2005－CVIM－149（36）
2005／5／13

画像認識におけるカーネル学習法

西田 健次[1], 栗田 多喜夫
産業技術総合研究所 脳神経情報研究部門

概要　カーネル学習法は、非線形識別関数を効率よく構成する手法であり、カーネルトリックとも呼ばれている。サポートベクターマシンは、現在知られている多くのパターン認識手法の中でも認識性能の優れた手法であると考えられているが、カーネルトリックによって非線形識別関数を構成できるようになったことが、その性能向上に大きく貢献している。カーネル学習法とサポートベクターマシンに代表される線形識別手法を組み合わせることにより高性能な識別器を構成する事が可能になったが、未学習データに対する認識性能（汎化性能）を更に向上するためには変数選択などの手法が重要な役割を果たす。本稿では、サポートベクターマシンを中心にカーネル学習法について概説し、汎化性能向上のための変数選択手法などを紹介する。さらに、画像認識への応用例も紹介する。

Kernel Methods in Image Understanding

Kenji Nishida and Takio Kurita
Neurosceince Research Institute,
National Institute of Advanced Industrial Science and Technology

Abstract
Kernel method, which is also called Kernel Trick, is known to be one of the best scheme to extend linear classfier systems to nonlinear classifier systems. Support vector machine (SVM) is recognized as one of the best models for two class classification among the many methods, since its performace is drastically improved by kernel trick. Although we can build a high performance classifier system with combination of kernel method and linear classification method such as SVM, feature selection is still important to obtain high performance for unlearned data. This paper reviews kernel methods centering on the SVM and introduces some feature selection methods. Some examples of applications for image understanding are also introduced.

1　はじめに

特徴ベクトルの線形分離可能性に基づく線形識別手法は、パターン認識の基本的手法として広く使われている。一方、実際の認識問題には、本質的に非線形性を持ったものも多く、線形識別手法のみでは十分な性能を得る事は困難になっている。カーネル学習法は、特徴ベクトルを非線形変換して、その空間で線形の識別を行うことで、問題の非線形性に対応しつつ従来の線形識別手法の利点を生かすもので、別名「カーネルトリック」とも呼ばれている。

サポートベクターマシン (Support Vector Machine, SVM)[1, 2, 3] は、現在知られている様々なパターン認識手法の中でも最も認識性能の優れた学習モデルの一つと考えられているが、カーネル学習法と組み合わせる事により非線形の識別関数が構成できる。しかも「マージン最大化」という基準を用いることで、未学習サンプルに対しても高い認識性能（汎化性能）を得ることができる。また、現在では、サポートベクター

[1]kenji.nishida@aist.go.jp

マシンのためのソフトウェアツールも手軽に利用できる。そのため、訓練用のデータを準備するだけで、誰でも、比較的簡単に、サポートベクターマシンを用いた非線形の識別器を実現することができる。画像認識の分野でも、顔の検出や歩行者の検出等の対象認識 [4, 5, 6]、文字認識 [7] 等に利用され、高い認識性能が得られることが報告されている。

カーネル学習法は、サポートベクターマシンだけでなく、多変量データ解析等の線形モデルで表される手法を非線形に拡張するためにも利用することができる [8, 9, 10]。すでに、カーネル主成分分析、カーネル判別分析、カーネル部分空間法、カーネル正準相関分析等が提案されている。

本稿では、サポートベクターマシンを中心に、カーネル学習法 [9] の考え方を概説し、汎化性能を向上させる手法、および、その画像認識への応用例について紹介する。

2 カーネル学習法

線形識別手法は、理論的背景がしっかりしており、また、長年の研究成果の蓄積もあり、パターン認識の基本的手法となっている。しかし、本質的に非線形な問題に対しては能力が不足しており、その適用は困難であった。一方、非線形な識別問題を直接扱おうとすると、特定の問題に有効な手法は開発できても、汎用性のある識別手法を開発することが困難であった。そこで、特徴ベクトルを非線形変換して、その空間で線形の識別を行うことで、線形識別手法を非線形へ拡張するカーネル学習法が提案されてきた。本節では、カーネル学習法の基礎となる「カーネルトリック」について概説し、これと組み合わせる事で高い識別性能を実現したサポートベクターマシンと判別分析について解説する。

2.1 カーネルトリック

一般に、線形分離可能性はサンプル数が大きくなればなるほど難しくなり、逆に、特徴空間ベクトルの次元が大きくなるほど易しくなる。例えば、特徴ベクトルの次元が訓練サンプルの数よりも大きいなら、どんなラベル付けに対しても線形分離可能である。しかし、高次元への写像を行うと、次元の増加に伴い汎化能力が落ちてしまう。また、難しい問題を線形分離可能にするためには、訓練サンプルと同程度の大きな次元に写像しなければならないので、結果的に膨大な計算量が必要となってしまう。

今、元の特徴ベクトル $\boldsymbol{x}$ を非線形の写像 $\phi(\boldsymbol{x})$ によって変換し、その空間で線形識別を行うことを考えてみよう。例えば、写像 ϕ として、入力特徴を 2 次の多項式に変換する写像を用いるとすると、写像した先で線形識別を行うことは、もとの空間で 2 次の識別関数を構成することに対応する。一般には、こうした非線形の写像によって変換した特徴空間の次元は大きくなりがちである。しかし、サポートベクターマシンの場合には、幸いにも、目的関数 L_D や識別関数が入力パターンの内積のみに依存した形になっており、内積が計算できれば最適な識別関数を構成することが可能である。つまり、もし非線形に写像した空間での二つの要素 $\phi(\boldsymbol{x}_1)$ と $\phi(\boldsymbol{x}_2)$ の内積が

$$\phi(\boldsymbol{x}_1)^T\phi(\boldsymbol{x}_2) = K(\boldsymbol{x}_1, \boldsymbol{x}_2) \tag{1}$$

のように、入力特徴 $\boldsymbol{x}_1$ と $\boldsymbol{x}_2$ のみから計算できるなら、非線形写像によって変換された特徴空間での特徴 $\phi(\boldsymbol{x}_1)$ や $\phi(\boldsymbol{x}_2)$ を陽に計算する代わりに、$K(\boldsymbol{x}_1, \boldsymbol{x}_2)$ から最適な非線形写像を構成できる。ここで、このような K のことをカーネルと呼んでいる。このように高次元に写像しながら、実際には写像された空間での特徴の計算を避けて、カーネルの計算のみで最適な識別関数を構成するテクニックのことを「カーネルトリック」と呼んでいる。

実用的には、K は計算が容易なものが望ましい。例えば、多項式カーネル

$$K(\boldsymbol{x}_1, \boldsymbol{x}_2) = (1 + \boldsymbol{x}_1^T\boldsymbol{x}_2)^p \tag{2}$$

Gauss カーネル

$$K(\boldsymbol{x}_1, \boldsymbol{x}_2) = \exp\left(\frac{-||\boldsymbol{x}_1 - \boldsymbol{x}_2||^2}{2\sigma^2}\right) \tag{3}$$

シグモイドカーネル

$$K(\boldsymbol{x}_1, \boldsymbol{x}_2) = \tanh\left(a\boldsymbol{x}_1^T\boldsymbol{x}_2 - b\right) \tag{4}$$

などが使われている。

2.2 サポートベクターマシン

サポートベクターマシンは、1960 年代に Vapnik 等が考案した Optimal Separating Hyperplane を起源とし、1990 年代になってカーネル学習法と組み合わせた非線形の識別手法へと拡張された。カーネルトリッ

クにより非線形の識別関数が構成できるように拡張したサポートベクターマシンは、現在知られている手法の中でも最もパターン認識性能の優秀な学習モデルの一つである。ただし、サポートベクターマシンは、基本的には 2 つのクラスを識別する識別器を構成するための学習法であり、文字認識などの多クラスの識別器を構成するためには、複数のサポートベクターマシンを組み合わせるなどの工夫が必要となる。ここでは、まず、サポートベクターマシンを中心にカーネル学習法を用いて訓練サンプルから非線形の識別器を構成する方法について概説する。一般に、カーネル学習法を用いて学習された識別器が、訓練サンプルに含まれていない未学習データに対しても高い識別性能を発揮できるためには、汎化能力を向上させるための工夫が必要である。サポートベクターマシンでは、「マージン最大化」という基準を用いることでこれを実現している。これは、結果的には、不必要なパラメータが値を持たないように学習の評価基準にペナルティ項を追加する shrinkage 法の一種と考えることができる。

サポートベクターマシンは、単純パーセプトロン（ニューロンのモデルとして最も単純な線形しきい素子）を用いて、2 クラスのパターン識別器を構成する手法である。ここでは、サポートベクターマシンの一般的な定義に従って、線形しきい素子として、入力特徴ベクトルに対し、識別関数（線形識別関数）

$$y = \mathrm{sign}(\boldsymbol{w}^T\boldsymbol{x} - h) \tag{5}$$

により 2 値 (± 1) を出力するモデルを用いて説明する。ここで、$\boldsymbol{w}$ はシナプス荷重に対応するパラメータであり、h はしきい値である。また、関数 $\mathrm{sign}(u)$ は、$u > 0$ のとき 1 をとり、$u \leq 0$ のとき -1 をとる符号関数である。このモデルは、入力ベクトルとシナプス荷重の内積がしきい値を超えれば 1 を出力し、超えなければ -1 を出力する。これは、幾何学的には、識別平面により、入力特徴空間を 2 つに分けることに相当する。今、2 つのクラスを C_1,C_2 とし、各クラスのラベルを 1 と -1 に数値化しておくとする。また、訓練サンプル集合として、N 個の特徴ベクトル $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N$ と、それぞれのサンプルに対する正解のクラスラベル $t_1, \ldots, t_N$ が与えられているとする。また、この訓練サンプル集合は、線形分離可能であるとする。すなわち、線形しきい素子のパラメータをうまく調整することで、訓練サンプル集合を誤りなく分けることができると仮定する。

訓練サンプル集合が線形分離可能であるとしても、

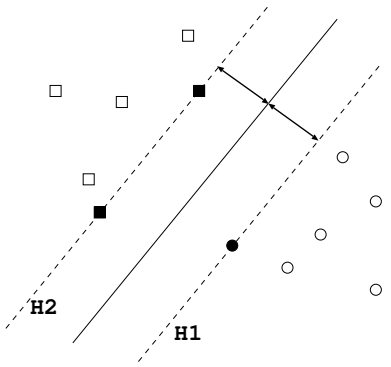

図 1: 線形しきい素子の分離超平面とマージン (◯がクラス 1 のサンプルで、□がクラス -1 のサンプルを示す。●と■はサポートベクターを示す。)

一般には、訓練サンプル集合を誤りなく分けるパラメータは一意には決まらない。サポートベクターマシンでは、訓練サンプルをすれすれに通るのではなく、なるべく余裕をもって分けるような識別平面が求められる。具体的には、最も近い訓練サンプルとの余裕をマージンと呼ばれる量で測り、マージンが最大となるような識別平面を求める。もし、訓練サンプル集合が線形分離可能なら、

$$t_i(\boldsymbol{w}^T\boldsymbol{x}_i - h) \geq 1, \quad i = 1, \ldots, N \tag{6}$$

を満たすようなパラメータが存在する。これは、H1: $\boldsymbol{w}^T\boldsymbol{x} - h = 1$ と H2: $\boldsymbol{w}^T\boldsymbol{x} - h = -1$ の 2 枚の超平面で訓練サンプルが完全に分離されており、2 枚の超平面の間にはサンプルがひとつも存在しないことを示している。線形識別関数の性質についての説明で触れたように、識別平面とこれらの超平面との距離（マージンの大きさ）は、$\frac{1}{||\boldsymbol{w}||}$ となる。したがって、マージンを最大とするパラメータ $\boldsymbol{w}$ と h を求める問題は、結局、制約条件

$$t_i(\boldsymbol{w}^T\boldsymbol{x}_i - h) \geq 1, \quad (i = 1, \ldots, N) \tag{7}$$

の下で、目的関数

$$L(\boldsymbol{w}) = \frac{1}{2}||\boldsymbol{w}||^2 \tag{8}$$

を最小とするパラメータを求める問題と等価になる。この最適化問題は、数理計画法の分野で 2 次計画問題として知られており、さまざまな数値計算法が提案されている。ここでは、双対問題に帰着して解く方法を紹介する。まず、Lagrange 乗数 $\alpha_i (\geq 0), i = 1, \ldots, N$

を導入し、目的関数を

$$L(\boldsymbol{w}, h, \boldsymbol{\alpha}) = \frac{1}{2}||\boldsymbol{w}||^2 - \sum_{i=1}^{N} \alpha_i \{t_i(\boldsymbol{w}^T \boldsymbol{x}_i - h) - 1\} \tag{9}$$

と書き換える。パラメータ $\boldsymbol{w}$ および h に関する偏微分から停留点では、

$$\boldsymbol{w} = \sum_{i=1}^{N} \alpha_i t_i \boldsymbol{x}_i \tag{10}$$

$$0 = \sum_{i=1}^{N} \alpha_i t_i \tag{11}$$

という関係が成り立つ。これらを上の目的関数の式に代入すると、制約条件、

$$\sum_{i=1}^{N} \alpha_i t_i = 0 \tag{12}$$

$$0 \leq \alpha_i, \quad i = 1, \ldots, N \tag{13}$$

の下で、目的関数

$$L_D(\boldsymbol{\alpha}) = \sum_{i=1}^{N} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{N} \alpha_i \alpha_j t_i t_j \boldsymbol{x}_i^T \boldsymbol{x}_j \tag{14}$$

を最大とする双対問題が得られる。これは、 Lagrange 乗数 α_i (≥ 0), $i = 1, \ldots, N$ に関する最適化問題となる。その解で α_i^* が 0 でない、すなわち、 $\alpha_i^* > 0$ となる訓練サンプル $\boldsymbol{x}_i$ は、先の 2 つの超平面 $\boldsymbol{w}^T \boldsymbol{x} - h = 1$ か $\boldsymbol{w}^T \boldsymbol{x} - h = -1$ のどちらかにのっている。このことから、 α_i^* が 0 でない訓練サンプル $\boldsymbol{x}_i$ のことを「サポートベクター」と呼んでいる。これが、サポートベクターマシンの名前の由来である。直感的に理解できるように、一般には、サポートベクターは、もとの訓練サンプル数に比べてかなり少ない。つまり、沢山の訓練サンプルの中から小数のサポートベクターを選び出し、それらのみを用いて線形しきい素子のパラメータが決定されることになる。

実際、双対問題の最適解 $\alpha_i^*(i \geq 0)$、および停留点での条件式から、最適なパラメータ $\boldsymbol{w}^*$ は、

$$\boldsymbol{w}^* = \sum_{i \in S} \alpha_i^* t_i \boldsymbol{x}_i \tag{15}$$

となる。ここで、 S はサポートベクターに対応する添え字の集合である。また、最適なしきい値 h^* は、 2 つの超平面 $\boldsymbol{w}^T \boldsymbol{x} - h = 1$ か $\boldsymbol{w}^T \boldsymbol{x} - h = -1$ のどちらかにのっているという関係を利用して求めることができる。すなわち、任意のサポートベクター $\boldsymbol{x}_s, s \in S$ から

$$h^* = \boldsymbol{w}^{*T} \boldsymbol{x}_s - t_s \tag{16}$$

により求まる。

また、最適な識別関数を双対問題の最適解 $\alpha_i^*(i \geq 0)$ を用いて表現すると

$$\begin{aligned} y &= \mathrm{sign}(\boldsymbol{w}^{*T} \boldsymbol{x} - h^*) \\ &= \mathrm{sign}(\sum_{i \in S} \alpha_i^* t_i \boldsymbol{x}_i^T \boldsymbol{x} - h^*) \end{aligned} \tag{17}$$

となる。すなわち、 $\alpha_i^* = 0$ となる多くの訓練サンプルを無視し、 $\alpha_i^* > 0$ となる識別平面に近い少数の訓練サンプルのみを用いて識別関数が構成される。ここで、重要な点は、「マージン最大化」という基準から自動的に識別平面付近の少数の訓練サンプルのみが選択されたことであり、その結果として、未学習データに対してもある程度良い識別性能が維持できていると解釈できる。 すなわち、サポートベクターマシンは、マージン最大化という基準を用いて、訓練サンプルを撰択することで、モデルの自由度を抑制するようなモデル撰択が行われていると解釈できる。

2.2.1 ソフトマージン

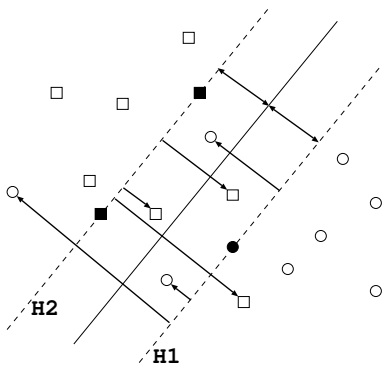

図 2: ソフトマージン (◯がクラス 1 のサンプルで、□がクラス -1 のサンプルを示す。●と■はサポートベクターを示す。)

上述のサポートベクターマシンは、訓練サンプルが線形分離可能な場合についての議論であるが、パターン認識の実問題で線形分離可能な場合は稀である。したがって、実際的な課題にサポートベクターマシンを使うには、さらなる工夫が必要である。まず考えられるのは、多少の識別誤りは許すように制約を緩める方

法である。これは、「ソフトマージン」と呼ばれている。

ソフトマージン法では、マージン $\frac{1}{||\boldsymbol{w}||}$ を最大としながら、図2に示すように、幾つかのサンプルが超平面 H1 あるいは H2 を越えて反対側に入ってしまうことを許す。反対側にどれくらい入り込んだかの距離を、パラメータ $\xi_i(\geq 0)$ を用いて、 $\frac{\xi_i}{||\boldsymbol{w}||}$ と表すとすると、その和

$$\sum_{i=1}^{N} \frac{\xi_i}{||\boldsymbol{w}||} \tag{18}$$

はなるべく小さいことが望ましい。これらの条件から最適な識別面を求める問題は、制約条件

$$\xi_i \geq 0, \quad t_i(\boldsymbol{w}^T\boldsymbol{x}_i - h) \geq 1 - \xi_i, \quad (i = 1, \ldots, N) \tag{19}$$

の下で、目的関数

$$L(\boldsymbol{w}, \boldsymbol{\xi}) = \frac{1}{2}||\boldsymbol{w}||^2 + \gamma \sum_{i=1}^{N} \xi_i \tag{20}$$

を最小とするパラメータを求める問題に帰着される。ここで、あらたに導入したパラメータ γ は、第1項のマージンの大きさと第2項のはみ出しの程度とのバランスを決める定数である。

この最適化問題の解法は、基本的には線形分離可能な場合と同様にふたつの制約条件に対して、 Lagrange 乗数 α_i、および、 ν_i を導入し、目的関数を

$$\begin{aligned} L(\boldsymbol{w}, h, \boldsymbol{\alpha}, \boldsymbol{\nu}) &= \frac{1}{2}||\boldsymbol{w}||^2 + \gamma \sum_{i=1}^{N} \xi_i \\ &\quad - \sum_{i=1}^{N} \alpha_i \{t_i(\boldsymbol{w}^T\boldsymbol{x}_i - h) - (1 - \xi_i)\} \\ &\quad - \sum_{i=1}^{N} \nu_i \xi_i \end{aligned} \tag{21}$$

と書き換える。パラメータ $\boldsymbol{w}$、 h、 ν_i に関する偏微分を 0 とする停留点では、

$$\boldsymbol{w} = \sum_{i=1}^{N} \alpha_i t_i \boldsymbol{x}_i \tag{22}$$

$$0 = \sum_{i=1}^{N} \alpha_i t_i \tag{23}$$

$$\alpha_i = \gamma - \nu_i \tag{24}$$

という関係が成り立つ。これらを目的関数の式に代入すると、制約条件

$$\sum_{i=1}^{N} \alpha_i t_i = 0 \tag{25}$$

$$0 \leq \alpha_i \leq \gamma, \quad i = 1, \ldots, N \tag{26}$$

の下で、目的関数

$$L_D(\boldsymbol{\alpha}) = \sum_{i=1}^{N} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{N} \alpha_i \alpha_j t_i t_j \boldsymbol{x}_i^T \boldsymbol{x}_j \tag{27}$$

を最大とする双対問題が得られる。線形分離可能な場合には、最適解 α_i^* の値により、平面 H1 および H2 上の訓練サンプル (サポートベクター) とそれ以外のサンプルに分類されたが、ソフトマージンの場合には、さらに、 H1 および H2 をはさんで反対側にはみ出すサンプルが存在する。それらは、同様に、最適解 α_i^* の値により区別することができる。具体的には、 $\alpha_i^* = 0$ なら、平面 H1 あるいは H2 の外側に存在し、学習された識別器によって正しく識別される。また、 $0 < \alpha_i^* < \gamma$ の場合には、対応するサンプルは、ちょうど平面 H1 あるいは H2 の上に存在するサポートベクターとなり、これも正しく識別される。 $\alpha_i^* = \gamma$ の場合には、対応するサンプルはサポートベクターとなるが、 $\xi_i \neq 0$ となり、平面 H1 あるいは H2 の内側に存在することになる。

2.2.2 カーネルサポートベクターマシン

式 (14) や式 (27) の目的関数 L_D は、

$$\begin{aligned} L_D(\boldsymbol{\alpha}) &= \sum_{i=1}^{N} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{N} \alpha_i \alpha_j t_i t_j \phi(\boldsymbol{x}_i)^T \phi(\boldsymbol{x}_j) \\ &= \sum_{i=1}^{N} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{N} \alpha_i \alpha_j t_i t_j K(\boldsymbol{x}_i, \boldsymbol{x}_j) \end{aligned} \tag{28}$$

のように内積をカーネルで置き換えた形に書ける。また、式 (17) から最適な識別関数は、

$$\begin{aligned} y &= \mathrm{sign}(\boldsymbol{w}^{*T}\phi(\boldsymbol{x}) - h^*) \\ &= \mathrm{sign}(\sum_{i \in S} \alpha_i^* t_i \phi(\boldsymbol{x}_i)^T \phi(\boldsymbol{x}) - h^*) \\ &= \mathrm{sign}(\sum_{i \in S} \alpha_i^* t_i K(\boldsymbol{x}_i, \boldsymbol{x}) - h^*) \end{aligned} \tag{29}$$

のようにサポートベクターマシンの内積をカーネルで置き換えた形に書ける。ここで、この式にシグモイドカーネルを代入すると、いわゆる3層の多層パーセプトロンと同じ構造となる。また、 Gauss カーネルを代入すると、 Radial Basis Function (RBF) ネットワークと同じ構造になり、構造的には従来のニューラルネットワークと同じになる。しかし、カーネルトリックを

用いて非線形に拡張したサポートベクターマシンでは、中間層から出力層への結合荷重のみが学習により決定され、前段の入力層から中間層への結合荷重は固定で、訓練データから機械的に求められる。また、中間層のユニット数が非常に大きく、訓練サンプル数と同じになる。つまり、カーネルトリックを用いて非線形に拡張したサポートベクターマシンでは、入力層から出力層への結合荷重を適応的に学習により求めない代わりにあらかじめ中間層に非常に多くのユニットを用意することで複雑な非線形写像を構成しようとする。

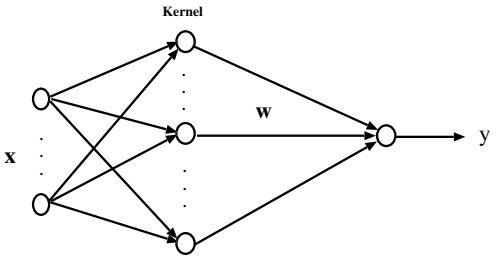

図 3: サポートベクターマシンの構造

カーネル学習法と組み合わせて非線形の識別関数が構成できるように拡張することで、カーネルサポートベクターマシンは、現在知られている多くのパターン認識手法の中でも最もパターン認識性能の良い学習モデルのひとつと考えられている。図 4に非線形のサポートベクターマシを用いて構成した識別器の例を示す。

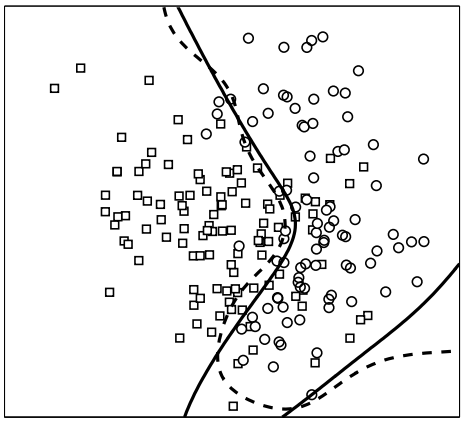

図 4: サポートベクターマシンによる識別例 (◯と□の識別課題であり、実線が Bayes 推定による識別境界、点線が SVM による識別境界である。)

2.3 カーネル判別分析

カーネルトリックを用いると、多変量データ解析等の線形モデルで表される手法を非線形に拡張することができる [8, 9, 10]。すでに、カーネル主成分分析、カーネル判別分析、カーネル部分空間法、カーネル正準相関分析等が提案されている。ここでは、カーネル判別分析について簡単に紹介する。

今、特徴ベクトルを $\boldsymbol{x} = (x_1, \ldots, x_M)^T$ を K 個のクラスに識別する課題について考えよう。この時、線形判別分析は、特徴ベクトルの空間から線形判別写像

$$\boldsymbol{y} = A^T \boldsymbol{x} \tag{30}$$

により、写された空間でのクラス内の平均的な散らばりがなるべく小さく、クラス間のちらばりがなるべく大きくなるような係数行列 $A = [a_{ij}]$ を求める問題として定式化される。判別写像の良さの評価としては、判別基準が用いられる。判別基準は、いくつかの等価な基準が知られているが、ここでは、

$$J = \boldsymbol{tr}(\hat{\Sigma}_T^{-1} \hat{\Sigma}_B) \tag{31}$$

を用いるものとする。ここで、$\hat{\Sigma}_T$ および $\hat{\Sigma}_B$ は、それぞれ、新特徴 $\boldsymbol{y}$ 上で定義された分散共分散行列およびクラス間平均分散共分散行列である。

判別基準 (31) を最大とする最適な係数行列 A は、固有値問題

$$\Sigma_B A = \Sigma_T A \Lambda \quad (A^T \Sigma_T A = I) \tag{32}$$

の解として求まる。ここで、 Λ は固有値を対角要素とする対角行列である。また、 Σ_T および Σ_B は、それぞれ、入力特徴 $\boldsymbol{x}$ 上で定義された分散共分散行列およびクラス間平均分散共分散行列であり、

$$\begin{aligned} \Sigma_T &= \sum_{i=1}^{N} (\boldsymbol{x}_i - \bar{\boldsymbol{x}}_T)(\boldsymbol{x}_i - \bar{\boldsymbol{x}}_T)^T \\ \Sigma_B &= \sum_{k=1}^{K} \omega_k (\bar{\boldsymbol{x}}_k - \bar{\boldsymbol{x}}_T)(\bar{\boldsymbol{x}}_k - \bar{\boldsymbol{x}}_T)^T \end{aligned} \tag{33}$$

のように定義される。ここで、 $\omega_k = N_k/N$, $\bar{\boldsymbol{x}}_k$, および $\bar{\boldsymbol{x}}_T$ は、それぞれ、クラス C_k の先見確率、 $\boldsymbol{x}$ のクラス C_k の平均ベクトル、および、全平均ベクトルである。

線形判別分析では、式 (30) のように線形写像を構成したが、カーネル判別分析では、非線形の変換 $\Phi(\boldsymbol{x})$ により特徴を抽出し、それらの線形結合で判別写像を

構成する。ここでは、簡単のため 1 次元の判別特徴を抽出する場合について考えよう。すなわち、

$$y = \boldsymbol{a}^T \Phi(\boldsymbol{x}) \tag{34}$$

のような変換を考える。この変換の結合重み $\boldsymbol{a}$ は訓練サンプルの線形結合によって、

$$\boldsymbol{a} = \sum_{i=1}^{N} \alpha_i \Phi(\boldsymbol{x}_i) \tag{35}$$

のように書ける。これを上式に代入すると

$$\begin{aligned} y &= \sum_{i=1}^{N} \alpha_i \Phi(\boldsymbol{x}_i)^T \Phi(\boldsymbol{x}) \\ &= \sum_{i=1}^{N} \alpha_i K(\boldsymbol{x}_i, \boldsymbol{x}) \\ &= \boldsymbol{\alpha}^T \boldsymbol{k}_i(\boldsymbol{x}) \end{aligned} \tag{36}$$

となる。ただし、 $\boldsymbol{k}_i(\boldsymbol{x}) = (K(\boldsymbol{x}_1, \boldsymbol{x}), \ldots, K(\boldsymbol{x}_N, \boldsymbol{x}))^T$ は、カーネル特徴を並べたベクトル（カーネル特徴ベクトル）である。

これらの関係からわかるように、判別基準

$$J = \frac{\boldsymbol{\alpha}^T \Sigma_B^{(K)} \boldsymbol{\alpha}}{\boldsymbol{\alpha}^T \Sigma_W^{(K)} \boldsymbol{\alpha}} \tag{37}$$

を最大とするパラメータ $\boldsymbol{\alpha}$ を求める問題は、カーネル特徴ベクトルに基づいて線形判別分析を行うことと等価となり、固有値問題

$$\Sigma_B^{(K)} \boldsymbol{\alpha} = \Sigma_W^{(K)} \boldsymbol{\alpha} \lambda \tag{38}$$

の解として求まる。ここで、 $\Sigma_B^{(K)}$ および $\Sigma_W^{(K)}$ は、それぞれ、カーネル特徴ベクトルに関する平均クラス間分散共分散行列および平均クラス内分散共分散行列である。

判別分析は、識別に有効な低次元の特徴を抽出する手法であり、汎化性能は比較的良いが、カーネル判別分析の場合には、汎化性能を向上させる工夫が必要となることもある。最も簡単で良く知られている方法は、平均クラス間分散共分散行列の対角要素に適当な定数を加えて、

$$\tilde{\Sigma}_W^{(K)} = \Sigma_W^{(K)} + \alpha I \tag{39}$$

のようにする手法である。これは、各特徴に平均 0 の正規ノイズを加えるのと同様の効果があり、数値計算を安定化させる。

3 カーネル学習法の汎化性能向上手法

カーネル学習法にサポートベクターマシンや判別分析などの識別手法を組み合わせる事により、強力な非線形識別器が構成できる。しかし、どんなに強力な識別器であっても、入力特徴ベクトル中に予測モデルにとって不要な特徴が含まれていると、それらは識別性能を阻害する要因となる。そこで、学習に有効な特徴の部分集合をを選択する変数選択法はカーネル学習法においても重要である。また、訓練サンプル数が膨大な場合、全サンプルを使用して識別器を構成しようとすると、計算時間が膨大にかかり、同時に過学習により汎化性能を低下させる可能性もある。このような場合には、訓練サンプルの部分集合を適切に選択するサンプル選択法とも言える手法が有効となる。

3.1 変数選択法

変数選択のためには、すべての特徴の部分集合に対して、予測性能を評価する必要がある。しかし、部分集合の数は、特徴の数が増えると指数関数的に増大する。したがって、特徴の数が多い場合には、すべての部分集合に対して評価することは現実的では無い。そのため、比較的良い特徴の部分集合を探策する手法が提案されている。

単純な方法としては、 Forward stepwise selection あるいは、 Backward stepwise selection と呼ばれる手法がある。 Forward stepwise selection は、最初、特徴 1 個のみのモデルからはじめて、特徴を 1 個づつ追加して行くことで、最も良い特徴の組を選び出す。逆に、 Backward stepwise selection は、全ての特徴を含むモデルから特徴を 1 個づつ取り除いて行くことで、最も良い特徴の組を選び出す。

遺伝的アルゴリズムを用いて特徴の組を選択する手法は [15] で提案されているが、 [16] では訓練セットに主成分分析をかけることで固有ベクトルを特徴として抽出し、遺伝的アルゴリズムで良い特徴の組を選び出している。

[17] では、 Latent Semantic Indexing[18] という手法で特徴抽出を行い、線形サポートベクターマシンのマージン w に対するそれぞれの特徴の貢献度を

$$w_k^2 = \sum_{i,j=1} m \alpha_i \alpha_j y_i y_j x_{ki} x_{kj} \tag{40}$$

として、 w_k^2 が小さい値の特徴を取り除くことにより、有効な特徴の組を残そうとしている。

[19] では、入力データをシンボル化した後、クラス情報との相互エントロピーを計算し、相互エントロピーの低いもの（他のシンボルとの相関の強いもの）を除いていく事で、有効な特徴のみを残すようにしている。

3.2 サンプル選択法

サンプル選択法については、必ずしも画像認識に用いられたものではないが、画像認識課題に対しても有効と考えられるものをいくつか紹介する。

[23] では、入力データを自己組織化マップ（ＳOM）[21] によってクラスタリングすることで、元のデータの分布などを反映しつつカーネルの中央値とするサンプル数を削減している。ＳOMの様に元データからボトムアップにサンプルを削減しただけでは、目的とする識別、あるいは、関数回帰に有用なサンプルが選択されるとは限らないため、 [23] では、関数回帰の誤差を評価し、誤差を小さくする方向にカーネルの中央値の調整を行う事を併用している。

[20] では、カーネルによって非線形変換された空間をより少ない特徴ベクトルで再現するため、特徴ベクトル選択法 (Feature Vector Selection) を提案している。この手法では、まず、一つの特徴ベクトルを選択し、次に、そのベクトルとは超空間中での方向が最も異なるものを二つ目の特徴ベクトルとして選択する。残りの特徴ベクトルから、既に選択された特徴ベクトルの線形結合で表現できる特徴ベクトルは取り除き、表現できない（線形結合との誤差が大きい）特徴ベクトルは新たに選択するという手順を繰り返す。その結果、元の全ての特徴ベクトルによって表現された空間は、選択された特徴ベクトルの線形結合で近似されることになる。

[22] では、データマイニングでの大量データの処理のために、圧縮とサンプリングを組み合わせた squashing と呼ばれる手法を提案している。 squashing は、まず、データを領域ごとにグループ化し、領域内のモーメントを計算する。次に、各領域中のデータのモーメントで近似データを生成する。これにより、グループ化された領域の代表点から全てのデータが近似として計算できることになる。

3.3 選択基準

ここで紹介した変数選択法は、何らかの方法で識別率を基準として変数（特徴）の選択を行っており、 supervised selection を行っていると考える事ができる。

図 5: サンプル画像と特徴、および、局所領域の例

一方、サンプル選択法では、元々のサンプル数が膨大であることを前提としているためか、識別率や回帰誤差などの結果を基準とするよりは、選択したサンプルが元のサンプル集合の空間を再現できるかどうかを基準としている。いわば、 unsupervised selection である。膨大なデータに対しての学習を避けるという意味では、識別器を構成する前にサンプルの選択を行う事が必要ではある。しかし、その選択が識別器の学習に有効なものでないと、高い識別性能を実現する事は難しくなる。そこで、 [23] のように unsupervised selection に識別結果による supervised な修正を加える手法が有効になると考えられる。

4 カーネル学習法の画像認識への応用

4.1 特徴選択と Soft-Margin SVM の Boosting を用いた歩行者検出

特徴選択とサポートベクターマシンを組み合わせて歩行者検出課題に適用した例を紹介する。 [24] は、特徴選択と AdaBoost[7] を組み合わせた歩行者識別の手法を提案している。この手法では、画像情報から二種類の特徴（ヒストグラム均一化特徴とエッジ特徴）を抽出し、それぞれの特徴画像を 4×8 画素の領域から順次領域を拡大しながら画像全体を走査し、それぞれ 100 個の部分画像を生成し、これを局所領域とする (図

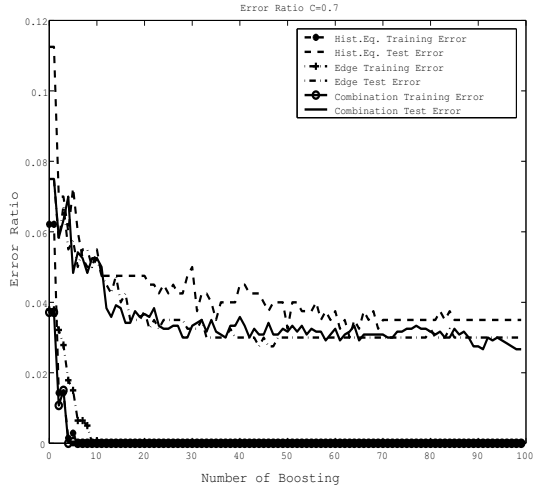

図 6: 100 局所領域でのエラー率

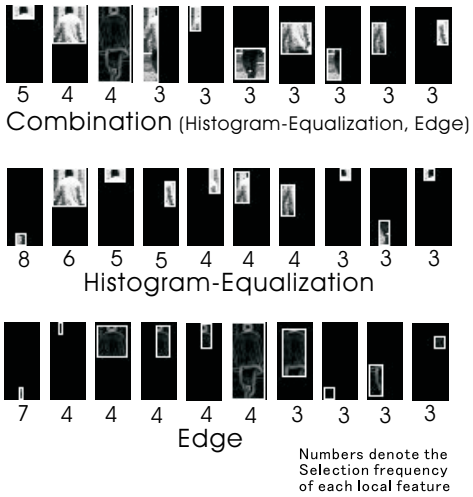

図 7: 選択された局所領域の例

5)。 Boosting の i 段目では、 $i-1$ 段目で定められた入力重みで、全ての特徴、局所領域に対する弱識別器を訓練され、その中で最もエラー率の低い弱識別器を i 段目の弱識別器とする。次段の入力重みと識別器の重みは、その識別器のエラー率を元に計算される。

図 6は、 Boosting100 段までのエラー率を示したもので、ヒストグラム均一化特徴でのテストエラーは 3.5%、エッジ特徴でのテストエラーは 3.0% となっている。 Boosting の各段で、ヒストグラム均一化特徴とエッジ特徴から訓練エラーの低い方を選択するようにした場合のテストエラーは 2.75% となり、複数の特徴から最適なものを選択する事によって識別器の汎化性が向上したことが示されている。図 7は、選択された局所領域の例を示す。この結果は、一種類の特徴を使用した場合よりも複数（二つ）の特徴を組み合わせた方が、小さな局所領域からある程度大きな局所領域までが満遍なく選択されて傾向があることを示す。特徴量の組み合わせにより、より「良い」局所領域が選択されると考えられる。

4.2 サポートベクタートラッキング

対象の追跡（トラッキング）は、通常は輝度情報やオプティカルフローを元に、対象が存在する尤度の高い領域を求めるものである。ここでは、対象の存在する尤度推定にサポートベクターマシンを利用したサポートベクタートラッキング [25] を紹介する。

領域 $\boldsymbol{I}$ での対象物の存在を識別するサポートベクターマシンは、

$$\sum_{j=1}^{i} y_i \alpha_j k(\boldsymbol{I}, \boldsymbol{x_j}) + b_i \tag{41}$$

で表される。ここで、 x_j はサポートベクター、 y_i は符号、 α_j はラグランジュ係数、 $k(\boldsymbol{I}, \boldsymbol{x_j})$ はカーネル関数で、 $\boldsymbol{I}$ はテストしようとする画像領域を示す。この式が正であれば、領域内に対象が存在し、負であれば存在しないことになる。

追跡すべき対象物の推定位置の初期値を $\boldsymbol{I_{init}}$、最終的な推定位置を $\boldsymbol{I_{final}}$ とした時、一次のテイラー展開により最終的な推定位置は (42) 式で表される。

$$\boldsymbol{I_{final}} = \boldsymbol{I_{init}} + u\boldsymbol{I_x} + v\boldsymbol{I_y} \tag{42}$$

ここで、 $\boldsymbol{I_x}$、 $\boldsymbol{I_y}$ は、初期推定領域 $\boldsymbol{I_{init}}$ の x 方向、 y 方向の微分値であり、 u、 v は動きのパラメータである。

最終的な推定位置は、領域内に対象が存在する尤度を最大化するものであるから、

$$\sum_{j=1}^{i} y_j \alpha_j k(\boldsymbol{I_{final}}, \boldsymbol{x_j}) = \max\{\boldsymbol{I} | \sum_{j=1}^{i} y_j \alpha_j k(\boldsymbol{I}, \boldsymbol{x_j})\} \tag{43}$$

となるような $\boldsymbol{I_{final}}$ に隣接する領域 $\boldsymbol{I}$ を見つければ良い事になる。

車両の追跡に適用した例では、輝度値の二乗誤差を用いたトラッキング方式では 10 フレーム程度で追跡が失敗する場合や隣の車両を間違えて追跡してしまう場合でも、サポートベクタートラッキングは対象を見失わないことが示されている。

5 おわりに

本稿では、サポートベクターマシンを中心に汎化性能の高い非線形の識別器を構成するための手法としてカーネル学習の話題について紹介した。カーネル学習法とサポートベクターマシンや判別分析を組み合わせる事により強力な識別器を構成する事が可能であるが、より高い汎化性を実現するためには適切な変数選択やサンプル選択を行う事が重要である。特に、膨大なデータを扱おうとする場合には、あらかじめサンプル数を削減するサンプル選択の重要性が高くなってくると考えられる。

参考文献

[1] V.N.Vapnik, *Statistical Learning Theory*, John Wiley & Sons (1998).

[2] 赤穂, 津田, “サポートベクターマシン — 基本的仕組みと最近の発展 —,” 数理科学, No.444, pp.52-58 (2000).

[3] 前田, “痛快！サポートベクトルマシン - 古くて新しいパターン認識手法 –,” 情報処理, Vol.42, No.7, pp.676-683 (2001).

[4] E.Osuna and R.Freund and F.Girosi, “Training Support Vector Machines: an Application to Face Detection,” Proc. of IEEE Computer Society Conf. on Computer Vision and Pattern Recognition, pp.130-136, 1997.

[5] C.P.Papageorgiou and M.Oren and T.Poggio, “A General Framework for Object Detection,” Proc. Fifth Int’l Conf. on Computer Vision, 1998.

[6] A.Mohan and C.P.Papageorgiou and T.Poggio, “Example-Based Object Detection in Images by Components,” IEEE Trans. on Pattern Analysis and Machine Intelligence, Vo.23, No.4, pp.349-361, 2001.

[7] T.Hastie, R.Tibshirani, J.Friedman, *The Elements of Statistical Learning - Data Mining, Inference, and Prediction*, Springer-Verlag, 2001.

[8] K.R.Muller, S.Mika, G.Ratsch, K.Tsuda, B.Scholkopf, “An introduction to kernel-based learning algorithms,” IEEE Trans. On Neural Networks, Vol.12, No.2, pp.181-201, 2001.

[9] B.Scholkpf and A.J.Smola, “Learning with Kernels,” The MIT Press, 2002.

[10] 麻生、津田、村田、” パターン認識と学習の統計学,” 岩波書店, 2003.

[11] D.M.Gavlira, “Pedestrian Detection from a Moving Vehicle”, *Proc. of Europian Conference on Computer Vision*, pp.37-49, 2000.

[12] B.Schölkopf, S.Mika, C.J.C,Burges, P.Knirsch, KR.Müller, G.Rätsch, and A.J.Smola, “Input Space Versus Fueature Space in Kernel-Based Methods”, *Trans. on Neural Networks*, val.10, No.5, pp.1000-1017, 1999.

[13] 堀田一弘、三島健稔、栗田多喜夫, “未知の画像に対する識別率を用いた顔検出のための特徴点の順序付け,” 電子情報通信学会論文誌, Vol.J84-D-II, No.8, pp.1781-1789, 2001.

[14] T.Kurita, T.Taguchi, “A modification of kernel-based Fisher discriminant analysis for face detection,” Proc. of the Fifth Inter. Conf. on Automatic face and Gesture Recognition, 20-21 May 2002, Washington, D.C., pp.300-305, 2002.

[15] 栗田多喜夫、 “遺伝的アルゴリズムによる線形重回帰分析における説明変数の選択の試み,” 情報処理学会全国大会講演予稿集, 4P-08, 1994.

[16] Z. Sun, G. Bebis and R. Miller, “Object detection using feature subset selection”, *Pattern Recognition*, Vol.36, pp.2165-2176, (2004).

[17] K.Shima, M.Todoriki and A.Suzuki, “SVM-based feature selection of latent semantic features”, *Pattern Recognition Letters*, Vol.25, pp. 1051-1057, (2004).

[18] S.Deerwester, V.Dumais, G.W.Furnas and T.K. Landauer, “Indexing by latent semantic analysis”, J. Amer. Soc. Inform. Sci., Vol.41, pp.391-297 (1990).

[19] R.Kumar, V.K.Jayaraman and B.D.Kulkarni, “An SVM classifier incorporating simulteneous noise reduction and Feature selection: ilustarative case examples”, *Pattern Recognition*, Vol.38, pp.41-49, (2004).

[20] G.Baudat and F.Anouar, “ Feature vector selection and projection using kernels”, *Neurocomputing*,Val.55, PP.21-38, (2003).

[21] T.Kohonen, *Self-Organizing Maps*, Springer, 2001.

[22] W.DuMouchel, C.Volinsky, T.Johnson, C.Cortes and D.Pregibon, “Squashing Flat Files Flatter,” KDD-99, pp.6-15 (1999).

[23] K.Nishida, T.Takahashi, and T.Kurita,“A Topographic Kernel-Based Regression Method,” Proceedings of 6th Joint Conference on Information Sciences, pp.521-524 2002.

[24] 西田, 栗田, “特徴選択と Soft-Margin SVM の Boosting を用いた歩行者検出”, 電子情報通信学会技術報告, PRMU2004-187(2005-2), pp.49-54, (2005).

[25] S.Avidan, “Support Vector Tracking”, IEEE *Trans. on Pattern Analysis and Machine Intelligence*, Vol.25, No.6, pp.1064-1072, (2004).