解説

数理計画の理論と実装

数理計画法とサポートベクターマシン

矢島　安敏

1. はじめに

近年Support Vector Machine (SVM) を用いた判別法が，文書の分類や画像の認識といった様々の分野に応用され，非常に有力な判別法と考えられるようになってきた．以前では，SVMは従来の判別法と比べると最適化問題を解く必要があるなど計算が複雑なため，大規模データに対しては適用できないと考えられていた．しかし，最近では計算機能力の進歩と様々な最適化技術を取り入れた高速アルゴリズムの登場で，データマイニング手法の一つとして，マイニングパッケージにも取り入れられるようになってきている．

本稿では，まず次節において，Vapnikによる標準的なSVMの定式化を示すとともに，現在まで提案されているいくつかのバリエーションを紹介する．これらに共通して用いられているアイディアは，ridgeあるいはlassoと呼ばれる回帰分析の分野では古くから用いられているものであることを紹介する．節3では，SVMの最適化アルゴリズムについて述べる．特に大規模データにも適応可能なアルゴリズムについて考える．節4では，カーネル関数を使った非線形な判別について考える．

2. SVMによる判別

2.1 SVMの定式化

N個の属性を持ったM個のデータが与えられている．各データ$j=1,2,\cdots,M$をN次元空間$\mathbb{R}^N$の点と考えて，N次の行ベクトルA_jで表すこととする．さらに，各点jには2値のラベル$y_j\in\{-1,+1\}$が与えられている．このとき，ラベルの値に従って点を判別する2クラスの判別問題を考える．

SVMはある種の線形回帰を使った判別手法と考えられる．N次元の法線ベクトルwおよび実数bで定まる線形関数$f(x)=x^Tw-b$を考え，関数値$f(A_j)$ができるだけラベルy_jと一致するようw,bを定めることを考える．最も簡単な方法は，最小二乗回帰をすればよく，

(2.1) | 最小化　$\sum_{j=1}^{M}\{y_j-(A_jw-b)\}^2$

を解きw,bを求めればよい．

ところが，一般にはこのようにして定めた関数fは，回帰に用いたデータでは$f(A_j)$とy_jとの差が小さいが，未知のデータに対してラベルを予測する際には必ずしもよい結果を導かない．このような過学習の現象を防ぐため，例えば，wの要素のいくつかを0に固定する，すなわち適切な属性選択を行い，限られた属性のみで回帰を行うことによって予測精度の向上が達成される．多くの場合，適当な基準を使い属性の削除や追加を繰り返すことによって適切な属性を選択することが行われている．

一方，ある属性を「使う」あるいは「使わない」と離散的に選択するのではなく，wのノルムをある値以下に制限した制約のもとy_jとA_jw-bの差の二乗和を最小化するridge回帰[6]と呼ばれる手法がある．すなわち，ある正のパラメータsを適当に定めた上で，以下の問題

(2.2)
$$\left|\begin{array}{ll}\text{最小化} & \sum_{j=1}^{M}\{y_j-(A_jw-b)\}^2\\ \text{制約} & \|w\|^2\le s\end{array}\right.$$

の最適化を行う．この問題は，sに対応した正のパラメータCを適当に定めれば，単純な凸2次関数最小化

(2.3)

| 最小化　$1/2\|w\|^2+C\sum_{j=1}^{M}\{y_j-(A_jw-b)\}^2$

に帰着可能であり，通常の回帰の場合と同様に線形方程式を一回解く手間でw,bの算出が行える．一方，wの1ノルムを使い

(2.4) | 最小化　$\|w\|_1+C\sum_{j=1}^{M}\{y_j-(A_jw-b)\}^2$

としたものはlasso (Least Absolute Shrinkage and

やじま　やすとし
東京工業大学
〒152-8552 目黒区大岡山2-12-1

Selection Operator)[10]と呼ばれている．この目的関数は微分不可能であり，ridge回帰の場合のように単純な最小化問題として解くことはできない．しかし，lassoで求めたwの要素には，値が厳密に0となるものが多く含まれることが知られており[5]，ridge回帰に比べて属性選択を行った場合と似た性質の解を求めることが可能と考えられている．

さて，以上の定式化では，いずれもy_jとA_jw-bの差をできるだけ0にすることを目指してきた．しかし，判別問題の場合にはこの考え方はあまり適切とは言えない．判別に際しては，クラスの差がより明確に分かれることが求められ，$y_j=1$のデータA_jに対しては$y_j\leq f(A_j)$となり，逆に$y_j=-1$のデータA_jに対しては$y_j\geq f(A_j)$となるようy_jとの差が生じても問題はないと考えるのが自然であろう．すなわち，

$$(2.5)\quad \xi_j=\begin{cases}\max\{0, y_j-(A_jw-b)\}, & \text{if } y_j=1\\ \max\{0, -y_j+(A_jw-b)\}, & \text{if } y_j=-1\end{cases}$$

と定まるペナルティξ_jを0に近づけることが目標である．ちなみに，$y_j=\pm 1$であることに注意して変形すれば，(2.5)は

$$(2.6)\quad \xi_j=\max\{0, 1-y_j(A_jw-b)\}$$

と一つの式で表現できる．さらに記号を簡略化するために，各データA_jを第j行ベクトルとするM行N列の行列Aを

$$A=\begin{bmatrix}A_1\\ A_2\\ \vdots\\ A_M\end{bmatrix}$$

と定める．また，各ラベルの値を要素とするM次元ベクトルyおよびM次対角行列Yを

$$y=\begin{pmatrix}y_1\\ y_2\\ \vdots\\ y_M\end{pmatrix},\quad Y=\begin{bmatrix}y_1 & 0 & \cdots & 0\\ 0 & y_2 & \cdots & \vdots\\ \vdots & \vdots & \ddots & 0\\ 0 & 0 & \cdots & y_M\end{bmatrix}$$

と定義する．さらに，eを要素が全て1のベクトル，また$\xi=(\xi_1, \xi_2, \cdots, \xi_M)^T$と定める．このとき，ridge回帰で用いた問題(2.3)の「誤差の二乗和」の部分を(2.5)で定めたペナルティξ_jを使い変形すると，次の最適化問題

$$(2.7)\quad \left|\begin{array}{ll}\text{最小化} & 1/2\|w\|^2+C\|\xi\|^2\\ \text{制約} & \xi\geq e-YAw+yb,\quad \xi\geq 0\end{array}\right.$$

が得られる．この問題は，2-norm SVMと呼ばれる定式化である．なお，最適解の性質よりξに対する非負制約は冗長であることがわかる．

一方，ペナルティξ_jはかならずしも二乗する必要はなく，単純な和を最小にする定式化

$$(2.8)\quad \left|\begin{array}{ll}\text{最小化} & 1/2\|w\|^2+Ce^T\xi\\ \text{制約} & \xi\geq e-YAw+yb,\quad \xi\geq 0\end{array}\right.$$

も可能である．この形がVapnik[12]の提案したSVMの定式化で，1-norm SVMと呼ばれているものである．また，lassoで用いた問題(2.4)に対しても同様にペナルティξ_jの和の最小化を考えれば，

$$(2.9)\quad \left|\begin{array}{ll}\text{最小化} & \|w\|_1+Ce^T\xi\\ \text{制約} & \xi\geq e-YAw+yb,\quad \xi\geq 0\end{array}\right.$$

を得る．なお，この問題は線形計画問題に帰着可能である．

SVMによる判別では，上で述べた最適化問題を解き，その最適解(w^*, b^*)により判別関数を$f(x)=x^Tw^*-b^*$と定めることとなる．また，未知のデータxに対しクラスを予測する場合には，関数値$f(x)$を求め，その値が1あるいは-1のどちらかに近いかでクラスの予測を行うこととなる．すなわちこれは，$f(x)$の正負により判別を行うことにほかならない．

なお，単純な回帰(2.1)やridge回帰(2.3)は無制約の最小化問題であり，最適化は線形方程式系の求解程度の手間で行えるのに対して，SVMの場合では，問題(2.7)や(2.8)は一般の凸2次計画問題，また，(2.9)の場合は線形計画問題となってしまうため，なんらかの最適化計算が必要である．特に，大規模問題に対しては，問題の特殊構造を使った最適化手法が不可欠であり，数値的最適化研究の成果の貢献が期待されている分野であろう．次節では，近年提案されたいくつかの最適化手法について述べる．

3. SVMに対する最適化手法

まず初めに，前節で導入したSVMの定式化(2.7)あるいは(2.8)に対する最適化アルゴリズムを考える．ほとんどのアルゴリズムでは，双対問題に対して最適化を行う．$\alpha\in\mathbb{R}^M$を双対変数，またM次の正方行列$Q=YAA^TY$を定めれば，問題(2.8)の双対問題は次の凸2次最小化問題

$$(3.1)\quad \left|\begin{array}{ll}\text{最小化} & W(\alpha)=\frac{1}{2}\alpha^TQ\alpha-e^T\alpha\\ \text{制約} & y^T\alpha=0,\quad 0\leq\alpha\leq eC\end{array}\right.$$

となる．一方，問題(2.7)の場合では，双対問題は

$$(3.2)\quad \left|\begin{array}{ll} \text{最小化} & \frac{1}{2}\alpha^T(Q+C^{-1}I)\alpha-e^T\alpha \\ \text{制約} & y^T\alpha=0,\quad 0\leq\alpha \end{array}\right.$$

と書くことができる．ただし，I は単位行列を表す．

問題 (3.1) および (3.2) ともに，Karush-Kuhn-Tucker (KKT) 条件より次の関係を使い双対問題の解より主問題の解を得ることが可能である．すなわち，双対問題の最適解 α^* と主問題の最適解 (w^*, b^*) の間には $w^*=A^TY\alpha^*$ という関係がある．また，b^* は双対問題の制約式 $y^T\alpha=0$ に対応した主問題の変数である．問題 (2.8) と (3.1) の場合では，$\alpha_j^*>0$ となる任意の添え字 j に対し，$b^*=y_j-A_j^Tw^*$ という関係があり，問題 (2.7) と (3.2) の場合では，やはり $\alpha_j^*>0$ となる任意の添え字 j に対し，$b^*=y_j-A_j^Tw^*-C^{-1}\alpha_j^*$ という関係がある．

さて，主問題 (2.7)，(2.8) また双対問題 (3.1)，(3.2) など今まで述べてきた多くの問題は，いずれも凸2次計画問題であることから，一般的には内点法 [11] などの高速なアルゴリズムを用いて最適化可能であると考えられる．しかし，データマイニングに見られる多くの問題では，データの次元 (N) はさほど大きくないものの，データの数 (M) が極めて大きな場合を扱わなくてはならない．例えば双対問題 (3.1) では，$M\times M$ の大型な行列 AA^T を扱わなくてはならない．さらに，この後の節4で導入する非線形判別を行う場合には，AA^T の部分がカーネル行列と呼ばれるものに置き換わり，その結果，行列 Q はほぼ完全に稠密となってしまう．M が数万を超えるような状況では，パソコンのような計算機では，Q を主記憶に配列として保持することは不可能である．そこで，双対問題の制約式が一本のみであることや，最適解では多くの変数が0といった特徴を用いたアルゴリズムが提案され，M が数万を超える規模の問題まで扱えるようになってきている．また，これらのアルゴリズムを実装した多くのソフトウエアパッケージも提供されており，SVM^{light} [7]，$SVMTorch$ [3] あるいは $LIBSVM$ [2] などがよく用いられているようである．

3.1 working set を用いた分割法

この節では，まず問題 (3.1) に対するアルゴリズムを考える．現在のところ，問題を小さな部分問題に分割して解く [3, 7] 等の working set を使ったアプローチが大規模問題に対して効率的なアルゴリズムを与えている．この方法は，変数の添え字集合を二つの集合 B，R に分割し，R に属する変数は適当な値に固定し，B の変数のみの部分問題を繰り返し解くものである．ここで最も重要な点は，working set と呼ばれる添え字集合 B の構成方法である．

今，一般的にアルゴリズムの k 回目の繰り返しにおいて，問題 (3.1) のある実行可能な解 α^k が得られているとする．このとき，できるだけ目的関数 $W(\alpha)$ の減少が大きくなるように q 個の変数を選び出し B を構成することが望ましい．これには，関数 $W(\alpha)$ の $\alpha=\alpha^k$ での最急降下方向 $-\nabla W(\alpha^k)=e-Q\alpha^k$ を求め，このベクトルの絶対値の大きな要素に対応する添え字で B を構成するとよいと思われる．そこで，現在の点 α^k を $d\in\mathbb{R}^M$ 方向に更新すると考え，d を変数とした以下の最適化問題

$$\left|\begin{array}{ll} \text{最大化} & (e-Q\alpha^k)^Td \\ \text{制約} & y^Td=0,\quad -e\leq d\leq e, \\ & d_i\geq 0 \text{ if } \alpha_i^k=0,\quad d_i\leq 0 \text{ if } \alpha_i^k=C, \\ & |\{d_i|d_i\neq 0\}|\leq q \end{array}\right.$$

を解く．ただし，d には現在の点 α^k から d 方向に点を更新しても問題 (3.1) の実行可能領域をはみ出すことがないように，制約が付けてある．この問題は制約領域の端点が全て整数となっている連続ナップザック問題である．よって，ベクトル $e-Q\alpha^k\in\mathbb{R}^M$ の要素を適切にソートすれば，整数の最適解 $d^*\in\{\pm1, 0\}^M$ を求めることが可能である．d^* を用いて working set を $B=\{i|d_i^*=\pm1\}$ と構成する．

B，R を使って変数を

$$\alpha=\begin{pmatrix}\alpha_B\\ \alpha_R\end{pmatrix},\quad y=\begin{pmatrix}y_B\\ y_R\end{pmatrix},$$

$$Q=\begin{bmatrix}Q_{BB} & Q_{BR}\\ Q_{RB} & Q_{RR}\end{bmatrix}$$

と分割すれば，部分問題は

$$(3.3)\quad \left|\begin{array}{ll} \text{最小化} & \frac{1}{2}\alpha_B^TQ_{BB}\alpha_B-(e-Q_{BR}\alpha_R^k)^T\alpha_B \\ \text{制約} & y_B^T\alpha_B=-y_R^T\alpha_R^k,\quad 0\leq\alpha_B\leq eC \end{array}\right.$$

となり，この最適解を $k+1$ 回目の解 α^{k+1} とする．この部分問題は q の大きさが十分小さければ，内点法 [11] などを使い極めて高速に最適化が可能である．なお，M が数千を超えるような問題に対しても，q は数十程度でよいパフォーマンスが得られることが報告されている [8]．

さて，以上の枠組みを実装するにあたっては，次の点に注意が必要であろう．まず，q が M に比べて非常に小さいことから，問題 (3.3) の計算の手間はほとんど無視することができるので，1回の繰り返しの

中で最も手間を必要とする部分は$Q\alpha^k$の計算となる．特に，Qを主記憶に配列で保持できない場合には，Qの$i-j$要素も毎回A_iとA_jから計算しなくてはならない．そこで，部分問題で変更されたのはq個の変数のみであることに注目し，まず$\Delta\alpha=\alpha^{k+1}-\alpha^k$を求めてから，$Q\alpha^{k+1}=Q\alpha^k+Q\Delta\alpha$とすべきである．$\Delta\alpha$の非ゼロ要素がたかだか$2q$個であるから，$Q\Delta\alpha$の計算は，$Q$のたかだか$2q$列のみを使うだけでよく，$Q$が保持されていない場合でも十分に高速に計算可能である．

なお，このアルゴリズムには，最適解での0の要素の減少に伴い効率が低下すること，また，高い精度で最適解を得ることが困難であるなど問題点もあるが，実用的観点からは効率的な手法であると考えられる．

3.2 Lagrangian SVM

現実の問題の中には，Mは大きいものの，それに比べてNが小さな問題も存在する．このような場合には双対問題 (3.1) や (3.2) を解くことが必ずしも得策ではない．ここでは，データの次元Nがデータ数Mに比べ非常に小さい場合有効であると考えられるLagrangian Support Vector Machine (LSVM) [9]について簡単に述べる．

まず，定式化 (2.7) をさらに変形させ，目的関数にb^2を加えた次の問題

$$(3.4)\quad \left|\begin{array}{ll} \text{最小化} & 1/2\|w\|_2^2+b^2+C\|\xi\|^2 \\ \text{制約} & \xi\geq e-YAw+yb \end{array}\right.$$

を考える．この場合，目的関数はstrictlyに凸となり，また双対問題は，

$$(3.5)\quad \left|\begin{array}{ll} \text{最小化} & \frac{1}{2}\alpha^T(YAA^TY+C^{-1}I+ee^T)\alpha-e^T\alpha \\ \text{制約} & \alpha\geq 0 \end{array}\right.$$

と凸2次関数を非負象限で最適化する非常に単純な問題となる．以降簡単のため，M行$N+1$列の行列を$H=Y[A-e]$，また$\bar{Q}=HH^T+C^{-1}I$と置き，問題 (3.5) を

$$\text{最小化}\left\{\frac{1}{2}\alpha^T\bar{Q}\alpha-e^T\alpha|\alpha\geq 0\right\}$$

と書くことにする．このときKKT条件から，次の反復式

$$(3.6)\quad \alpha^{k+1}=\bar{Q}^{-1}(e+\max\{0,\bar{Q}\alpha^k-e-\alpha^k\mu\})$$

を導くことができ，実数μを適当に定めれば，点列$\{\alpha^k\}$は最適解α^*に収束することが示されている[9]．この方法の最大のポイントは，反復を行うためには$\bar{Q}=HH^T+C^{-1}I$の逆行列を1回計算すればよい点である．さらに，

$$(3.7)\quad \bar{Q}^{-1}=C\left(I-H\left(H^TH+\frac{I}{C}\right)^{-1}H^T\right)$$

となる関係を使えば，$\bar{Q}^{-1}$の計算はN次行列$H^TH+C^{-1}I$の逆行列の計算に帰着し，データ数Mが大きくても$\bar{Q}^{-1}$を算出することが可能である．また，反復計算 (3.6) の実行には行列Hを主記憶に保持すればよく，Hの疎性（もしあれば）も利用可能である．

この節で述べた二つのアルゴリズム以外にも，Nが小さい状況ならば，Mが相当に大きな場合でも，例えば，線形計画問題 (2.9) であれば，標準的な単体法や内点法で解くことも可能である．また，$Q=(YA)(YA)^T$と分解できることを使えば，2次計画問題 (3.1) や (3.2) を内点法[4]で解くことは，線形計画問題 (2.9) を解く場合と計算上大きな差はなく，大規模問題にも十分適応可能である

4. 非線形な判別関数の構成

4.1 カーネル関数

SVMが多くの問題に対して高い判別力を示すことができるのは，双対問題 (3.1) などを使い，非線形な判別関数が構成できる点にある．そこで，この節ではカーネル関数を用いたSVMによる非線形判別について述べる．

まず適当な非線形写像$\phi:\mathbb{R}^N\to\mathcal{F}$を使い各データ$A_j$をより高い次元の空間$\mathcal{F}$へ非線形に変換する．以降では，特にこの空間$\mathcal{F}$を特徴空間と呼ぶことにする．非線形SVMの基本的な考え方は，この変換された$\mathcal{F}$の点$\phi(A_1),\phi(A_2),\cdots,\phi(A_M)$に対して問題 (2.7) や (2.8) を適用し，$\mathcal{F}$上での線形判別関数を求めることによって，非線形な判別を実現しようとするものである．

簡単な非線形変換の例を示す．今，$N=2$属性からなるデータ$x=(x_1,x_2)$から，属性の積を新たな属性とする変換を考え，写像ϕを

$$(4.1)\quad \phi:(x_1,x_2)\mapsto(x_1^2,x_2^2,\sqrt{2}x_1x_2)$$

と3属性のデータに変換するものとする．そこで，$\mathcal{F}=\mathbb{R}^3$の空間で$w_1x_1^2+w_2x_2^2+w_3x_1x_2-b$と線形な判別関数を構成すれば，元の$\mathbb{R}^2$の空間では2次関数となり，結果的に非線形の判別関数が得られることとなる．一般に，d個の積を要素とする変換を行えばd次多項式の判別関数を求めることが可能である．

一方，このようにN属性のデータに対してd個の積を考えた場合，変換後の次元は${}_{N+d-1}C_d$と指数的に増大してしまう．そのため，例えば回帰分析などで同様なことを行うと，$d=2$程度でも過学習となり，予測精度の低下を引き起こしてしまう．しかし，SVMではwのノルムをコントロールしているため，必要な属性のみによる予測精度の高い判別関数を構成することが可能である．一方で，問題(2.7)や(2.8)は極めて大規模な最適化問題となり，したがって双対問題が重要な役割を果たすこととなる．

今，変換後の点$\phi(A_i)$と$\phi(A_j)$の内積の値を$i-j$要素とするM次正方行列を$\mathcal{K}$とすれば，変換された点に対する1-norm SVMの双対問題は，(3.1)の目的関数を$\alpha^T Y\mathcal{K}Y\alpha-e^T\alpha$としたものにほかならない．すなわち，双対問題を定めるためには，$\phi(A_i)$と$\phi(A_j)$の内積の値のみが必要である．そこで，SVMではカーネル関数と呼ばれる特殊な関数を用いA_i, $A_j\in\mathbb{R}^N$から直接$\phi(A_i)$と$\phi(A_j)$の内積を求め双対問題を定めている．例えば，$\mathbb{R}^N$の元x, zに対し，カーネル関数として$\mathcal{K}(x,z)=(x^Tz)^2$を用いたとする．このとき，

$$(x^Tz)^2=\sum_{i=1}^{N}x_i^2z_i^2+\sum_{i<j}(\sqrt{2}\,x_ix_j)(\sqrt{2}\,z_iz_j)$$

となることより，$\mathcal{K}$は，(4.1)で示した写像をN変数に拡張した変換

$$\phi:(x_1,x_2,\cdots,x_n)\mapsto$$
$$(x_1^2,x_2^2,\cdots,x_n^2,\sqrt{2}x_1x_2,\sqrt{2}x_1x_3,\cdots,\sqrt{2}x_{n-1}x_n)$$

の内積を与えていることがわかる．カーネル関数を使えば特徴空間の点$\phi(x)$を構成することなく内積の計算が可能となるのである．

これ以外にも，カーネル関数には様々なものが知られている．$\mathcal{K}(x,z)=(x^Tz+1)^d$となるカーネル関数をpolynomial kernelと呼び，これを用いれば，d次以下の多項式の判別関数が構成可能である．属性の積の組合せが指数的に増加しても，その内積はカーネル関数により元の属性数に比例する計算量で算出が可能である．その他にもSVMでよく用いられる代表的なカーネル関数として，RBF kernel $\mathcal{K}(x,x')=\exp(-\|x-x'\|^2/\sigma^2)$，やsigmoid kernel $\mathcal{K}(x,x')=\tanh(\varkappa x^Tx'-\Theta)$(ただし$d$は自然数のパラメータ，$\sigma$, $\varkappa$, Θは実数のパラメータである)などがあり，様々な分野へ応用され有効であると考えられている．

以上述べたように，SVMで非線形の判別関数を構成する際には，稠密で大規模行列$\mathcal{K}$を持った双対問題(3.1)や(3.2)の最適化が必要となり，節3で述べたようなアルゴリズムの研究が行われている．一方，LSVMにおいてカーネルを使った非線形判別を行う場合には，(3.7)のような分解ができず，$\bar{Q}^{-1}$を陽に保持して(3.6)で点を更新しなくてはならない．次節では，大規模行列$\mathcal{K}$を用いないで，$\phi(A_j)$を低い次元の実ベクトルとして近似的に表現することによる，判別アルゴリズムを考える．

4.2 特徴空間の近似表現

まず行列$\mathcal{K}$と非線形写像ϕとの関係について考える．実数$\lambda_1\geq\lambda_2\geq\cdots\geq\lambda_M\geq0$を$\mathcal{K}$の固有値，また$p_1$, $p_2,\cdots,p_M\in\mathbb{R}^M$を対応した固有ベクトルで長さ1に正規化されているものとする．このとき，0でない固有値の中から，大きなもの$S(<M)$個とそれに対応する固有ベクトルを使い，M行S列の行列

$$D_S=[\sqrt{\lambda_1}\,p_1\sqrt{\lambda_2}\,p_2\cdots\sqrt{\lambda_S}\,p_S]$$

を定める．行列$D_SD_S^T$はランクがSの行列でFrobenius normの意味で行列$\mathcal{K}$の最良の近似となっている．

D_Sの各要素と空間$\mathcal{F}$の間には次のような関係を導くことができる．行列D_Sの$j-k$要素をd_{jk}とし，次のようにして定まる$\mathcal{F}$の元

$$(4.2)\quad \mathcal{V}_k=\frac{\sum_{j=1}^{M}d_{jk}\phi(A_j)}{\lambda_k},\quad k=1,2,\cdots,S$$

を考える．今，行列D_Sの第k列ベクトルを$d_k\in\mathbb{R}^M$と記し，また特徴空間$\mathcal{F}$の内積を$\langle\cdot,\cdot\rangle$と書くこととする．このとき，簡単な計算により，任意の$k, k'=1,2,\cdots,S$に対して$\langle\mathcal{V}_k,\mathcal{V}_{k'}\rangle=\frac{1}{\lambda_k\lambda_{k'}}d_k^T\mathcal{K}d_{k'}$と変形できる．もし$k\neq k'$ならば明らかに$d_k^T\mathcal{K}d_k=0$となるので，$\langle\mathcal{V}_k,\mathcal{V}_{k'}\rangle=0$であり，また$d_k^T\mathcal{K}d_k=\lambda_k\|d_k\|^2=\lambda_k^2$より$\langle\mathcal{V}_k,\mathcal{V}_k\rangle=1$を得る．ゆえに，

$$(4.3)\quad \mathcal{V}=\{\mathcal{V}_1,\mathcal{V}_2,\cdots,\mathcal{V}_S\}$$

は特徴空間$\mathcal{F}$の正規直交基底となる．ここで，$\mathcal{V}$で張られる$\mathcal{F}$の部分空間を$\mathcal{F}_S$と表す．このとき，任意の点A_j, $j=1,2,\cdots,M$に対してベクトル$\phi(A_j)$の$\mathcal{F}_S$への射影は

$$(4.4)\quad \langle\phi(A_j),\mathcal{V}_1\rangle\mathcal{V}_1+\cdots+\langle\phi(A_j),\mathcal{V}_S\rangle\mathcal{V}_S$$

と表現することができる．また，ベクトル(4.4)の基底$\mathcal{V}$に関する座標ベクトル

$$(\langle\phi(A_j),\mathcal{V}_1\rangle\langle\phi(A_j),\mathcal{V}_2\rangle\cdots\langle\phi(A_j),\mathcal{V}_S\rangle)$$

はD_Sの第j行ベクトルにほかならない．そこで，このS次ベクトルを$\phi(A_j)$の近似と考え，線形判別を行うことを考える．

さらに一般的に，任意の点 $x\in\mathbb{R}^N$ に対し，非線形写像 $\phi(x)$ の基底 $\mathcal{V}$ に関する S 次元の座標ベクトルを $[x]_{\mathcal{V}}$，すなわち

$$[x]_{\mathcal{V}}=(\langle\phi(x),\mathcal{V}_1\rangle\langle\phi(x),\mathcal{V}_2\rangle\cdots\langle\phi(x),\mathcal{V}_S\rangle)^T\in\mathbb{R}^S$$

と記すことにすれば，$[x]_{\mathcal{V}}$ の第 k 成分は

$$\langle\phi(x),\mathcal{V}_k\rangle=\left\langle\phi(x),\frac{\sum_{j=1}^{M}d_{jk}\phi(A_j)}{\lambda_k}\right\rangle$$

$$=\frac{\sum_{j=1}^{M}d_{jk}\mathcal{K}(x,A_j)}{\lambda_k}\tag{4.5}$$

と関数 $\phi(x)$ を陽に定めることなく，カーネル関数 $\mathcal{K}(x,A_j)$ だけから求めることができる．

以上のように，行列 $\mathcal{K}$ の固有値と固有ベクトルを使えば，$\mathcal{F}$ の点 $\phi(A_j)$ を S 次元の実ベクトル $[A_j]_{\mathcal{V}}$ として近似的に表現することが可能なり，この S 次元空間で線形判別を求めることにより，結果的にもとの N 次元空間での非線形判別を行うことが可能となる．筆者等は，データとして A の代わりに D_S を用い，S 次元空間での線形判別を線形計画問題 (2.9) を解いて求めたところ，通常のSVMによる非線形判別に近い判別力のある関数を，効率よく構成できることを確認している．また，LSVMによる定式化を使うのであれば，(3.7) の最終項の逆行列部分が

$$\left(H^TH+\frac{I}{C}\right)^{-1}=\left(\begin{bmatrix}D_S^TD_S & -D_S^Te\\ -e^TD_S & M\end{bmatrix}+\frac{I}{C}\right)^{-1}$$

となるが，固有ベクトルの性質より $D_S^TD_S=I$ であることを使えば，逆行列の計算も容易に行うことができる．

5. おわりに

本稿では，SVMのいくつかのバリエーションをその定式化とともに紹介した．標準的に広く用いられているSVMでは，カーネルを用いた非線形判別を行うため，2次の双対問題 (2.7) や (2.8) が導入された．しかし，線形判別を行うのであれば，必ずしもこの2次計画問題を解く必要はなく，より単純な問題 (3.5)，あるいは線形計画問題 (2.9) でも十分に効率のよい判別関数を構成することが可能である．さらに，上で説明したように特徴空間の点を低い次元に近似的に表現することを行えば，LSVMや線形計画法によっても非線形な判別関数が構成可能である．

本稿では，2クラスの判別問題のみを取り上げたが，3クラス以上の多クラスの分類問題に対しても2クラスの場合の考え方を拡張したM-SVM[1]と呼ばれる手法も提案されている．この定式化でも，双対問題が凸2次計画問題へと帰着され，カーネル関数を使った非線形な多クラス判別が可能である．しかし，M-SVMで使われる2次計画問題は，その構造が (2.8) などと比べればより複雑であり，本稿で述べたような特殊アルゴリズムの構築は望めないであろう．しかし，節4で述べた方法で，特徴空間 $\mathcal{F}$ の点を S 次元の点で近似的に表現し，線形判別の問題へと帰着させてしまえば，問題 (2.9) とほぼ同様な線形計画問題[13]により非線形な多クラスの判別を行うことも可能である．

参考文献

[1] E. J. Bredensteiner and K. P. Bennett, *Multicategory classification by support vector machines*, Computational Optimization and Applications, 12, 1999, pp. 53-79.

[2] C.-C. Chang and C.-J. Lin, *LIBSVM : a library for support vector machines*, 2001. Software available at http://www.csie.ntu.edu.tw/~cjlin/libsvm/

[3] R. Collobert and S. Bengio, *SVMTorch : Support vector machines for large-scale regression problems*, Journal of Machine Learning Research, 1, 2001, pp. 143-160.

[4] M. C. Ferris and T. S. Munson, *Interior point methods for massive support vector machines*, Technical Report 00-05, Computer Science Department, University Wisconsin, 2000.

[5] T. Hastie, R. Tibshirani, and J. Friedman, *The elements of statistical learning*, Springer Series in Statistics, Springer-Verlag, New York, 2001. Data mining, inference, and prediction.

[6] A. E. Hoerl and R. W. Kennard, *Ridge regression : Biased estimation for nonorthgonal problems*, Technometrics, 12, 1970, pp. 55-67.

[7] T. Joachims, Making large-scale support vector machine learning practical, in Advances in Kernel Methods, B. Schölkopf, C. Burges, and A. Smola, eds., The MIT Press, 1999, pp. 169-184.

[8] T. Joachims, *Learning to Classify Text Using Support Vector Machines*, Kluwer Academic Publishers, Boston, 2002.

[9] O. L. Mangasarian and D. R. Musicant, *Lagrangian support vector machines*, J. Mach. Learn. Res., 1, 2001, pp. 161-177.

[10] R. Tibshirani, *Regression shrinkage and selection via the lasso*, J. Roy. Statist. Soc. Ser. B, 58, 1996, pp. 267-288.

[11] R. J. Vanderbei, *LOQO : An interior point code for quadratic programming*, Optimization Methods and Software, 11, 1999, pp. 451-484.

[12] V. N. Vapnik, *The nature of statistical learning theory*, Statistics for Engineering and Information Science, Springer-Verlag, New York, 2000.

[13] Y. Yajima, *Linear programming approaches for multicategory support vector machines*, Technical Report 2002-6, Department of Industrial Engineering and Management, Tokyo Institute of Technology, 2002.