

# LS-SVM 群を複合した多クラス画像識別器のロバスト化

野瀬 裕馬<sup>1,a)</sup> 川上 肇<sup>2,b)</sup>

**概要:** 本論文は、多クラス識別器でロバスト画像識別を実現するための方法に関して、その有効性を議論している。1 bit の誤り訂正機能が備わる ECOC を用いて 2 クラス識別器を複合化した多クラス識別器に対して、ロバスト推定の手法を応用した識別面のロバスト学習と、2 クラス識別器を LS-SVM で構成した場合に有効な識別方法が提案されている。LS-SVM 群を複合化した画像識別器と L1-SVM 群を複合化した画像識別器とにこれらの方法を適用して、提案方法の性能を評価した。

**キーワード:** ロバスト画像識別, LS-SVM, L1-SVM, 複合, 誤り訂正出力コード

## Classification Methods for Robust Image Matcher using Multiple LS-SVMs

YUMA NOSE<sup>1,a)</sup> HAZIMU KAWAKAMI<sup>2,b)</sup>

**Abstract:** This paper discusses feasibility of the method for a robust image classification by a multi-class classifier. For a multi-class classifier constructed with multiple 2-class classifiers using ECOC capable of correcting 1 bit error, a robust method for learning decision boundaries based on robust estimation and a classification method for the multi-class classifier constructed with multiple LS-SVMs are proposed. We evaluate capability of these methods by applying them to an image classifier constructed with multiple LS-SVMs and one done with multiple L1-SVMs.

**Keywords:** Robust image recognition, LS-SVM, L1-SVM, multiple classifier systems, error-correcting output code.

### 1. はじめに

2 クラスの識別問題に対する識別能力と応用性の高い識別器として、SVM が提案されている。これを複合的に組み合わせた SVM 群で多クラスを識別する方法も盛んに研究されている [1], [2], [3]。特に、SVM の発展型として提案された LS-SVM (Least Square Support Vector Machine) [4] を 2 クラスの識別器に用いて多クラス識別器を構成すれば、汎化能力が高まる可能性が報告されている [5]。しか

しながら、このような多クラス識別器のロバスト化に関する研究は極めて少ないのが現状である。

そこで我々は、LS-SVM 群を複合して構成した多クラス識別器のロバスト化を下記の方針で検討している [6]。複合のために、誤り訂正出力コード (error-correcting output code:ECOC) [9], [10], [11], [12] の考え方を導入する。そのうえで、ECOC の特性を考慮したロバストな識別器の学習方法と LS-SVM の特性を考慮した識別方法を提案している。まず、1 bit の誤り訂正機能が備わる ECOC を導入したうえで、学習時に 2 bit 以上のエラーを生じさせるデータを訓練データから除外した後に再学習することにより、識別器のロバスト化を図る。次に、LS-SVM では特徴空間にある分離超平面付近のマージン領域に多くの入力データが正しく写像されるので、マージン領域内で正しく識別されるデータに対して入力データと各クラスとの距離が増加

<sup>1</sup> 龍谷大学大学院理工学研究科電子情報学専攻  
Department of Electronics and Information, Graduate School  
of Science and Technology, Ryukoku University

<sup>2</sup> 龍谷大学理工学部電子情報学科  
Department of Electronics and Information,  
Faculty of Science and Technology, Ryukoku University  
a) t12m031@mail.ryukoku.ac.jp

b) kawakami@rins.ryukoku.ac.jp

	制約条件 $i=1, \dots, l$	目的関数 $L(\omega) \rightarrow$ 最小
ハード マージン SVM	$y_i(\phi(x_i) \cdot \omega - h) \geq 1$	$\frac{1}{2} \ \omega\ ^2$
LS-SVM	$y_i(\phi(x_i) \cdot \omega - h) + e_i = 1$	$\frac{1}{2} \ \omega\ ^2 + C \sum_{i=1}^l e_i^2$

図 1 SVM と LS-SVM の比較表

Fig. 1 Comparison between SVM and LS-SVM.

しないように識別計算を設計する。

ソフトマージンを導入した SVM でも訓練データの一部がマージン領域内に分布した状態で学習が終了するので、提案方法は有効であると思われる。そこで、4クラスの画像群を変動が混入する悪環境下で識別する問題を、LS-SVM 群を複合した識別器と L1-SVM 群を複合した識別器のそれぞれで処理することにより、提案法の有効性を調べた。

## 2. LS-SVM

以下ではクラスの個数を 2 とする。  $l$  個の観測データが与えられているとき、それぞれの観測データは、特徴ベクトル  $x_i \in R^d, i=1, \dots, l$  とそれに割り当てられたクラス  $y_i \in \{1, -1\}$  の組から成るとする。  $x_i$  の各要素は計測された各特徴量を表す。図 1 に示すように、ハードマージン SVM(以下 SVM) では同図に示す目的関数を

$$\text{制約条件: } y_i(\phi(x_i) \cdot \omega - h) \geq 0, i=1, \dots, l \quad (1)$$

の下で最小化することで、特徴ベクトルのクラスを決定する識別面  $H_0: D(x) = 0$  を求める。ここで、  $D(x) = \phi(x) \cdot \omega - h$  は決定関数、  $\phi(\bullet)$  は  $d$  次元の入力空間から  $\tilde{d}$  次元の特徴空間への写像、  $\omega$  は  $\tilde{d}$  次元係数ベクトル、  $h$  はバイアス項である。LS-SVM は SVM を 2 つの点で改良し、学習過程の効率化を図る。まず、1 つ目の改良は制約条件の不等号を等号に変えることである。そのために、  $y_i = 1$  としたとき、クラス 1 の分離超平面  $\phi(x) \cdot \omega - h = 1$  からもう一方のクラス -1 の分離超平面  $\phi(x) \cdot \omega - h = -1$  に向かって測った特徴ベクトル  $x_i$  の変位をエラー変数  $e_i, i=1, \dots, l$  で表し、これを下記のように制約条件に導入する：

$$\text{制約条件: } y_i(\phi(x_i) \cdot \omega - h) = 1 - e_i, i=1, \dots, l \quad (2)$$

導入したエラー変数  $e_i$  を図 2 に矢印で例示する。ここで、+印はそれぞれの SVM におけるクラス 1、青い○はクラス-1を表す。2 つ目の改良はエラー変数の 2 乗和を付加した目的関数を導入することである。

$$\text{目的関数: } L(\omega) = \frac{1}{2} \|\omega\|^2 + C \sum_{i=1}^l e_i^2 \quad (3)$$

これらの 2 つの改良により、目的関数の最小化問題が非常に簡単になる。ここで新たに導入したパラメータ  $C$  は適当な正の定数である。

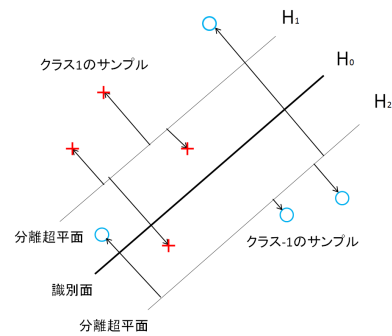


図 2 LS-SVM でのエラー変数  $e_i$

Fig. 2 Error variable  $e_i$  in LS-SVM.

クラス番号	0	1	2	3
2進数	00	01	10	11
SVM <sub>0</sub>	-1	-1	1	1
SVM <sub>1</sub>	-1	1	-1	1

図 3 4 クラスコーディング (誤り訂正機能なし)

Fig. 3 4 class coding (without error correction).

## 3. LS-SVM の複合

1 つの SVM では 2 つのクラスしか識別することが出来ない。そこで、2 クラス識別器である SVM を複合的に用いることによって多クラス識別器を構成する。  $k (= 2^n)$  個までのクラスを識別できる識別器を構成するために、例えば  $n$  個の SVM を用いる。クラス名を 2 進数  $X_1, \dots, X_n$  でコード化し、その  $j$  番目のビットに割り当てられた  $j$  番目の SVM, SVM <sub>$j$</sub>  で  $j$  番目のビット  $X_j$  を識別する。したがって、各 SVM はそれぞれ異なった決定関数を持つことになる。特徴ベクトル  $x_i$  に対する各 SVM の出力が 1 なら 1、-1 なら 0 として 1 ビットを決定する。  $n$  個の LS-SVM 群から出力される  $n$  bit のコードを多クラス識別器の出力とする。4 クラス識別器を構成する LS-SVM 群とクラス名との関係を図 3 に例示する。上記の考え方に基いて構成される多クラス識別器に対して雑音が混入する状況下でのロバスト性が備わるように、誤り訂正出力コードでクラス名をコード化する方法が提案されている。そこで、1 bit の誤り訂正機能が備わる ECOC の使用を前提として、その方法を LS-SVM の特性に合わせて改良する方法を導入する [6]。

### 3.1 ロバスト化学習

SVM の学習では一部の訓練データだけで識別面が構成されるけれども、LS-SVM の学習ではすべての訓練データによって識別面が構成される。この LS-SVM に固有の特性は、不適切な訓練データが識別面の構成に悪影響を及ぼす可能性を示唆する。そこで、ECOC を導入した多クラス識別器に合った方法でロバスト推定法 [7] の考え方を下記のように適用することにより、この悪影響を除去する：

**Step1** 1 bit の誤り訂正機能が備わるようにクラス名を

$j \backslash i$	0	1	2	3
SVM <sub>0</sub>	1	-1	0	0
SVM <sub>1</sub>	1	0	-1	0
SVM <sub>2</sub>	1	0	0	-1
SVM <sub>3</sub>	0	1	-1	0
SVM <sub>4</sub>	0	1	0	-1
SVM <sub>5</sub>	0	0	1	-1

図 4 4 クラスに対する統一のコーディング

Fig. 4 The unifying coding for 4 classes.

コード化した後、全ての訓練データを用いて各 LS-SVM の学習を実行する。その結果、各 LS-SVM が個別に識別面を構成する。

**Step2** 訓練データに対する多クラス識別器の出力コードに着目する。この識別器には 1 bit の誤り訂正機能しか備わっていないので、その出力コードに 2 bit 以上の誤りを生じさせる訓練データは、誤りが生じたビットの識別を担当している LS-SVM の学習に悪影響を及ぼす外乱データであると判断する。

**Step3** すべての訓練データから、**Step2** で外乱データであると判断されたデータを除去したデータのセットを新たに訓練データとして、全ての LS-SVM の再学習を実行し、以上で、多クラス識別器の学習を終了する。

### 3.2 ECOC を用いた識別

図 4 を用いて 4 クラスの統一のコーディング例を示す。i 番目のクラス名を n bit の ECOC でコード化するために、同図に示すように j 番目の SVM (SVM<sub>j</sub>) が識別すべき 2 値のクラス名 g<sub>ij</sub> に対してドントケアを表す 0 を含めた統一のコーディング [8] がある。このようなコードを用いた上で、SVM に特徴ベクトル x が入力された時、次のようにして x のクラスを決定する。まず、x のクラスが i であると仮定する。次に、その仮定に対する SVM<sub>j</sub> が表す決定関数値 D<sub>j</sub>(x) のエラー ε<sub>ij</sub>(x) を次式で定義する：

$$\varepsilon_{ij}(x) = \begin{cases} 0 & (g_{ij} = 0) \\ \max(1 - g_{ij}D_j(x), 0) & (g_{ij} \neq 0) \end{cases} \quad (4)$$

ε<sub>ij</sub>(x) の作用は次の通りである。g<sub>ij</sub> = 0 は無視すべきドントケア出力であるので ε<sub>ij</sub>(x) = 0 とする。g<sub>ij</sub> ≠ 0 のとき、g<sub>ij</sub>D<sub>j</sub>(x) ≥ 1 とすると SVM<sub>j</sub> が表す分離超平面の正しいクラス側に x はあり、しかも決定関数値 D<sub>j</sub>(x) が 1 以上あることになる。したがって、エラーは ε<sub>ij</sub>(x) = 0 とする。一方、g<sub>ij</sub>D<sub>j</sub>(x) < 1 のとき、x は SVM<sub>j</sub> が表す識別面の間違った側にあるか、或いは正しい側でも決定関数値が 1 より小さい領域にある。したがって、エラーを ε<sub>ij</sub>(x) = 1 - g<sub>ij</sub>D<sub>j</sub>(x) とする。その上で、クラス i と特徴ベクトル x との距離を

$$d_i(x) = \sum_{j=1}^k \varepsilon_{ij}(x) \quad (5)$$

で定義し、これにより、特徴ベクトル x を次のクラスに分

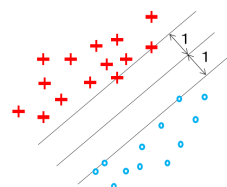


図 5 SVM に対する訓練データの分布

Fig. 5 Distribution of training data for SVM.

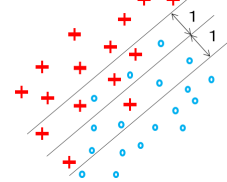


図 6 LS-SVM に対する訓練データの分布

Fig. 6 Distribution of training data for LS-SVM.

類する。

$$\arg \min_{i \in \{0, 1, \dots, n-1\}} d_i(x) \quad (6)$$

### 3.3 LS-SVM に合わせたエラーの改造

2 クラス識別器に通常の SVM を用いて上記の方法を実行する場合、各 SVM は図 5 に例示するように、決定関数値が 1 より小さくなる領域内には訓練データが存在しない状態で識別面の学習が完了する。これと対比して、2 クラス識別器に LS-SVM を用いた場合を考える。この場合、図 6 に例示するように、識別面の正しい側で決定関数値が 1 より小さくなる領域に訓練データが多数存在する状態で識別面の学習が完了する。その結果、クラスが未知のデータを識別する際にも、識別面の正しい側で決定関数値が 1 より小さくなる可能性が高い。この場合、識別結果は正しいのでペナルティを与えるべきではない。そこで、2 クラス識別器に LS-SVM を用いた場合のエラー ε<sub>ij</sub>(x) の計算では、式 (4) にある 1 - g<sub>ij</sub>D<sub>j</sub>(x) をパラメータ M ∈ [0, 1] を用いて M - g<sub>ij</sub>D<sub>j</sub>(x) に変更したエラー関数：

$$\varepsilon_{ij}(x) = \begin{cases} 0 & (g_{ij} = 0) \\ \max(M - g_{ij}D_j(x), 0) & (g_{ij} \neq 0) \end{cases} \quad (7)$$

を式 (5) と式 (6) に併せて使用する方法を提案する。

## 4. 実験

まず、§3.1 で説明したロバスト化学習と §3.3 に示した方法を導入して構成した LS-SVM 群による 4 クラス識別器を構成する。そのうえで、それに備わるロバスト性を調べるために、変動が混入する状況で、図 7~ 図 10 に例示する自然のテクスチャ画像を識別する実験を行った。次に、§3.3 に示した方法の特性を調べるために、ソフトマージン SVM の一例である L1-SVM 群を複合した識別器によっても同様の実験を行った。ここで L1-SVM では目的関数と制約条件が次のように設定される：

$$\text{目的関数: } L(\omega) = \frac{1}{2} \|\omega\|^2 + C \sum_{i=1}^l e_i \quad (8)$$

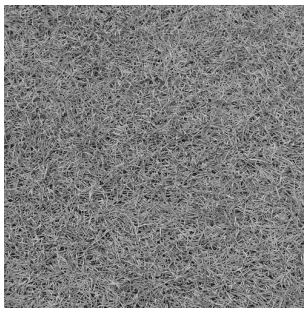


図 7 クラス 0 (芝生)

Fig. 7 Class 0(Grass).



図 8 クラス 1 (石)

Fig. 8 Class 1(Stone).

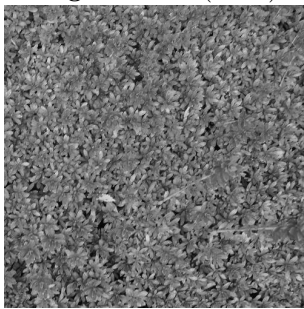


図 9 クラス 2 (短い木)

Fig. 9 Class 2(Short tree).



図 10 クラス 3 (落ち葉)

Fig. 10 Class 3(Fallen leaves).

制約条件:  $y_i(\phi(x_i) \cdot \omega - h) \geq 1 - \varepsilon_i, i = 1, 2, \dots, l(9)$   
なお、実験では  $\sigma = 10$  の RBF カーネルを使用する。

#### 4.1 方法

実験手順の概要は以下の通りである：

- (1) 訓練データの生成 クラス  $j$  の訓練データとして用意したテクスチャ画像  $I_j$  の全面に亘って、 $41 \times 41$  画素の部分画像群を 625 枚切り出し、各部分画像から抽出した 1681 個の特徴ベクトル群でクラス  $j$  の訓練データを構成する。
- (2) 識別器の学習 図 4 に従って、6 個の 2 クラス識別器を複合して 4 クラス識別器を構成し、各識別器に対して、上記の訓練データを用いたロバスト化学習を行う。
- (3) テストデータの生成 クラス  $j$  のテストデータとして用意したテクスチャ画像  $J_j$  の各画素に、一様乱数  $\in [0, z]$  を外乱として混入させた後、訓練データと同様にして、その画像からクラス  $j$  のテストデータを構成する。ここで、パラメータ  $z (\geq 0)$  は外乱の大きさを表すので、 $z$  を変動強度と呼ぶ。
- (4) 評価方法 §3.3 に示した方法を使用することにより、学習済みの 4 クラス識別器でテストデータの特徴ベクトルを識別した後、その識別率  $\mu$  を次式で算出する：

$$\mu = \frac{\text{正しく識別された特徴ベクトル数}}{\text{テストデータに含まれる特徴ベクトル数}} \quad (10)$$

上記において特徴ベクトルは下記の手順で生成した：

- (1) 白色化 PCA 部分空間 すべての訓練データの全ての部分画像を、画素値を並べてベクトル化した後、これらを主成分分析する。その結果求まる大きさ 1 の固有ベクトル  $u$  のうち、対応する固有値  $\lambda$  が大きい上位 35

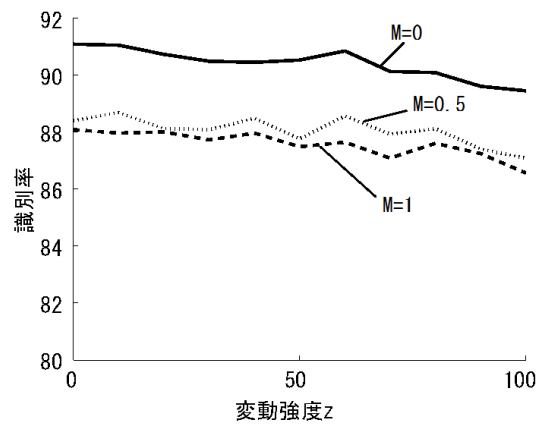


図 11 LS-SVM の識別率 (ロバスト化学習有り) ( $C=500$ )

Fig. 11 Classification ratio of LS-SVM with the robust learning( $C=500$ ).

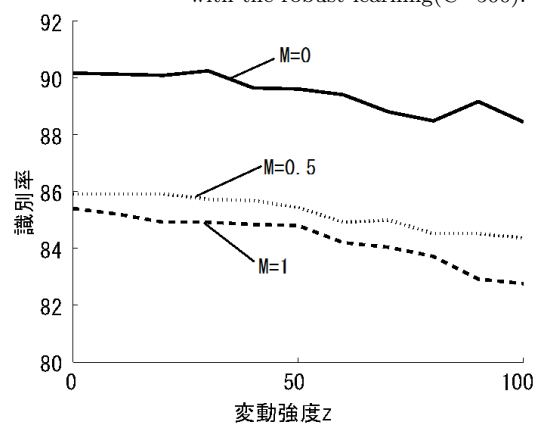


図 12 LS-SVM の識別率 (ロバスト化学習無し) ( $C=500$ )

Fig. 12 Classification ratio of LS-SVM without the robust learning( $C=500$ ).

個の固有ベクトル群で構成した  $\{\sqrt{\lambda}u\}$  を基底とする空間を構成する。この空間を白色化 PCA 部分空間と呼ぶ。

- (2) 特徴ベクトルの構成 訓練データとテストデータとなる各部分画像の画素値を並べたベクトルを白色化 PCA 部分空間に写像してそれぞれの特徴ベクトルとする。

#### 4.2 実験結果

まず、ロバスト化学習を行った後に、式 (7) で  $M = 0, 0.5, 1.0$  に設定したうえで、テストデータに混入させる外乱の変動強度  $z$  を 0 から 100 まで増加させながら、LS-SVM 群で画像識別実験を行った。式 (3) の目的関数で  $C = 500$  とした時の識別率  $\mu$  を、変動強度  $z$  を横軸、識別率  $\mu$  を縦軸とした平面上的グラフにして図 11 に示す。次に、ロバスト化学習の効果調べるために、これを用いない学習を適用して同様の実験を行った。その結果を図 12 に示す。さらに L1-SVM 群で構成した識別器でも同様の実験を行ったときの識別率を図 13 と図 14 に示す。

#### 4.3 考察

LS-SVM では、図 11 と図 12 とからロバスト化学習により識別率が改善されていることがわかる。さらに、 $M$  を

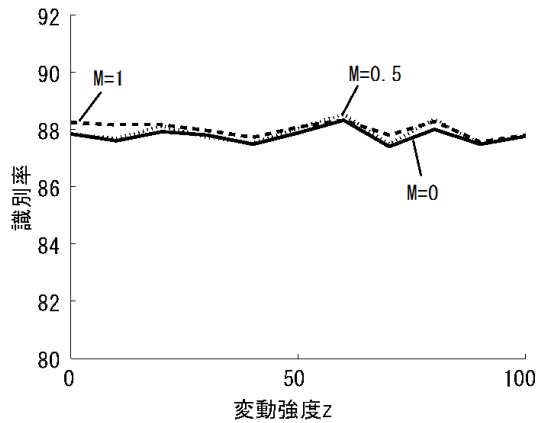


図 13 L1-SVM の識別率 (ロバスト化学習有り) (C=500)

Fig. 13 Classification ratio of L1-SVM with the robust learning(C=500).

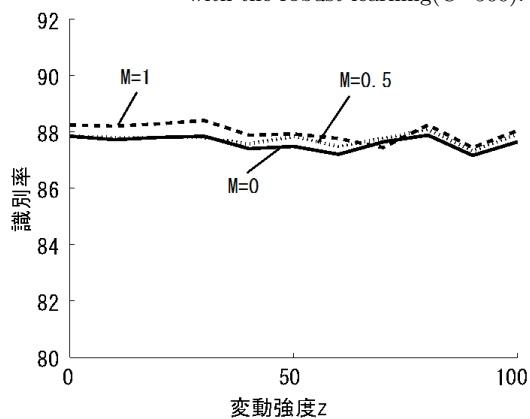


図 14 L1-SVM の識別率 (ロバスト化学習無し) (C=500)

Fig. 14 Classification ratio of L1-SVM without the robust learning(C=500).

1 から減少させて  $M = 0$  にすることで、識別率が高まっている。これにより、§3.3 に示した方法は LS-SVM に有効であることがわかる。これらの効果の現れ方を調べるために、変動強度  $z = 0$  としたクラス 0 のテストデータに対する LS-SVM<sub>0</sub> の決定関数値の散布図を求めた。ここで、LS-SVM<sub>0</sub> はこれらを 1 に分類するよう学習している。図 15 はロバスト化学習を行ったときの散布図、図 16 はロバスト化学習を用いずに通常の学習を行ったときの散布図である。これらの図より、ロバスト化学習によりマージン領域内に写像されるテストデータが減少していることがわかる。この効果で識別率が改善されていると思われる。

対比して、L1-SVM では図 13 と図 14 とから、ロバスト化学習の効果は現れていない。また、 $M$  を変化させても識別率の変化は少ない。したがって、§3.3 の改造は L1-SVM に対して有効でないように思える。 $C = 500$  とした場合に L1-SVM 群を複合した識別器にロバスト化学習の効果が見れないことは、訓練データに対する識別率が高いことを意味する。図 17 にクラス 0 と 1 の訓練データに対する通常に学習を行った L1-SVM<sub>0</sub> の決定関数値を散布図で示す。同図により、L1-SVM<sub>0</sub> はすべての訓練データを正しく識

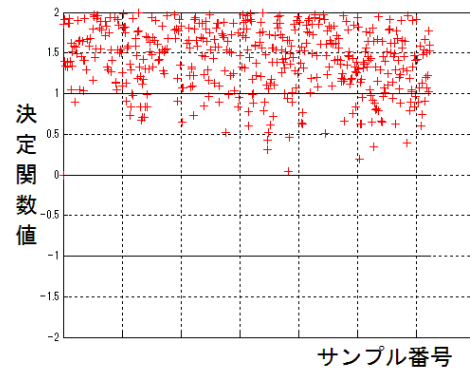


図 15 クラス 0 のテストデータに対する LS-SVM<sub>0</sub> の出力 (ロバスト化学習有り) (C=500,z=0)

Fig. 15 Output of LS-SVM<sub>0</sub> for the class 0 test data with the robust learning(C=500,z=0).

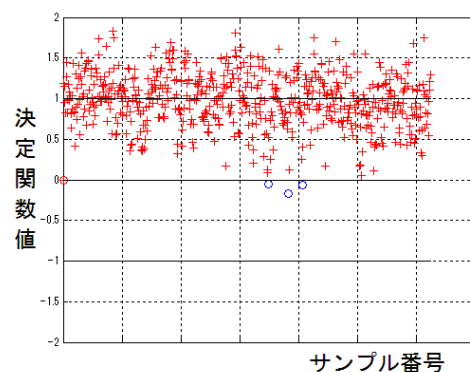


図 16 クラス 0 のテストデータに対する LS-SVM<sub>0</sub> の出力 (ロバスト化学習無し) (C=500,z=0)

Fig. 16 Output of LS-SVM<sub>0</sub> for the class 0 test data without the robust learning(C=500,z=0).

別している事がわかる。そこで、学習時に誤認識が生じる状態で実験を続けた。図 18 に、 $C = 1$  としたときのクラス 0 と 1 の訓練データに対する通常に学習した L1-SVM<sub>0</sub> の決定関数値を散布図で示す。また、このときの識別率を図 19 と図 20 に示す。図 19 はロバスト化学習を L1-SVM 群の学習に適用したときのテストデータに対する識別率、図 20 は通常の学習を適用したときの同様の結果である。これらの図より、L1-SVM 群のロバスト化学習で識別率は劣化しているけれども、 $M$  を減少させることで識別率は改善されていることがわかる。これにより、§3.3 に示す方法は L1-SVM 群で構成した識別器に対しても有効であると考えられる。ロバスト化学習による劣化は、除去される訓練データが多すぎる結果、学習不足になったのではないと思われる。

## 5. むすび

LS-SVM 群の複合により構成した多クラス識別器をロバスト化するための学習方法と識別方法を、L1-SVM 群の複合により構成した識別器にも適用し、変動が混入する悪環境下で性能比較実験を行った。その結果、LS-SVM 群で構成した識別器に提案法の効果が現れること、L1-SVM でも

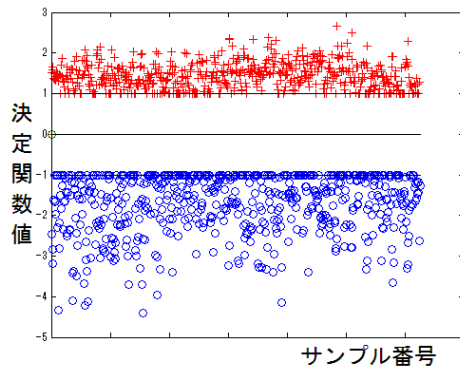


図 17 訓練データに対する L1-SVM<sub>0</sub> の出力 (ロバスト化学習無し) (C=500)

Fig. 17 Output of L1-SVM<sub>0</sub> for the training data without the robust learning(C=500).

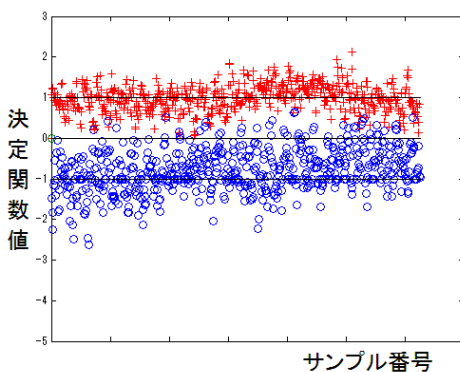


図 18 訓練データに対する L1-SVM<sub>0</sub> の出力 (ロバスト化学習なし) (C=1)

Fig. 18 Output of L1-SVM<sub>0</sub> for the training data without the robust learning(C=1).

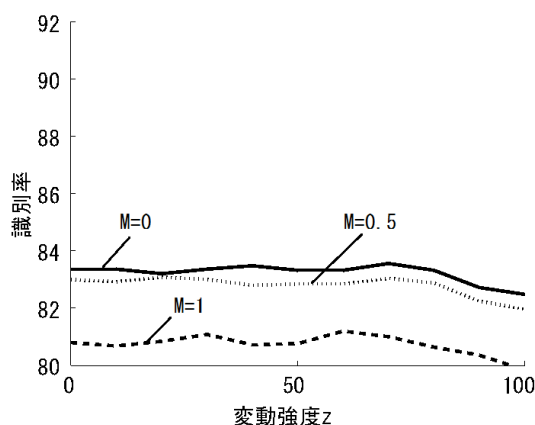


図 19 L1-SVM の識別率 (ロバスト化学習有り) (C=1)

Fig. 19 Classification ratio of L1-SVM with the robust learning(C=1).

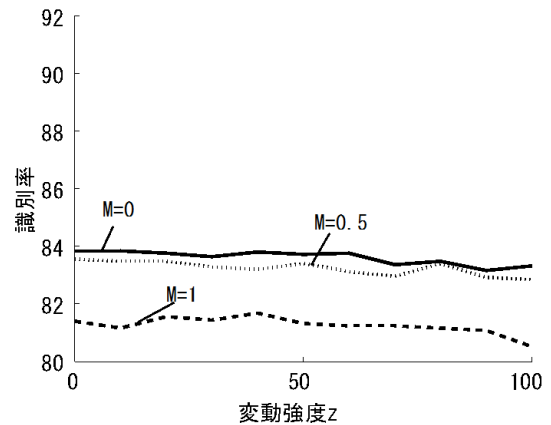


図 20 L1-SVM の識別率 (ロバスト化学習無し) (C=1)

Fig. 20 Classification ratio of L1-SVM without the robust learning(C=1).

参考文献

- [1] Kai Lienemann, Thomas Plotz, and Gernot A. Fink, On the Application of SVM-Ensembles Based on Adapted Random Subspace Sampling for Automatic Classification of NMR Data, M. Haindl, J. Kittler, and F. Roli(Eds.):MCS2007,LNCS4472,pp.42-51,(2007)
- [2] Albert D. Shieh and David F. Kamm, Ensembles of One Class Support Vector Machines, J. A. Benediktsson, J. Kittler, and F. Roli(Eds.):MCS2009, LNCS5519,pp.181-190,(2009)
- [3] Kai ming Ting and Lian Zhu, Boosting Support Vector Machines Successfully,J. A. Benediktsson, J. Kittler, and F. Roli(Eds.):MCS2009, LNCS5519,pp.509-518, (2009)
- [4] J. A. K. Suykens, T. Van Gestel, J. De Brabanter, B. De Moor, J. Vandewalle, "Least Squares Support Vector Machines", World Scientific Pub. Co., Singapore, 2002.
- [5] 阿部重夫, "パターン認識のためのサポートベクトルマシン入門", 森北出版, 2011.
- [6] 清水 紀貴, and 川上 肇,LS-SVM 群の複合によるロバスト画像識別, 信学技報 IEICE Technical Report PRMU2011-143(2011-2012), pp.107-112,(2011-2012)
- [7] Frank R. Hampel, Elvezio M. Ronchetti, Peter J. Rousseeuw, Werner A. Stahel, "Robust Statistics The Approach Based on Influence Functions", Wiley Series in Probability and Statistics, 1986.
- [8] E. L. Allwein, R. E. Schapire, and Y. Singer, Reducing multiclass to binary, "A unifying approach for margin classifiers", Journal of Machine Learning Research, 1:113-141,2000.
- [9] T. G. Dietterich and G. Nakiri: "Solving multiclass learning problems via error-correcting output codes", Journal of Artificial Intelligence Research, 2:263-286,1995.
- [10] Elizabeth Tapia, Jose C. Gonzalez, Alexander Huttermann, and Javier Garcia, Beyond Boosting : Recursive ECOC Learning Machines, F.Roli, J. Kittler, and T. Windatt(Eds.):MCS2004,LNCS 3077,pp.62-71,(2004)
- [11] Claudio Marrocco, Paolo Simeone, and Francesco Tortorella, Embedding Reject Option in ECOC Through LDPC Codes, M. Haindl, J. Kittler, and F. Roli(Eds.):MCS2007,LNCS4472,pp.333-343,(2007)
- [12] Sergio Escalera, Oriol Pujol, and Petia Radeva, Recoding Error-Correcting Output Codes, J. A. Benediktsson, J. Kittler, and F. Roli(Eds.):MCS2009, LNCS5519,pp.11-21,(2009)

ソフトマージンの動作を調節すれば効果が現れることがわかった. 今後, ロバスト化学習やエラー関数の最適化を行い, 安定に識別することができるクラス数を増やすことが課題となる.