サポートベクターマシンによる統計的判別

2003MM120 山田 俊哉
指導教員: 田中 豊

1 はじめに

2クラス分類問題を解く方法において、Vladmir N Vapnik等により提唱されたサポートベクターマシンという学習機械が提唱された。今回はサポートベクターマシンの理論への理解を深めると同時に、統計解析フリーソフトRへの実装を目的とし、サポートベクターマシンを実行する新たなプログラムを作成する。

2 サポートベクターマシン

2.1 導入

サポートベクターマシン(Support Vecter Machine. SVM) は2クラスの分類問題を解くために作られた学習機械(学習アルゴリズム)である。SVMが優れている理由に、クラス分類を行う識別面を一意的に決定するためのに「マージン最大化」と言う明確な基準が設けられている点と、カーネル学習法により非線形の判別問題への拡張することができる、と言う2点が挙げられる [1]。ここで「マージン」とは各群の先端の学習データ同士のユークリッド距離である。

2.2 線形分離可能な学習データに関するSVM

学習データの集合が以下のように与えられたとする。

$$(\mathbf{x}_1, t_1), (\mathbf{x}_2, t_2), \cdots, (\mathbf{x}_n, t_n) \tag{1}$$

ここで$\mathbf{x}_i = (x_1, x_2, \cdots, x_n)^T$ は個体の特徴ベクトル$t_i \in \{-1, 1\}$はクラスラベルである。この学習データの入力に対し、SVMは次の識別関数を持つ

$$f(x) = sign\big(\mathbf{w}^T\mathbf{x} - b\big) \tag{2}$$

ここで、$\mathbf{w}$とbは識別面$g(x)$を決定しているパラメータ。$sign(y)$関数は$y > 0$のとき1をとり$y \leq 0$のとき-1をとる符号関数である。学習データが線形分離可能である場合$H1 : \mathbf{w}^T\mathbf{x} - b = -1$と$H2 : \mathbf{w}^T\mathbf{x} - b = 1$の2枚の超平面で学習データが完全に分離されていることを示す。よって線形分離可能な場合マージンは$\gamma = 1/\|\mathbf{w}\|$となる。このマージンを最大化する問題を計算上簡単な形にすると線形分離可能な場合SVMは以下のような最適化問題になる。

$$\min_w \frac{1}{2}\|\mathbf{w}\|^2 \tag{3}$$

$$s.t. \qquad t_i(\mathbf{w}^T\mathbf{x}_i - b) \geq 1 \ i = 1, \ldots, n \tag{4}$$

この問題を今回は双対問題に帰着して解く。まず、Lagrange乗数$\boldsymbol{\lambda} = (\lambda_1, \cdots, \lambda_n)$を導入し各パラメータの偏微分を0とした条件式を用いると、最適化問題は以下のようになる

$$\max_\lambda L_D(\boldsymbol{\lambda}) = \sum_{i=1}^n \lambda_i - \sum_{i=1}^n \sum_{j=1}^n \lambda_i\lambda_j t_i t_j \mathbf{x}_i^T\mathbf{x}_j \tag{5}$$

$$s.t. \sum_{i=1}^n \lambda_i t_i = 0, \quad \lambda_i \geq 0, \, i = 1, \cdots, n \tag{6}$$

$\boldsymbol{\lambda}$のみに関する最大化問題になる。

2.3 サポートベクター

前節で求めたλ_i*において$\lambda_i* = 0$に対応する学習データ$\mathbf{x}_i$はパラメータ$\mathbf{w}$の決定に関与していない。つまり、全ての学習データの中で$\lambda_i* > 0$となる一部の学習データ(ベクトル) を「Support Vector」と呼び、SVMの名前の由来にもなっている。また各クラスに属するサポートベクターを$\mathbf{x}_s^+$、$\mathbf{x}_s^-$とおくと最適なbは以下の式より求められる。

$$b = -\frac{\min_{t_i=1}(\mathbf{w}^T\mathbf{x}_s^+) + \max_{t_i=-1}(\mathbf{w}^T\mathbf{x}_s^-)}{2} \tag{7}$$

2.4 線形分離不可能なデータへの拡張

線形分離可能な学習データは自然界には珍しく実用に向かず、実際は非線形で複雑な識別面を持つ場合が多い。それに対応する考えられる方法は「ソフトマージン法」と「カーネル法」である [1]。

2.4.1 ソフトマージン法

ソフトマージン法では、マージン$1/\|\mathbf{w}\|$を最大化しながら、識別面の反対側に入る事を許す。反対側にどれくらい入り込んだかの距離を、$\xi_i(\geq 0)$を用いて、$\xi_i/\|\mathbf{w}\|$とあらわす。反対側に入り込んだ距離の和は最小が望ましいため最適な識別面は

$$\min_{w,\xi_i} \frac{1}{2}\|\mathbf{w}\|^2 + \alpha\sum_{i=1}^n \frac{\xi_i}{\|\mathbf{w}\|} \tag{8}$$

$$s.t. \qquad t_i(\mathbf{w}^T\mathbf{x}_i - b) \geq 1 - \xi_i \ i = 1, \cdots, n \tag{9}$$

のうような最適化問題になる。ここでパラメータαはマージンの大きさの項とはみ出しの距離の項とのバランスを調整する重みのパラメータである。これによって以下のような最適化問題が得られる。

$$\max_\lambda L_D(\boldsymbol{\lambda}) = \sum_{i=1}^n \lambda_i - \sum_{i=1}^n \sum_{j=1}^n \lambda_i\lambda_j t_i t_j \mathbf{x}_i^T\mathbf{x}_j \tag{10}$$

$$s.t. \qquad \sum_{i=1}^n \lambda_i t_i = 0, 0 \leq \lambda_i \leq \alpha, \, i = 1, \cdots, n \tag{11}$$

2.4.2 カーネル法

非線形で複雑な識別面に対応する方法として、特徴ベクトルを非線形の写像を行うことにより、線形分離可能性の高い高次元の状態にし識別する方法である。元の特徴ベクトル$\mathbf{x_i}$を非線形の写像$\phi(\mathbf{x_i})$によって変換すると、元々、式(5) は入力データの内積に依存しているため、非線形に写像した$\phi(\mathbf{x_i})$と$\phi(\mathbf{x_j})$の内積が$\phi(\mathbf{x_i})^T\phi(\mathbf{x_j}) = K(\mathbf{x_i}, \mathbf{x_j})$のように元の特徴ベクトルからカーネルと呼ばれる$K(\mathbf{x_i}, \mathbf{x_j})$が計算できれば高次元空間で$\phi(\mathbf{x_i})^T\phi(\mathbf{x_j})$を計算しなくても良い。このカーネルトリックを用いると目的関数$L_D(\boldsymbol{\lambda})$と識別関数$f(x)$は

$$\max_{\lambda} L_D(\boldsymbol{\lambda}) = \sum_{i=1}^{n} \lambda_i - \sum_{i=1}^{n}\sum_{j=1}^{n} \lambda_i \lambda_j t_i t_j K(\mathbf{x}_i, \mathbf{x}_j) \quad (12)$$

$$f(x) = sign((\sum_{i=1}^{n} \lambda_i t_i K(\mathbf{x}_i, \mathbf{x}_j) - b) \quad (13)$$

となる。カーネルトリックにより識別面のパラメータbを求める。wは求めないで識別関数が計算できる。

3 統計ソフトRへの実装

3.1 アルゴリズム

前述の最適化問題を解く為の手法には目的関数がどれもが凸関数である事を利用し最急降下法の一つである慣性法を用いる。慣性法を用いた場合、$\boldsymbol{\lambda}$の更新式は以下のようになる。

$$\lambda_i(k+1) = \lambda_i(k) + \eta\left(-(1-\omega)\eta\frac{\partial L_D(\lambda)}{\partial \lambda_i} + \omega D_{k-1}\right) \quad (14)$$

3.2 Rプログラミング

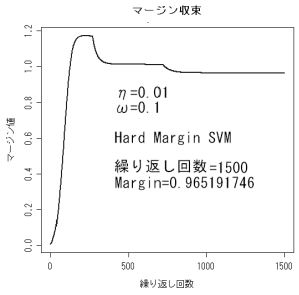
統計ソフトRで実際に動くプログラミングを作成した。実際にSVMが判別する様子を探るため、Rに収録されている”iris”データを用いて判別した。

3.2.1 線形分離可能な場合

線形分離可能な場合irisデータの2種(setosaとversicolor)を判別対象とし、各群5例ずつ学習データを抽出した。$\eta = 0.01$,$\omega = 0.1$とし1500回計算を行った場合、求められた識別面のパラメータは

$$\mathbf{w}* = (-0.1613600, -0.5132494, 0.8025875, 0.3739237)^T$$
$$b = -3.516565^{\times}10^{-10} \quad (15)$$

となり、これをirisデータの上記2種100例に適用したところ完全に判別できた。マージンの収束状況は横のようになり、サポートベクター（以下sv）に近い学習データがsvから外れるときマージンの値が減少している様子が見て取れる。

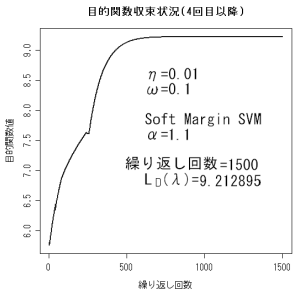


3.2.2 ソフトマージンを用いた場合

ソフトマージンを用いてSVMを実行する場合、判別対象をirisデータの2種(versicolorとvirginica)とし、各群5例ずつ学習データを抽出した。$\eta = 0.01$,$\omega = 0.1$,$\alpha = 0.5$とし1500回計算を行った場合、求められた識別面のパラメータは

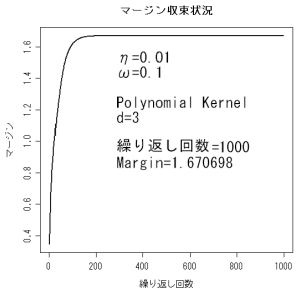
$$\mathbf{w}* = (-0.9325721, -0.4181110, 1.1666994, 0.7041798)^T$$
$$b = 0.01093106 \quad (16)$$

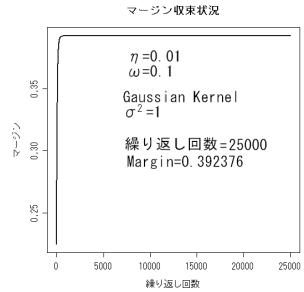
となり、これを用いるとirisデータの上記2種を95% で判別した。目的関数の収束状況だが計算の始め4回程度は非常に大きな値より急激に減少するため右図には4回目以降の目的関数の収束状況を示すが。右図において一度いびつな収束が見られるがこれは3つのλがこの時期に立て続けに収束した影響である事が解っている。



3.2.3 カーネル法を用いた場合

カーネル法を用いる場合、判別対象をiris2種(versicolorとvirginica)を標準化したデータを用い、各群5例ずつ学習データを抽出した。カーネルは2種類、多項カーネル、Gaussカーネルを用いた。以下には多項カーネル(図左)を用いた場合のマージンの収束状況、Gaussカーネルを用いた場合の(図右)を示す。





多項カーネルは次数3計算回数1000回、Gaussカーネルは分散1、計算回数25000回として両カーネルともに$\eta = 0.01$,$\omega = 0.1$で計算した。どちらもマージンは収束しているが、その収束速度には大きな違いがあり、多項カーネルを用いた場合、誤判別が2% Gaussカーネルを用いた場合3%出たため、データの構造によってカーネルの取り方を考える必要があることがわかる。

4 おわりに

本研究ではSVMの基本的な理論と、プログラム作成データへの応用を試みた。プログラムの学習精度としては既存の判別分析と大差が無いような判別結果となってしまった。しかし、今回プログラムを実行するにあたり、ソフトマージンのパラメータ、カーネルのパラメータ、学習係数を恣意的に設定した。実際にどのパラメータが最適であるかを今回は実験的に探っていったため判別性能は今ひとつ低いものとなってしまったが、このパラメータの取り方や学習データの選出などの課題も浮き彫りになったと思われる。今後は本研究で示したSVMの可能性を広げつつ新たな判別手法として用いることができると思われる。

参考文献

[1] Nello Cristianini,John Shawe-Taylor,大北剛　訳：サポートベクターマシン入門, 共立出版(2005)