■2 群（画像・音・言語）- 2 編（パターン認識とビジョン）

1 章 パターン識別・分類

（執筆者：佐藤真一）［2011 年 2 月 受領］

■概要■

本章では，パターン識別並びに分類，すなわち広くパターン認識と呼ばれる技術について概説する．パターン認識とは，与えられた「パターン」を特定の「カテゴリ」あるいは「クラス」に分類する技術をいう．これは，我々人類を含む生物にとって必要欠くべからざる能力である．我々が人の顔を認識したり，サルが仲間の鳴き声で危険を察知したり，ハエが遠くにある食物を察知して的確に飛んで集まったりするのに重要な役割を果たす．すなわち，画像のみならず，音響，触覚，嗅覚などの情報をパターンとして観測し，適切なクラスに分類し，それに基づき判断を行って行動の制御などを行っている．パターン認識は我々を取り巻く様々な計算機システムにも重要な要素技術となってきている．デジタルカメラの顔検出機能，はがきの宛名の自動読み取りシステム，カーナビゲーションシステムなどの音声認識システムから，天気予報，株価予測などでも中心的な役割を果たしている．

パターン認識，特に統計的パターン認識は，長い研究の歴史があり，学問体系として確立している．それでもなお，計算機によるパターン認識は，人間をはじめとする様々な生物が無意識に行っている柔軟なパターン認識に及ばない面がある．また，本質的に困難な問題として，実世界中の情報（画像，音響など）を適切に獲得する問題，並びにそこから適切な特徴量を抽出する問題がある．本章では，各種メディア情報から特徴量を抽出する技術には，基本的には触れない立場を取っている．これらの技術については，ほかの関連する章を参照されたい．本章では，特徴量がパターンとして観測可能であることを前提として，パターン認識研究の歴史も一部踏まえ，パターン認識の現状の技術水準を一通り概観することを目的としている．

【本章の構成】

本章では，パターン認識概論（1-1 節），特徴圧縮技術（1-2 節），パーセプトロン，ニューラルネットワーク，判別分析，SVM，Boosting，部分空間法を主な例とした様々なパターン認識手法（分類器）の紹介（1-3 節），そしてパターン認識手法の性能を評価する手法（1-4 節）について述べる．

■2 群 画像・音・言語- 2 編 パターン認識とビジョン- 1 章 パターン識別・分類

1-1 パターン認識概論

（執筆者：佐藤真一）[2011 年 2 月 受領]

パターン認識とは，入力として与えられたパターンを，あらかじめ定義したいくつかのクラスに適宜割り当てる技術をいう．例えば，文字の画像を入力として，これをアルファベット 26 文字のいずれか（26 クラス）に割り当てる場合に相当する．パターン識別，パターン分類とも，パターン認識とほぼ同様の意味として使われる．

図 1·1 に，パターン認識処理の一般的な流れを示す．まずは入力として画像や音響情報が与えられると，カメラやマイク，及び AD 変換器などから構成されるセンサにより電気信号やディジタル情報へと変換される．その後は通常計算機プログラムにより処理される．前処理ではノイズ除去や画像の大きさの正規化などが行われる．特徴抽出では識別に必要な特徴量を算出する．その結果は通常ベクトルとみなされ，特徴ベクトルと呼ばれる．具体的には，画像の場合には色ヒストグラム，エッジ検出並びに方向性特徴，ウェーブレット係数など，音響情報の場合にはケプストラムやゼロ交差率などが使われる．ここまでの処理は，対象のメディア（画像，音響など）に強く依存しているため，本章では扱わない．

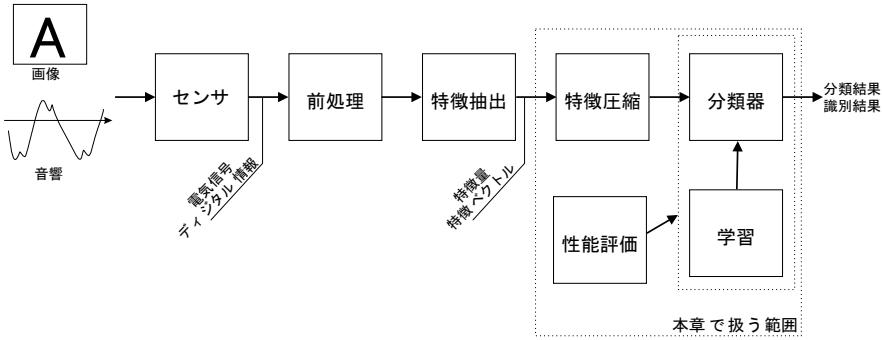


図 1·1　パターン認識処理の流れ

得られた特徴ベクトルには，識別には撹乱となってしまう要素や相互に相関している要素などが含まれている場合があり，分類器の性能に悪影響を及ぼす場合がある．特徴圧縮では，これらに対処し，より識別性の高い特徴量の選択，特徴ベクトルの相関の除去などを行い，より望ましい，通常より低次元の特徴ベクトルへの圧縮を行う．1-2 節では，こうした手法について述べる．分類器では，特徴ベクトルを所定のクラスへと分類し，クラス情報を出力する．この際，通常は学習用に用意した正解のクラス情報つきの特徴ベクトル群を用い，学習により，適切な分類を行うような分類器を構築する．1-3 節では，様々な分類器，それぞれの学習方法，並びにそれらの特性などについて述べる．また，分類器の性能は利用する分類器の種類，分類するパターンの特性，学習用に用意したデータなどにより変わってくるが，

その適切な性能評価もパターン認識では重要な問題である．1-4 節では，分類器の性能を測る評価尺度，信頼性の高い性能評価を行うための技法などについて述べる．

■2 群 - 2 編 - 1 章

1-2 特徴圧縮

（執筆者：仙田修司）[2011 年 2 月 受領]

パターン認識のための特徴とは，特徴抽出などの処理によって得られる多次元データである．特徴は高次元のデータとなることがあるため，冗長な情報を削除して低次元のデータに変換する特徴圧縮（特徴選択と呼ぶ場合もある）を行うと取扱いが容易になる．ここでは，特徴圧縮の手法として最もよく用いられる主成分分析（Principal Component Analysis，以下 PCA）[2]について述べる．なお，KL 展開（Karhunen-Loève expansion）と呼ばれる手法も，特徴圧縮という観点では PCA と同じ手法である．

PCA では，d 次元ベクトル $\mathbf{x} = (x_1, x_2, \cdots, x_d)^T$（$T$ は転置を表す）で表された原特徴を，c ($< d$) 次元の低次元特徴 $\mathbf{y} = (y_1, y_2, \cdots, y_c)^T$ に線形変換する．すなわち，$\mathbf{y} = \mathbf{A}^T\mathbf{x}$ となる $d \times c$ 行列 $\mathbf{A}$ を求める．ただし，$\mathbf{A}$ を正規直交基底とするために，$\mathbf{A}^T\mathbf{A} = \mathbf{I}$ の制約を設ける．

特徴圧縮のための基準として，圧縮された特徴の分散をなるべく大きくするもの（$\mathbf{y}$ の分散の最大化）と，圧縮による誤差をなるべく小さくするもの（$\mathbf{Ay}$ と $\mathbf{x}$ の 2 乗誤差の最小化）とが考えられるが，どちらの基準でも同じ $\mathbf{A}$ が得られる．以下では，分散最大化を考える．

n 個の特徴 $\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_n$ があるとき，これらを変換した $\mathbf{y}_1, \mathbf{y}_2, \cdots, \mathbf{y}_n$ の分散 σ^2 は，

$$\sigma^2 = \frac{1}{n}\sum_{i=1}^{n}(\mathbf{y}_i - \frac{1}{n}\sum_{j=1}^{n}\mathbf{y}_j)^T(\mathbf{y}_i - \frac{1}{n}\sum_{j=1}^{n}\mathbf{y}_j) \tag{1・1}$$

で表されるから，$\mathbf{y}_i = \mathbf{A}^T\mathbf{x}_i$ を代入し，原特徴の平均を $\mathbf{m} = \frac{1}{n}\sum_{j=1}^{n}\mathbf{x}_j$ とおけば，

$$\sigma^2 = \frac{1}{n}\sum_{i=1}^{n}\{\mathbf{A}^T(\mathbf{x}_i - \mathbf{m})\}^T\{\mathbf{A}^T(\mathbf{x}_i - \mathbf{m})\} \tag{1・2}$$

$$= \frac{1}{n}\sum_{i=1}^{n}\mathrm{Tr}[\{\mathbf{A}^T(\mathbf{x}_i - \mathbf{m})\}\{\mathbf{A}^T(\mathbf{x}_i - \mathbf{m})\}^T] \tag{1・3}$$

$$= \mathrm{Tr}[\mathbf{A}^T\{\frac{1}{n}\sum_{i=1}^{n}(\mathbf{x}_i - \mathbf{m})(\mathbf{x}_i - \mathbf{m})^T\}\mathbf{A}] \tag{1・4}$$

$$= \mathrm{Tr}(\mathbf{A}^T\mathbf{S}\mathbf{A}) \tag{1・5}$$

と変形できる．ただし，Tr( ) は行列のトレース（対角成分の和），$\mathbf{S}$ は原特徴 $\mathbf{x}_i$ の共分散行列 $\frac{1}{n}\sum_{i=1}^{n}(\mathbf{x}_i - \mathbf{m})(\mathbf{x}_i - \mathbf{m})^T$ である．$\mathbf{A}^T\mathbf{A} = \mathbf{I}$ の制約条件から，ラグランジュの未定乗数法より，

$$L(\mathbf{A}, \mathbf{\Lambda}) = \sigma^2 - \mathrm{Tr}\{(\mathbf{A}^T\mathbf{A} - \mathbf{I})\mathbf{\Lambda}\} = \mathrm{Tr}(\mathbf{A}^T\mathbf{S}\mathbf{A}) - \mathrm{Tr}\{(\mathbf{A}^T\mathbf{A} - \mathbf{I})\mathbf{\Lambda}\} \tag{1・6}$$

を最大化すればよい．ただし，$\mathbf{\Lambda}$ は未定乗数を表す c 次元の対角行列である．式 (1・6) を $\mathbf{A}$ で偏微分すると，$2\mathbf{S}\mathbf{A} - 2\mathbf{A}\mathbf{\Lambda}$ となり，これを 0 とおくことで $\mathbf{S}\mathbf{A} = \mathbf{A}\mathbf{\Lambda}$ を得る．これは，$d \times d$ 次元の対称行列 $\mathbf{S}$ の固有値問題である．このときの $\mathbf{A}$ 及び $\mathbf{\Lambda}$ を，それぞれ，$\mathbf{A}^*$ 及び $\mathbf{\Lambda}^*$ とすれば，σ^2 の最大値は $\mathrm{Tr}(\mathbf{A}^{*T}\mathbf{S}\mathbf{A}^*) = \mathrm{Tr}(\mathbf{A}^{*T}\mathbf{A}^*\mathbf{\Lambda}^*) = \mathrm{Tr}(\mathbf{\Lambda}^*)$ となり，$\mathbf{\Lambda}^*$ は $\mathbf{S}$ の固有値 $\{\lambda_1, \lambda_2, \cdots, \lambda_c\}$ の大きいものから順に c 個を対角要素とする対角行列，$\mathbf{A}^*$ は各固有値に対応する固有ベクトル $\{\mathbf{u}_1, \mathbf{u}_2, \cdots, \mathbf{u}_c\}$ を並べたものとなる．ただし，$\mathbf{A}^{*T}\mathbf{A}^* = \mathbf{I}$ の条件から，

各固有ベクトルは大きさ 1 に正規化されているものとする．

ここまでの議論では，圧縮された特徴の次元数 c は与えられているものとしたが，原特徴の分布に応じて決定することもできる．具体的には，分散の最大値 $\mathrm{Tr}(\mathbf{\Lambda}^*) = \sum_{i=1}^{c} \lambda_i$ に対して，全固有値の和との比 $(\sum_{i=1}^{c} \lambda_i)/(\sum_{i=1}^{d} \lambda_i)$ を累積寄与率と呼び，これが一定値（例えば 0.9）以上となる最小の c を選択する．ただし，PCA は識別におけるクラスの概念をもたないため，寄与率の低い次元が識別に影響を及ぼさないとは限らない点に注意が必要である．例えば，**図 1·2** に示すように，PCA ではデータ全体の分布を表現するような主軸（第 1 固有ベクトル $\mathbf{u}_1$）が選ばれるが，これは明らかに二つのクラスを分離するのに適していない．クラスの分布を考慮して識別に有効な変換を行う方法については，本章 3-2 節の判別分析で述べる．

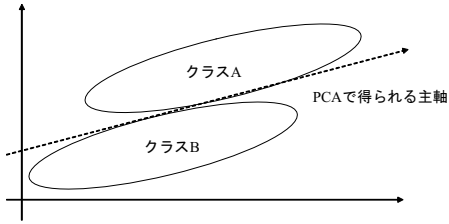


図 1·2　PCA で得られる主軸が識別に有効とは限らない例

ところで，画像を入力ベクトルとするなど，原特徴の次元数 d がデータ数 n よりも非常に大きい場合，原特徴の共分散行列 $\mathbf{S}$ は最大でも $n-1$ 個の固有値しかもたないため，$d \times d$ 次元の固有値問題を解くのは無駄が多い．PCA の導出において，原点中心となるように平均ベクトルを引いてから原特徴を並べたデータ行列を $\mathbf{X} = (\mathbf{x}_1 - \mathbf{m}, \mathbf{x}_2 - \mathbf{m}, \ldots, \mathbf{x}_n - \mathbf{m})^T$ とおけば，$\mathbf{S} = \frac{1}{n}\mathbf{X}^T\mathbf{X}$ と表せることから，$\mathbf{SA} = \mathbf{A\Lambda}$ の両辺に $n\mathbf{X}$ をかけて変形すれば，$(\mathbf{XX}^T)(\mathbf{XA}) = (\mathbf{XA})(n\mathbf{\Lambda})$ が得られる．これは，$n \times n$ 行列 $\mathbf{XX}^T$ の固有値問題となり，$n < d$ の場合には $d \times d$ 行列 $\mathbf{S}$ の固有値問題よりも計算が簡単になる[2)]．$\mathbf{XX}^T$ の固有ベクトルを並べた行列を $\mathbf{A}' = \mathbf{XA}$，固有値を要素とする対角行列を $\mathbf{\Lambda}' = n\mathbf{\Lambda}$ とおけば，$\mathbf{X}^T\mathbf{A}' = \mathbf{X}^T\mathbf{XA} = n\mathbf{SA} = n\mathbf{A\Lambda} = \mathbf{A\Lambda}'$ より，$\mathbf{A} = \mathbf{\Lambda}'^{-1}\mathbf{X}^T\mathbf{A}'$ となる．ただし，固有ベクトルには定数倍の自由度があるため，求める線形変換 $\mathbf{A}^*$ は，$\mathbf{A}^{*T}\mathbf{A}^* = \mathbf{I}$ の条件を満たすように定数部分を調整すると，$\mathbf{A}^* = \mathbf{\Lambda}'^{-\frac{1}{2}}\mathbf{X}^T\mathbf{A}'$ が得られる．ただし，$\mathbf{A}'$ についても，$\mathbf{A}'^T\mathbf{A}' = \mathbf{I}$ を満たすよう規格化されているものとする．

ここまでは線形変換による線形 PCA について述べたが，近年，カーネル関数を利用したカーネル PCA と呼ばれる非線形手法が注目されている[2, 35)]．カーネル関数による非線形化は，本章 3-4 節の SVM でも用いられており，非線形変換 $\boldsymbol{\phi}(\mathbf{x})$ によって原特徴空間での非線形問題を高次元空間での線形問題としてとらえる手法である．$\boldsymbol{\phi}(\mathbf{x})$ で定義された高次元空間での線形 PCA は，カーネル関数と呼ばれる高次元空間での内積演算 $K(\mathbf{x}, \mathbf{y}) = \boldsymbol{\phi}(\mathbf{x})^T\boldsymbol{\phi}(\mathbf{y})$ によって構成することができるため，$\boldsymbol{\phi}(\mathbf{x})$ を直接計算する場合に比べて大幅に計算効率が良くなる．このような手法はカーネルトリックと呼ばれている．

■2 群 - 2 編 - 1 章

1-3 分類器

1-3-1 パーセプトロン

（執筆者：仙田修司）［2011 年 2 月 受領］

パーセプトロンは，Rosenblatt が提案[29]した視覚と脳の機能をモデル化したパターン分類機械であり，感覚（S）層，連合（A）層，応答（R）層の 3 層構造からなる．S 層と A 層はランダムに結合されており，A 層と R 層の間でのみ学習を行うことから，以下では，A 層を入力層，R 層を出力層とする 2 層パーセプトロンについて述べる．

2 層パーセプトロンでは，**図 1･3** に示すように，入力となる特徴ベクトル $(x_1, x_2, \ldots, x_d)^T$（$T$ は転置を表す）に対して，出力を $z = f(w_0 + \sum_{i=1}^{d} w_i x_i)$ と定義する．ここで，$\{w_1, w_2, \ldots, w_d\}$ は，入力に対する重み係数，w_0 は定数項，関数 $f(\cdot)$ は出力値を得るための（非線形）関数である．関数 $f(\cdot)$ には，しきい値関数（$a \geq 0$ のとき $f(a) = 1$，それ以外のとき $f(a) = 0$），シグモイド関数 $f(a) = 1/(1 + e^{-a})$，恒等関数 $f(a) = a$ などが用いられる．また，拡張特徴ベクトル $\mathbf{x} = (1, x_1, x_2, \ldots, x_d)^T$ と，拡張重みベクトル $\mathbf{w} = (w_0, w_1, w_2, \ldots, w_d)^T$ を定義することによって，出力は $z = f(\mathbf{w}^T\mathbf{x})$ と表記できる．

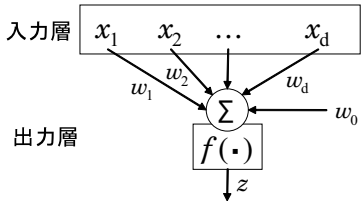


図 1･3　2 層パーセプトロンの構成

2 層パーセプトロンの学習では，n 個の学習ベクトル $\{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n\}$ と，それに対応する教師出力データ $\{z_1, z_2, \ldots, z_n\}$ が与えられたとき，出力の 2 乗誤差 $J = \frac{1}{2}\sum_{i=1}^{n}\{z_i - f(\mathbf{w}^T\mathbf{x}_i)\}^2$ が最小となるように重みベクトル $\mathbf{w}$ を決定する．$f(\cdot)$ が恒等変換 $f(a) = a$ の場合，最急降下法によって $\mathbf{w}$ の更新式を求めると，ρ を学習のための定数として，

$$\mathbf{w} \Leftarrow \mathbf{w} + \rho\frac{\partial J}{\partial \mathbf{w}} = \mathbf{w} + \rho\sum_{i=1}^{n}(z_i - \mathbf{w}^T\mathbf{x}_i)\mathbf{x}_i \tag{1･7}$$

が得られ，これは Widrow-Hoff の学習規則と呼ばれている．$f(\cdot)$ をしきい値関数とした場合には，式 (1･7) は Rosenblatt が提案した誤分類パターンのみを学習する方式と一致し，この場合，学習データが線形分離可能な場合には正しい重みに収束することが知られている．

また，学習ベクトルを並べた行列を $\mathbf{X} = (\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n)^T$，教師データを並べたベクトルを $\mathbf{z} = (z_1, z_2, \ldots, z_n)^T$ と表記すると，$f(\cdot)$ が恒等変換のとき，$J = \frac{1}{2}(\mathbf{z} - \mathbf{w}^T\mathbf{X}^T)^T(\mathbf{z} - \mathbf{w}^T\mathbf{X}^T)$ と書くことができ，$\partial J/\partial \mathbf{w} = 0$ を解くことによって，J を最小とする $\mathbf{w}^*$ は，

$$\mathbf{w}^* = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{z} \tag{1･8}$$

のように，解析的に求めることができる．これは重回帰分析と同じ定式化となる．

1-3-2 判別分析

（執筆者：仙田修司）［2011 年 2 月 受領］

判別分析は，クラス分けされた学習データが与えられた際に，クラス間をよく分離するための基準（判別関数）を得る手法である．判別関数として線形関数を用いる線形判別分析（Linear Discriminant Analysis，以下 LDA）について，まず，2 クラスの場合を述べる．

LDA では，d 次元特徴ベクトル $\mathbf{x} = (x_1, x_2, \ldots, x_d)^T$ に，d 次元の係数ベクトル $\mathbf{A} = (a_1, a_2, \ldots, a_d)^T$ をかけた値 $z = \mathbf{A}^T\mathbf{x}$ によって，与えられた 2 クラスが最も分離するように $\mathbf{A}$ を求める．このために，z のクラス内分散を σ_W^2，クラス間分散を σ_B^2 としたとき，両者の比 σ_B^2/σ_W^2 を最大化するように $\mathbf{A}$ を決定する手法を，フィッシャー判別分析[7]と呼ぶ．

i 番目のクラス（$i = 1, 2$）に関して，学習データの集合を C_i，学習データ数を n_i とし，$z = \mathbf{A}^T\mathbf{x}$ のクラス内平均を $m_i = \frac{1}{n_i}\sum_{\mathbf{x}\in C_i}\mathbf{A}^T\mathbf{x}$，全平均を $m = (n_1 m_1 + n_2 m_2)/(n_1 + n_2)$ とすれば，

$$\sigma_W^2 = \sum_{i=1,2}\{\frac{1}{n_i}\sum_{\mathbf{x}\in C_i}(\mathbf{A}^T\mathbf{x} - m_i)^2\},\ \sigma_B^2 = \sum_{i=1,2}(m_i - m)^2 \tag{1·9}$$

と表せる．$\mathbf{x}$ のクラス内平均を $\mathbf{m}_i = \frac{1}{n_i}\sum_{\mathbf{x}\in C_i}\mathbf{x}$，全平均を $\mathbf{m} = (n_1\mathbf{m}_1 + n_2\mathbf{m}_2)/(n_1 + n_2)$ とすれば，特徴空間におけるクラス内共分散行列 $\mathbf{S_W}$ とクラス間共分散行列 $\mathbf{S_B}$ は，

$$\mathbf{S_W} = \sum_{i=1,2}\{\frac{1}{n_i}\sum_{\mathbf{x}\in C_i}(\mathbf{x} - \mathbf{m}_i)(\mathbf{x} - \mathbf{m}_i)^T\},\ \mathbf{S_B} = \sum_{i=1,2}(\mathbf{m}_i - \mathbf{m})(\mathbf{m}_i - \mathbf{m})^T \tag{1·10}$$

と定義され，$m_i = \mathbf{A}^T\mathbf{m}_i$，$m = \mathbf{A}^T\mathbf{m}$ であることから，$\mathbf{A}^T\mathbf{S_W}\mathbf{A} = \sigma_W^2$，$\mathbf{A}^T\mathbf{S_B}\mathbf{A} = \sigma_B^2$ であることが分かる．よって，評価基準 σ_B^2/σ_W^2 を最大化するには，$\mathbf{A}$ に定数倍の自由度があることを考慮すると，$\sigma_W^2 = \mathbf{A}^T\mathbf{S_W}\mathbf{A} = 1$ の制約の下で，$\sigma_B^2 = \mathbf{A}^T\mathbf{S_B}\mathbf{A}$ を最大化すればよい．ラグランジュの未定乗数法より，

$$L(\mathbf{A}, \lambda) = \mathbf{A}^T\mathbf{S_B}\mathbf{A} - \lambda(\mathbf{A}^T\mathbf{S_W}\mathbf{A} - 1) \tag{1·11}$$

を $\mathbf{A}$ で偏微分して 0 とおけば，$(\mathbf{S_B} + \mathbf{S_B}^T)\mathbf{A} - \lambda(\mathbf{S_W} + \mathbf{S_W}^T)\mathbf{A} = 0$ となり，共分散行列は対称行列であるから，$\mathbf{S_B}\mathbf{A} = \lambda\mathbf{S_W}\mathbf{A}$ が得られる．$\mathbf{S_W}$ が逆行列をもつとき，

$$(\mathbf{S_W}^{-1}\mathbf{S_B})\mathbf{A} = \lambda\mathbf{A} \tag{1·12}$$

と変形できることから，λ は行列 $\mathbf{S_W}^{-1}\mathbf{S_B}$ の最大固有値（$\sigma_B^2/\sigma_W^2 = \lambda$ を最大化するため），$\mathbf{A}$ は対応する固有ベクトルとなる．

クラス数が K (≥ 2) の場合，$\mathbf{A}$ を $d \times (K-1)$ 行列として線形変換 $\mathbf{z} = \mathbf{A}^T\mathbf{x}$ を定義すれば，2 クラスの場合と同様な議論が行える．これを正準判別分析と呼ぶ．$\mathbf{z}$ におけるクラス内共分散を $\mathbf{S_{ZW}}$，クラス間共分散を $\mathbf{S_{ZB}}$ とすると，評価基準 $\mathrm{Tr}(\mathbf{S_{ZW}}^{-1}\mathbf{S_{ZB}})$ を最大化する $\mathbf{A}$ を求めればよい．2 クラスの場合と同様な手順により，結局，$\mathbf{S_W}^{-1}\mathbf{S_B}$ の固有値を大きいものから $K-1$ 個取ってきた λ_i と，それらに対応する固有ベクトル $\mathbf{u}_i$ ($i = 1, 2, \ldots, K-1$) を使って，$\mathrm{Tr}(\mathbf{S_{ZW}}^{-1}\mathbf{S_{ZB}})$ の最大値は $\sum_{i=1}^{K-1}\lambda_i$，そのときの線形変換は $\mathbf{A} = (\mathbf{u}_1, \mathbf{u}_2, \ldots, \mathbf{u}_{K-1})$ となる．

なお，判別分析は各クラスが分離しやすい空間を構成する手法であり，パターン識別のためにクラス間の境界が必要な場合は，別の方法で定める必要がある．

1-3-3 ニューラルネットワーク

（執筆者：佐藤真一）［2011 年 2 月 受領］

脳は膨大な数の神経細胞のネットワークによる情報処理システムとして機能しており，パターン認識などの知的処理を極めて柔軟に実現可能である．ニューラルネットワークとは，ここに着想を得，ソフトウェアなどで人工的に実現した神経細胞（ニューロン）並びにそのネットワークにより，柔軟なパターン認識などの知的処理の実現を目的とした計算モデルである．ニューラルネットワークに関する研究には，ニューロンや脳の機能や動作原理の解明という生理学的側面，人間の知能の解明という認知心理学的側面，ニューラルネットワークにより実現される並列分散処理の原理の解明という情報科学的側面などがある．

ニューロンは，ほかの複数のニューロンから信号を受け取り，そのパターンに基づき信号を発し，更にほかのニューロンに信号を伝えるという役割をもつ．ニューロン間の信号を伝達する部分をシナプスと呼ぶ．ニューロンの動作のモデル化を，実際の生理モデルに従って精密に行おうという研究もあるが，McCulloch-Pitts モデルは入力信号の線形和と閾値関数により構成される単純なモデルである[28]．また，Hebb は，ニューロンが興奮した（＝信号強度が高い）場合，寄与した入力のシナプス結合強度を強めることにより，学習が可能であることを示した[17]．この二つがニューラルネットワークの基本原理である．

ニューラルネットワークは，結合形態に基づき大きく二つに分けられる．一つは階層型ネットワークと呼ばれ，ネットワークへの入力となる入力層，出力となる出力層と，その間に存在する複数の中間層から構成される．ニューロン間の接続は入力層から中間層を経由して出力層へと一方向に限られており，フィードフォワードネットワークとも呼ばれる．一方，逆向きの結合（フィードバック結合）も許す場合は相互結合型ネットワーク，あるいはリカレント型ネットワークと呼ばれる．階層型ネットワークでは入力が与えられると出力が静的に得られるが，相互結合型ネットワークでは，入力に対して徐々に安定状態に収束したり，振動状態に至ったりするダイナミズムをもつ．また，学習様式に基づき，教師あり学習と教師なし学習とに分けられる．教師あり学習では，入力信号と併せて望ましい出力信号が与えられ，これらに基づいて学習が行われる．一方，教師なし学習では入力のみから自律的に学習が行われる．

教師あり学習を行う階層型ネットワークの例が，前出のパーセプトロンであり，入力層，一層の中間層，そして出力層の三層からなる．パーセプトロンは線形識別のみ可能であったが，中間層を複数にすることによって複雑な識別面が実現可能となる．現在広く使われるのは多数の中間層をもつ階層型ネットワークである．その学習には，パーセプトロンの学習と類似しているが，より高速で正確な学習が可能な一般化デルタルール[30]が用いられる．また，多数の階層での学習を実現するため，まず出力層から学習を開始し，順次前段階の層の出力誤り（実際の出力と望ましい出力の差）を推定しながら，出力層から入力層に向けて逆向きに，各層で一般化デルタルールに基づく学習を行う誤差逆伝搬アルゴリズム（error backpropagation algorithm, BP 法とも呼ばれる）[31]が利用される．教師なし学習を行う階層型ネットワークとして有名なのが自己組織化マップ（Self Organizing Maps, SOM）である[23]．SOM では，競合学習により相互に類似した入力に対して同一の出力を与えるようなパターン分類を行うと

同時に，出力ニューロン（ユニット）の幾何的な配置も考慮して近傍ユニットの結合重みを同時に変更するという学習を行うことにより，入力のトポロジーを保存した変換が実現される．例えば，入力の多次元ベクトル間の類似度によるトポロジーをある程度保存したまま，各ベクトルを 2 次元空間に配置することなどが可能である．このほか，階層型ネットワークとしては，層間の結合に視覚野のような局所性を取り入れ，文字認識などへの有効性が示されたネオコグニトロンが知られている[14]．

相互結合型ネットワークの例としてよく知られているのが Hopfield モデルである[19]．フィードバック結合をもつことにより，ネットワークの状態変化は無限に連続し得るが，ネットワーク間の結合重みが一定の条件に従う場合，必ず収束することが示されている．Hopfield モデルでは，各ユニットの入力に従い出力は決定論的に確定するが，入力に従って出力を確率的に決定する（入力により出力変化の確率を決める）モデルが考えられており，ボルツマンマシンと呼ばれている[18]．Hopfield モデルでは，局所最適解に陥ってしまうという問題があった．一方ボルツマンマシンでは，確率を決定する式に温度と呼ばれるパラメータ T が存在し，状態変化の初期段階では T を大きくすることにより，局所安定点ではない不安定な状態を取ることを可能とし，徐々に T を小さくすることにより安定状態に収束させることにより，時間をかければ大局的な最適解を得られることが保障されている．徐々に温度パラメータを下げることから，焼きなまし法（simulated annealing）と呼ばれる．これらの相互結合型ネットワークの応用として，Hopfield モデルによる連想記憶や巡回セールスマン問題の近似解法[20]，ボルツマンマシン（あるいは焼きなまし法）による画像復元[15]などが知られている．

1-3-4 SVM

（執筆者：仙田修司）

サポートベクターマシン（Support Vector Machines, 以下 SVM）は，Vapnik らによって考案[37]された統計的学習理論に基づく学習アルゴリズムであり，近年，様々な分野で広範囲に応用されている．SVM の特長として，ニューラルネットワークと同様に非線形最適化能力をもちながら大域的最適化により学習結果が一意に定まる点，学習データに対する識別能力だけでなく未学習データに対する汎化能力に優れている点，簡単に利用可能な汎用ライブラリが公開されている点などが挙げられる[3]．最初に，最も単純な SVM として，線形分離可能なデータを対象として 2 クラス分類を行う線形 SVM について述べる．

d 次元特徴ベクトル $(x_1, x_2, \ldots, x_d)^T$（$T$ は転置を表す）に対して，クラスを表すラベル $y \in \{-1, 1\}$ が与えられているとする．線形識別関数 $f(\mathbf{x})$ は，d 次元重みベクトルを $\mathbf{w}$，定数項を b とすれば，$f(\mathbf{x}) = \mathbf{w}^T\mathbf{x} + b$ と表せる．n 個の学習データ $\mathbf{x}_i$ と教師ラベル y_i に対して，

$$y_i f(\mathbf{x}_i) = y_i(\mathbf{w}^T\mathbf{x}_i + b) \geq 1,\ i = 1, 2, \ldots, n \tag{1·13}$$

が成立するように $\mathbf{w}$ と b を学習すれば，未知データに対する $f(\mathbf{x})$ の正負によって分類すべきクラスが決められる．学習データが線形分離可能であればこのような解は多数存在し得るが，線形 SVM では識別境界に最も近い学習データから識別境界までの距離（マージンと呼ばれる，**図 1·4** 参照）を最大化することによって，未知データに対してもなるべく良い分離が得られるように学習を行う．このときの識別境界に最も近い学習データをサポートベクタと呼ぶ．特徴ベクトル $\mathbf{x}$ から識別境界 $f(\mathbf{x}) = 0$ までの距離は $|f(\mathbf{x})|/\sqrt{\mathbf{w}^T\mathbf{w}}$ で表され，式 (1·13) が成立するときの $|f(\mathbf{x})|$ の最小値は 1 であるから，そのときのマージンは $1/\sqrt{\mathbf{w}^T\mathbf{w}}$ となり，

これを最大化することは $\mathbf{w}^T\mathbf{w}$ の最小化と同値である．式 (1･13) の制約の下で $\mathbf{w}^T\mathbf{w}$ を最小化するには，ラグランジュ乗数 $\alpha_i \geq 0,\ i = 1, 2, \ldots, n$ を導入することで，

$$L(\mathbf{w}, b, \boldsymbol{\alpha}) = \frac{1}{2}\mathbf{w}^T\mathbf{w} - \sum_{i=1}^{n} \alpha_i\{y_i f(\mathbf{x}_i) - 1\} \tag{1･14}$$

を最小化することによって求められる．導出は省略するが，結局，サポートベクターに対応する α_i のみが $\alpha_i > 0$ となり（それ以外の α_i は 0），重みベクトル $\mathbf{w} = \sum_{i=1}^{n} y_i\alpha_i\mathbf{x}_i$ となる．

ここまでは，学習データが線形分離可能であるとしてきたが，そうでない場合は式 (1･13) を満たす解が存在しない．そこで，学習データごとに $\xi_i \geq 0$ だけのはみ出しを許して，

$$y_i f(\mathbf{x}_i) \geq 1 - \xi_i,\ i = 1, 2, \ldots, n \tag{1･15}$$

を新たな制約条件とし，はみ出しの総和をペナルティとして加えた $\frac{1}{2}\mathbf{w}^T\mathbf{w} + c\sum_{i=1}^{n}\xi_i$ を最小化する手法をソフトマージン最適化と呼ぶ．c は学習時に与える定数である．なお，ソフトマージン最適化の解は，ソフトマージンがない $\xi_i = 0$ の場合（ハードマージンと呼ばれる）において，$\alpha_i < c$ の制約を加えた場合と同値であることが知られている．

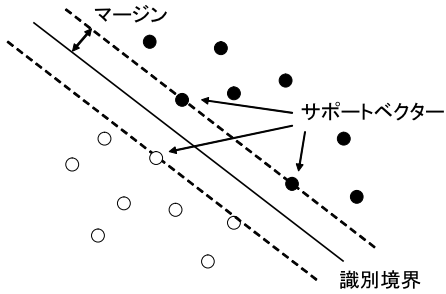


図 1･4　2 クラス線形 SVM におけるマージンの例

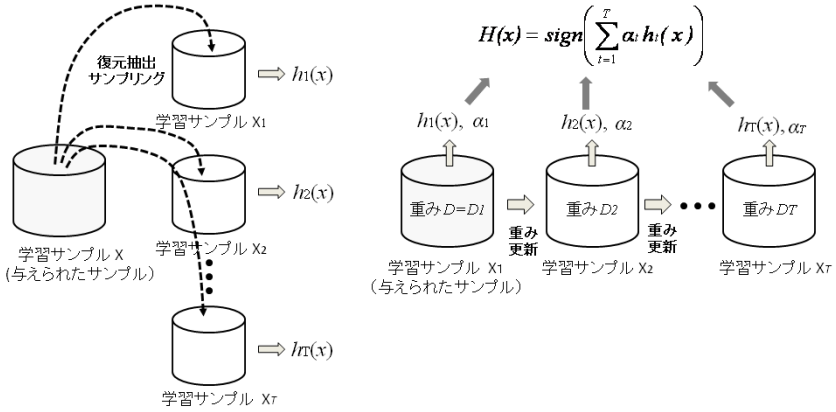
次に，SVM を非線形識別器へと拡張するために，カーネルトリックと呼ばれる手法を導入する．特徴ベクトルを入力とする非線形変換 $\boldsymbol{\phi}(\mathbf{x})$ によって，特徴空間を $\boldsymbol{\phi}(\cdot)$ がつくる高次元空間に写像し，この高次元空間上で線形 SVM を行うことで特徴空間上での非線形識別が達成できる．実際には高次元空間への変換は行わず，高次元空間での内積計算を定義したカーネル関数 $K(\mathbf{x}, \mathbf{z}) = \boldsymbol{\phi}(\mathbf{x})^T\boldsymbol{\phi}(\mathbf{z})$ を用いる手法がカーネルトリックである．代表的なカーネル関数として，多項式カーネル $K(\mathbf{x}, \mathbf{z}) = (\mathbf{x}^T\mathbf{z} + \beta)^\gamma$，RBF（Radial Basis Function）カーネル $K(\mathbf{x}, \mathbf{z}) = e^{-\beta(\mathbf{x}-\mathbf{z})^T(\mathbf{x}-\mathbf{z})}$，シグモイドカーネル $K(\mathbf{x}, \mathbf{z}) = \tanh(\beta\mathbf{x}^T\mathbf{z} + \gamma)$ などがある．カーネル関数は，非線形識別問題を線形識別問題に変換するために導入するため，データの分布に応じて最適なものを選択する必要がある．

1-3-5 アサンブル学習法

（執筆者：福井和広）［2011 年 2 月 受領］

アサンブル学習法とは，単一の識別器の識別性能を最大限に高める替わりに，識別性能が

低い識別器（弱識別器，あるいは弱仮説）を複数用意して，これらをうまく結合して，高い識別性能をもつ識別器（強識別器，あるいは最終仮説）を実現する方式の総称である．これまでに様々なアンサブル学習法が提案されているが，以下では代表的な方法であるバギング法とブースティング法について説明する．**図 1･5** に示すように両者では，各弱識別器の生成法とそれらの結合法が異なる．



(a) **バギング法**, (b) **ブースティング法**

図 1･5　代表的なアンサブル学習法

(1) バギング法

バギング法はアンサブル学習法のなかでは比較的単純な方式である．図 1･5(a) に示すように，与えられた学習サンプル X から復元抽出により T 個の学習サンプルセット X_1~X_T を生成し，これらの学習サンプルセットからそれぞれ独立に T 個の弱識別機 $h_i(x), (i = 1\sim T)$ を学習する．ここで復元抽出とは，抽出したサンプルを元に戻して再度ランダムに抽出する方法である．識別は T 個の弱識別器の多数決，あるいはそれらから出力される識別関数値の重み平均値に基づいて行われる．

(2) ブースティング法

ブースティング法では，弱識別器を並列で学習するバギング法に対して，図 1･5(b) に示すように，誤認識した学習サンプル i の重み $D(i)$ を大きくしながら，複数の弱識別器を逐次的に学習する．ブースティング法の代表的な方法が AdaBoost[8]である．この方法の流れは以下のようになる．まず与えられた学習サンプル X_1 を用いて弱識別器 $h_1(x)$ を学習した後，この識別器 $h_1(x)$ で学習サンプル X_1 を識別する．この際，誤識別した学習サンプルを重点的に学習するために，そのサンプルの重みを大きくした新たな学習サンプルセット X_2 を生成する．続いて学習サンプルセット X_2 を用いて学習した弱識別器 $h_2(x)$ で X_2 を識別し，先と同様に誤認識したサンプルの重みを大きくした学習サンプルセット X_3 を生成する．以下，同様の

処理を繰り返して，T 個の弱識別器を学習する．強識別器は T 個の弱識別器の重み付き結合により構成する．以下では，m 個の学習サンプル $(x_1, y_1), \ldots, (x_m, y_m)$ が与えられたとして，AdaBoost のアルゴリズムを示す．ここで x_i は i 番目のサンプル，$y_i \in \{+1, -1\}$ は x_i が属するクラスラベルである．

step0 各サンプルの重み $D_1(i)$ を，$D_1(i) = 1/m$ として均一に設定する．

step1 For $t = 1, \ldots, T$：

1-1 学習サンプル X_t に対する誤り率 $\epsilon_t = \sum_{i:y_i \neq h_t(x_i)} D_t(i)$（誤認識したサンプルの重みの総和）が最小となる弱識別器を $h_t(x)$ とする．

1-2 弱識別器 $h_t(x)$ の信頼度 $\alpha_t = \frac{1}{2} ln(\frac{1-\epsilon_t}{\epsilon_t})$ を求める．

1-3 サンプル i の重み $D_t(i)$ を次のルールにより更新する．

$$D_{t+1}(i) = \frac{D_t(i)}{Z_t} \times \begin{cases} exp(-\alpha_t) & \text{if } h_t(x_i) = y_i \\ exp(\ \alpha_t) & \text{if } h_t(x_i) \neq y_i \end{cases}$$

$$= \frac{D_t(i) exp(-\alpha_t y_i h_t(x_i))}{Z_t}$$

ここで Z_t は正規化量で $Z_t = \sum_{i=1}^{m} D_t(i) exp(-\alpha_t y_i h_t(x_i))$ である．

Step2 Step1 で得られた T 個の弱識別器を信頼度 α_t で重み平均した識別器 $g(x) = \sum_{t=1}^{T} \alpha_t h_t(x)$ を構成し，その出力値の符号 $H(x) = sign(g(x))$ に基づいて $\{+1, -1\}$ の 2 クラス識別を行う．

AdaBoost のアルゴリズムは一見するとヒューリスティックな方法に見えるが，指数損失関数を最小化するという考えに基づいており，理論基盤はしっかりしている[8]．設定すべきパラメータは学習する弱識別器の数 T のみであり，使いやすい方法である．この T に関しても，大きくとっても過学習を起こしにくいとされており，設定が比較的容易である．これまでに AdaBoost の様々な拡張方式が提案されている．例えば，Real AdaBoost[33]や，損失関数を指数関数から別のタイプの関数に変更した MadaBoost[4]や LogitBoost[5]などが挙げられる．更に AdaBoost は 2 クラス問題を対象としているが，複数クラス問題へ拡張した方法，AdaBoost.M1，AdaBoost.M2[9]が提案されている．

AdaBoost の応用範囲は極めて広いが，コンピュータビジョンにおける代表的な適用事例は顔検出[38]であろう．この事例では顔検出を入力画像を顔と非顔クラスの 2 クラスに識別する問題としてとらえ，これに AdaBoost を適用している．ここでは弱識別器として Haar-like 特徴と呼ばれる単純な特徴量に基づいて判別を行う関数が使われている．Haar-like 特徴は画像の一部に隣り合うように設定された二つの矩形領域の平均輝度差として得られる．この方法の大まかな流れは以下のようになる．まず事前に前述の矩形の位置やサイズを多様に変化させることで，多様かつ大量な Haar-like 特徴を用意しておく．学習では，これらの用意された Haar-like 特徴から識別器の誤認識率を最も小さくするものを一つ選択し，これに基づく識別器を第 1 の弱識別器とする．以下，先に説明した AdaBoost のアルゴリズムに基づいて，

Haar-like 特徴を逐次選択していき，T 個の弱識別器を得る．最終的な強識別器はこれらの弱識別器の重み付き結合により構成される．Haar-like 特徴を逐次選択していく過程は顔と非顔を識別するために有効な識別特徴を抽出する過程とも見なせ，AdaBoost が特徴抽出にも有効であることが分かる．

1-3-6 部分空間法

（執筆者：福井和広）[2011 年 2 月 受領]

部分空間法とその一連の拡張法を用いたパターン識別について概説する．これまでに様々なタイプのパターン識別法が提案され，画像認識へ適用されてきたが，実使用に耐えられた識別法はそれ程多くない．部分空間法はそのような方法の一つであり，商用の文字認識や顔認識システムの識別エンジンとして実際に使用されている．部分空間法が日本人により提案されてから，30 年以上が経過した現在でも，理論拡張が精力的に試みられている[24, 13]．

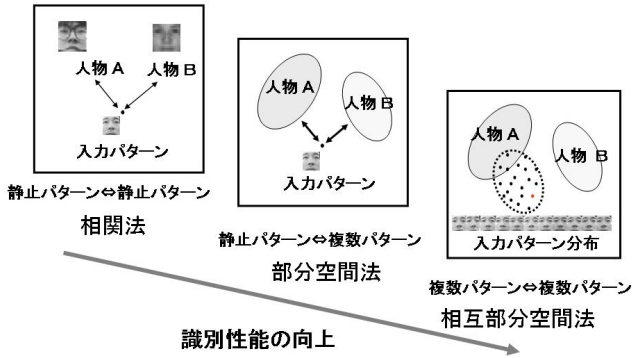


図 1･6　相関法から相互部分空間法へ

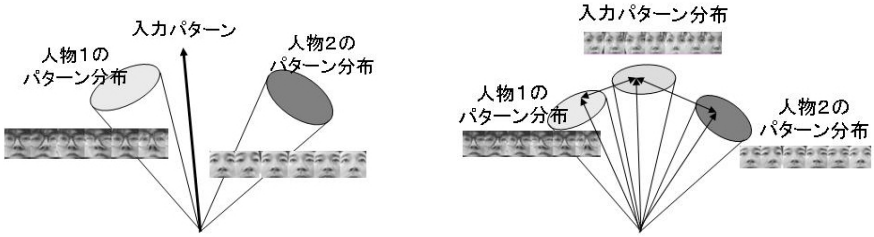
(1) 角度ベースの識別法

画像認識では，通常，画像パターンを 1 行に展開したベクトルとして扱う．つまり，$d \times d$ ピクセル の画像パターンは $d \times d$ 次元ベクトル空間のベクトルと見なす．これにより二つの画像の類似度は，両者に対応する二つのベクトルのなす角度（相関）に基づいて定義されることになる．角度ベースの識別法を整理すると，**図 1･6** のようになる．最も単純な角度ベースの識別法は，入力ベクトルと辞書ベクトルの角度を類似度とする正規化相関法である．この方法は実装が容易なこともあり幅広く使われている．しかしながら，1 枚の辞書パターンだけでそのクラスのパターン変形を十分に表現することは難しく，一般には高い識別性能は期待できない．これに対して，部分空間法ではパターン分布を辞書とすることで，パターンの変形に対する吸収能力を大幅に高めた．一般に，同一クラスに属する画像パターンの分布は高次元ベクトル空間における低次元の部分空間に局在していることが知られている．部分空間法ではこの特性に着目して，画像パターン分布をベクトル空間における低次元部分空間で表す．これにより類似度は入力ベクトルと部分空間のなす角度で定義されることになる．この正規化相関法から部分空間法への自然な拡張を更に進めると，入力側もパターン分布として

部分空間で表す識別法が考えられる．この方法は相互部分空間法[27]と呼ばれており，パターン変形に対する吸収能力を更に高めたものになっている．

(2) 部分空間法による識別

図 1･7(a) に部分空間法の概念図を示す．この図では顔認識への適用例が示されており，各人物に対してそれぞれ辞書部分空間が用意されている．まず学習フェーズでは各人物ごとに，様々な撮影条件で撮影した複数の顔パターンを収集し，これらからその人物の辞書部分空間を生成する．識別フェーズでは入力顔パターン（ベクトル）と各クラス部分空間とのなす角度（あるいは射影長）を求めて，最も小さい角度（射影長が長い）のクラスに入力顔パターン（ベクトル）を識別する．各クラス部分空間を張る基底ベクトルは，そのクラスに属する学習パターンセットから計算される自己相関行列の固有ベクトルとして求まる．部分空間の最適な次元数を理論的に求める方法は確立されておらず，寄与率に基づいて実験的に決める必要がある．



(a) **部分空間法，** (b) **相互部分空間法**

図 1･7　部分空間法ベースの方法の概念図

(3) 相互部分空間法による識別

相互部分空間法の概念図を図 1･7(b) に示す．この方法では，入力側もベクトルの代わりに部分空間を用いて，入力部分空間と各クラス部分空間のなす角度に基づいて両者の類似度を測る．この角度は正準角と呼ばれ，M 次元部分空間 $\mathcal{P}$ と N 次元部分空間 $\mathcal{Q}$（便宜上，$M \geq N$ と仮定）の間には N 個の正準角が定義できる．第 1 正準角 θ_1 は両者のなす最小角である．第 2 正準角 θ_2 は最小正準角 θ_1 に直交する方向において計った最小角，第 3 正準角 θ_3 は第 2 正準角 θ_2 に直交する方向で計った最小角である．以下同様に N 個の正準角が順次求まる．これらの正準角は比較する二つの部分空間の正規直交基底ベクトルのみから容易に計算できる．識別時には入力パターン分布から入力部分空間を生成し，各クラス部分空間とのなす正準角を測り，最も小さい正準角に該当するクラスに入力パターン分布を識別する[11]．

(4) 部分空間法・相互部分空間の性能改善

部分空間法・相互部分空間法において生成されるクラス部分空間はパターン分布を最小自乗の観点で最良近似する空間にはなっているが，識別の観点からは必ずしも最適とはいえない．そこで他クラスとの関係を考慮することで，識別性能を向上させる様々な拡張が試みら

れている．例えば，部分空間法については，多クラスとの差分ベクトルを識別に考慮する混合類似度法，各クラス部分空間を事前にお互いにできるだけ直交しておく直交部分空間法，主成分分析の際に，他クラスのパターンの射影成分をできるだけ小さくする制限を加えた相対主成分分析法[39]，誤認識したサンプルに基づいて各クラス部分空間の関係を逐次修正する学習部分空間法[24]などである．相互部分空間法についても同様の拡張がなされており，各クラス部分空間を直交化したうえで相互部分空間法を適用する直交相互部分空間法（白色化相互部分空間法）[21, 22]や，有効な識別特徴のみから構成される一般化差分部分空間へ射影した後に相互部分空間法を適用する制約相互部分空間法[11]などが提案されている．他クラスとの関係を考慮する方法だけではなく，各クラスパターン分布の分散を表す重みを各クラス部分空間の基底ベクトルに対して導入することで識別性能を高めた重み付き部分空間法[24]も提案されている．更に，近年，非線形サポートベクタマシンでも使われているカーネル関数を用いた非線形識別への拡張法も行われ，カーネル部分空間法[26, 36]，カーネル相対主成分分析法，非線形核相互部分空間法[32]，カーネル非線形制約相互部分空間法[11]，カーネル非線形直交相互部分空間法などが提案されている．これらの非線形拡張法により，3 次元物体の多視点画像パターン分布のように非線形構造が強く，線形部分空間ベースでは識別が難しかったパターン分布に対しても高い識別性能が実現できるようになった[24, 13, 11]．

部分空間法・相互部分空間法の理論に関しては解説[24, 13]に詳しい．更に最新動向及び応用事例については，2006 年から毎年開催されている部分空間法に関するワークショップ *Subspace* 2006-2009 が参考になる．

■2 群 - 2 編 - 1 章

1-4 評価法

（執筆者：福井和広）[2011 年 2 月 受領]

識別器の性能を有限個のサンプルから高精度で推定することは重要な課題である[16]．一般に性能は誤認識で測られる場合が多く，確率密度関数と事前確率が既知の場合には Bayes 識別器が与える誤認識率が有効な指標となる．しかしながら，一般に収集できるサンプル数には限りがあり，確率密度関数の高精度な推定は困難である．仮にこれらが分かったとしても，特徴ベクトルの次元が高くなると，確率密度関数の多重積分が計算困難となる．以下では四つの方法を紹介するが，この内，再代入法は誤認識の下限を知るうえでは有効であるが，実際的な評価指標としてはあまり使われない．利用できるサンプル数が少ない場合には，交差確認，あるいはブートストラップ法の使用が好ましい．

1-4-1 再代入法（Resubstitution method）

この方法では識別器の学習に用いた学習サンプルを，評価サンプルとしても使用する．容易に実施できるが，学習と評価に同じサンプルを用いるので，真の誤認識率よりも低い方へ偏った推定値が得られる．

1-4-2 分割法（Hold-out method）

この方法では全サンプルをランダムに学習サンプルと評価サンプルに分割して，学習サンプルで識別器を学習した後に，残りの評価サンプルで性能評価を行う．この方法は分割法（Hold-out method, 略して H 法）と呼ばれる．十分な数の学習サンプルが確保できる場合には有効な方法であるが，少ないサンプルに適用した場合には，サンプルの一部のみを学習サンプルとして利用するために，真の誤認識率よりも高い方へ偏った推定値が得られる．これを解決するためには，限られたサンプルを効率的に利用する必要があり，次に紹介する交差確認法やブートスラップ法[6]が有効である．

1-4-3 交差確認法（Cross validation method）

分割法の欠点を改善する交差確認法（CV 法）の流れを以下に示す．

1. サンプル集合を K 個の部分サンプル集合に分割する．
2. K 個の内，$K-1$ 個の部分サンプル集合に含まれる全サンプルを用いて識別器を学習した後，残りの 1 個の部分サンプル集合に対する誤認識率を求める．
3. 評価用として取り出す部分サンプル集合を変えて上記と同様の評価を K 回行い，得られた K 個の誤認識率の平均値で性能評価を行う．

上記のようにサンプル集合を K 個のサンプル部分集合に分割する場合を K-Fold Cross-Validation と呼ぶ．また抜き出すサンプル部分集合の要素数が 1 の場合は，一つ抜き法（Leave-one-outmethod，あるいはジャックナイフ法）[25]と呼ばれ，性能評価ではよく使われる方法となっている．CV 法は分割法に比べて，得られる推定値の真値からの偏りが小さいという利

点があり，十分なサンプル数が使えない場合には，CV 法を使うことが好ましい．ただし，得られる識別率の分散が大きい点と，識別器を K 回学習するために計算量が多い点は問題として残る．

1-4-4 ブートストラップ法（Bootstrap method）

真値からの偏りと分散を小さくする方法としてブースストラップ法（BS 法）を紹介する．このアルゴリズムの流れは以下のようになる[6]．

1. n 個のサンプルからなる X_0 から復元抽出により n 個のサンプルを抽出する．この合成されたサンプル集合をブートストラップサンプル集合 $X_{B(n)}$ と呼ぶ．

2. ブートストラップサンプル集合 $X_{B(n)}$ を用いて識別器 $h_{B(n)}(x)$ を学習した後に，同じ $X_{B(n)}$ に対する誤識別率 $e_{B(n)}$ を求める．

3. 識別器 $h_{B(n)}(x)$ の学習サンプル X_0 に対する誤識別率 e_B を求める．

4. 先に求めた誤識別率 $e_{B(n)}$ と e_B から，$\delta = e_B - e_{B(n)}$ を求める．

5. 上記１．～４．の操作を B 回繰り返して B 個の δ を求め，それらの平均を δ^* とする．

6. 最終的に誤識別率の推定値 R_B は次式から求まる．

$$R_B = R_0 + \delta^*$$

ここで R_0 はサンプル X_0 で学習した識別器を，再代入法，つまり同じサンプル X_0 に対して適用して得られた誤識別率である．

繰り返し回数 B は事前に設定する必要があるが，シミュレーションによると一つの目安として 200 が与えられている．

■参考文献

1) 麻生英樹, 津田宏治, 村田　昇, “パターン認識と学習の統計学,” 統計科学のフロンティア 6, 岩波書店, 2003.
2) C.M. Bishop “Pattern Recognition and Machine Learning,” Springer, 2006.（C.M. ビショップ “パターン認識と機械学習,” シュプリンガー・ジャパン, 2007.）
3) N. Cristianini and J. Shawe-Taylor, “An Introduction to Support Vector Machines and Other Kernel-based Learning Methods,” Cambridge University Press, 2000.（N. クリスティアニーニ, J.S. テイラー, “サポートベクターマシン入門,” 共立出版, 2005.）
4) C. Domingo, O. Watanabe, “MadaBoost: A modification of AdaBoost,” COLT’00, pp.180-189, 2000.
5) Jerome Friedman, Trevor Hastie, Robert Tibshirani, “Aditive logistic regression: A statistical view of boosting,” The Annals of Statistic, vol.28, no.2, pp.337-407, 2000.
6) B. Efron, “Bootstrap Methods: Another Look at the Jackknife,” Annals of Statistics, vol.7, no.1, pp.1-26, 1979.
7) R.A. Fisher, “The use of multiple measurement in taxonmic problems,” Annals of Eugenics 7, 1936.

8) Yoav Freund, Robert E. Schapire, “A decision-theoretic generalization of on-line learning and an application to boosting,” Proceedings of the Second European Conference on Computational Learning Theory table of contents, Lecture Notes in Computer Science, vol.904, pp.23-37, 1995.
9) Yoav Freund, Robert E. Schapire, “Experiments with a New Boosting Algorithm,” Proceedings of the Thirteenth International Conference on Machine Learning, pp.148-156, 1996.
10) フロインド ヨアブ, シャピリ ロバート, 安倍直樹, “ブースティング入門（< 特集 > 計算学習理論の進展と応用可能性）,” 人工知能学会, vol.14, no.5, pp.771-780, 1999.
11) 福井和広, 山口　修, “部分空間法の理論拡張と物体認識への応用,” 情報処理学会論文誌コンピュータビジョンとイメージメディア, vol.46, no.SIG 15（CVIM 12）, pp.21-34, 2005.
12) Kazuhiro Fukui, Osamu Yamaguchi, “The Kernel Orthogonal Mutual Subspace Method and Its Application to 3D Object Recognition,” ACCV2007, pp.467-476, 2007.
13) 福井和広, “部分空間法の今昔（いまむかし）（下）: 最近の技術動向 : 相互部分空間法への拡張とその応用事例,” 情報処理学会誌, vol.49, no.6, pp.680-685, 2008.
14) K. Fukushima, “Neocognitron: A Self-Organizing Neural Network Model for a Mechanism of Pattern Recognition Unaffected by Shift in Position,” Biol. Cybern., vol.36, no.4, pp.193-202, 1980.
15) S. Geman and D. Geman, “Stochastic Relaxation, Gibbs Distributions, and the Bayesian Restoration of Images,” IEEE Trans. Pattern Analysis and Machine Intelligence, vol.6, no.6, pp.721-741, 1984.
16) 浜本義彦, “統計的パターン認識入門,” 森北出版, 2009.
17) D.O. Hebb, “The Organization of Behavior,” Willey, 1949.
18) J. Hinton and T. Sejnowski, “Learning and Relearning in Boltzmann Machines,” in D.E. Rumelhart, J.L. McClelland and the PDP Research Group eds., Parallel Distributed Processing, vol.1, pp.282-317, The MIT Press, 1986.
19) J.J. Hopfield, “Neural Networks and Physical Systems with Emergent Collective Computational Abilities,” Proc. of the National Academy of Sciences of the USA, vol.79, no.8 pp.2554-2558, 1982.
20) J.J. Hopfield and D.W. Tank, “ “Neural” Computation of Decisions in Optimization Problems,” Biol. Cybern., vol.52, no.3, pp.141-152, 1985.
21) 河原智一, 西山正志, 山口　修, “直交相互部分空間法を用いた顔認識,” 情処研報 CVIM-151, pp.17-24, 2005.
22) Tomokazu Kawahara, Masashi Nishiyama, Tatsuo Kozakaya, Oamu Yamaguchi, “Face Recognition based on Whitening Transformation of Distribution of Subspaces,” ACCV Workshop Subspace2007, pp.97-103, 2007.
23) T. Kohonen, “Self-Organizing Maps,” Springer-Verlag, 1995.
24) 黒沢由明, “部分空間法の今昔（いまむかし）（上）: 歴史と技術的俯瞰 : 誕生から競合学習との出会いまで,” 情報処理学会誌, vol.49, no.5, pp.566-572, 2008.
25) Peter A. Lachenbruch and M. Ray Mickey, “Estimation of Error Rates in Discriminant Analysis,” Technometrics, vol.10, no.1, pp.1-11, 1968.
26) 前田英作, 村瀬　洋, “カーネル非線形部分空間法によるパターン認識,” 信学論 (D-II), vol.J82-D-II, no.4, pp.600-612, 1999.
27) 前田賢一, 渡辺貞一, “局所的構造を導入したパターン・マッチング法,” 信学会論 (D), vol.J68-D, no.3, pp.345-352, 1985.
28) W.S. McCulloch and W. Pitts, “A Logical Calculus of the Ideas Immanent in Nervous Activity,” Bulletin of Mathematical Biophysics, vol.5, pp.115-133, 1943.
29) F. Rosenblatt, “The Perceptron : A Probabilistic Mode for Information Storage and Organization in the Brain,”Psychological Review, vol.65, no.6, pp.386-408, 1958.
30) D.E. Rumelhart, G.E. Hinton and R.J. Williams, “Learning Internal Representations by Error Propagation,” in D.E. Rumelhart, J.L. McClelland and the PDP Research Group eds., Parallel Distributed

Processing, vol.1, pp.318-362, The MIT Press, 1986.
31) D.E. Rumelhart, G.E. Hinton and R.J. Williams, “Learning Representations by Back-Propagating Errors,” Nature, vol.323, pp.533-536, 1986.
32) 坂野　鋭, 武川直樹, 中村太一, “核非線形相互部分空間法による物体認識,” 信学論 (D-II), vol.J84-D-II, no.8, pp.1549-1556, 2001.
33) R.E. Schapire, Y. Singer, “Improved Boosting Algorithms Using Confidencerated Predictions,” Machine Learning, pp.297-336, 1999.
34) B. Schölkopf, J.C. Platt, J. Shawe-Taylor, A.J. Smola and R.C. Williamson, “Estimating the Support of High-Dimensional Distribution,” Neural Computation, vol.13, no.7, pp.1443-1471, 2001.
35) B. Schölkopf, A.J. Smola and K.R. Müller, “Nonlinear principal component analysys as a kernel eigenvalue problem,” Neural Computation, vol.10, no.5, pp.1299-1319, 1998.
36) 津田 宏治, “ヒルベルト空間における部分空間法,” 電子信学論 (D-II), vol.J82-D-II, no.4, pp.592-599, 1999.
37) V.N. Vapnik, “The Nature of Statistical Learning Theory,” Springer, 1995.
38) P. Viola, Michael and J. Jones, “Robust real-time face detection,” IJCV Vol.57, no.2, pp.137-154, 2004.
39) Y. Washizawa, K. Hikida, T. Tanaka, Y. Yamashita, “Kernel Relative Principal Component Analysis for Pattern Recognition,” Proc. of Joint IAPR Iinternational Workshops on Syntactical and Structural Pattern Recognition and Statistical Pattern Recognition（SSPR/SPR 2004）, pp.1105-1113, Lisbon, 2004.
40) 八木康史, 斎藤英雄 編, “コンピュータビジョン最先端ガイド ＜ 1 ＞ Level Set,Graph Cut,Particle Filter,Tensor,AdaBoost（CVIM チュートリアルシリーズ）,” 第 5 章, 2008.