

タイトル	矢印黙読時における脳波のサポート・ベクター・マシンによる特徴選択
著者	角森, 史浩; 山ノ井, 高洋; 高柳, 浩; KAKUMORI, Fumihiro; YAMANOI, Takahiro; TAKAYANAGI, Hiroshi
引用	北海学園大学工学部研究報告(38): 141-153
発行日	2011-02-14

矢印黙読時における脳波の サポート・ベクター・マシンによる特徴選択

角 森 史 浩*・山ノ井 高 洋**・高 柳 浩***

Feature Selection by Use of Support Vector Machine for EEG Data on Silent Reading of Arrows

Fumihiko KAKUMORI*, Takahiro YAMANOI** and Hiroshi TAKAYANAGI***

Abstract

To develop a brain computer interface the authors have investigated the brain activity during human recognition of arrows. The authors have recorded electroencephalograms (EEGs) from subjects in reading silently four types of arrows presented on CRT. Each of arrows has directions of upward, downward, leftward or rightward. The support vector machine is applied to discriminate the single trial EEGs. Power spectols of nineteen channels of EEGs had been used to discriminate the four types of symbols. Results of the discriminant rates by use of the present system were more than 60%. These results are fairly good rates to the single trial EEG data.

1 はじめに

ヒト脳機能に関する様々な研究によれば、視覚刺激に関する処理は後頭葉でなされ、高次処理の機能は左脳と右脳とで大まかには分化されていると言われる [1][2]。山ノ井らは先行研究 [3][4][5] において言語および空間認知に関する脳活動部位の推定を行い、詳細な時空間的脳活動のモデルを得た。さらに先行研究において逆の向きに直線移動する視覚刺激に対する脳活動の比較を行った結果、事象関連電位 (event-related potential: ERP) のピーク時の極性反転、および高次脳活動での前頭葉での脳活動を確認した [4]。さらに、山ノ井らは先行研究において、向きを示す単語と記号に関する脳内処理部位の推定および比較を行うため、被験者が視覚刺激として提示された向きを示す単語 (漢字: 上, 下, 左, 右) と記号 (矢印: ↑, ↓, ←, →) とを観察する際の脳波 (electro-encephalograms: EEGs) を計測し、等価電流

* (株)ジャパンテクニカルソフトウェア Japan Technical Software

** 北海学園大学工学部電子情報工学科 Department of Electronics and Information Engineering, Faculty of Engineering, Hokkai-Gakuen University

*** (株)情報科学センター Information Science Center (ISRC)

双極子推定 (equivalent current dipole source localization : ECDL) 法 [7][8] を試み、脳活動の時空間的な比較を行った [6]. この結果、潜時400ミリ秒前後で漢字と矢印とで共通した脳活動が前頭葉で観察された. さらに、これらの潜時で逆の向きを示す視覚刺激に対して得られたERPを比較した結果、ERPのピーク極性の反転を確認した. この事実に着目し、山ノ井らは、ピーク極性の反転により、被験者がイメージした矢印の種類を判別できると考え、ブレイン・コンピュータ・インタフェース (brain computer interface : BCI) への利用を試みた [7][8]. その結果、全被験者で80%を超える判別率を得た.

特に本研究の目的は (BCI) であるため、先行研究の知見を基に、シングルトライアルEEGに対してのさらなる判別率向上を目指し、本研究では、全てのチャンネルおよび周波数を用いて特徴選択を施し、シングルトライアルEEGに対する識別率を高めることを考えた. 全チャンネルを用いた特徴選択ならば、現状のEEGデータで算出し得る最高に近い識別率を得られると考えたからである. 矢印の黙読時のEEGの全チャンネルのパワースペクトルに対して識別と特徴選択を行なった. 識別方法として、現在最も汎化性能 (未知のデータに対する識別能力) の高い識別器といわれるサポート・ベクター・マシン (SVM : support vector machine) [9][10][11] を用いた. また、次元数が増加するほど汎用能力が悪くなるいわゆる「次元の呪い」を避けるために、特徴選択をすることにした. 特徴選択によりEEG計測に用いるチャンネルや周波数を絞れば今後のBCI開発にも役立つ. 本研究の目的は、全チャンネルのEEGに対してSVMを用いた特徴選択を行うことにより、矢印の黙読時のEEGから得られる最高の識別率を得ることである.

2 Forward Stepwise Selectionによる特徴選択

対象が k 個の特徴量 (属性) のベクトルで記述されている場合に機械学習の手法を適用するとしたとき、 k 個の特徴の全ては利用せず、その中で有用なものを選び出すことを特徴選択という. 特徴選択は“目的変数と無関係な特徴量を使わないことで予測精度を向上させること”と“学習された関数を定性的に解釈しやすくすること”の二つの目的で行われる. 本研究では前者の予測精度を向上させることの目的で特徴選択を行う.

特徴選択の実現手法はフィルタ法 (filter法) とラッパー法 (wrapper法) の二つに分けられる. フィルタ法は目的変数と各特徴量の情報ゲインなどの特徴の良さの基準を使って選択する手法である. ラッパー法は特徴量の部分集合を使って実際に学習アルゴリズムを適用し、交差検定などで求めた汎化誤差を最小にする特徴量の部分集合を選択する手法である. フィルタ法とラッパー法を比較すると、ラッパー法の方が予測精度は高い. しかしながら、ラッパー法は実際に学習アルゴリズムを適応する必要があるため計算量が膨大であり、フィルタ法の方がはるかに高速である.

本研究では特徴選択の方法としてラッパー法の一つであるForward Stepwise Selectionを用い

た. Forward Stepwise Selectionは, すべての特徴について一つずつ識別を試みて, 識別率の最も良い特徴を追加してゆく方法である.

3 サポート・ベクター・マシン (SVM)

SVMは, 2値分類問題($y \in \{-1, 1\}$)を解くために考案された学習アルゴリズムである. SVMは基本的には線形の識別器であるが, カーネル関数と凸二次計画法の最適化法との組み合わせにより, 非線形の識別器に拡張されている. さらに, SVMは非線形の識別を扱えるものの局所解の問題がないという利点がある.

3.1 線形サポート・ベクター・マシン

入力空間 $X \in R^n$ およびデータ集合 $\mathbf{x}_1, \dots, \mathbf{x}_l$ が与えられるとき, 線形SVMの識別関数は次式で表される.

$$f(\mathbf{x}) = \mathbf{w}^T \mathbf{x} + b \quad (3.1)$$

$$\text{sgn}(f(\mathbf{x})) = \begin{cases} 1, & f(\mathbf{x}) > 0 \\ -1, & \text{otherwise} \end{cases} \quad (3.2)$$

SVMは (3.2) 式によって, \mathbf{x} をクラス 1 とクラス -1 に分類する. (3.1) 式は, 自由度として係数 \mathbf{w} と非負値である b をパラメータとして有している. ここで, 係数 \mathbf{w} は線形識別器の重みベクトルと呼ばれ, 定数 b は非負値でバイアス項と呼ばれるパラメータである. SVMでは学習によって次式を満たす \mathbf{w} と b を求める必要がある.

$$\begin{aligned} & \text{minimise}_{\mathbf{w}, b} \quad \mathbf{w}^T \mathbf{w}, \\ & \text{subject to} \quad y_i (\mathbf{w}^T \mathbf{x}_i + b) \geq 1, \quad i = 1, \dots, l. \end{aligned} \quad (3.3)$$

ここで, $y_i \in \{-1, 1\}$ は \mathbf{x}_i に対応するクラスインデックスであり, 学習データである. この最適化問題は数理計画法の凸二次計画法として知られており, 双対問題に帰着して解くことができる. (3.1) 式を, ラグランジュ乗数 $\alpha_i \geq 0$ を導入して書き換えると,

$$f(\mathbf{x}) = \sum_{i=1}^l y_i \alpha_i \mathbf{x}_i^T \mathbf{x} + b \quad (3.4)$$

となり, (3.3) 式の最適化問題は,

$$\begin{aligned} & \text{maximise} \quad \mathbf{W}(\boldsymbol{\alpha}) = \sum_{i=1}^l \alpha_i - \frac{1}{2} \sum_{i,j=1}^l y_i y_j \alpha_i \alpha_j (\mathbf{x}_i^T \mathbf{x}_j), \\ & \text{subject to} \quad \sum_{i=1}^l y_i \alpha_i = 0, \alpha_i \geq 0, i = 1, \dots, l. \end{aligned} \quad (3.5)$$

と置き換えることができる.

3.2 線形ソフトマージン・サポート・ベクター・マシン

前述の線形SVMは、学習データを完全に分離できる超平面が存在することが前提となっている。しかし現実問題そのような超平面は存在しない場合が多い。ソフトマージンSVMは次式で表現される緩和変数（スラック変数）

$$\xi_i \geq 0, \quad i = 1, \dots, l$$

を導入して、(3.3) 式を満たさない学習データが存在してもよいようにするSVMである。

緩和変数 ξ_i を使って (3.3) 式を次式のように緩和することができる。

$$y_i (\mathbf{w}^T \mathbf{x}_i + b) \geq 1 - \xi_i, \quad i = 1, \dots, l, \quad (3.6)$$

このとき、解くべき最適化問題は

$$\text{minimise}_{\xi, \mathbf{w}, b} \quad \mathbf{w}^T \mathbf{w} + C \sum_{i=1}^l \xi_i^2,$$

$$\text{subject to } y_i (\mathbf{w}^T \mathbf{x}_i + b) \geq 1 - \xi_i, \quad i = 1, \dots, l,$$

$$\xi_i \geq 0, \quad i = 1, \dots, l. \quad (3.7)$$

となる。(3.7) 式の最適化問題を、ラグランジュ乗数 $\alpha_i \geq 0$ を導入して書き換えると、

$$\text{maximise } W(\boldsymbol{\alpha}) = \sum_{i=1}^l \alpha_i - \frac{1}{2} \sum_{i,j=1}^l y_i y_j \alpha_i \alpha_j (\mathbf{x}_i^T \mathbf{x}_j),$$

$$\text{subject to } \sum_{i=1}^l y_i \alpha_i = 0, C \geq \alpha_i \geq 0, i = 1, \dots, l. \quad (3.8)$$

となる。ここで、 C はマージンの大きさと学習データのはみ出しの程度のバランスを決める定数である。以降 C を便宜上マージン・パラメータと呼ぶ。マージン・パラメータの値はSVMの識別能力に大きく影響を与えることが知られており、事前に適切な値を設定しなければならない。

3.3 非線形ソフトマージン・サポート・ベクター・マシン

線形SVMを非線形SVMへ拡張する方法としてカーネルトリックと呼ばれる手法がある。線形サポート・ベクター・マシンによる識別の前処理として、入力ベクトル $\mathbf{x}_1, \dots, \mathbf{x}_l$ を次式のように高次元特徴空間に写像し、その特徴空間で識別を行うという方法を考える。

$$\Phi : \mathbf{x}_i \rightarrow \mathbf{z}_i \quad (3.9)$$

すなわち,

$$\Phi(\mathbf{x}_i) = \mathbf{z}_i$$

このとき, (3.4) 式は次式のようになる.

$$f(\mathbf{z}) = \sum_{i=1}^l y_i \alpha_i \mathbf{z}_i^T \mathbf{z} + b \quad (3.10)$$

また, (3.8) 式の最適化問題は, 次式のようになる.

$$\begin{aligned} \text{maximise } W(\boldsymbol{\alpha}) &= \sum_{i=1}^l \alpha_i - \frac{1}{2} \sum_{i,j=1}^l y_i y_j \alpha_i \alpha_j (\mathbf{z}_i^T \mathbf{z}_j), \\ \text{subject to } \sum_{i=1}^l y_i \alpha_i &= 0, C \geq \alpha_i \geq 0, i = 1, \dots, l. \end{aligned} \quad (3.11)$$

ただし,

$$\mathbf{z}_i^T \mathbf{z} = \Phi(\mathbf{x}_i)^T \Phi(\mathbf{x})$$

ここで,

$$K(\mathbf{x}_i, \mathbf{x}) = \Phi(\mathbf{x}_i)^T \Phi(\mathbf{x}) \quad (3.12)$$

となるような関数を導入する. この関数 K は一般にカーネル (kernel) 関数と呼ばれ, 高次元特徴空間へのベクトル \mathbf{x}_i の写像とベクトル \mathbf{x} の写像との内積を計算することに等しい.

カーネル関数として, 一般に以下の関数がよく使われる.

- ・ ガウシアンカーネル (Gaussian kernel)

$$K(\mathbf{x}_i, \mathbf{x}) = \exp\left(-\frac{\|\mathbf{x}_i - \mathbf{x}\|^2}{\sigma^2}\right) \quad (3.13)$$

ガウシアンカーネルはRBF (radial bases function) カーネルとも呼ばれる.

- ・ d 次多項式カーネル (polynomial kernel)

$$K(\mathbf{x}_i, \mathbf{x}) = (\mathbf{x}_i^T \mathbf{x} + c)^d \quad (3.14)$$

- ・ シグモイドカーネル (sigmoid kernel)

$$K(\mathbf{x}_i, \mathbf{x}) = \tanh(c \mathbf{x}_i^T \mathbf{x} + \theta) \quad (3.15)$$

一般に、高次元特徴空間へ写像してから内積を計算する場合とカーネル関数を計算する場合を比較すると、カーネル関数を用いたほうが計算量が少い。そのため、カーネル関数を用いた手法はカーネルトリックと呼ばれる。(3.13) 式の σ , (3.14) 式の c と d , (3.15) 式の c と θ はカーネル・パラメータと呼ばれる定数であり、事前に設定する必要がある。カーネル・パラメータの値はSVMの識別能力に大きく影響を与えることが知られており、適切に設定する必要がある。

カーネル関数を用いると (3.10) 式は次式のように書き換えることができる。

$$f(\mathbf{z}) = \sum_{i=1}^l y_i \alpha_i K(\mathbf{x}_i, \mathbf{x}) + b \quad (3.16)$$

また、(3.11) 式の最適化問題は、次式のように書き換えることができる。

$$\begin{aligned} \text{maximise } W(\boldsymbol{\alpha}) &= \sum_{i=1}^l \alpha_i - \frac{1}{2} \sum_{i,j=1}^l y_i y_j \alpha_i \alpha_j K(\mathbf{x}_i, \mathbf{x}_j), \\ \text{subject to } \sum_{i=1}^l y_i \alpha_i &= 0, C \geq \alpha_i \geq 0, i = 1, \dots, l. \end{aligned} \quad (3.17)$$

(3.17) 式により、 \mathbf{x} の高次元特徴空間での識別問題（非線形の識別問題）を扱うことができる。

また、カーネル関数にはバイアス b を埋め込むことができる。入力空間 X を、1次元追加した拡張空間 \hat{X} （この空間では新たなベクトルを、ある固定の t に対し、 $\hat{x} = (x, b/t)$ と表記する）に埋め込むことを考えた場合、“単位重みベクトル \mathbf{w} とバイアス b により表現される X 上の線形関数”は、“重みベクトル $\hat{\mathbf{w}} = (\mathbf{w}, b/t)$ とゼロのバイアスにより表現される \hat{X} 上の線形関数”と同値になり、この拡張次元をカーネル関数 K により誘導された特徴空間に加えることは、 t^2 を加算して適応カーネル（adapted kernel）

$$\hat{K}(\mathbf{x}_i, \mathbf{x}) = K(\mathbf{x}_i, \mathbf{x}) + t^2 \quad (3.18)$$

を作ることと同値であり、この方法における定数 t の安全な選択は

$$t = R = \max_{1 \leq i \leq l} (\|\mathbf{x}_i\|^2) \quad (3.19)$$

であるという。

3.4 サポート・ベクター・マシンの学習

SVMの学習方法として、勾配登り（gradient ascent）や逐次最小化アルゴリズム（Sequential Minimal Optimization : SMO）アルゴリズムが知られている。本研究では勾配登りを採用した。勾配登りは、最急勾配登りアルゴリズム（steepest ascent algorithm）、あるいは最急降下法

(steepest descent method) と呼ばれる手法である。ここでは本研究で採用した勾配登りによる学習について説明する。

バイアス b をカーネル関数に埋め込むことにより、固定値として設定すれば、(3.17) 式の最適化問題の双対は

$$\begin{aligned} \text{maximise } W(\boldsymbol{\alpha}) &= \sum_{i=1}^l \alpha_i - \frac{1}{2} \sum_{i,j=1}^l y_i y_j \alpha_i \alpha_j K(\mathbf{x}_i, \mathbf{x}_j), \\ \text{subject to } 0 &\leq \alpha_i \leq C, \quad i = 1, \dots, l. \end{aligned} \quad (3.20)$$

となる。

勾配登りにより最適解 $\hat{\boldsymbol{\alpha}}$ を求めることを考えると、 α_i の更新は適当な学習率 η_i を用いて、

$$\hat{\alpha}_i \leftarrow \alpha_i + \eta_i \frac{\partial W(\boldsymbol{\alpha})}{\partial \alpha_i} \quad (3.21)$$

となり、また、

$$\frac{\partial W(\boldsymbol{\alpha})}{\partial \alpha_i} = 1 - y_i \sum_{j=1}^l \alpha_j y_j K(\mathbf{x}_i, \mathbf{x}_j) \quad (3.22)$$

である。ここで、最適化する問題は α_i の 2 次関数であるので学習率 η_i は

$$\eta_i = \frac{1}{K(\mathbf{x}_i, \mathbf{x}_j)} \quad (3.23)$$

とすれば、(5.22) 式の導関数は最適な $\hat{\boldsymbol{\alpha}}$ により収束する。したがって、勾配登りによる SVM の学習は、 $\boldsymbol{\alpha}$ に対して

$$\hat{\alpha}_i \leftarrow \alpha_i + \frac{1}{K(\mathbf{x}_i, \mathbf{x}_j)} \left(1 - y_i \sum_{j=1}^l \alpha_j y_j K(\mathbf{x}_i, \mathbf{x}_j) \right) \quad (3.24)$$

という更新を繰り返すことで行われる。

本研究における SVM の学習の終了条件はすべての i ($1, \dots, l$) に対して、

$$0 \leq \alpha_i \leq C,$$

$$y_i f(\mathbf{x}_i) = \begin{cases} \geq 1 & \alpha_i = 0 \text{ の点に対して} \\ = 1 & 0 \leq \alpha_i \leq C \text{ の点に対して} \\ \leq 1 & \alpha_i = C \text{ の点に対して} \end{cases} \quad (3.25)$$

という基準 (Karush–Kuhn–Tucker 条件: KKT 条件) を満たすか、あるいは $\boldsymbol{\alpha}$ の更新量が一定値を下回ることにした。

4 粒子群最適化アルゴリズムによるSVMのハイパー・パラメータの最適化

ここでのSVMのハイパー・パラメータはマージン・パラメータとカーネル・パラメータを指す。マージン・パラメータとカーネル・パラメータの値はSVMの識別能力に大きく影響を与えることが知られており、これらの値は適切に設定しなければならない。SVMのハイパー・パラメータに応じた性能の評価は交差検定 (cross validation : CV) がよく知られている。本研究ではSVMのハイパー・パラメータを最適化する方法として、CVでの識別率を評価値とする粒子群最適化 (Particle Swarm Optimization : PSO) [12][13][14] を採用した。

PSOは、最適化問題に応用可能な群知能の一つであり、多次元空間での解が点や面で表される問題の最適解を探索する汎用的な極小化技術である。焼きなまし法等の他のグローバルな極小化戦略と比較したときのPSOの利点はローカルな極小値がある問題に対して非常に強いことである。

基本的なPSOでは、多次元空間を数多くの粒子 (particle) が動き回る。各粒子は位置ベクトル \mathbf{x}_i 、速度ベクトル \mathbf{v}_i と、その粒子が最高の適応度を獲得した場所 \mathbf{p}_i を記憶し、粒子全体における最高の適応度の場所 \mathbf{p}_g の情報を各粒子で共有する。各粒子は仮説に基づいて空間内にはばらまかれ、それぞれに初期速度が与えられる。各粒子は空間内を移動し、一定時間間隔で適応度に基づいて評価され、よりよい適応度をもつ粒子を中心として群れを構成するように加速を行う。

本研究において用いたPSOでは各粒子の速度 \mathbf{v}_i の更新は次式で行われる。

$$\mathbf{v}_i \leftarrow w\mathbf{v} + c_1r_1(\mathbf{p}_i - \mathbf{x}) + c_2r_2(\mathbf{p}_g - \mathbf{x}) \quad (3.26)$$

次に各粒子の位置 \mathbf{x}_i の更新は次式で行われる。

$$\mathbf{x}_i \leftarrow \mathbf{x}_i + \mathbf{v}_i \quad (3.27)$$

ただし、 w は慣性重み、 r_1 と r_2 は区間 $[0, 1]$ の一様乱数である。また、 c_1 は “cognitive”， c_2 は “social” と呼ばれるパラメータであり、自己の最良位置およびグループの最良位置への探索に対する重み付けを表現している。慣性重み w は多くの場合 1 より若干小さい値が最適であり、“cognitive” と “social” は多くの場合 1 に近い値が最適である。

5 SVMを用いた特徴選択の実装

本研究におけるSVMを用いた特徴選択の実装について説明する。本研究では向きを表す記号黙読実験で得られたEEGデータに対して本研究では 5-fold CV を評価基準とする特徴選択を試みる。

5.1 データの前処理

識別対象のデータは向きを表す記号黙読実験で得た矢印の黙読時のEEGである。識別に用いるEEGの潜時は、過去の研究においてECDが確認された潜時を参考にして200msから700msとした。周波数帯は8 Hzから30Hz (α 波帯と β 波帯)とした。識別に用いるチャンネルは国際10-20電極配置法(図3.1)における全ての頭皮上電極19ch (Fp1, Fp2, F3, F4, C3, C4, P3, P4, O1, O2, F7, F8, T3, T4, Fz, Cz, Pz)分とした。まず、EEGに対してノイズの除去を目的に8 Hzから30HzのFIRフィルタによるバンドパスフィルタをかけた。本研究では、とりあえずFIRフィルタを用いその次元数は250とした。次に、フィルタをかけた後のEEGのうち200msから700msの区間を切り出し、パワースペクトルを算出した。パワースペクトルの周波数分解能はEEGの区間が500ms間であることから2 Hzである。したがって1 chにつき12次元(8 Hz, 10Hz, 12Hz, 14Hz, 16Hz, 18Hz, 20Hz, 22Hz, 24Hz, 26Hz, 28Hz, 30 Hz)のデータが得られ、19ch分では228次元のデータが得られる。

5.2 識別に用いるSVM

識別に用いるSVMはガウシアンカーネルによる非線形ソフトマージンSVMとする。SVMは2群の識別器であるため、4群の識別に対応するために、複数のSVMを組み合わせて識別を行った。複数のSVMによる識別は以下の手順で行った。

- (1) “↑”群と“↓”群をトレーニングデータとするSVMで
識別対象のシングルトライアルを識別する。
- (2) 手順(1)で識別された群と“←”群をトレーニングデータとするSVMで
識別対象のシングルトライアルを識別する。
- (3) 手順(2)で識別された群と“→”群をトレーニングデータとするSVMで
識別対象のシングルトライアルを識別する。
- (4) 手順(3)で識別された群をシングルトライアルに対する識別結果とする。

SVMはCVによる識別におけるテストデータ集合に対して識別を行う度にPSOによる最適化を行う。PSOで最適化するハイパー・パラメータは(3.7)式におけるマージン・パラメータ C と(3.13)式におけるカーネル・パラメータ σ^2 の2つである。PSOの各粒子の適応度の評価はEEGデータに対する4群の5-fold CV識別率である。したがって、上記の4群の識別の手順

において、SVMのハイパー・パラメータは同一の値を用いる。すなわち、手順（1）におけるSVMの C および σ^2 の値と、手順（2）および手順（3）におけるSVMの C および σ^2 の値はそれぞれ等しい。

PSOの各粒子の位置 $x_i (\in \mathbf{R}^2)$ の初期位置は2次元とも $[-2^{16}, 2^{16}]$ の範囲を取る一様乱数とした。また、局所解に陥ることを避けるため、速度 v_i の更新が(3.26)式に従う粒子（便宜上、通常粒子と呼ぶ）と共に、速度の各成分が乱数に従う粒子（乱数粒便宜上、乱数粒子と呼ぶ）を用いた。各粒子の位置更新の反復は、全粒子の最良の適応度が一定回数更新されない場合に打ち切るようにした。通常粒子の慣性重み w は1.0とし、 c_1 （“cognitive”）と c_2 （“social”）は0.8とした。

5.3 特徴選択

本研究における特徴選択の手法はラッパー法のforward stepwise selectionである。PSOの過程において、最良の5-fold CV識別率を記録した特徴を選択してゆく。

6 特徴選択の結果

前述の前処理を施したEEGデータに対して、PSOで最適化する非線形SVMを用いて、Forward Stepwise Selectionでの特徴選択を施した。特徴選択は各データに対して3度行った。そのうち2度は、PSOの通常粒子数3、乱数粒子数2、更新打ち切り回数5とし、1度はPSOの通常粒子数8、乱数粒子数2、更新打ち切り回数10とした。ただし、今回の計算は、特徴選択の途中で計算を中断した。また、特徴選択を行うたびに、最高識別率とそのときの特徴数が異なっている。

特徴選択の途中結果を表5.1、表5.2、表5.3に示す。表中の太字は各データの3度の特徴選択のうちの最高識別率である。特徴選択の過程での最高識別率は49.17%から62.50%である。

表6.1 特徴選択結果1（通常粒子数：3，乱数粒子数：2，更新打ち切り回数：5）

データ名	中断時の 特徴数	最高識別率時の 特徴数	最高識別率 [%]
HY	100	22	56.67
MM	101	30	52.50
MY 1	98	21	62.50
MY 2	109	12	51.67
NM	99	64	55.83
SI 1	97	5	49.17
SI 2	102	87	57.50

表6.2 特徴選択結果2 (通常粒子数：3, 乱数粒子数：2, 更新打ち切り回数：5)

データ名	中断時の特徴数	最高識別率時の特徴数	最高識別率 [%]
HY	97	50	60.83
MM	112	19	53.33
MY 1	104	26	53.33
MY 2	100	33	59.17
NM	92	49	57.50
SI 1	102	29	54.17
SI 2	122	118	57.50

表6.3 特徴選択結果3 (通常粒子数：8, 乱数粒子数：2, 更新打ち切り回数：10)

データ名	中断時の特徴数	最高識別率時の特徴数	最高識別率 [%]
HY	74	68	59.17
MM	83	24	50.83
MY 1	77	27	54.17
MY 2	88	17	52.50
NM	83	53	57.50
SI 1	73	28	49.17
SI 2	88	25	49.17

7 考察

まず、特徴選択するたびに結果が異なるのは無視できない問題である。結果が異なる原因はPSOの最適化の精度が足りないか、あるいは初期値が乱数であることに由来する局所最適解に陥ったことが考えられる。分析を行うたびに選択される特徴が異なるならば、選択された特徴にあまり意味はない。このことから、乱数での初期値に依存しない解析方法を採用すべきだと考えられる。また、PSOによる最適化の精度を上げるには、粒子数を増やすことと、更新の回数を増やすことが挙げられる。しかしながら、最適化を行いながらの特徴選択は膨大な計算時間を要し、粒子数や更新回数を増やすことは、計算時間の更なる増大に直結する。本研究でも計算に時間を費やしすぎ、最終的に分析を中断せざるを得なかった。このことから、粒子数や更新回数を増やすことでの精度の向上は、現実的ではない。

以上より、PSOによる最適化を行いながらのSVMによる特徴選択は、数100次元を扱うような特徴選択には適さないと考えられる。その対処として、初期値が乱数に依存せず、膨大な計

算時間を要さない解析方法を考える必要がある。

謝辞

本研究は平成19年3月に終了した文部科学省私立大学学術研究高度化推進事業ハイテク・リサーチ・センター整備事業に伴う北海学園大学ハイテク・リサーチ・センター研究プロジェクト「視覚・画像・音声・言語情報の高度化と知的計測制御技術への応用」ならびに、平成19年度新規に採択された文部科学省私立大学戦略的研究基盤形成支援事業に伴う北海学園大学ハイテク・リサーチ・センター研究プロジェクト「電磁・光センシングを主体とする生体関連情報の先進的計測・処理技術の開発と応用」の一環として行われた。本研究を行うにあたり協力を頂いた被験者および援助を頂いた方々に謝意を表す。

参考文献

- [1] R. A. McCarthy, E. K. Warrington (相馬芳明, 本田仁視監訳), 認知神経心理学, 医学書院, 1996
- [2] 中川敦子, 単漢字の処理における左右半球機能の検討—漢字の構造と音訓を通して—, 神経心理学, 第18巻, 第2号, pp.120–128, 2002
- [3] T. Yamanoi, T. Yamazaki, J.–L. Vercher, E. Sanchez, M. Sugeno, “Dominance of recognition of words presented on right or left eye –Comparison of Kanji and Hiragana–”, to appear in *Modern Information Processing, From Theory to Applications*, B. Bouchon–Meunier, G. Coletti and R.R. Yager Eds., Elsevier Science B.V., pp.407–416, 2006
- [4] Takahiro YAMANOI, Hisashi TOYOSHIMA, Toshimasa YAMAZAKI, Shin–ichi OHNISHI : Localization of brain activity during perception of circle movement by use of equivalent current dipole analysis, 2004 IEEE international Conference on Fuzzy Systems Proceedings VOLUME 1, pp.321–324, 2004
- [5] 豊島恒, 山ノ井高洋, 山崎敏正, 大西真一: 時空間等価電流双極子推定法によるRDS両眼立体視の脳内処理部位の解析, 知能と情報 (日本知能情報ファジィ学会誌), Vol. 18, No. 1, pp. 102–110, 2006
- [6] 豊島恒, 山ノ井高洋, 山崎敏正, 大西真一, 菅野道夫: 向きを表す単語と記号に対する時空間的脳活動の比較, 知能と情報 (日本知能情報ファジィ学会誌), Vol. 18, No. 3, pp. 425–433, 2006
- [7] 山ノ井高洋, 豊島恒, 山崎敏正, 菅野道夫: 向きを表す記号イメージング時EEGによるマイクロロボット制御, 第24回ファジィシステムシンポジウム講演論文集, WD2–1, pp. 200–203, 2008
- [8] Takahiro Yamanoi, Hisahi Toyoshima, Toshimasa Yamazaki, Shin–ichi Ohnishi, Michio Sugeno, and Elie Sanchez : Micro Robot Control by Use of Electroencephalograms from Right Frontal Area, *Journal of Advanced Computational Intelligence and Intelligent Information*, vol.13, No.2, pp.68–75, 2009
- [9] 栗田多喜夫: サポートベクターマシン入門, 2002
<http://www.neurosci.aist.go.jp/~kurita/lecture/svm/svm.html>
- [10] Nello Cristianini (著), Jhon Shawe–Taylor (著), 大北 剛 (訳): サポートベクターマシン入門, 共立出版, 初版2005
- [11] 小野田 崇 (著), 人工知能学会 (編集), JSAI= (編集): サポートベクターマシン (知の科学), オーム社, 初版2007
- [12] 田中 謙次: ブレインコンピュータインターフェースのためのモデル選択に関する研究, 筑波大学大学院博士課程システム情報工学研究科修士論文

<http://www.cs.tsukuba.ac.jp/H18Syuron/200520978.pdf>, 2007

[13] 伊庭斉志 (著), 人工知能学会 (編集), JSAI= (編集): 進化論的計算手法 (知の科学), オーム社, 初版2005

[14] 高濱 徹行: 集団的降下法による最適化 — 差分進化と粒子群最適化 —

<http://www.ints.info.hiroshima-cu.ac.jp/~takahama/documents/pdescent/pdescent.html>, 2007