

2 群及び 3 群のデータセットを用いた
複数機械学習手法の実装と評価

s013038

齋藤 浩

応用情報学講座

田中研究室

第 1 章	序論	4
1.1	研究の背景	4
1.2	研究の概要	4
第 2 章	分類手法	5
2.1	用語説明	5
2.2	代表ベクトルからのユークリッド距離による分類	5
2.2.1	代表ベクトルとユークリッド距離	5
2.2.2	2次元2群データのユークリッド距離による分類の例	5
2.3	ベイズ識別	7
2.3.1	ベイズの定理	7
2.3.2	ベイズの定理の証明	7
2.3.3	事後確率最大化識別(ベイズ識別)	7
2.3.4	多次元正規分布	8
2.3.5	分散共分散行列の RANK と行列式	8
2.3.6	ベイズ識別の実際の計算	8
2.3.7	2次元2群データのベイズ識別による分類の例	9
2.4	線形 SVM	10
2.4.1	SVM とは	10
2.4.2	基本的な考え方	10
2.4.3	ラグランジュ未定乗数法	11
2.4.4	最急降下法	12
2.4.5	2次元2群データの線形 SVM による分類の例	12
2.5	非線形 SVM	14
2.5.1	カーネルトリック	14
2.5.2	カーネル関数	14
2.5.3	2次元2群データの非線形 SVM による分類の例	15
第 3 章	誤り率を小さくする手法	16
3.1	フィルタによるブースティング	16
3.1.1	実装したブースティングのフローチャート	16
3.1.2	誤り率の減少する理由	16
3.2	adaboost	17
3.2.1	実装した adaboost のフローチャート	18
第 4 章	実験環境	20
4.1	動作コンピュータ	20
4.2	Matlab について	20

4.3	分類するテストデータ.....	20
第5章	分類実験.....	22
5.1	実験の方法.....	22
5.2	分類の実験と考察.....	22
5.2.1	テストデータ1 [直線]の分類の結果.....	22
5.2.2	テストデータ2 [2群ガウス]の分類の結果.....	23
5.2.3	テストデータ3 [正弦曲線]の分類の結果.....	25
5.2.4	テストデータ4 [円]の分類の結果.....	28
5.2.5	テストデータ5 [3群ガウス].....	30
第6章	考察とまとめ.....	32
第7章	謝辞.....	34
第8章	文献.....	34

第1章 序論

1.1 研究の背景

分類は文字認識、音声認識、画像認識などさまざまな領域で利用・応用されており、分類の基本原理を理解することは重要である。

分類とは最終的に異なる 2 つの間に境界を設けることである。それらの代表的な方法として、代表ベクトルによるユークリッド距離、事後確率最大化識別(ベイズ識別)があり、機械学習の手法として近年注目されている SVM(Support Vector Machine)[1]がある。機械学習とは分類を計算により自動的に行うことである。

また、分類の精度を向上させるためにブースティングおよび adaboost[2]などの手法が考案されている。これらは、弱い識別器を組み合わせることで、強い識別器を作ることはいかに、という考えによるものである。

1.2 研究の概要

分類の代表的な手法である代表ベクトルからのユークリッド距離、ベイズ識別、近年注目されている強力な分類手法である SVM を利用した分類を実装し、それらを誤り率に関して比較をおこなった。また、弱い分類手法を重ね合わせることで強い分類手法を構成するブースティングおよび adaboost について実装し、ブースティング、adaboost による分類性能向上性について検証する。

具体的には、いくつかのテスト用に生成したデータに対し、代表ベクトルからのユークリッド距離、ベイズ識別、線形 SVM、非線形 SVM によってテストデータの分類を行い、分類の誤り率を計算する。また、それぞれに対し、フィルタによるブースティングおよび adaboost を適用し、誤り率の変化を調べた。

第2章 分類手法

2.1 用語説明

特徴ベクトル x_i は 1 行 N 列からなるベクトルとする。以後ベクトルと記す。また、ベクトル x_i が属するクラスのクラスラベルを y_i とする。

クラス c に属するベクトルとは、対応するクラスラベル $y_i = c$ であるベクトルの集合とする。

データとは、ベクトルの集合とする。

学習とは分類器が必要とするパラメータをデータから求めることである。

教師データとは、学習に利用するベクトルの集合とする。

試験データとは、学習に利用しないベクトルの集合とする。

x^T とは行ベクトル x の転置である。

2.2 代表ベクトルからのユークリッド距離による分類

2.2.1 代表ベクトルとユークリッド距離

各クラスのデータが、クラスごとに局所的にまとまったものである場合には、各クラスを 1 つの代表ベクトルであらわすことができると考えられる。具体的には、クラス c に属する教師データから相加平均ベクトル μ_c を生成し、これをクラス c の代表ベクトルとする。各クラスの代表ベクトルと試験データ内のデータとのユークリッド距離の 2 乗 D_c を計算し、この距離 D_c がもっとも短くなるクラス c へ分類する。距離 D_c の計算は以下の式によって計算する[3]。

$$D_c(x) = \|x - \mu_c\|^2 = (x - \mu_c)(x - \mu_c)^T \quad \text{eq. 1}$$

以後ユークリッド距離による分類と記す。

2.2.2 2次元2群データのユークリッド距離による分類の例

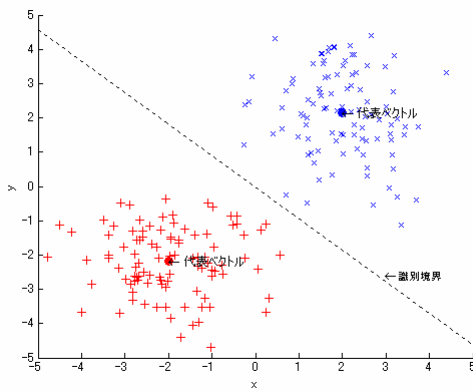


図 1

グラフの軸 縦軸 y 、横軸 x

正規乱数により作成

青 \times 平均 $(x, y) = (1.9850, 2.1398)$

x の分散 0.8685

y の分散 1.5714

赤 $+$ 平均 $(x, y) = (-1.9830, -2.1760)$

x の分散 1.2231

y の分散 0.9300

図 1 2次元ベクトルのユークリッド距離による識別境界は1次直線になる。

2.3 ベイズ識別

2.3.1 ベイズの定理

事象 A_1, \dots, A_K を互いに排反な全事象の分割とする。このとき、任意の事象 B に対して、

$$p(A_i | B) = \frac{p(B | A_i) p(A_i)}{\sum_{i=1}^K p(B | A_i) p(A_i)} \quad \text{eq. 2}$$

となる。

2.3.2 ベイズの定理の証明

事象 A_1, \dots, A_K を互いに排反な全事象の分割とする。このとき、任意の事象 B に対して、

$$\begin{aligned} p(A_i \cap B) &= p(B | A_i) p(A_i) = p(A_i | B) p(B) \\ p(B) &= \sum_{i=1}^K p(B | A_i) p(A_i) \end{aligned}$$

であるから、

$$\begin{aligned} p(A_i | B) &= \frac{p(B | A_i) p(A_i)}{p(B)} \\ &= \frac{p(B | A_i) p(A_i)}{\sum_{i=1}^K p(B | A_i) p(A_i)} \end{aligned}$$

となる[5]。

2.3.3 事後確率最大化識別(ベイズ識別)

$p(c)$	クラス c の事前確率
$p(c x)$	クラス c の事後確率
$p(x c)$	クラス c における特徴ベクトル x の確率

クラス c の個数 $|c| = K$

確率 $p(c)$ および $p(x | c)$ が既知であれば、 $p(c | x)$ はベイズの定理を用いて以下のように求めることができる。

$$p(c | x) = \frac{p(x | c) p(c)}{\sum_{c=1}^K p(x | c) p(c)} \quad \text{eq. 3}$$

事後確率 $p(c | x)$ を最大にする c を選択する判別方式を事後確率最大化識別(ベイズ識別)という[3]。

ただし、真の事前確率 $p(c)$ および $p(x|c)$ は一般に不明であり、教師データから推定する必要がある。次節で $p(x|c)$ の推定方法を説明する。

2.3.4 多次元正規分布

クラス c であるベクトル x がクラスごとに局所的にまとまった分布になっていれば、その分布 $p(x|c)$ を多次元正規分布によってモデル化することは適切であると考えられる。以下の式が、多次元正規分布の式である[3]。

$$p(x|c) = \frac{1}{(2\pi)^{N/2} |\Sigma_c|^{1/2}} \exp\left\{-\frac{1}{2}(x - \mu_c)\Sigma_c^{-1}(x - \mu_c)^T\right\} \quad \text{eq. 4}$$

N …ベクトル x の次元

Σ_c …クラス c の分散共分散行列

$|\Sigma|$ … Σ の行列式

Σ^{-1} … Σ の逆行列

2.3.5 分散共分散行列の RANK と行列式

多次元正規分布でモデル化する場合、 $p(x|c)$ の計算において、クラス c の分散共分散行列(以後 Σ_c と記す)の行列式および逆行列を算出する必要がある。

乱数によってベクトルを発生させたプログラムによる検証の結果によれば、 N 個以下の N 次元ベクトルの集合を元に Σ_c を計算した場合、 Σ_c の RANK は $N-1$ 以下となる。そして Σ_c の RANK がベクトルの次元 N 未満ならば、 Σ_c の行列式は 0 あるいは 0 に近い値をとる。また、行列式と逆行列の関係より、

正方行列 A の行列式が 0 A の逆行列は存在しない

が、成立する。

以上より、 $p(x|c)$ は分母に Σ_c の行列式を、 e の肩に Σ_c の逆行列を含むため、 Σ_c の行列式が 0 あるいは 0 に近い値をとる場合、 $p(x|c)$ が計算不能あるいは非常に信頼性のない非常に大きな値になってしまう。よって Σ_c の RANK に配慮したデータを用意する必要がある。

2.3.6 ベイズ識別の実際の計算

$p(c|x)$ の分母の部分 $\sum_{c=1}^K p(x|c)p(c)$ はすべてのクラスに共通しているから、

$p(c|x)$ の大小は分子部分のみを比較するだけでよい。また、分子部分からベクトル x とクラスにかかわる部分のみを取り出すと、

$$\log(p(c)) - \frac{1}{2} \left\{ (x - \mu_c) \Sigma_c^{-1} (x - \mu_c)^T + \log(|\Sigma_c|) \right\} \quad \text{eq. 5}$$

となり、この大小を比較すればよい。

2.3.7 2次元2群データのベイズ識別による分類の例

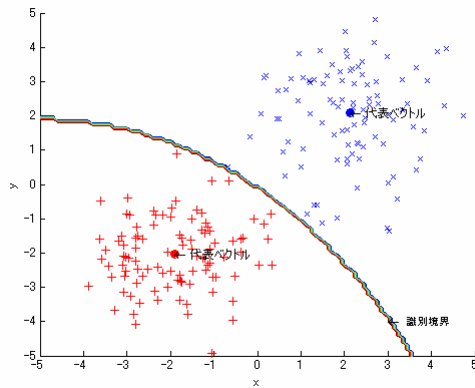


図 2

グラフの軸 縦軸 y、横軸 x

正規乱数により作成

青 x、赤 + の事前確率はともに 0.5

青 x 平均 $(x, y) = (2.1208, 2.0790)$

分散共分散行列 $\Sigma_{\text{青}x} = \begin{pmatrix} 1.2363 & 0.0980 \\ 0.0980 & 2.1142 \end{pmatrix}$

赤 + 平均 $(x, y) = (-1.9038, -2.0433)$

分散共分散行列 $\Sigma_{\text{赤}+} = \begin{pmatrix} 1.0704 & 0.1011 \\ 0.1011 & 1.0091 \end{pmatrix}$

図 2 2次元ベクトルの2次元ベイズ識別による識別境界は2次曲線になる

2.4 線形 SVM

SVMとはサポートベクターマシン(Support Vector Machine)の略であり、2クラスの分類問題を解くためにつくられた学習機械である。以下の説明は[1][3]を参考にした。

2.4.1 SVMとは

x をデータ、対応するクラスラベルを y (ただし、 $y = \{1, -1\}$) とするとき、線形 SVM の識別関数は以下のように表される。

$$f(x) = xw^T + b \quad \text{eq. 6}$$

x …ベクトル(1行N列)

w …重みパラメータ(1行N列)

b …バイアス項(スカラ値)

$f(x) = 0$ を満たす点の集合が識別境界となる。また、分類は、 $f(x)$ の正負の符号で行う。

2.4.2 基本的な考え方

分類の目標は教師データを正しく分類することではなく、未知のベクトルを正しく分類することである。この目的のために、SVMでは、識別境界を2つのクラスの中央を通るように決定する。これをマージン最大化という。

例として $x_i = (x_{i1}, x_{i2})$, $y_i = \{-1, 1\}$, $i = 1, \dots, M$, $w = (w_1, w_2)$ という2クラス2次元のベクトルである場合を考える。識別関数 $f(x) = 0$ としたとき、識別境界

は x_{i1}, x_{i2} 平面上で $x_{i2} = -\frac{w_1}{w_2} x_{i1} - \frac{b}{w_2}$ という直線で表される。このときパラメータ w, b は定数倍しても表現する識別境界がまったく変わらないため以下のような制限を加える。

$$\min_{i=1, \dots, M} |x_i w^T + b| = 1 \quad \text{eq. 7}$$

|a|…aの絶対値

この制限により、教師データと識別境界の最小距離は

$$\min_{i=1, \dots, M} \frac{|x_i w^T + b|}{\|w\|} = \frac{1}{\|w\|}$$

$\|a\| \dots a$ のノルム

と表現することができる。上の式を最大化する事は、 $\|w\|$ を最小化することと等しく、これを扱いやすくするために、以下のように定式化する。

$$\min_w \frac{1}{2} \|w\|^2 \quad \text{eq. 8}$$

制約条件 $|x_i w^T + b| \geq 1$

この目的関数を計算しやすくするため、ラグランジュ未定乗数法を用いる。

2.4.3 ラグランジュ未定乗数法

ラグランジュ未定乗数法とは、制約条件のもとで、ある関数の極値を求める方法である。条件 $h_i(x) = 0, i = 1, \dots, M$ のもとで、 $f(x)$ が極値をとる x を求めるための、ラグランジュ乗数 $\lambda_i (\geq 0)$ を導入したラグランジュ関数は以下ようになる。

$$L(x, \lambda) = f(x) - \sum_{i=1}^M \lambda_i h(x_i)$$

$$\frac{\partial L}{\partial \lambda_i} = -h_i(x) = 0, \quad \text{ただし } i = 1 \dots M$$

ラグランジュ未定乗数法により、SVM の目的関数は

$$L(w, b, \lambda) = \frac{1}{2} \|w\|^2 - \sum_{i=1}^M \lambda_i \{y_i ((x_i w^T + b) - 1)\}$$

を得る。最適化問題を解くには、 w と b に対して最小化し、 λ に対して最大化すればよい。最適解においては、 L の勾配が 0 になるので以下の式が成立する。

$$\frac{\partial L}{\partial b} = \sum_{i=1}^M \lambda_i y_i = 0$$

$$\frac{\partial L}{\partial w} = w - \sum_{i=1}^M \lambda_i y_i x_i = 0 \quad \text{eq. 9}$$

以上より、 λ のみに関する最大化問題になる。

$$\max_{\lambda} \sum_{i=1}^M \left\{ \lambda_i - \frac{1}{2} \sum_{i,j=1}^M (\lambda_i \lambda_j y_i y_j x_i x_j^T) \right\} \quad \text{eq. 10}$$

制約条件 $\lambda_i \geq 0, \sum_{i=1}^M \lambda_i y_i = 0$

適切な λ から w を求めるに eq. 9 を用いる。また b は各クラスの $\lambda_i > 0$ となる x_i (これをサポートベクターという) と w を用いて、eq. 7 より

$$b = -\frac{1}{2} (x_A w^T + x_B w^T) \quad \text{eq. 11}$$

x_A … クラスAのサポートベクター
 x_B … クラスBのサポートベクター

により求めることができる。

2.4.4 最急降下法

関数 $f(x_1, \dots, x_N)$ に対し、 $f(x)$ を最大化(最小化)するような $x = [x_1, \dots, x_N]$ を求める方法。最初に $x = [x_1, \dots, x_N]$ に適当な初期値を設定し、以下のような方法で x を更新していく。

$$\begin{aligned} \text{最大化 } x_i^{\text{new}} &= x_i^{\text{old}} - \eta \frac{\partial f(x)}{\partial x_i} \\ \text{最小化 } x_i^{\text{new}} &= x_i^{\text{old}} + \eta \frac{\partial f(x)}{\partial x_i} \end{aligned}$$

η は小さな値であり、 $\eta = 0.01$ とした。

2.4.5 2次元2群データの線形SVMによる分類の例

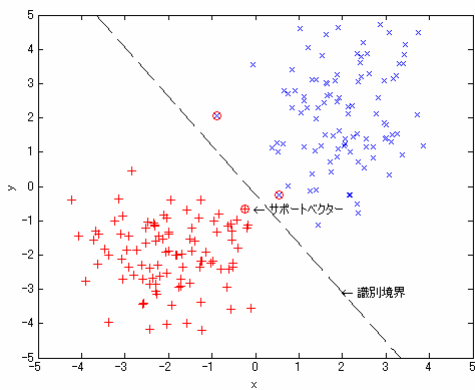


図 3

グラフの軸 縦軸 y、横軸 x

正規乱数により作成

青 x 平均 $(x, y) = (2.0066, 2.1371)$

x の分散 0.8375

y の分散 2.0821

赤 + 平均 $(x, y) = (-1.9427, -2.0228)$

x の分散 0.9948

y の分散 1.0239

図 3 線形 SVM による識別境界。 で囲まれた点はサポートベクターである。

2.5 非線形 SVM

線形 SVM はカーネルトリックを適用することによって非線形に拡張できる。非線形 SVM は多層パーセプトロン(MLP)と同様の非線形の識別器だが、MLP とことなり局所解の問題がないという利点がある。以下の説明は[1][3]を参考にした。

2.5.1 カーネルトリック

特殊な関数を利用することにより、高次元での内積の計算を回避する手法である。線形 SVM の識別関数は eq. 9 より重み w を展開すると以下ようになる。

$$f(x) = x[\sum_{i=1}^M \lambda_i y_i x_i]^T + b = \sum_{i=1}^M (\lambda_i y_i x x_i^T) + b$$

M 次元ベクトル $x = [x_1, \dots, x_M]$ から次元の高い N 次元空間への写像は存在し、 $\Phi(x)$ とすると、

$$f(\Phi(x)) = \Phi(x)[\sum_{i=1}^M \lambda_i y_i \Phi(x_i)]^T + b = \sum_{i=1}^M \{\lambda_i y_i \Phi(x) \Phi(x_i)^T\} + b$$

とできる。ここで、

$$K(x, x_i) = \Phi(x) \Phi(x_i)^T$$

を満たすカーネル $K(x, x')$ が存在すると仮定すると、高次元空間での線形 SVM は $K(x, x_i)$ によって書くことができ以下ようになる。

$$f(x) = \sum_{i=1}^M \lambda_i y_i k(x, x_i) \quad \text{eq. 12}$$

このとき、写像 Φ の具体的な式及び内積計算を省くことができる。これをカーネルトリックという。

2.5.2 カーネル関数

x が連続値の場合のカーネル関数として以下の関数が知られている。

多項式カーネル

$$K(x_1, x_2) = (x_1 * x_2^T)^p \quad \text{eq. 13}$$

p はパラメータであり、問題に応じて決定する必要がある。

ガウシアンカーネル

$$K(x_1, x_2) = \exp\left(-\frac{\|x_1 - x_2\|^2}{\sigma^2}\right) \quad \text{eq. 14}$$

σ は経験的に決定するパラメータであり、問題に応じて決定する必要がある。
本件ではこのガウシアンカーネルを使用し、適当な値として、 $\sigma^2 = 2$ とした。

2.5.3 2次元2群データの非線形SVMによる分類の例

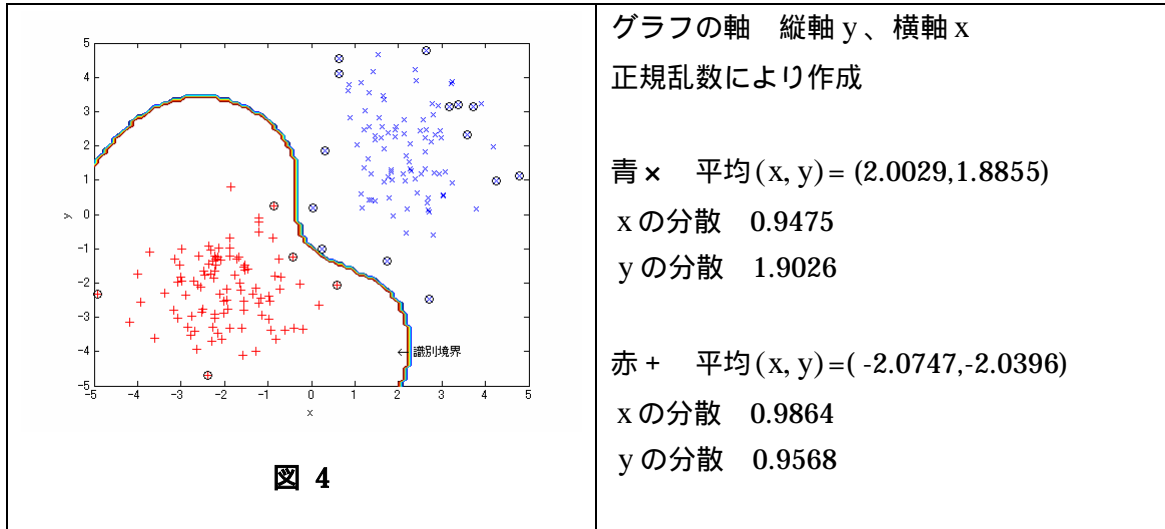


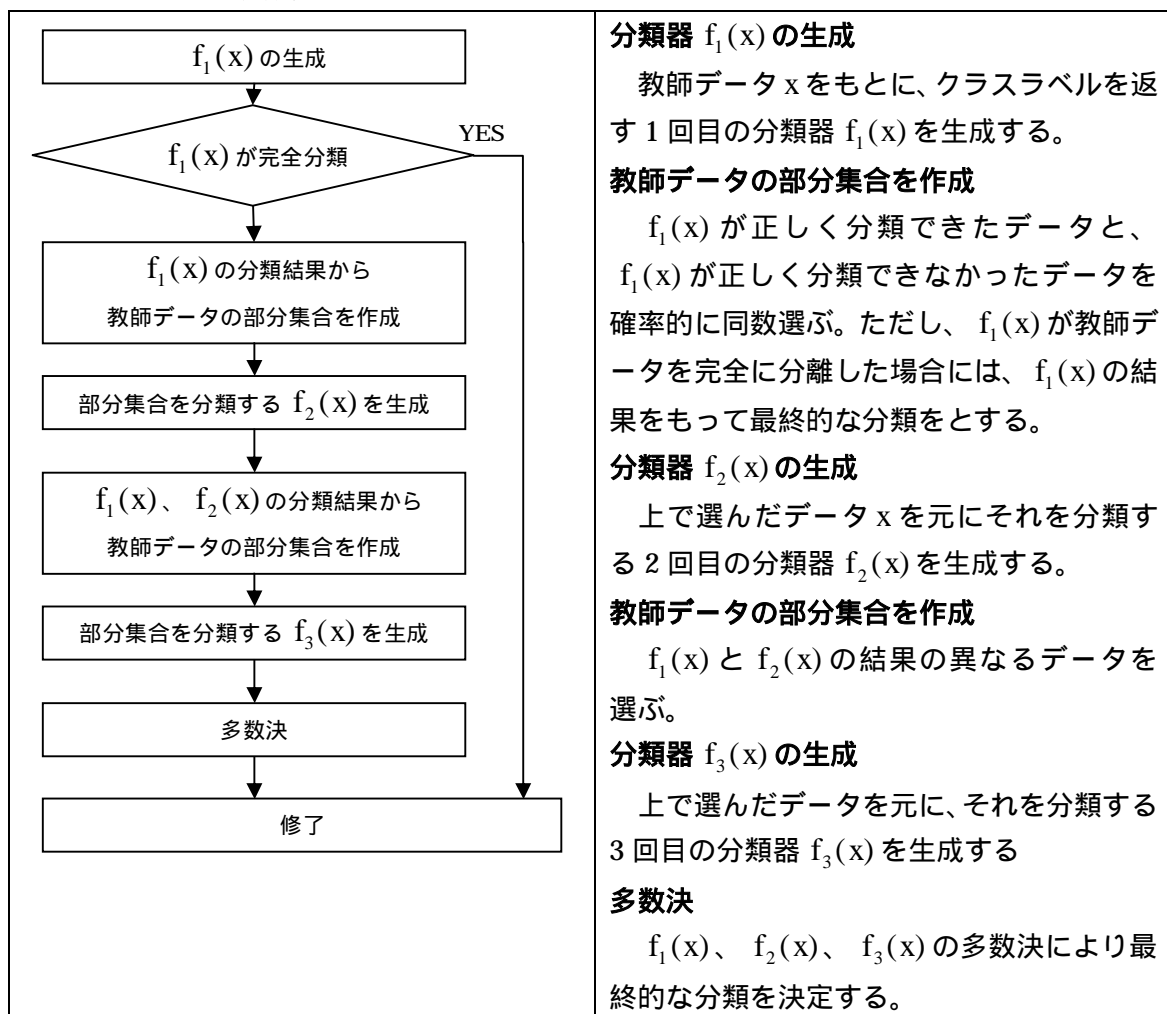
図 4 非線形 SVM による分類。曲線により分類されている。

第3章 誤り率を小さくする手法

3.1 フィルタによるブースティング

何らかの分類手法を用いて分類器を 3 つ生成し、その多数決によってより誤り率を小さくする手法である[3]。1 つ目と 2 つ目の分類器が、次の学習に用いる教師データをふるいにかけて選んでいるように考えることができるので、フィルタによるブースティングと呼ばれている。以後ブースティングと記す。また、この手法を入れ子のようにして繰り返し利用することも可能である。

3.1.1 実装したブースティングのフローチャート



3.1.2 誤り率の減少する理由

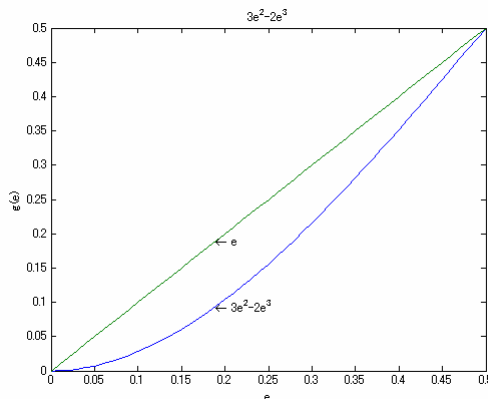
- 1 つ目の分類器 $f_1(x)$ が誤って分類する確率を Er_1 、
- 2 つ目の分類器 $f_2(x)$ が誤って分類する確率を Er_2 、
- 3 つ目の分類器 $f_3(x)$ が誤って分類する確率を Er_3 とするとき、最終結果が誤り

である確率 $g(Er)$ は、以下のようになる。

$$g(Er) = (1 - Er_1)Er_2Er_3 + Er_1Er_2 + Er_1(1 - Er_2)Er_3$$

$Er_1 = Er_2 = Er_3 = e$ とすると、

$$g(e) = 3e^2 - 2e^3$$



となる。詳細は[1][3]にある。

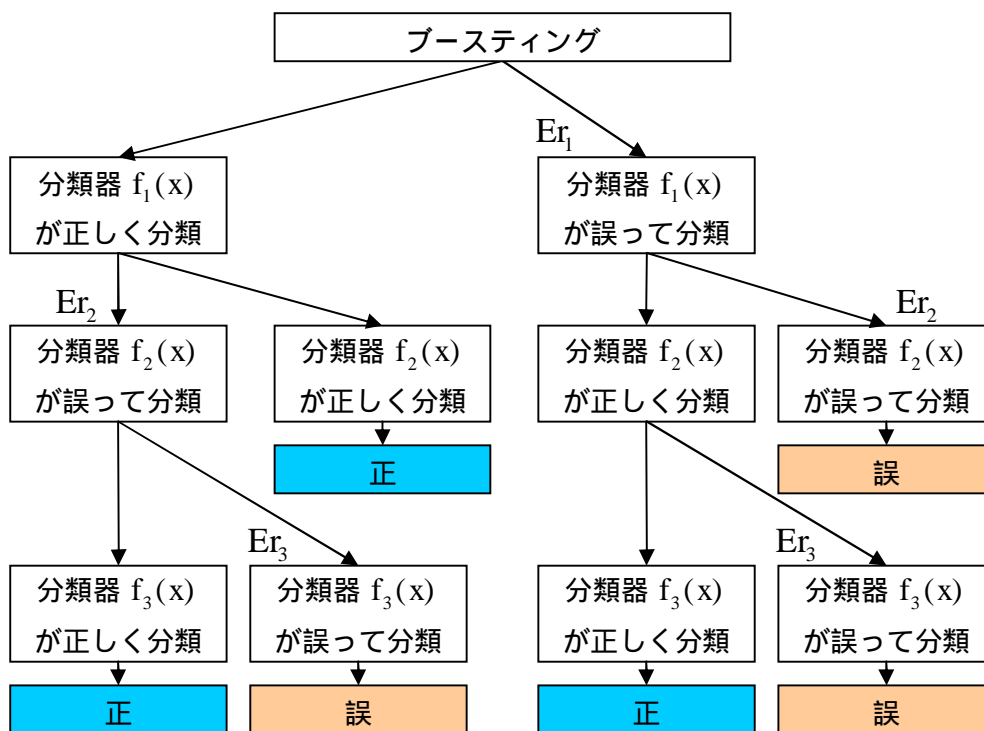
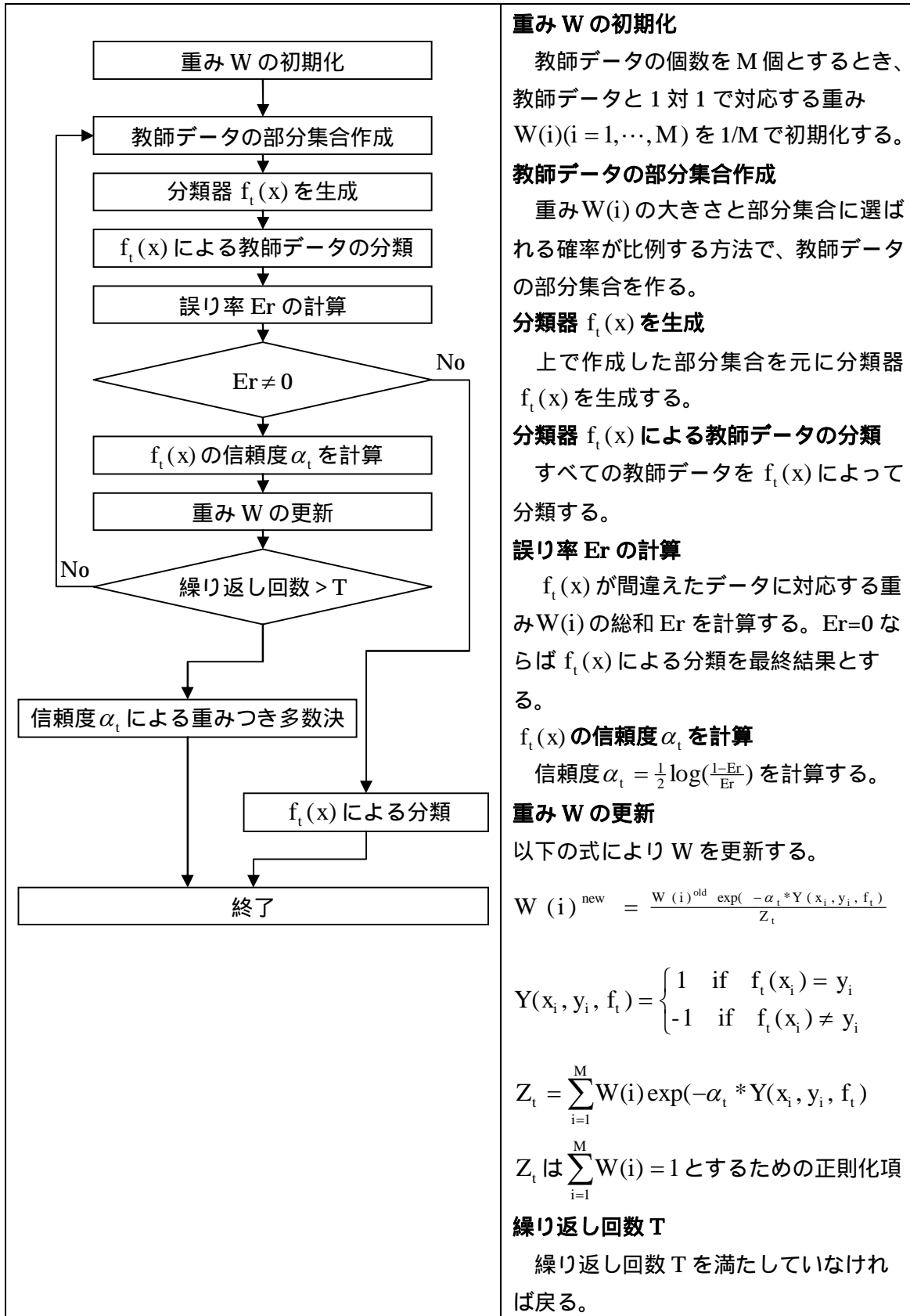


図 5 ブースティングの分類の流れ

3.2 adaboost

ブースティングは単純な多数決だが、adaboost は繰り返しの中で生成する分類器の分類の正確さにより重み付きの多数決を行う。以下の説明は、[2][3][4]を参考にした。

3.2.1 実装した adaboost のフローチャート



重み W の初期化

教師データの個数を M 個とするとき、教師データと 1 対 1 で対応する重み $W(i) (i=1, \dots, M)$ を $1/M$ で初期化する。

教師データの部分集合作成

重み $W(i)$ の大きさと部分集合に選ばれる確率が比例する方法で、教師データの部分集合を作る。

分類器 $f_t(x)$ を生成

上で作成した部分集合を元に分類器 $f_t(x)$ を生成する。

分類器 $f_t(x)$ による教師データの分類

すべての教師データを $f_t(x)$ によって分類する。

誤り率 Er の計算

$f_t(x)$ が間違えたデータに対応する重み $W(i)$ の総和 Er を計算する。 $Er=0$ ならば $f_t(x)$ による分類を最終結果とする。

$f_t(x)$ の信頼度 α_t を計算

信頼度 $\alpha_t = \frac{1}{2} \log \left(\frac{1-Er}{Er} \right)$ を計算する。

重み W の更新

以下の式により W を更新する。

$$W(i)^{\text{new}} = \frac{W(i)^{\text{old}} \exp(-\alpha_t * Y(x_i, y_i, f_t))}{Z_t}$$

$$Y(x_i, y_i, f_t) = \begin{cases} 1 & \text{if } f_t(x_i) = y_i \\ -1 & \text{if } f_t(x_i) \neq y_i \end{cases}$$

$$Z_t = \sum_{i=1}^M W(i) \exp(-\alpha_t * Y(x_i, y_i, f_t))$$

Z_t は $\sum_{i=1}^M W(i) = 1$ とするための正規化項

繰り返し回数 T

繰り返し回数 T を満たしていなければ戻る。

信頼度 α_t による重みつき多数決

c クラス分類の場合、クラスラベル $y = \{y_1, \dots, y_i, \dots, y_c\}$ それぞれに対し、

$\sum_{f_t(x)=y \text{ となる } t}^T \alpha_t$ が最大となる y を最終結果と

する。

第4章 実験環境

4.1 動作コンピュータ

CPU	Pentium4 3GHz
メモリ	512MB
OS	WindowsXP sp2
使用ソフト	Matlab7

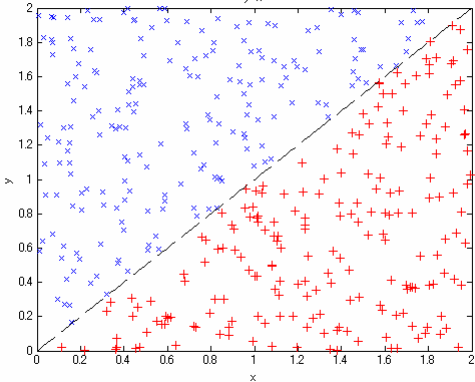
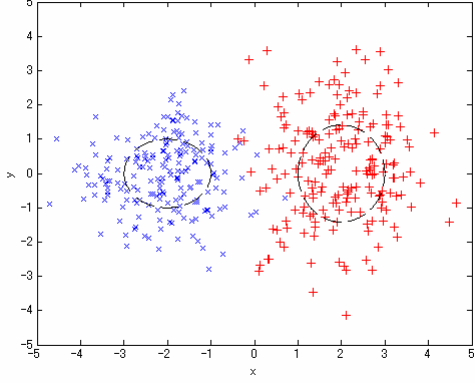
4.2 Matlab について

Matlab は行列を扱うことのできるプログラムであり、行列やベクトルを扱う関数・計算を容易に記述できる。また行列を基にグラフの作成と表示ができる。

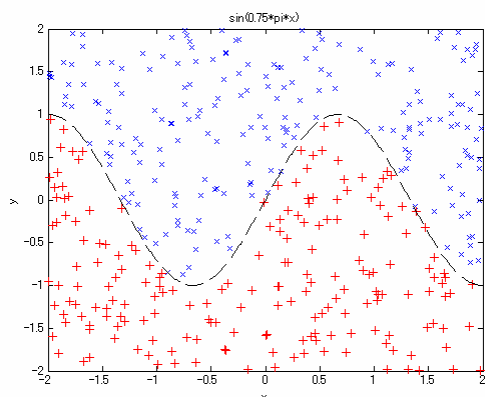
また、m ファイルに関数やスクリプトを記述でき、複雑な数値計算を簡単に行うことができる。

4.3 分類するテストデータ

テストデータ 1 から 5 はすべて 1000 個の 2 次元ベクトルからなる(図はその内の 400 個を表示)。

<p>テストデータ 1 [直線]</p> 	<p>範囲(0,2)内の一様乱数の $y \geq x$ の領域による分類</p> <p>青 x (49.86%) 平均(x,y)=(0.6681,1.3264) 分散共分散行列 $\begin{pmatrix} 0.2228 & 0.1135 \\ 0.1135 & 0.2285 \end{pmatrix}$</p> <p>赤+ (50.14%) 平均(x,y)=(1.3329,0.06702) 分散共分散行列 $\begin{pmatrix} 0.2265 & 0.1143 \\ 0.1143 & 0.2210 \end{pmatrix}$</p>
<p>テストデータ 2 [2 群ガウス]</p> 	<p>正規乱数による分類</p> <p>青 x (50%) 平均(x,y)=(-2.0056,0.0115) 分散共分散行列 $\begin{pmatrix} 1.0109 & 0.0114 \\ 0.0114 & 1.016 \end{pmatrix}$</p> <p>赤+ (50%) 平均(x,y)=(1.9793,0.0083) 分散共分散行列 $\begin{pmatrix} 1.0556 & -0.0181 \\ -0.0181 & 2.0385 \end{pmatrix}$</p>

テストデータ 3 [正弦曲線]



(-2,2)内の一様乱数の、 $y \geq \sin(0.75 * \pi * x)$ の領域による分類

青 x (50.4%)

平均(x,y)=(0.0266,0.8803)

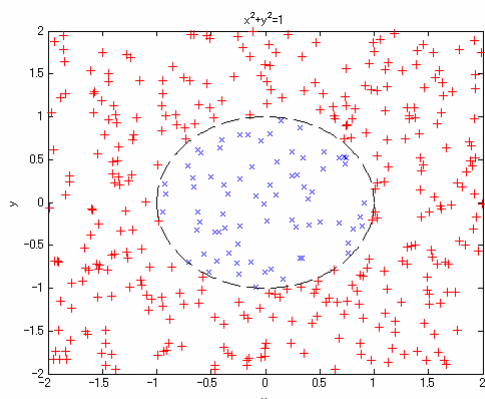
分散共分散行列 $\begin{pmatrix} 1.3350 & -0.0449 \\ -0.0449 & 0.5670 \end{pmatrix}$

赤+(49.6%)

平均(x,y)=(-0.0397,-0.8971)

分散共分散行列 $\begin{pmatrix} 1.3455 & -0.0544 \\ -0.0544 & 0.5721 \end{pmatrix}$

テストデータ 4 [円]



(-2,2)内の一様乱数の、 $x^2 + y^2 \leq 1$ の領域による分類

青 x (19.64%)

平均(x,y)=(0.0015,0.0171)

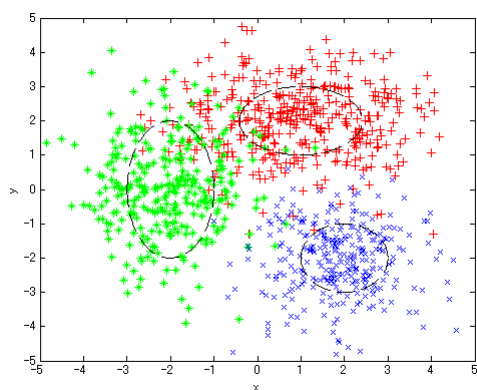
分散共分散行列 $\begin{pmatrix} 0.2534 & -0.0020 \\ -0.0020 & 0.2501 \end{pmatrix}$

赤+(80.36%)

平均(x,y)=(-0.0029,-0.0122)

分散共分散行列 $\begin{pmatrix} 1.6065 & -0.0064 \\ -0.0064 & 1.5923 \end{pmatrix}$

テストデータ 5 [3群ガウス]



正規乱数による分類

青 x (30%)

平均(x,y)=(-2.0191,-2.0070)

分散共分散行列 $\begin{pmatrix} 1.0404 & -0.0110 \\ -0.0110 & 1.0263 \end{pmatrix}$

赤+(40%)

平均(x,y)=(1.0097,2.0082)

分散共分散行列 $\begin{pmatrix} 2.0494 & 0.0122 \\ 0.0122 & 1.0082 \end{pmatrix}$

緑*(30%)

平均(x,y)=(-2.0016,0.0044)

分散共分散行列 $\begin{pmatrix} 1.0136 & -0.0202 \\ -0.0202 & 2.0074 \end{pmatrix}$

第5章 分類実験

5.1 実験の方法

分類手法は、

- ユークリッド距離(ユークリッド)
- ベイズ識別法(ベイズ)
- 線形 SVM
- 非線形 SVM

の4通りに対し、

- 各手法をそのまま適用したもの(標準)
- 各手法に、ブースティングを適用したもの(boost)
- 各手法に、ブースティングを2回適用したもの(2回 boost)
- 各手法に内部ループ5回の adaboost を適用したもの(ada5)
- 各手法に内部ループ10回の adaboost を適用したもの(ada10)
- 各手法に内部ループ20回の adaboost を適用したもの(ada20)

の6通りの合計24通りの分類を行った。()は表における表記である。

誤り率は、教師データ800件、試験データ200件を5回行う、5交差確認法により教師データ、試験データそれぞれの誤り率を計算し平均を結果とした。誤り率の計算方法はそれぞれ、教師データの誤り分類数 / 教師データ数、試験データの誤り分類数 / 試験データ数である。ただし、非線形 SVM に関しては、処理時間の関係上、教師データ160件、試験データ40件を5回行う5交差確認法による。また、データの件数の異なる非線形 SVM を除いて、交差確認法に用いるデータはどの手法にも差のないように同一のものを選ぶようにした。

5.2 分類の実験と考察

5.2.1 テストデータ1 [直線]の分類の結果

表1 テストデータ1 [直線]の教師データについての誤り率

	ユークリッド	ベイズ	線形 SVM	非線形 SVM
標準	.0105	.0055	.0095	.0150
boost	.0170	.0070	.0080	.0162
2回 boost	.0170	.0005	.0040	.0100
ada5	.0195	.0090	.0008	.0037
ada10	.0198	.0055	.0000	.0000
ada20	.0160	.0038	.0000	.0000

表 2 テストデータ 1 [直線]の試験データについての誤り率

	ユークリッド	ベイズ	線形 SVM	非線形 SVM
標準	.008	.005	.008	.025
boost	.016	.005	.006	.025
2 回 boost	.018	.002	.006	.020
ada5	.021	.009	.002	.030
ada10	.017	.006	.002	.020
ada20	.020	.008	.001	.025

標準の方法でも正しく分類している様子が見える。標準の手法では、訓練データ・試験データともにベイズ識別による分類が最も誤り率が小さくなっている。また、線形 SVM が adaboost によってより正しく分類できるようになっている。

5.2.2 テストデータ 2 [2 群ガウス]の分類の結果

表 3 テストデータ 2 [2 群ガウス]の教師データについての誤り率

	ユークリッド	ベイズ	線形 SVM	非線形 SVM
標準	.0267	.0270	.3478	.0363
boost	.0273	.0267	.5812	.0325
2 回 boost	.0260	.0233	.3718	.0225
ada5	.0265	.0275	.0698	.0213
ada10	.0275	.0278	.0568	.0163
ada20	.0270	.0270	.0473	.0075

表 4 テストデータ 2 [2 群ガウス]の試験データについての誤り率

	ユークリッド	ベイズ	線形 SVM	非線形 SVM
標準	.026	.027	.352	.075
boost	.027	.029	.577	.050
2 回 boost	.026	.039	.383	.070
ada5	.028	.032	.065	.085
ada10	.029	.027	.063	.075
ada20	.026	.032	.055	.075

ユークリッド距離とベイズ識別は標準でも誤り率は低く、また、ブースティン

グ・adaboost の影響を受けていないようである。線形 SVM の分類が非常に悪い。教師データについて平均を取る前の表 5 によれば、誤り率 0.5 を超えるものが多い。実際、図 6 のように分類に失敗している場合を確認した。しかし、adaboost によって誤り率を小さくすることに成功している。また非線形 SVM は教師データではブースティング・adaboost によって誤り率が小さくなっているが、試験データでは、教師データの場合と比べ、誤り率が高くなっている。

表 5 テストデータ 2 [2 群ガウス] 線形 SVM の分類 教師データ

LineSVM	1 回目	2 回目	3 回目	4 回目	5 回目	平均
std	0.975	0.035	0.05	0.05	0.6288	0.34776
boost	0.9425	0.9063	0.6312	0.39	0.0362	0.58124
2 回 boost	0.035	0.0925	0.7388	0.0488	0.9437	0.37176
ada5	0.0963	0.0712	0.0963	0.0338	0.0512	0.06976
ada10	0.035	0.0788	0.085	0.0475	0.0375	0.05676
ada20	0.045	0.0612	0.0413	0.05	0.0388	0.04726

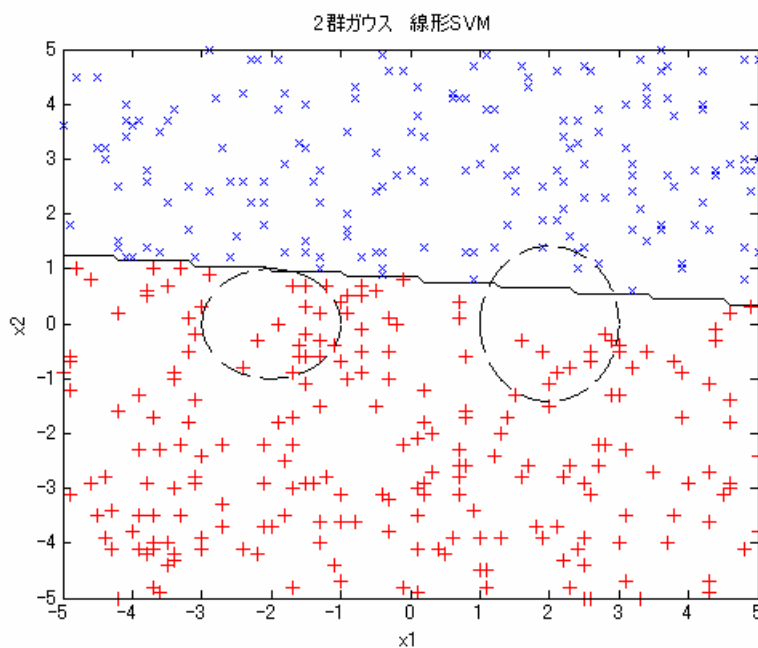


図 6 2 群ガウス 線形 SVM

5.2.3 テストデータ 3 [正弦曲線]の分類の結果

表 6 テストデータ 3 [正弦曲線]の教師データについての誤り率

	ユークリッド	ベイズ	線形 SVM	非線形 SVM
標準	.1427	.1445	.4315	.0238
boost	.1405	.1338	.3753	.0125
2 回 boost	.1408	.1118	.3305	.0088
ada5	.1420	.1425	.3280	.0013
ada10	.1442	.1400	.2218	.0000
ada20	.1417	.1380	.2388	.0000

表 7 テストデータ 3 [正弦曲線]の試験データについての誤り率

	ユークリッド	ベイズ	線形 SVM	非線形 SVM
標準	.145	.144	.441	.040
boost	.146	.136	.393	.065
2 回 boost	.142	.116	.327	.035
ada5	.146	.148	.319	.030
ada10	.139	.142	.223	.035
ada20	.145	.141	.244	.045

ベイズ識別では 2 回ブースティングを適用したものの誤り率が最も小さくなっている(図 7)。非線形 SVM の特性がよくでており、図 8 のように分類できている。線形 SVM は標準で誤り率が高いが、ブースティング、adaboost によって誤り率が小さくなっており図 9 のように分類できる場合もある。

ユークリッド距離による分類と線形 SVM による分類のそれぞれに adaboost を適用した場合、分離境界が大きく異なる点(図 9 図 10)に注目したい。

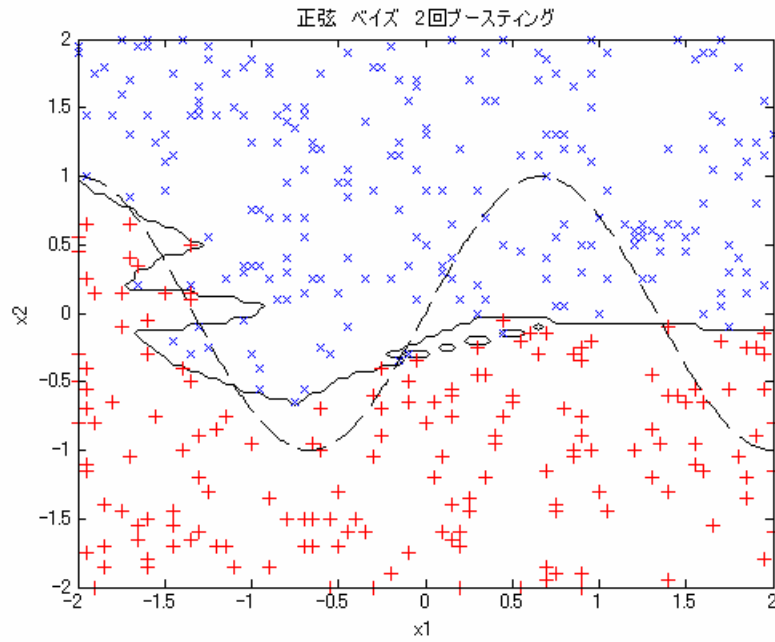


図 7 正弦 ベイズ識別 2回ブースティング

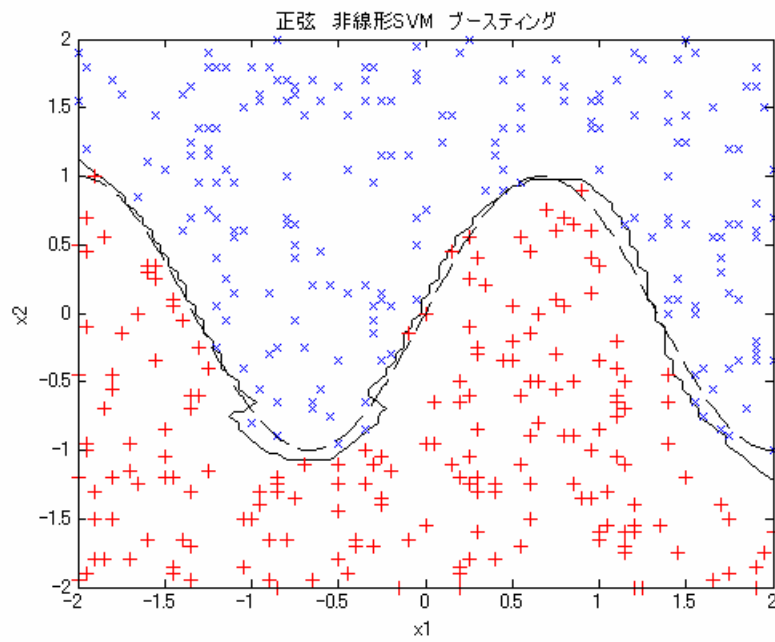


図 8 正弦 非線形 SVM ブースティング

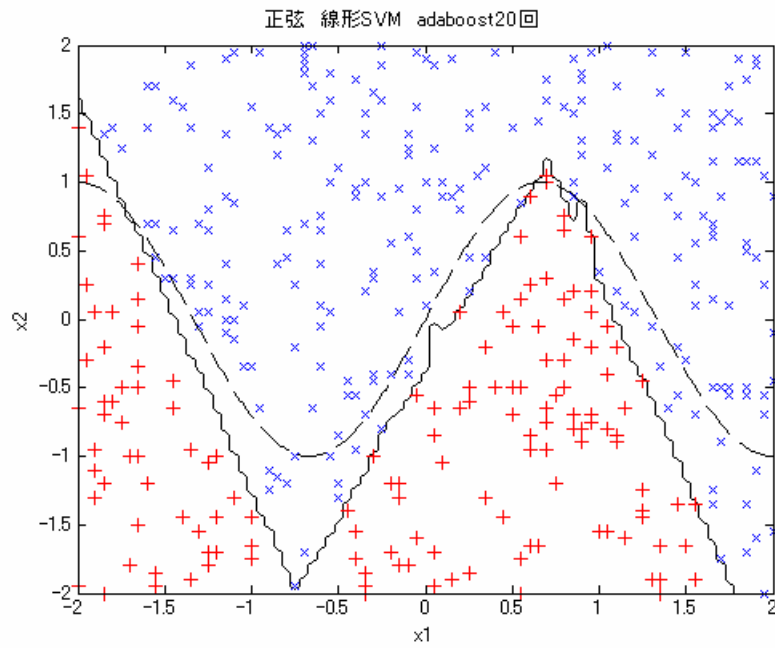


図 9 正弦 線形 SVM adaboost20 回

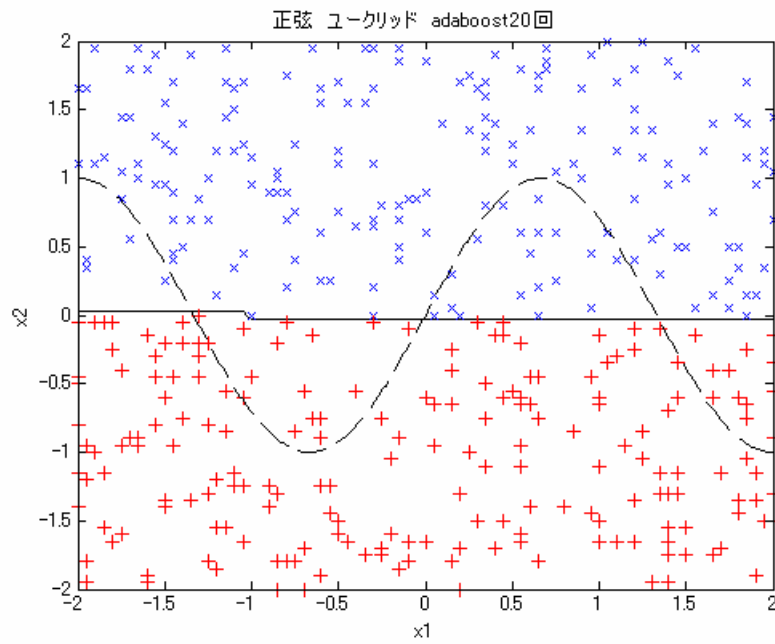


図 10 正弦 ユークリッド距離 adaboost20 回

5.2.4 テストデータ 4 [円]の分類の結果

表 8 テストデータ 4 [円]の教師データについての誤り率

	ユークリッド	ベイズ	線形 SVM	非線形 SVM
標準	.4710	.1492	.5257	.0500
boost	.4702	.0413	.5540	.0237
2 回 boost	.4703	.0275	.5368	.0163
ada5	.4705	.0278	.3610	.0275
ada10	.4620	.0145	.3340	.0287
ada20	.4600	.0063	.3640	.0100

表 9 テストデータ 4 [円]の試験データについての誤り率

	ユークリッド	ベイズ	線形 SVM	非線形 SVM
標準	.504	.153	.486	.040
boost	.501	.042	.513	.020
2 回 boost	.489	.021	.552	.040
ada5	.484	.021	.375	.045
ada10	.502	.017	.321	.040
ada20	.487	.009	.380	.050

adaboost の適用によって線形 SVM はユークリッド距離による分類よりも誤り率は小さくなっているが、どちらも使い物にはならない。標準のベイズ識別による分類の誤り率は高いが、ブースティングや adaboost を適用したベイズ識別の誤り率は非常に小さくなっている。

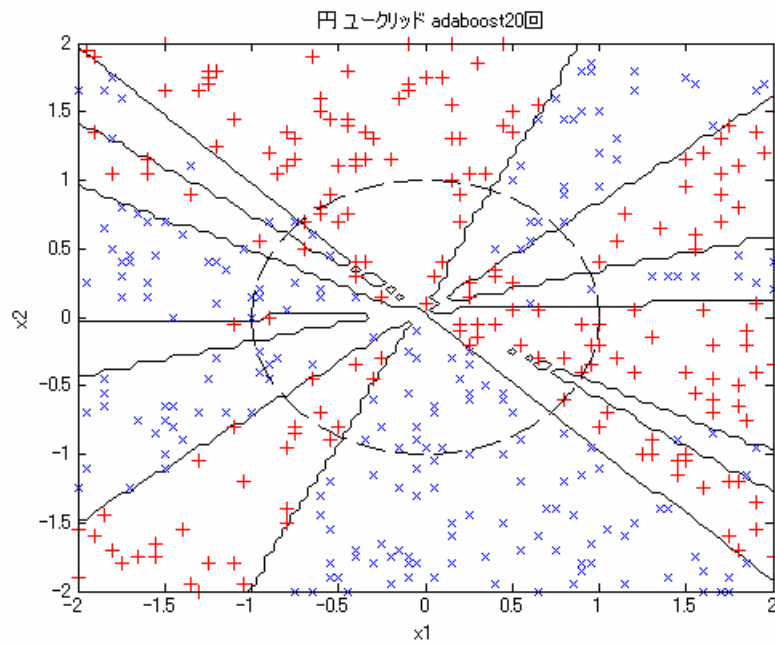


図 11 円 ユークリッド距離 adaboost20回

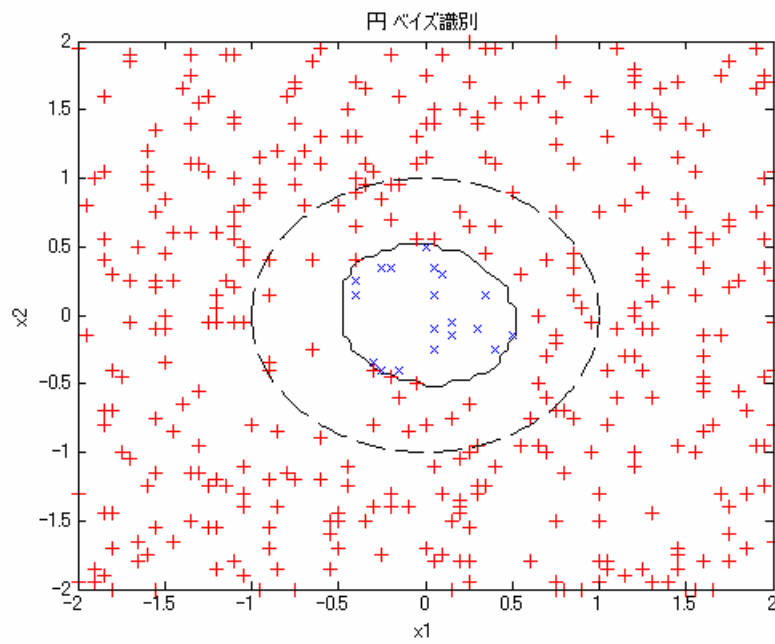


図 12 円 ベイズ識別

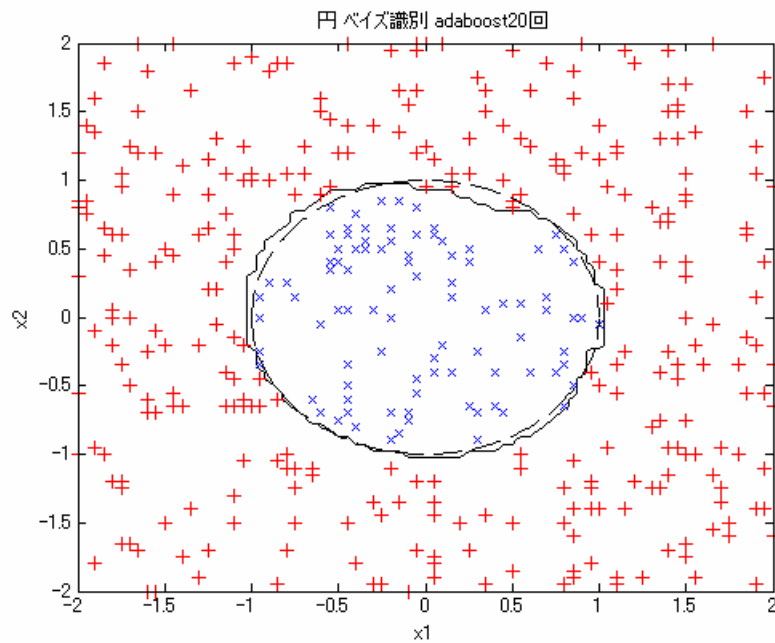


図 13 円 ベイズ識別 adaboost20 回

5.2.5 テストデータ 5 [3 群ガウス]

表 10 テストデータ 5 [3 群ガウス]の教師データについての誤り率

	ユークリッド	ベイズ	線形 SVM	非線形 SVM
標準	.0760	.0692	.4888	.0563
boost	.0778	.0695	.4460	.0413
2 回 boost	.0790	.0708	.3552	.0275
ada5	.0755	.0717	.4235	.0450
ada10	.0768	.0725	.4005	.0225
ada20	.0748	.0707	.3215	.0100

表 11 テストデータ 5 [3 群ガウス]の試験データについての誤り率

	ユークリッド	ベイズ	線形 SVM	非線形 SVM
標準	.078	.072	.469	.105
boost	.086	.076	.445	.110
2 回 boost	.087	.080	.337	.125
ada5	.078	.073	.432	.140
ada10	.078	.065	.393	.110
ada20	.076	.070	.355	.125

ユークリッド距離による分類とベイズ識別による分類(図 14)は正しく分類している。非線形 SVM(図 15)は教師データではうまく分類しているが、やはり試験データに対して誤り率が高くなっている。

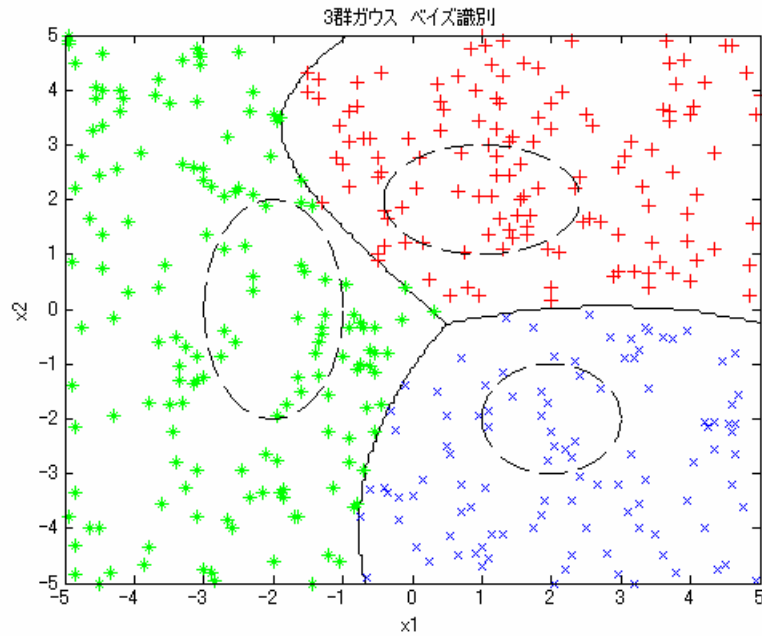


図 14 3群ガウス ベイズ識別

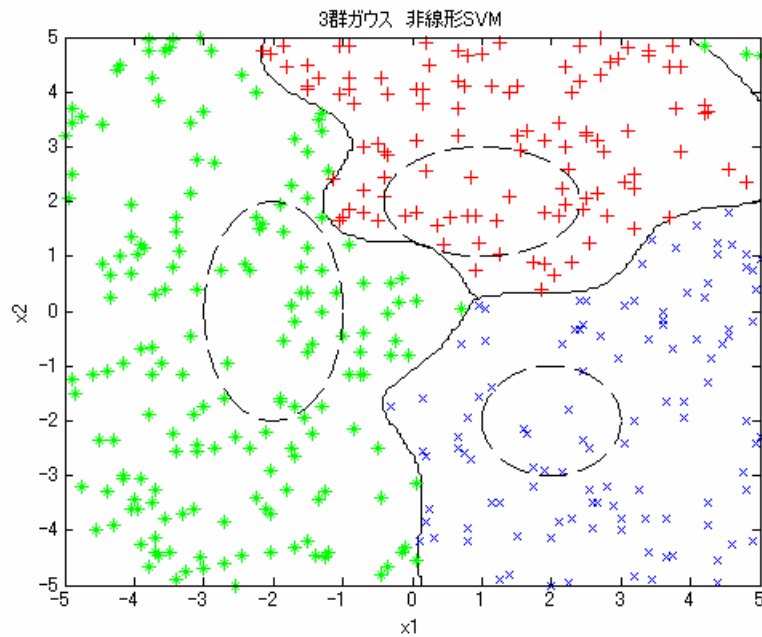


図 15 3群ガウス 非線形 SVM

第6章 考察とまとめ

試験データについてまとめると表 12 から表 14 が得られる。

表 12 adaboost20 回により標準よりも誤り率が小さくなったもの

	ユークリッド	ベイズ	線形 SVM	非線形 SVM
直線	×	×		×
2 群ガウス	×	×		×
正弦曲線	×			×
円				×
3 群ガウス				×

線形 SVM と adaboost は相性が良いようである。

表 13 標準の分類誤り 0.1 以下

	ユークリッド	ベイズ	線形 SVM	非線形 SVM
直線				
2 群ガウス			×	
正弦曲線	×	×	×	
円	×	×	×	
3 群ガウス			×	×

分類手法ごとに得意なデータ分布がある。非線形 SVM は強力だが、ガウス分布に対して弱いようである。

表 14 adaboost20 回後の分類誤り 0.1 以下

	ユークリッド	ベイズ	線形 SVM	非線形 SVM
直線				
2 群ガウス				
正弦曲線	×	×	×	
円	×		×	
3 群ガウス			×	×

ベイズ識別が[円]の分布を分類できるようになっている。

ユークリッド距離による分類や線形 SVM による分類は、どちらもデータを直線によって分離境界を引けることができるかが重要であると考えられる。データ 1[直線]はどちらも正しく分類できており、データ 2[2 群ガウス]とデータ 3[3 群ガウス]も必然的に誤りを含むが直線で分離できるため誤り率は低い。一方、データ 3[正弦曲線]やデータ 4[円]は直

線では分離できないため、誤り率が高くなっている。

ユークリッド距離はその計算の特性上図 11 のように代表ベクトルの位置が分類境界を左右している。よってデータ 4[円]の図 11 のように代表ベクトルに縛られてしまうこともありうる。一方、線形 SVM は代表ベクトルに左右されないため、精度強化手法によって、データ 3[正弦曲線]を図 9 のように比較的よく分類できるという結果を得ることができた。

ベイズ識別による分類は分布の前提を多次元正規分布としているため、その分布に従うデータ 2[2 群ガウス]とデータ 3[3 群ガウス]に対しては非常に誤り率が低い。その上、試験データの誤り率と教師データの誤り率は非常に近いものとなっている。また、データ 4[円]に対してブースティングや adaboost を適用した場合に非常に良い結果を出している。

非線形 SVM は非常に強力な分類器であった。今回用意したすべてのデータセットの教師データに対する誤り率はおおよそ 0.05 以下となった。しかし、計算に時間がかかるため他の分類手法とデータ数の異なる条件でしか実験できなかったのが残念である。また、試験データの誤り率が、訓練データの誤り率より特に悪いことについて、今回実装した非線形 SVM はたった 1 つの教師データの有無によって分類境界が大きく変わってしまう可能性があるため、どこにでも出現する可能性があるガウス乱数に対応できなかったと考えられる。また、他の分類手法と異なり、教師データ数が少ないことも影響している可能性がある。

今回実装したブースティングや adaboost によって線形 SVM はデータ 1[直線]を正しく分類している。また、複雑なデータ 3[正弦曲線]を図 9 のように分類していること、および表 12 から、線形 SVM と adaboost の相性が良く、組み合わせによって非常に柔軟な分類器を構成できると考えられる。また、ブースティングや adaboost を適用したベイズ識別がデータ 4[円]に対して非常に良い結果を出しており、このような特殊な分布を分類する良い組み合わせを発見できた。一方、理想的なガウス分布に従うデータをユークリッド距離またはベイズ識別による分類には今回実装した精度強化手法は効果がなかった。

今回の研究により、分類の誤り率は、データの分布と使用した分類手法に依存している部分があることがわかる。また、精度強化手法を利用する場合も、用いる分類手法やデータの分布によっては効果のない場合があることも確認された。分類の誤りを少なくするには、直線で分離できる分布、多次元正規分布に従う分布、曲線で分離できる分布など、データの分布や境界線にあわせた適切な分類手法を選ぶことが必要である。今回の実験結果は分類手法を選ぶ際の参考になると考える。

第7章 謝辞

本研究を進めるにあたり、ゼミを中心に最後まで温かく熱心なご指導、ご助言を頂きました田中章司郎教授に深く感謝の意を表すとともに心より御礼申し上げます。また、院生の長田さん、喜吉代さん、同じ学部生である塚定さん、川島さん、齋藤さん、関本さん、高橋さんにもいろいろとご協力、ご助言いただいたことに御礼申し上げます。なお、本研究で作成したプログラム、発表資料などのすべての著作権を田中章司郎教授に譲渡いたします。

第8章 文献

- [1] Klaus-Robert Müller, Sebastian Mika, Gunnar Rätsch, Koji Tsuda, and Bernhard Schölkopf (2001) “An Introduction to Kernel-Based Learning Algorithms” IEEE TRANSACTIONS ON NEURAL NETWORKS, VOL. 12, NO. 2, MARCH 2001 18
- [2] Robert E. Schapir and Yoram Singer (1999) “Improved Boosting Algorithms Using Confidence-rated Predictions” Machine Learning, 37(3):297-336, 1999 [3]
- [3] 麻生英樹 津田直治 村田昇 “統計科学のフロンティア 6 パターン認識と学習の統計学” (岩波書店 2003)
- [4] Richard O. Duda, Peter E. Hart, David G. Stork Pattern Classification, Second Edition(2000)
- [5] 稲垣宣生 山根芳知 吉田光雄 “統計学入門” (裳華房 1992)