アンサンブル学習

上 田 修 功†

パターン識別器の性能は，設計に使用した学習データだけでなく，未知データに対する識別性能（汎化性能）で評価される．換言すれば，汎化性能の高い識別器の学習法の研究が実用上重要となる．その 1 つの有力なアプローチとしてアンサンブル学習がある．本論文では，アンサンブル学習に関する従来の代表的手法を，その基本的な考え方，具体的なアルゴリズムを含めサーベイを行う．さらに，最新の話題として，異種情報を効果的に融合する従来とは別のタイプのアンサンブル学習についても解説する．

Ensemble Learning

Naonori Ueda†

The classification performance of a classifier is evaluated in terms of minimizing the classification errors for unseen test data. In other words, it is crucial to develop methods for learning classifiers with high generalization ability for the test data. Ensemble learning is a promising approach for this purpose. The aim of this paper is to give a review on literature dealing with the ensemble learning. The basic ideas and their learning algorithms of the conventional representative ensemble learning methods are explained. Moreover, another type of ensemble methods that effectively combine heterogeneous information are also included as the recent results.

1. はじめに

パターン認識は，あらかじめ定められた有限個のクラス（カテゴリ）に入力データ（パターン）を識別（通常は排他的分類）する問題で，文字認識，音声認識，テキスト分類，医療診断などの分類問題，決定問題など多岐にわたる．パターン認識器は，通常，識別器（classifier）と呼ばれ，あらかじめクラスラベルが付与されたデータ（学習データ）を用いて設計される．識別器の性能は未学習データ（テストデータ）での識別性能，すなわち，汎化性能で評価される．換言すれば，汎化性能の高い識別器の構成法の研究がパターン認識における重要研究課題となっている．

一般に，線形識別器で作られる識別境界は，学習データの変動に対して安定であるが，自由度が低いため，複雑な識別境界が必要な問題には十分でない．一方，多層ニューラルネットワークのような非線形識別器は，複雑な識別境界を構成できるものの，自由度が高いため，学習データの変動に対する識別境界の変動が大きくなる．それゆえ，少数の学習データで学習した場合，それらに過度に適合したクラス境界が構成され，テストデータでの識別性能が劣化するという過学習の問題に悩まされる．このように識別器の性能向上には，直面している分類タスクと学習データ数に応じた識別器の自由度の適切な制御が必要となる．

一方，近年，単一の識別器の性能向上よりも，複数の識別器を何らかの形で融合させて汎化性能の向上を図るアンサンブル学習法が提案され，実用面での有効性が示されている．“集合的予測戦略”[25] や “集合機械”[9),17)] がその先駆的成果である．特に，弱識別器，すなわち，線形識別器のような単一では識別能力が低い識別器ですら，アンサンブルすることで，驚くべき汎化性能の高い識別器が構成できるというブースティング法[7)] が提案され注目を集めた．アンサンブル学習では，どのような識別器をどのように融合するかで様々な手法が考えられる．本論文では，この観点で従来の代表的なアンサンブル学習を体系的に整理し，かつ，それらの基本的な考え方，具体的なアルゴリズムについて解説する．アンサンブル学習は識別問題に限定されないが，本論文ではパターン認識への応用を意識し，識別問題のためのアンサンブル学習法に焦点をあてる．

† 日本電信電話株式会社 NTT コミュニケーション科学基礎研究所
NTT Communication Science Laboratories, Nippon Telegraph and Telephone Corporation

以下の本論文では，まず，2 章で従来のアンサンブル学習の形態を整理し，3 章で個々の形態での代表的な手法を説明する．4 章では，異種情報を融合して汎化性能の向上を図る別のタイプのアンサンブル学習法について説明する．5 章はまとめである．

2. アンサンブル学習の形態

識別器は，通常，識別関数☆1として表現される．そこで，まず，識別関数について説明し，次いで，既存のアンサンブル学習の形態の全体像を概観する．

2.1 識別関数

パターンは何らかの特徴抽出により変換された特徴 x として表現される☆2．C クラスの識別問題の場合，C 個の識別関数を用意し，入力 x に対し，$g_j(x) > g_i(x),\ \forall i \neq j$ のとき，x をクラス j と識別する．

ベイズ最適識別，すなわち，x に対するクラスの事後確率を最大化する識別の場合，識別関数は次式で計算されるクラス事後確率とすればよい．

$$g_j(x) = P(j|x) = \frac{P(j)p(x|j)}{\sum_{l=1}^{C} P(l)p(x|l)} \qquad (1)$$

ここに，$P(j)$ はクラス j が生起する事前確率，$p(x|j)$ はクラス j の確率モデル，すなわち，クラス j における x の確率分布を表す☆3．なお，2 クラス問題の場合，$g(x) = P(1|x) - P(2|x)$ として 1 つの識別関数 $g(x)$ の符号判定により識別を行うことができる．

クラス j の確率モデルとして，本来観測されない潜在変数 z を含む統計モデルも想定される．たとえば，時系列パターン認識で多用される隠れマルコフモデルでは，状態変数は観測されない．このような統計モデルの場合，クラス j の統計モデルも周辺化により $p(x|j;\theta_j) = \sum_z p(x,z|j;\theta_j)$ と表現できるため，以下の本論文での $p(x|j;\theta_j)$ は潜在変数モデルも含めたクラス j の一般的な統計モデルを意味するものとする．

2.2 アンサンブル学習の形態

既存の主要なアンサンブル学習は，おおよそ，**図 1** のように整理できる．各手法についての代表手法を図 1 に示す．

アンサンブル学習を大別すると，何らかの手段で複数の識別器を設計して，それらを何らかの手段で結合するアプローチ（図中では，“識別器の結合” と略記）と，パターンを異種特徴で表現し，個別にモデル化したものを結合するアプローチ（図中では，“特徴空間の結合” と略記）がある．後者は，広義のアンサンブル学習であり，具体的には，クラスラベルあり・なし混在データからの識別器の学習法，および，最大エントロピー法に基づく，異種特徴の最適結合法がある．

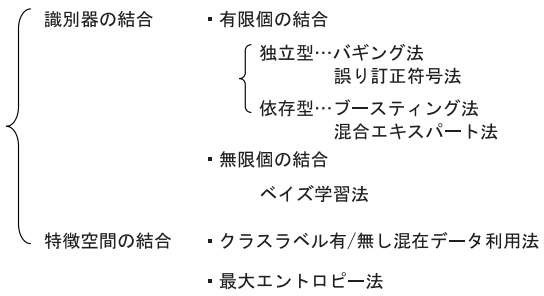


図 1 アンサンブル学習の分類
Fig. 1 Types of ensemble learning method.

“識別器の結合” を行う前者のアプローチの場合，どのような性質の識別器をどのように結合するかにより，図 1 に示すように，さらに，独立型と依存型に分類できる．前者では，複数の識別器が独立に学習され，個々の識別器は自由度の高い識別器が用いられる．バギング法，誤り訂正符号法がこの独立型に属する．後者では，複数の識別器が，“逐次” もしくは “同時” に学習され，個々の識別器は比較的単純な識別器☆4が用いられる．逐次学習タイプとしてブースティング法があり，同時学習タイプとして混合エキスパート法がある．

また，究極のアンサンブルとして，有限個ではなく，可能なすべての識別器の出力をその信頼性で重み付け平均した期待値を出力するアンサンブルアプローチがベイズ学習である．

3. アンサンブル学習法

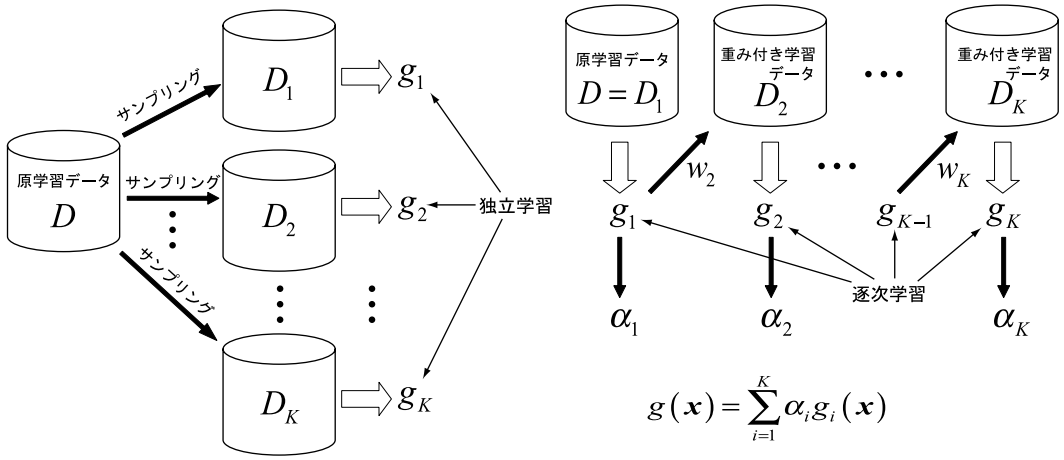
本章では，有限個の識別器の結合法である，バギング法，誤り訂正符合法，ブースティング法，混合エキスパート法の各々の手法を説明する．

3.1 バギング法

アンサンブル学習の最も初期の手法として，単純に複数の識別器を個別に学習し，新たな入力に対する識別は，それらの識別器の多数決，もしくは，識別関数値の単純平均値に基づく単純アンサンブル学習[9),17)]がある．

バギング法[1)]は，単純アンサンブル学習の考え方とまったく同じであるが，学習データを分割するのでは

☆1 判別関数とも呼ばれる．
☆2 通常は，ベクトルであるが，近年，流行りのカーネルアプローチでは，x は木やストリングといった多様な表現法が用いられる．
☆3 本論文では，確率関数を大文字の P，確率密度関数を小文字の p で表記することとする．
☆4 弱学習器，弱識別器とも呼ばれる．



（a）バギング法　　（b）ブースティング法

図 2 バギング法およびブースティング法．バギング法では，K 個の識別器が独立に学習されるのに対し，ブースティング法では逐次的に識別器が学習される

Fig. 2 Bagging and boosting methods. In bagging method, K classifiers are trained independently, while in boosting, K classifiers are successively trained.

なく，**図 2** (a) に示すように，学習データから復元抽出して K 組の学習データを作成し，ある識別器を K 通りの方法で独立に学習させ，最後にそれらを統合するという手法である☆．

統合は，各識別関数値の（重み付き）平均とする方法と，全識別器の多数決（majority voting）とする方法があり，通常の応用では後者が多用されている．重み付けの方法として誤り率最小化基準に基づく最適重み付け手法も提案されている[21]．

独立な識別器の識別結果の多数決により識別誤りが改善される直観的な理由を以下に説明する．今，K 個（K は奇数とする）の識別器の単純アンサンブル学習を考える．各識別器が統計的に独立とし，かつ，識別器の判定の誤り確率を一律 θ とすると，K 個の分類器のうち，k 個の分類器の判定が誤る確率 $P(k)$ は，簡単に 2 項分布：

$$P(k) = {}_K C_k \theta^k (1-\theta)^{K-k} \qquad (2)$$

となる．$K = 21$，$\theta = 0.3$ の場合の $P(k)$ を**図 3** に示す．このとき，アンサンブルによりクラス判定が誤る確率は，21 個のうち，過半数の 11 個が誤る確率ゆえ，図中の塗り潰した部分の和 $(P(11) + \cdots + P(21))$ となる．グラフからこの値は，単一の分類器の判定誤り確率 $\theta = 0.3$ に比べはるかに小さいことが分かる．これがアンサンブルの効果である．

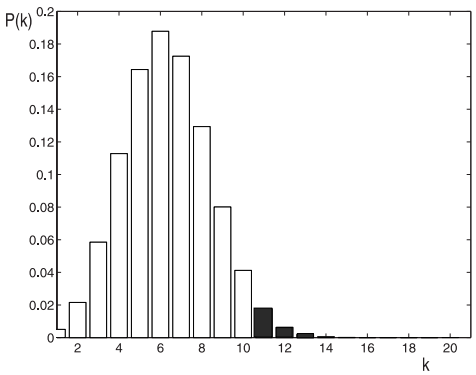


図 3 誤り確率

Fig. 3 Error probability.

回帰問題（関数近似問題）の場合，予測誤差（真値と予測器の自乗誤差）は，予測器の真値との差（偏り）の自乗と予測器の分散の和として表現される．単純アンサンブル学習の場合，K 個の予測器のアンサンブルにより，予測誤差は $1/K$ となる[17),19)] が，偏りはアンサンブルにより変化しない[20)]．一方，1 章で述べたように，予測器が複雑になるにつれて，予測の偏りは小さく，分散は大きくなる☆☆．したがって，線形予測器のような偏りの小さな予測器をアンサンブルしても予測誤差の改善は小さく，分散の大きな，非線形予測器のような複雑な予測器をアンサンブルした方がよいことになる．

☆ バギング（bagging）とは，“bootstrap aggregating” の略．

☆☆ “偏りと分散のジレンマ” と呼ばれる．

識別器の場合，予測誤差は，通常，0/1 損失（識別誤りのときのみ 1）で計られるため，自乗誤差に基づく上記回帰問題の議論はそのまま適用できない．Friedman は，分類問題に対し，境界誤差，境界バイアス，境界分散なる独自の定義を与え，識別問題でのアンサンブル学習の近似解析を行っている．詳細は文献 5) に譲るが，結論からいうと，単純アンサンブル識別器は，学習データ変動に対するクラス境界の変動を安定化させる効果があることが理論的に明らかにされている．それゆえ，回帰問題同様，バギング法は，ニューラルネットのような自由度の高い識別器のアンサンブルに効果的であり，線形識別関数をバギングしてもその効果は期待できない．

3.2 ブースティング法

ブースティング（boosting）とは，“高める，上げる” という意味で，ブースティング法は文字どおり，逐次的に識別器を学習させ，識別性能を向上させる手法である．特徴は，弱学習器をアンサンブルの対象とする点である．弱学習器とは，ランダムな識別よりはまし程度の低品質な識別器を指す．ブースティング法の著名なアルゴリズムが以下に示す AdaBoost アルゴリズム[7] である．

AdaBoost algorithm

Step 1.［初期化］ データ重みを $w_{1,1} = \cdots = w_{1,N} = \frac{1}{N}$ と一様分布に初期化する．$i \leftarrow 1$.

Step 2.［ブースティング］ 以下を実行.

2.1 $\{w_{i,n}\}_{n=1}^{N}$ を用いて，弱仮説器 $g_i(x)$ を構成☆.

2.2 重み付き識別誤り率を算出.

$$\mathrm{err}_i = \sum_{n=1}^{N} w_{i,n} I(y_n \neq g_i(x_n)) \tag{3}$$

ただし，$I(u)$ は u が真（偽）のとき，$I(u) = 1(0)$ を返す 2 値関数とする.

2.3 識別器重みを算出.

$$\alpha_i = \frac{1}{2} \log \frac{1 - \mathrm{err}_i}{\mathrm{err}_i} \tag{4}$$

$i = K$ なら Step 3 へ.

2.4 $n = 1, \ldots, N$ に対し，データ重みを更新.

$$w_{i+1,n} \leftarrow \frac{w_{i,n} \exp\left\{ -\alpha_i y_n g_i(x_n) \right\}}{\sum_{n=1}^{N} w_{i,n} \exp\{-\alpha_i y_n g_i(x_n)\}} \tag{5}$$

$i \leftarrow i + 1$ とし，Step 2.1 へ.

Step 3.［出力］ 得られた K 個の弱仮説を重み付き平均したアンサンブル識別器：

$$g(x) = \sum_{i=1}^{K} \alpha_i g_i(x) \tag{6}$$

を生成．そして，新たな入力 x に対し，$g(x)$ の値の符号（正/負）により $+1$，-1 のクラス識別を行う.

上記アルゴリズム中の 2.1 は，重み $w_{i,n}$ を第 n サンプルの生起確率と見なして，学習データから N 回の復元抽出により N サンプルを抽出し，その N サンプルで弱仮説 $g_i(x)$ を設計することを意味する．AdaBoost アルゴリズムは 2 クラス問題（便宜上 2 クラスを $+1$ クラス，-1 クラスとする）のための手法である．この場合，特徴 x の帰属クラスは 2 値 $y \in \{+1, 1\}$ とする．また，$g_i(x)$ の出力は $+1/-1$ であり，実際には，線形識別関数などの単純な識別器を用いてクラス判定を行い $+1/-1$ の値を出力する．以下では特に混乱のない範囲で，$g_i(x)$ を識別関数と同一視するが，あくまで出力は 2 値 $\{+1, -1\}$ であることに注意.

図 2 (b) にブースティング法の学習の流れを示す．ブースティング法のもう 1 つの特徴は，学習データの各々に重みを導入している（サンプル重み）点である．最初は一様重みが付与され，識別を誤ったサンプルの重みを指数的に増やし（式 (5)），重みの大きなサンプルを優先的に学習される．また，識別関数に対する重み α_i も同時に計算される．両重みを算出しながら，図 2 (b) に示すように，逐次的に識別関数を学習していく．すなわち，g_1〜g_K が同一学習データで設計されるのではなく，$g_i(x)$ に対する学習データは $g_{i-1}(x)$ で分類誤りを起したデータができるだけ強調されるように個々の学習データに重みを与え，分類が容易なデータから適応的に識別していく学習戦略をとる☆☆．クラス未知のデータ x^* に対しては，式 (6) の識別関数を用いて，$g(x^*) = \sum_{i=1}^{K} \alpha_i g_i(x^*)$ の値の正負判定で識別される.

AdaBoost アルゴリズム中の式 (4)，(5) の関係，さ

☆ g_i は $+1$，-1 の 2 値の出力関数とするという意味で仮説器という表現をしている.

☆☆ AdaBoost の “Ada” は adaptive（適応的）の意味.

らに，AdaBoost により識別性能改善が実現できる理由を以下に説明する．

まず，天下り的ではあるが，識別関数 $g(x)$ の損失を次式の指数損失関数：

$$L(g) = \frac{1}{N}\sum_{n=1}^{N}\exp\{-y_n g(x_n)\} \tag{7}$$

で評価することとする．y_n が $g(x_n)$ により正しく識別されるには，y_n と $g(x_n)$ が同符号で，かつ，その判定の信頼度は $g(x_n)$ の絶対値が大きいほど大とするのが自然である．指数損失関数では，$-y_n g(x_n)$ の値が大きいほど，つまり，y_n が正しく識別され，かつ，その判定の信頼度が高いほど，損失を零に漸近させるようにしている．

さて，$g(x) = \sum_{i=1}^{k}\alpha_i g_i(x)$ とし，$g(x)$ に識別器 $g_{k+1}(x)$ を重み $\alpha_{k+1}(>0)$ で付加した $g(x) + \alpha_{k+1}g_{k+1}(x)$ の損失 $L(g(x) + \alpha_{k+1}g_{k+1}(x))$ が $L(g(x))$ より小さくなる条件を考察する．便宜上，$L_{k+1} = L(g(x) + \alpha_{k+1}g_{k+1}(x))$，$L_k(g(x)) = L_k$ と書くことする．$g_{k+1}(x) \in \{+1, -1\}$ に注意すると，簡単な計算により

$$\begin{aligned}
&L_{k+1}\\
&= \frac{1}{N}\sum_{n=1}^{N}\exp\{-y_n g(x_n)\}\exp\{-y_n\alpha_{k+1}g_{k+1}(x_n)\}\\
&= \frac{1}{N}\sum_{n:g_{k+1}(x_n)=y_n}\exp\{-\alpha_{k+1}\}\sum_{n=1}^{N}\exp\{-y_n g(x_n)\}\\
&\quad + \frac{1}{N}\sum_{n:g_{k+1}(x_n)=\neq y_n}\exp\{\alpha_{k+1}\}\sum_{n=1}^{N}\exp\{-y_n g(x_n)\}\\
&= \frac{1}{N}\exp\{-\alpha_{k+1}\}\sum_{n=1}^{N}\exp\{-y_n g(x_n)\}\\
&\quad + \frac{1}{N}(\exp\{\alpha_{k+1}\} - \exp\{-\alpha_{k+1}\})\\
&\quad \times \sum_{n:g_{k+1}(x_n)\neq y_n}\exp\{-y_n g(x_n)\}\\
&= \frac{Z_k}{N}\Big(\exp\{-\alpha_{k+1}\} + \big(\exp\{\alpha_{k+1}\} - \exp\{-\alpha_{k+1}\}\big)\\
&\quad \times \sum_{n:g_{k+1}(x_n)\neq y_n}\exp\{-y_n g(x_n)\}/Z_k\Big)\\
&= \frac{Z_k}{N}\exp\{-\alpha_{k+1}\}\\
&\quad + \frac{Z_k}{N}\big(\exp\{\alpha_{k+1}\} - \exp\{-\alpha_{k+1}\}\big)\mathrm{err}_{k+1}
\end{aligned} \tag{8}$$

を得る．ここで $Z_k = \sum_{n=1}^{N}\exp\{-y_n g(x_n)\}$ とする．err_{k+1} は仮説 g_{k+1} による分布 $\{w_{k+1,1}, \ldots, w_{k+1,N}\}$ から確率的に生成されるデータに対する誤り率を表し，式 (3) で $i = k+1$ とした場合に相当する．ただし，

$$w_{k+1,n} = \exp\{-y_n g(x_n)\}/Z_k \tag{9}$$

以上より，L_{k+1} を小さくするにはできるだけ err_{k+1} が小さくなるように，すなわち，仮説 g_{k+1} が正しくなるように識別関数を設計すればよいことが分かる．そして，重み α_{k+1} については，式 (8) を α_{k+1} に関して最小化すべく，$\partial L_{k+1}/\partial\alpha_{k+1} = 0$ より，

$$\alpha_{k+1} = \frac{1}{2}\log\frac{1-\mathrm{err}_{k+1}}{\mathrm{err}_{k+1}} \tag{10}$$

を得る．また，式 (9) より式 (5) が成立することも容易に確認できる．以上から，AdaBoost アルゴリズムにおける式 (3)，(4)，(5) の妥当性が示された．

また，式 (10) を式 (8) に代入すると，

$$L_{k+1} = 2L_k\{\mathrm{err}_{k+1}(1-\mathrm{err}_{k+1})\}^{1/2} < L_k \tag{11}$$

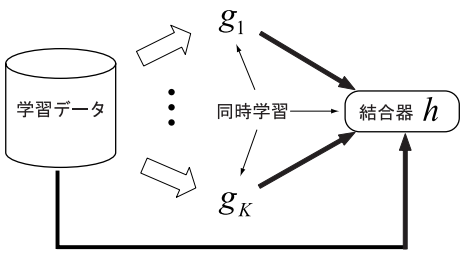
となり，新たな仮説の追加により損失を減少させることが可能となることが分かる．

以上の結果から，AdaBoost アルゴリズムは，式 (7) を損失関数とし，この損失を順次減少させる逐次型アンサンブル学習といえる．さらに，$\gamma_n = y_n g(x_n)$ とすると，アンサンブル識別関数 $g(x_n)$ が信頼度高く識別するほど，γ_n が大きくなることから，γ_n は識別の余裕度，すなわち，マージンと見なせる[6]．つまり，AdaBoost アルゴリズムは，0/1 損失を $\exp\{-\gamma_n\}$ なるマージンに基づく損失関数を最小化，等価的に，マージンを最大化するアルゴリズムと解釈できる．2 クラス問題を対象とし，かつ，マージン最大化を実現するという観点で，Boosting 法はサポートベクトルマシンの考え方と類似している．

AdaBoost アルゴリズムは，弱学習器の逐次結合で学習データの誤り率を単調に減少可能だが，汎化誤差も単調に減少するという保証はなく，過学習が生じることもある．K の最適な設定法は現在のところ提案されていないが，所詮は弱仮説器の結合ゆえ，K を大きくしすぎても過学習が深刻になることはないと報告されている．

3.3 混合エキスパート法

バギング法やブースティング法では，基本的には複数の識別関数の線形和として表現されるが，混合エキスパート法では，データ空間を複数のエキスパートが領域分担して学習し，最終的に 1 つの識別器が構成される[10]．すなわち，**図 4** に示すように，K 個の識別器 $g_1(x), \ldots, g_K(x)$ が同一の学習データで同時かつ互いに依存しながら学習する．依存関係は，結合器（ゲー



$$g(\boldsymbol{x}) = \sum_{i=1}^{K} h(\boldsymbol{x}) g_i(\boldsymbol{x})$$

図 4 混合エキスパート法
Fig. 4 Mixture of experts.

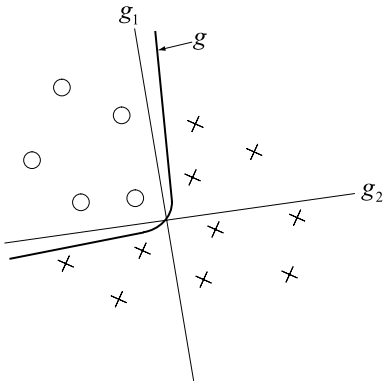


図 5 2 クラス問題に対する，2 個の線形識別関数を用いた混合エキスパート法による識別結果
Fig. 5 A recut by mixture of experts, each of which is a linear discriminant function for two class problem.

ト関数）$h(x)$ により決定される．統合識別関数は，

$$g(x) = \sum_{i=1}^{K} h(x) g_i(x) \qquad (12)$$

となる．Boosting 法では，統合重みは入力 x と独立であったのに対し，ここでは依存して統合される．

たとえば，**図 5** の 2 分類問題の場合，単一の線形識別関数 (g_1, g_2) では満足なクラス境界の構成が期待できないが，2 つの線形識別関数 g_1，g_2 を特徴空間上で，相補的に合成することにより，図中の非線形識別関数 g が構成できる．これが混合エキスパート法である．識別器と結合器は，パラメータを持つ確率モデルによりモデル化され，モデルパラメータは最尤推定により求められる．

混合エキスパート法も，自分の得意な領域の識別を優先するという点では，先の Boosting の考え方と通じるものがあるが，大きな相違点は，混合エキスパート法では，統合重みを入力 x に依存させることにより，同時に複数の識別関数を学習させる点にある．

Class	Code Word f_1 f_2 f_3
C_1	1 0 0
C_2	0 1 0
C_3	0 0 1

(a)

Class	Code Word f_1 f_2 f_3 f_4 f_5 f_6 f_7
C_1	1 0 0 1 1 0 1
C_2	0 1 0 1 0 1 1
C_3	0 0 1 0 1 1 0

(b)

図 6 3 クラスの場合の符号語の例．(a) 単一符号化 (b) 冗長符号化（誤り訂正符号法）
Fig. 6 An example of code words for three class problem. (a) Single coding, (b) redundant coding (error-correcting coding).

単純アンサンブルでは，せっかく単一の優れた識別器があっても☆アンサンブルによりその識別器の性能が鈍化されることが考えられる．一方，Boosting や混合エキスパート法では，相補的なアンサンブルゆえ，その問題が緩和されるといえる．

3.4 誤り訂正符合法

この手法は，一種の誤り訂正の考え方を取り入れた方法である[4]．

今，簡単のため 3 クラス問題を考える．識別関数を f_1，f_2，f_3 とする．誤り訂正符合法では 3 クラス問題を 2 分類問題に分解する．最も自然には，**図 6** (a) に示すように，クラス 1，2，3 に対し，各々 $C_1 = (1\,0\,0)$，$C_2 = (0\,1\,0)$，$C_3 = (0\,0\,1)$ とし（これらを符号語と見なす），f_k をクラス k に属するか否かの 2 分類器とすればよい．しかし，これでは，たとえば f_2 が分類誤りを起こして，クラス 1 に属するサンプルを $C_1 = (1\,0\,0)$ となるべきところを $C' = (1\,1\,0)$ としてしまうと，C' と C_1，C_2，C_3 とのハミング距離がすべて 1 ゆえ，もはや正しく復元できない．

そこで，図 6 (b) のように符号語に冗長ビットを付加し，それにともない 2 分類器 f_4〜f_7 を追加する☆☆．このとき，たとえば，f_4 はクラス 1 またはクラス 2 に属するか否かの 2 分類器に相当する．

☆ もちろん，汎化誤差の意味で，個別のどの識別関数が best かが分からないので，1 つに絞らず，重み付けしてアンサンブルしている．

☆☆ この例では 4 ビットの冗長性を持たせているが特にそうである必要はない．

誤り訂正符合法では，クラスの教師信号を，誤り訂正符号の考え方に従い，クラス間でハミング距離が大きくなるような符号語とする．いったん f_1〜f_7 が学習できれば，未知入力 x に対し $f_i(x),\ i = 1, \ldots, 7$ の値に基づいて 7 ビットの符号語が求まることになる．そしてこの符号語とハミング距離が最も近い符号語（図 6 (b) 各行）を選択し C_1，C_2，C_3 を決定する．

上記のように冗長ビットを付加することにより，たとえば，クラス f_2，f_4 に誤りが生じて，クラス 1 に属するサンプルを $C_1 = (1\,0\,0\,1\,1\,0\,1)$ となるべきところを $C' = (1\,1\,0\,0\,1\,0\,1)$ となったとしても，C' は C_1 に最もハミング距離が小さいので，クラス 1 と正しく復元可能となる．

誤り訂正符合法の基本的な考え方は，クラス間でのマージン（ここではハミング距離）を最大化する手法と見なせる．f_i として，サポートベクトルマシン（SVM）が用いられる．通常，SVM で多クラス問題を解く場合，図 6 (a) の one-against-all 法が採られているが，誤り訂正符合法は，さらに図 6 (b) のような拡張により性能向上を図る手法と位置づけることができる．その意味で，誤り訂正符合法は SVM で多クラス問題を解く際の手法と見なすこともできる．

3.5 ベイズ学習

バギング法では，個々の識別器の汎化性能はまったく考慮されず，すべて対等にアンサンブルされる．より正確には何らかの形で個々の識別関数に重みを付けてアンサンブルすることが望ましいと考えられる．Boosting 法や混合エキスパート法はその一方法を提供しているが，ベイズ学習は，ベイズ統計の枠組みでより一般的な重み付けの方法論を提供する．

統計モデルに基づく識別関数の場合，2 章で述べたように，第 j クラスの識別関数は，第 j クラスのクラス事後に相当する．そして，ベイズの定理より，

$$g_j(x;\theta) = P(j|x;\theta) = \frac{P(j)p(x|j;\theta_j)}{\sum_j P(j)p(x|j;\theta_j)} \quad (13)$$

として計算される．ここに θ_j はクラス j の統計モデル $p(x|j;\theta_j)$ のモデルパラメータである．たとえば，正規分布の場合，θ_j は平均ベクトル μ_j，共分散行列 Σ_j に相当する．また，$P(j)$ はクラス j の事前確率を表す．

最尤推定の場合，各クラスの学習データを用いて，尤度基準に従って

$$\theta_j^* = \arg\max_{\theta_j} \mathcal{L}(D_j;\theta_j),\ j = 1, \ldots, C \quad (14)$$

として推定される．C は総クラス数，$D_j = \{(x, y = j)\}$ は第 j クラスの学習データを表す．また，対数尤度関数は

$$\mathcal{L}(D_j;\theta_j) = \sum_{x \in D_j} \log p(x|j;\theta_j) \quad (15)$$

として計算される．式 (14) の最尤推定値は，一般に，学習データ数が少ない場合，推定値の信頼性が低いという問題があり，そのような場合，式 (13) の識別関数の汎化性能にも問題が生じる．ベイズ学習は，統計モデルベースの識別関数に対し，少数データでの最尤推定値の問題を緩和する有力な手法である．なお，いうまでもなく，ベイズ学習はアンサンブル学習として提案されたものではなく，ベイズ学習もアンサンブル学習と見なせるという観点で取り上げて説明していることに注意．

ベイズ学習ではパラメータ θ_j を確率変数として取り扱い，その事前分布 $p(\theta_j)$ も考える．そして新たな入力 x^* に対する，クラス事後確率は，$P(y|x, \theta_j)$ を，観測データ D_j を得た下での事後分布 $p(\theta_j|D_j)$ で重み付き平均した

$$P(y|x^*, D_j) = \int P(y|x^*, \theta_j)p(\theta_j|D_j)d\theta_j \quad (16)$$

として求める．注意すべきは，式 (16) の左辺は，式 (13) の最尤推定の場合と異なり，θ_j が θ_j の事後分布を用いて積分消去され，もはや θ_j の関数ではないという点である．

そして，パラメータの事後分布は，ベイズの定理から観測データ D_j に基づいて

$$p(\theta_j|D_j) = \frac{p(D_j|\theta_j)p(\theta_j)}{\int p(D_j|\theta_j)p(\theta_j)d\theta_j} \quad (17)$$

として求められる．事後分布は観測データを得た下でのモデルの信頼度と見なせ，信頼度の高い識別器が優先的にアンサンブルされる．

しかし，上記はあくまで形式的な説明で，実用上，式 (17) の計算は解析的に求まらないことが多く，マルコフ連鎖モンテカルロ法などのサンプリング手法の援用が必要となる．最近では，決定論的な近似手法でベイズ学習を行う変分ベイズ法も提案されている．ベイズ学習全般，マルコフ連鎖モンテカルロ法と変分ベイズ法との相違点，さらには，混合エキスパートモデルに対する変分ベイズ学習法に関する詳細については文献 22)〜24) を各々参照されたい．

4. 特徴空間の結合

以上は，識別器の統合によるアンサンブル学習であったが，本章では，特徴空間を統合する広義のアンサンブル学習について説明する．

4.1 ラベルあり・なしデータの統合

通常，分類器の設計は，クラスラベルが付与された教師あり学習となるが，クラスラベルは人手で行われるためコストがかかる．一方，ラベルなしデータは安価で容易に入手できる場合が多い．特に，テキスト分類では，クラスラベルなしテキストは web 上で大量に入手できる．こうした背景から，クラスラベルあり/なし混在データから分類器を学習する手法が提案され，その有効性が示されている[12),14),16),19)]．これらは静的なデータを対象としているが，最近，時系列データを対象とした研究もなされている[11)]．

識別関数が統計モデルの場合，式 (13) に示したように，クラス j の識別関数は，ラベルありデータ D_j を用いて式 (14) により推定できるが，クラスラベルあり/なし混在データの場合，少し工夫が必要となる．具体的には，不完全データからの最尤推定の数値解法である EM アルゴリズム[3)] を用いて推定する．以下では，その推定法の概略を説明しておく．詳細は文献 11)，16) を参照されたい☆

クラスラベルありデータを D^l，クラスラベルなしデータを D^u とするとき，データの独立同分布性を仮定すると，$D = (D^l, D^u)$ に対する尤度関数は $p(D;\theta) = p(D^l;\theta) \cdot p(D^u;\theta)$ と書ける．さらに，D^u に対しては，各ラベルなしデータの各々に対する潜在変数（ここではクラスインデックス）の集合 Z を用いて，$p(D^u;\theta) = \sum_Z p(D^u, Z;\theta)$ と書ける．すなわち，対数尤度 $\log p(D;\theta)$ はラベルあり／なしデータの対数尤度の直和となるので，それにともない EM アルゴリズムの Q 関数が直和表現でき[16)]，E ステップで，ラベルなしデータの潜在変数の事後分布，すなわち，帰属クラス事後分布が確率的に推定され，M ステップで，その事後分布に基づいてパラメータ更新が行われる．

より単純に，ラベルありデータのみで各クラスの分布を推定し，その後で，ラベルなしデータのクラスを識別して，ラベルありデータに加えて再学習するというナイーブな方法も考えられるが，学習の収束性が保証されず，また，EM アルゴリズムによる上記方法に比べ識別性能も低下することを補足しておく．

4.2 最大エントロピー原理による異種情報の統合

最大エントロピー原理は，確率分布を推定する一手法で，幅広い分野で用いられている．近年，自然言語処理における N グラムモデルの学習やテキスト分類などにも応用されている[2),18)]．また，最近，最大エントロピー原理を用いた異種情報の統合手法も提案されている[8)]．ここでの統合とは，異種情報の各々に適切なモデルを仮定し，それらを 1 つのモデルとして統合するための学習を意味し，確率モデルのアンサンブル学習と見なせ，これも広義のアンサンブル学習といえるので，以下ではこの異種情報の統合法について説明する．

観測データ $D = \{d\}$ において，$d = (x, y)$ の k 次元の特徴 x が $x = (x^1, x^2)$ なる 2 つの部分空間の直和として表現されているものとする☆☆．もちろん，単純に $x = (x^1, x^2)$ と単純に合わせた特徴量として識別器を構成することも可能だが，個別にモデル化した後，それらを最適に統合する方法も考えられる．通常，x^1 と x^2 が y が与えられた下で独立（条件付き独立）と仮定して，

$$p(x|y;\Theta) = p(x^1|y;\theta_y^1)p(x^2|y;\theta_y^2) \tag{18}$$

と分解する方法が採られる．つまり，クラス y が決まれば，x^1，x^2 はモデル $p(x^1|y;\theta_y^1)$，$p(x^2|y;\theta_y^2)$，から各々独立に生成されるというモデリングである．各モデルが独立ゆえ，θ_y^i は最尤推定，すなわち，対数尤度関数：$\sum_{x^i \in D_y} \log p(x^i|y;\theta_y^i)$ を最大化する $\hat{\theta}_y^i$ として求める．D_y は D の要素のうち，クラス y に属す観測データの集合を表す．

クラス事前確率を一様分布とすると，クラス y の識別関数は式 (18)，および，ベイズの定理から

$$\begin{aligned} P(y|x) &\propto p(x|y;\hat{\theta}_y) \\ &= p(x^1|y;\hat{\theta}_y^1)p(x^2|y;\hat{\theta}_y^2) \end{aligned} \tag{19}$$

となる．式 (19) では，x^1 と x^2 のモデルが対等に扱われており，一見，これで問題なさそうであるが，単純積として統合すると，特徴量の違いからスケールに偏りがある場合，たとえば，$p(x^1|y;\hat{\theta}_y^1)$ が $p(x^2|y;\hat{\theta}_y^2)$ に比べてつねに値が大きい場合，x^2 の特徴は識別に有効利用されないという問題が生じる．そこで，実用上，式 (19) の右辺の対数をとり $\lambda \log p(x^1|y;\hat{\theta}_y^1) + (1-\lambda)\log p(x^2|y;\hat{\theta}_y^2)$ と書き換え，交差検定法（cross-validation）により λ（> 0）を求める方法も提案されている[16)]．しかし，このアプローチでは，本来最適化すべき目的関数が不明瞭である．一方，以下で説明するように，最大エントロピー原理に従えば，エントロピー最大化という観点で最適な統合が実現できる．

☆ 文献 11) では，クラスラベルとは別に統計モデル自身が潜在変数を持つ，より一般的なケースに対する定式化となっている．

☆☆ 本議論は，3 つ以上の直和にも容易に拡張できるが，説明を簡単にするために 2 つの部分空間とする．

x^1，x^2 のモデルの対数 $\log p(x^i|y;\hat{\theta}_y^1)$，$i=1,2$ に対し，経験分布による期待値：

$$\frac{1}{|D_y|}\sum_{x^i\in D_y}\log p(x^i|y;\hat{\theta}_y^i),\quad i=1,2 \tag{20}$$

と，クラス事後確率 $P(y|x)$ と x の分布 $p(x)$ による期待値：

$$\sum_{x\in D}p(x)\sum_{y=1}^{C}P(y|x)\log p(x^i|y;\hat{\theta}_y^i),\ i=1,2 \tag{21}$$

が等しいという制約の下で，クラス事後確率のエントロピー：

$$-\sum_{x\in D}p(x)\sum_{i=1}^{2}P(y|x)\log p(y|x) \tag{22}$$

を最大とする $P(y|x)$ を求める枠組みが最大エントロピー原理である．すなわち，分布は制約条件を満たす範囲でできるだけ一様とする原理である．なお，表記 $|A|$ は集合 A の要素数を表す．

x の分布 $p(x)$ は未知ゆえ，経験分布，つまり，学習データで近似して，$p(x)\simeq\sum_{x'\in D}\delta(x-x')/|D|$ とし，ラグランジュ乗数法を用いて上記最適化問題を解くと，次式の分布を得る．

$$P(y|x;\Lambda)=\frac{\exp\{\sum_i\lambda_i\log p(x^i|y;\hat{\theta}_y^i)\}}{\sum_y\exp\{\sum_i\lambda_i\log p(x^i|y;\hat{\theta}_y^i)\}} \tag{23}$$

ただし，$\Lambda=(\lambda_1,\lambda_2)$．したがって，学習データを用いて，対数尤度関数：$\sum_{(x,y)\in D}\log P(y|x;\Lambda)$ を Λ に関して最大化することにより $\{\lambda_i\}$ の最尤推定値を求めれば，事後確率最大化の観点で，x^1 と x^2 のモデルの最適統合係数 λ_i が求まる．具体的な推定アルゴリズム，および，テキスト分類での実験などに関する詳細は文献 8) を参照されたい．

5. おわりに

本論文では，パターン識別における汎化性能向上の有力なアプローチであるアンサンブル学習について解説した．一般に多数決というのは，大多数の意見を尊重するという “決め方” であり，その結果が正しいかどうかは不明である．換言すれば，多数決は正解が不明な場合の妥当な決め方といえる．パターン識別の場合も，データの分布が完全に既知でない限り，手持ちの学習データで設計した識別器に対して，未来のテストデータに関する最適性を論じることは困難である．したがって，識別器の設計においても，複数の識別器を組み合わせたアンサンブル学習は人間社会に学ぶ妥当かつ自然なアプローチといえる．また，4 章で紹介した異種情報の統合の研究は未成熟であるが，今日のようなマルチメディア時代には不可欠な技術といえる．今後の進展が期待される．

参考文献

1) Breiman, L.: Bagging predictors, *Machine Learning*, Vol.24, No.2, pp.123–140 (1996).
2) Della Pietra, S., Della Pietra, V. and Lafferty, J.: Inducing features of random fields, *IEEE Trans. Pattern Anal. Machine Intell.* (*PAMI*), Vol.19, No.4, pp.380–393 (1997).
3) Dempster, A.P., Laird, N.M. and Rubin, D.B.: Maximum-likelihood from incomplete data via the EM algorithm, *Journal of the Royal Statistics. Society B* (methodological), Vol.39, pp.1–38 (1977).
4) Dietterich, T.G. and Bariri, G.: Solving multiclass learning problems via error-correcting outputs codes, *Journal of Artificial Intelligence Research*, Vol.2, pp.263–286 (1995).
5) Friedman, J.: On bias, variance, 0/1-loss, and the curse of dimensionality, *Data Mining and Knowledge Discovery*, Vol.1, No.1, pp.55–77 (1997).
6) Friedman, J., Haste, T. and Tibshirani, R.: Additive logistic regression: statistical view of boosting, *The Annal. of Statistics*, Vol.38, No.2, pp.337–374 (2000).
7) Freund, Y. and Schapire, R.: Decision-theoretic generalization of on-line learning and an application to boosting, *Journal of Computer and System Sciences*, Vol.55, pp.119–139 (1997).
8) 藤野昭典，上田修功，斎藤和巳：文書の構成要素モデルのアンサンブル学習に基づくテキスト分類，信学技報，Vol.104, No.349, NC2004-80, pp.69–74 (2004).
9) Hansen, L.K. and Salamon, P.: Neural network ensembles, *IEEE Trans. Pattern Anal. Machince Intell* (*PAMI*)., Vol.12, No.10, pp.993–1001 (1990).
10) Jacobs, R.A., Jordan, M.I., Nowlan, S.J. and Hinton, G.E.: Adaptive mixtures of local experts, *Neural Computation*, Vol.3, pp.79–87 (1991).
11) Inoue, M. and Ueda, N.: Exploitation of unlabeled sequences in hidden markov models, *IEEE Trans. Pattern Anal. Machine Intell.* (*PAMI*), Vol.25, No.12, pp.1570–1581 (2003).
12) Joachims, T.: Transductive Inference for Text

Classification using Support Vector Macines, *International Conference on Machine Learning* (*ICML*), pp.200–209 (1999).
13) Krogh, A. and Vedelsby, J.: Neural networks ensembles, cross validation, and active learning, *Advances in Neural Information Processing Systems 7* (*NIPS7*), MIT Press, Cambridge, MA (1995).
14) Miller, D.J. and Uyar, H.S.: A mixture of experts classifier with learning based on both labelled and unlabeled data, *Advances in Neural Information Processing Systems 9*, pp.571–577 (1997).
15) Raina, R., Shen, Y., Ng, A.Y. and McCallum, A.: Classification with hybrid generative/discriminative models, *Advances in Neural Information Processing Systems* (*NIPS*) *17* (2003).
16) Nigam, K., Mccallum, A.K., Thrun, S. and Mitchell, T.: Text classification from labeled and unlabeled documents using EM, *Machine Learning*, Vol.39, pp.103–134 (2000).
17) Perrone, M.P.: Improving regression estimates: Averaging methods for variance reduction with extensions to general convex measure optimization, Ph.D. Thesis, Brown University (1993).
18) Rosenfeld, R.: Adaptive statistical language modeling: A maximum entropy approach, Ph.D. thesis, Carnegie Mellon Univ. (1994).
19) Shahshahani, B. and Landgrebe, D.: The effect of unlabeled samples in reducing the small sample size problem and mitigating the Hughes phenomenon, *IEEE Trans. Geoscience and Remoto Sensing*, Vol.32, No.5, pp.1087–1095 (1994).
20) 上田修功，中野良平：アンサンブル学習の汎化誤差解析，電子情報通信学会論文誌（D-II），Vol.J80-D-II, No.9, pp.2512–2521 (1997).
21) Ueda, N.: Optimal linear combination of neural networks for improving classification performance, *IEEE Trans. Patten Anal. Machine Intell.* (*PAMI*), Vol.22, No.2, pp.207–214 (2000).
22) 上田修功：ベイズ学習［I］–［IV］，電子情報通信学会誌，Vol.85, No.4,6,7,8 (2002).
23) Ueda, N. and Gharamani, Z.: Bayesian model search for mixture models based on optimizing variational bounds, *Neural Networks*, Vol.15, pp.1223–1241 (2002).
24) 上田修功：ベイズ学習のアルゴリズム—高次元積分の近似手法，人工知能学会誌，特集（統計モデルと学習の数理），Vol.19, No.6, pp.656–663 (2004).
25) Vovk, V.G.: Aggregating Strategies, Proc. Computational Learning Theory, pp.371–386 (1990).

(平成 17 年 1 月 21 日受付)
(平成 17 年 7 月 18 日採録)

(担当編集委員　和田 俊和)

上田　修功（正会員）

昭和 33 年生．昭和 57 年大阪大学工学部通信工学科卒業．昭和 59 年同大学院修士課程修了．同年 NTT 電気通信研究所入所．平成 5 年より 1 年間米国 Purdue 大学客員研究員．現在，NTT コミュニケーション科学基礎研究所知能情報研究部長（創発学習研究グループリーダ兼務）．奈良先端科学技術大学院大学客員教授．統計的学習，パターン認識の研究に従事．博士（工学）．共著『よくわかるパターン認識』（オーム社），共著『統計科学のフロンティア，第 11 巻：新しい確率計算』（岩波書店）．平成 4 年日本神経回路学会研究奨励賞，平成 9 年電気通信普及財団賞（テレコムシステム技術賞），平成 12，16 年電子情報通信学会論文賞受賞，平成 15 年日本神経回路学会研究賞，平成 16 年船井ベストペーパー賞等受賞．電子情報通信学会，日本神経回路学会，IEEE 各会員．