では境界を正確に近似することができず，分類精度の低い仮説が生成されてしまう．

このように，LDA に ECOC を適用する際，列の選択は分類精度に大きく影響する可能性があるので，適切な列を選択することが重要となる．しかし，特にクラス数が多い場合，適切な列を見つけることは非常に困難である．そこで，精度の低い仮説の影響を低減するために重み付き投票を導入する．

訓練集合に対する仮説の分類精度が低ければ，未知の事例に対する分類精度も低くなる可能性が高いので，訓練集合に対する分類精度を重みとして用い，仮説を統合する際に重み付けを行う．これにより，最終仮説 $H(x)$ は以下のようになる．

$$H(x) = \arg\min_{c} \sum_{i=1}^{n} a_i HD(h_i(x),\ h_i(c)). \quad (1)$$

ここで，n は仮説の数であり，a_i は i 番目の列の仮説，すなわち h_i の分類精度である．また，$HD(h_i(x),\ h_i(c))$ は，事例 x に対する仮説 h_i の出力と，クラス c に割り当てられている h_i の値の間のハミング距離を表す．$h_i(c)$ は，クラス c の i 番目の codeword の値に等しい．

式 (1) の a_i を求める際，すべての訓練事例から分類精度を計算することが望ましい．しかし，訓練事例数が非常に多い場合は，すべての事例から分類精度を計算すると計算量が非常に大きくなる．このような場合，適応サンプリングによって計算量を抑えることができる．適応サンプリングを用いれば，少数の訓練事例から訓練事例全体の分類精度を推定し，この推定値を重みとして用いて仮説を統合することができる．

3 性能の比較

本章では，提案アルゴリズムと他の学習アルゴリズムを，分類精度については実験的に，計算量については理論的に比較する．比較対象とする学習アルゴリズムとして，多クラス問題に拡張された LDA（以後 multi-class LDA と呼ぶ）と，C4.5 を弱学習器とする AdaBoost を用いた．multi-class LDA は，2 クラスの LDA を拡張したものであり，多クラス問題に対しても，訓練事例の統計量を用いて線形関数で表される仮説を生成する．

3.1 実験による分類精度の比較

本節では，提案アルゴリズムと multi-class LDA の比較実験の結果を示す．また，アンサンブル学習の観点から，AdaBoost との比較結果も示す．AdaBoost の弱学習器には C4.5 を用いており，ラウンド数は 100 としている．さらに，Support Vector Machine (SVM) [5] との比較結果も示す．LDA が訓練事例の統計量から仮説を生成しているのに対し，SVM はマージン，すなわち境界面に最も近い事例と境界面の間の距離を最大化して仮説を生成している．SVM の出力する仮説は線形関数としている．以上の条件のもとで，10-fold cross-varidation paired t-test を行い，それぞれの分類精度を比較した．この検定の有意水準は 5% としている．

実験結果を **表 2** に示す．なお，t 欄における * は，t 検定により提案アルゴリズムと multi-class LDA の間に有意な差があると判定されたことを表している．また，Error 欄における ECOC は提案アルゴリズム，mLDA は multi-class LDA，Ada は AdaBoost，SVM は SVM による結果をそれぞれ表している．

提案アルゴリズムと multi-class LDA を比較すると，**balance-scale**，**car**，**nursery** の 3 種類のデータセットについて，提案アルゴリズムでは非常に高い分類精度となっている．

表 2: 分類精度

Data set	Error(%)				
	ECOC	mLDA	Ada	SVM	t
bal	10.40	22.85	34.88	12.16	*
car	29.28	37.85	12.78	32.59	*
der	5.17	4.08	4.64	7.93	-
thy	7.07	5.58	6.84	4.19	-
nur	15.05	30.10	6.92	25.86	*
pos	62.44	63.55	42.78	58.98	*
wav	24.99	19.40	21.15	19.04	-

これは，それぞれのデータセットで，各クラスの境界面が線形関数では非常に近似困難であるが，提案アルゴリズムでは ECOC を適用することにより仮説の表現能力が増し，境界面を比較的良く近似できているためであると考えられる．AdaBoost と比較すると，多くのデータセットについて AdaBoost は LDA に基づく両アルゴリズムよりも高い分類精度を実現している．これは，弱学習器である C4.5 の出力する仮説の表現能力が LDA に比べて高いためと考えられる．しかし，計算量の点からは，全体的に AdaBoost の計算量は大きく，計算時間も多く必要としている．SVM と比較すると，分類精度の点からは，全体的には SVM の方が良い結果となっている．しかし，線形 SVM を用いているため，multi-class LDA と同様，3 種類のデータセットについては，提案アルゴリズムは SVM と比べて高い分類精度となっている．また，計算量は提案アルゴリズムの方が SVM よりも概して小さい．

3.2 計算量の理論的比較

学習アルゴリズムの理論的な比較のため，ここでは計算量を考える．SVM および弱学習器として C4.5 を用いた AdaBoost を比較対象とする．それぞれの学習アルゴリズムの計算量を**表 3** に示す．表 3 において，n は事例数，d は属性数，k はクラス数を表す．また，n' は分類精度を推定する際，適応サンプリングによりサンプリングする事例数を表し，$n' \leq n$ である．SVM における反復回数は，二次計画問題を解く際に行われる反復の回数を表す．

表 3: 各アルゴリズムの計算量

ECOC	$O(k) \times \max\{O(nd^2), O(d^3), O(kn'd)\}$
mLDA	$\max\{O(nd^2), O(kd^2), O(d^3)\}$
Ada	$O(k) \times$ (ラウンド数) $\times O(n^2 \log n)$
SVM	$O(k) \times$ (反復回数) $\times O(nd)$

SVM や C4.5 の計算量は主に事例数に比例して増加するが，LDA の計算量は主に属性数に比例して増加する．したがって，計算量の点から，LDA に基づくアルゴリズムは，事例数に比べて属性数が小さい場合に効果的であると言える．しかし，LDA は他のアルゴリズムと比べて，オーダーには現れない定数部分が小さいので，本実験では概して LDA に基づくアルゴリズムの計算量は小さくなっている．

参考文献

[1] Fisher, R. A.: The use of multiple measurements in taxonomic problems, *Annual Eugenics*, Vol. 7, No. 2, pp. 179-188 (1936).

[2] Dietterich, T. G.: Ensemble Learning, The Handbook of Brain Theory and Neural Networks, Cambridge, MA: MIT Press (2002).

[3] Schapire, R. E.: Theoretical Views of Boosting, *Proceedings of Fourth Euro-COLT'99*, pp. 1-10 (1999).

[4] Dietterich, T. G. and Bakiri, G.: Solving multiclass learning problems via error-correcting output codes, *Artificial Intelligence Research 2*, pp. 263-286 (1995).

[5] Vapnik, V.: Three Remarks on the Support Vector Method of Function Estimation, *Advances in Kernel Methods - Support Vector Learning*, Cambridge, MA: MIT Press, pp. 25-41 (1999).