社団法人　情報処理学会　研究報告
IPSJ SIG Technical Report

2004－ＩＣＳ－135　(33)
2004／3／17

ブール関数の学習における ブーリアンカーネルを用いた特徴選択について

佐土原　健†

† 産業技術総合研究所
〒 305–8568 茨城県つくば市梅園 1–1–1 つくば中央第二
E-mail: †sadohara@computer.org

あらまし　本論文は，離散データ分類器の学習のための，変数間の依存関係を考慮した変数選択アルゴリズムを提案する．このアルゴリズムは，変数の組合せの重み付き線形和として学習された分類器から，ある変数を含む組合せを，全て除去して得られる分類器の制限を計算する．そして，制限による分類能力の低下が確認されないとき，その変数を分類に寄与しない変数として除去する．このような変数選択手法が，既存手法に比べて優れていることを，人工的に生成したデータを用いた実験により示す．また，本手法のテキスト分類問題に対する適用可能性を示唆する実験結果を示す．
キーワード　特徴選択，サポートベクトルマシン，分類学習，テキスト分類，データマイニング

Feature selection using Boolean kernels for the learning of Boolean functions

Ken SADOHARA†

† National Institute of Advanced Industrial Science and Technology (AIST)
AIST Tsukuba Central 2, 1–1–1 Umezono, Tsukuba-shi, Ibaraki, Japan
E-mail: †sadohara@computer.org

Abstract　This paper presents a variable selection algorithm for learning classifiers of discrete data that can take into account variable dependency. The algorithm restricts a learned classifier represented as a weighted linear sum of combinations of variables by removing combinations containing a variable. Then, the variable is identified as useless if the restriction does not degrade discriminative ability. The presented algorithm is shown to outperform some existing algorithms in the experiment on synthetic data sets. Furthermore, an encouraging result on the applicability of the algorithm toward text classification is also shown.
Key words　feature selection, support vector machine, classification, text categorization, data mining

1. 序　　論

近年，データマイニングが対象とするデータは，例えば，マイクロアレイ分析やテキスト分類等で用いられるデータのように，非常に多くの変数で記述されている場合が多い．そうした状況の下で，データの分析に寄与する変数や特徴を(注1)選択するための変数・特徴選択手法に関する研究が，再び注目を集めている[1],[2]．特に，分類学習において，変数選択は，データを記述する多くの変数の中から，分類に寄与する変数を選択する問題である．分類にとって十分かつ小さな変数の集合を選択することは，分類精度の向上が期待できるだけでなく，学習や分類に必要な計算資源を節約したり，データをより良く理解するためにも有用である．

本論文では，分類学習における変数選択を，より基本的に，ブール関数の学習における変数選択の問題として考察する．その理由は，離散データに対する分類学習は，本質的にブール関数の帰納学習の問題に帰着されるからである．さらに，数値データに対しても，可読性や計算資源の節約のために，変数が離散化される場合も多いことを考えれば，ブール関数の学習における変数選択を研究することの意義は大きい．

本論文が意図する応用領域は，テキスト分類[3]である．テキスト分類においては，テキストは典型的に，“bag of words”，つまり，テキストに現われる単語やフレーズの出現頻度などのベ

(注1)：本論文では，カーネル法を用いて，入力変数から派生する特徴を取り扱うため，“変数”と“特徴”という用語を区別して用いることにする．

クトルとして表現される．そして，ベクトルの各要素として，例えば，出現頻度がある閾値よりも多いか否かを表わすブール変数を考えれば十分な場合も多い[4]．

ブール関数の学習における変数選択においては，変数間の依存関係を考慮しなければならない．例えば，ブール式 $y = x_1\overline{x_2} \vee x_2x_3$ において，x_2 は y にとって必要不可欠な変数であるにも関わらず，x_2 の値を知ることは，全く情報利得をもたらさないので，変数の依存関係を考慮しなければ，x_2 は分類に不必要な変数であると判断してしまう危険性があるからである．このような変数の依存関係を考慮するために，本研究では，Support Vector Machine (SVM) [5], [6] に代表されるカーネル法[8] を用いた，変数選択アルゴリズムを提案する．

このアルゴリズムは，SVM を用いて，論理積が張る空間上で，論理積の重み付き線形和としてブール関数を学習する．そのような空間は一般に非常に高次元であるが，ブーリアンカーネルを用いることで，高次元空間においても効率良くブール関数を学習できることが知られている[9]〜[11]．本論文で提案する変数選択アルゴリズムは，このブーリアンカーネルを，学習に加えて，学習されたブール関数を分析して，ブール関数の出力に有用でない変数の同定を行う目的でも利用する．この分析において，学習された線形和から，ある変数を含む論理積を全て除去して得られる，制限された線形和を計算するために，ブーリアンカーネルが用いられ，制限によって分類能力を劣化させない変数は，分類に寄与しないと判断される．このような制限を計算するためには，非常に多くの論理積を除去せねばならず，計算量的に実行不能であるように思われるが，ブーリアンカーネルを用いることで，効率良く制限を計算することが可能になる．

本論文では，このような変数選択手法を，人工的に生成したデータセットを用いた計算機実験において，様々な角度から既存手法と比較検討し，提案手法が，既存手法よりも優れていることを示す．特に，SVM を用いた Recursive Feature Elimination (RFE) [12] と詳細な比較を行い，提案手法の方が，多くのデータの真偽値の決定に関与する，影響力の大きな変数を，誤って除去してしまう危険性が低く，従って，より高い分類精度を達成可能であることを示す．さらに，提案手法はテキスト分類問題に適用され，テキストの分類に有用かつ比較的小さな変数の集合を選択可能であることを示す．

2. ブーリアンカーネルと SVM

任意のブール関数 y は，等価な選言標準形 (Disjunctive Normal Form, DNF) を持つので，$sgn(f(\boldsymbol{x})) = y(\boldsymbol{x})$ を満す (注2) 論理積の重み付き線形和 f として表現可能である[9]．しかしながら，可能な論理積の数は一般に非常に多いので，このような論理積の線形和を，論理積が張る空間上で直接学習することは計算量的に困難である．これに対し，Support Vector Machines (SVMs) [5], [6] に代表されるカーネル法[8] を用いることで，このような高次元空間を陽に取り扱うことなく，f の学習を効率良く行うことが可能になる．

SVM は，与えられた入力変数から派生する特徴によって張られた特徴空間 Z 上で，線形識別関数 $f(\boldsymbol{z}) = \langle \boldsymbol{w} \cdot \boldsymbol{z} \rangle + b$ を学習する．ここで，ブール関数の学習の場合，$b = 0$ の識別関数のみを考えれば十分である[9] ので，以降，$b = 0$ であると仮定する．訓練データ $\{(\boldsymbol{x}_i, y_i)\}_{i=1}^n$ と，入力空間から特徴空間への写像 ϕ に対して，識別関数 f は，特徴空間上の点 $\boldsymbol{z}_i = \phi(\boldsymbol{x}_i)$ を次のように分離しなければならない:

$$y_i f(\boldsymbol{z}_i) \geqq 1 - \xi_i \quad \xi_i \geqq 0 \quad 1 \leqq i \leqq n. \tag{1}$$

ここで，ξ_i は線形分離不能の場合に対処するために導入された，分離の不完全性を許容するパラメータである．制約 (1) を満す識別関数は一般に複数存在し得るが，SVM は，$\|\boldsymbol{w}\|^2 + C\sum_{i=1}^n \xi_i$ を最小化する識別関数を学習する．このような最小化は，ある適切な正数 C の下で，汎化誤差を近似的に最小化することが知られている．

さらに，最適化の理論によれば，上記のような最適化問題は，次のような双対問題と等価であることが知られている．

$$\text{maximize} \quad \sum_{i=1}^n \alpha_i - \frac{1}{2}\sum_{i=1}^n\sum_{j=1}^n \alpha_i\alpha_j y_i y_j \langle \boldsymbol{z}_i \cdot \boldsymbol{z}_j \rangle \tag{2}$$

$$\text{subject to} \quad 0 \leqq \alpha_i \leqq C \quad (1 \leqq i \leqq n). \tag{3}$$

そして，このような最適化問題に対して，効率の良い解法が知られている[7]．また，最適解 $\alpha_1^*, \ldots, \alpha_n^*$ が得られるとき，最適な識別関数 $f^*(\boldsymbol{z})$ は，以下のように双対表現可能である．

$$f^*(\boldsymbol{z}) = \sum_{i=1}^n \alpha_i^* y_i \langle \boldsymbol{z}_i \cdot \boldsymbol{z} \rangle .$$

ここで，$\alpha_i^* \neq 0$ の場合，$\boldsymbol{x}_i$ は サポートベクトルと呼ばれる．

双対表現を用いる利点は，特徴空間の次元が大きい場合に，計算コストの大きい ϕ の計算を回避することができる点である．双対表現においては，ϕ が

$$\langle \boldsymbol{z}_i \cdot \boldsymbol{z}_j \rangle = \langle \phi(\boldsymbol{x}_i) \cdot \phi(\boldsymbol{x}_j) \rangle$$

のように内積の形でしか現われないので，もしも，この内積を，次のようにデータから直接計算することができれば，

$$K(\boldsymbol{x}_i, \boldsymbol{x}_j) = \langle \phi(\boldsymbol{x}_i) \cdot \phi(\boldsymbol{x}_j) \rangle ,$$

データを陽に特徴空間に写像することなしに，学習を行うことができる．このような関数 K を カーネル関数 と呼ぶ．

全ての可能な論理積が張る特徴空間に対しては，次のような関数を用いて，内積を計算できることが知られている[9], [10]:

$$K(\boldsymbol{u}, \boldsymbol{v}) \stackrel{\text{def}}{=} -1 + 2^{\mathrm{s}(\boldsymbol{u},\boldsymbol{v})}.$$

ここで，$\mathrm{s}(\boldsymbol{u}, \boldsymbol{v})$ は，ビット列 $\boldsymbol{u}$ と $\boldsymbol{v}$ において同じ値を持つビットの数を表わす．これは，積が 1 となる成分に対応する論理積は，$\boldsymbol{u}$ と $\boldsymbol{v}$ において同時に真とならなければならず，そうした論理積の数は，空論理積を除くと，$-1 + 2^{\mathrm{s}(\boldsymbol{u},\boldsymbol{v})}$ 個存在するためである．この最も一般的な特徴空間の幾つかの部分空間に対しても，カーネル関数が知られている．

(注2): 記述の簡便さのために，通常とは異なり，ブール関数の出力を $+1$ あるいは -1 としていることに注意されたい．

長さが高々 k である論理積が張る特徴空間に対しては，次の K^k を用いて内積を計算可能であることが知られている [10]:

$$K^k(\boldsymbol{u},\boldsymbol{v}) \stackrel{\text{def}}{=} \sum_{i=1}^{k} \begin{pmatrix} \mathrm{s}(\boldsymbol{u},\boldsymbol{v}) \\ i \end{pmatrix}.$$

また，否定を含まない長さが高々 k である論理積が張る特徴空間に対しては，次の K_m^k を用いて内積を計算可能であることが知られている [10]:

$$K_m^k(\boldsymbol{u},\boldsymbol{v}) \stackrel{\text{def}}{=} \sum_{i=1}^{k} \begin{pmatrix} \mathrm{sp}(\boldsymbol{u},\boldsymbol{v}) \\ i \end{pmatrix}.$$

ここで，$\mathrm{sp}(\boldsymbol{u},\boldsymbol{v})$ は，$\boldsymbol{u}$ と $\boldsymbol{v}$ において共に値 1 を持つビットの数を表わす．

前述した通り，これらのブーリアンカーネルは，論理積が張る高次元空間において，効率の良い学習を可能にするために導入されたが，次節では，学習された分類器を分析する目的でブーリアンカーネルを利用する．

3. 変数選択アルゴリズム

本節では，ブーリアンカーネルを用いて，学習した識別関数の制限を計算・評価することで，識別に寄与しない変数を除去するアルゴリズムを示す．特徴の部分集合 $V' \subseteq \{z_1,\ldots,z_\ell\}$ により張られた部分空間を Z' とするとき，任意の線形識別関数 $f(z_1,\ldots,z_\ell) = \sum_{i=1}^{\ell} w_i z_i$ に対して，$f'(z_1,\ldots,z_\ell) = \sum_{z_i \in V'} w_i z_i$ を，f の Z' への制限と呼ぶ．

具体的に，ブール変数 x_1, x_2, x_3, x_4 から構成される全ての論理積が張る空間 Z 上の識別関数とその制限を考えてみよう．ブール関数 $y = x_1\overline{x_2} \vee x_2x_3$ に対して，例えば，

$$\begin{aligned} f(x_1,x_2,x_3,x_4) &= x_1(1-x_2) + x_2x_3 \\ &\quad -(1-x_1)(1-x_2) - x_2(1-x_3) \end{aligned}$$

は，y の入力を正しく識別する関数である．ここで，Z から，変数 x_4 を含む論理積 (例えば $x_4, \overline{x_4}, x_4x_1, x_4\overline{x_1}, x_4x_1x_2, \ldots$) を取り除いて得られる Z の部分空間 Z_{-x_4} を考えると，f の Z_{-x_4} への制限 f' は，f と等価である．一方，Z から，変数 x_2 を含む論理積を取り除いて得られる Z の部分空間 Z_{-x_2} を考えると，f の Z_{-x_2} への制限 f'' は，恒等的に 0 となる．

本節で示す変数選択アルゴリズムは，これらの制限と元々の識別関数との識別能力を比較し，識別能力にほとんど変化を与えないような変数は，識別に寄与しない変数であると判断される．例えば，Z_{-x_4} への制限 f' は，明らかに f と同じ識別能力を持っているので，変数 x_4 は，識別に寄与しない変数である考えられる．一方，Z_{-x_2} への制限 f'' を考えると，f'' は，識別能力を全く失うので，f'' は，識別に寄与する変数であると考えることができる．

しかし，ある変数を含む論理積の数は非常に多いので，識別関数 f の Z_{-x} への制限を効率良く計算する方法は自明ではない．これに対して，以下の命題は，制限の計算にカーネル関数が利用可能であることを示している．

[命題 1] 特徴空間 Z に対するカーネル関数を K, Z の部分空間 Z' のカーネル関数 K' とするとき，以下のような Z 上の任意の線形関数 $f(\boldsymbol{x}) = \sum_{j=1}^{n} y_j\alpha_j K(\boldsymbol{x}_j,\boldsymbol{x})$, の Z' への制限は，$f'(\boldsymbol{x}) = \sum_{j=1}^{n} y_j\alpha_j K'(\boldsymbol{x}_j,\boldsymbol{x})$. と書ける．

カーネル関数を用いて，分類器の制限を計算するというアイデアは，文献 [11] でも用いられている．ここでは，分類に寄与する論理積の長さを同定する目的で，長さが k よりも大きな論理積を除去した制限を計算するために，ブーリアンカーネル K^k を用いている．

同様に，変数選択を行なうためには，ある変数を含む論理積を除去して得られる部分空間のカーネル関数が必要になる．このような，カーネル関数は，特定の変数が除去された入力ベクトルに対するブーリアンカーネルを考えることで得られる．可能な全ての論理積が張る特徴空間に対しては，$K(-x)(\boldsymbol{u},\boldsymbol{v}) \stackrel{\text{def}}{=} K(\boldsymbol{u}(-x),\boldsymbol{v}(-x))$ が，変数 x を含む論理積を全て除去して得られる部分空間に対するカーネル関数となる．ここで，$\boldsymbol{u}(-x)$ は，ベクトル $\boldsymbol{u}$ から x に対応する成分を取り除いたベクトルを表わす．長さが高々 k の論理積が張る特徴空間 Z に対しては，$K^k(-x)(\boldsymbol{u},\boldsymbol{v}) \stackrel{\text{def}}{=} K^k(\boldsymbol{u}(-x),\boldsymbol{v}(-x))$ が，部分空間 Z_{-x} のカーネル関数となる．また，長さが高々 k の否定を含まない論理積が張る特徴空間 Z' に対しては，$K_m^k(-x)(\boldsymbol{u},\boldsymbol{v}) \stackrel{\text{def}}{=} K_m^k(\boldsymbol{u}(-x),\boldsymbol{v}(-x))$ が，部分空間 Z'_{-x} のカーネル関数となる．

このように計算される識別関数の制限は，次に，その識別能力を元々の識別関数と比較される．今，学習された識別関数を，$f(\boldsymbol{x}) = \sum_{j\in SV} y_j\alpha_j K(\boldsymbol{x}_j,\boldsymbol{x})$ とする．ここで，SV は，サポートベクトルの集合である．このとき，f とその制限 $f'(\boldsymbol{x}) = \sum_{j\in SV} y_j\alpha_j K'(\boldsymbol{x}_j,\boldsymbol{x})$ との識別能力の差 D は以下のように定義される．

$$\begin{aligned} D(f,f') &\stackrel{\text{def}}{=} \sum_{i=1}^{n} y_i \left(f(\boldsymbol{x}_i) - f'(\boldsymbol{x}_i)\right) \\ &= \sum_{i=1}^{n} \sum_{j\in SV} y_iy_j\alpha_j \left(K(\boldsymbol{x}_j,\boldsymbol{x}_i) - K'(\boldsymbol{x}_j,\boldsymbol{x}_i)\right). \end{aligned}$$

表 1 は，これまでに述べた変数選択アルゴリズム示しており，このアルゴリズムを Feature Elimination with Restriction Kernel (FERK) と呼ぶことにする．

V: 変数の集合, $K(-M)$: M 中の変数を含む論理積を除去して得られる空間のカーネル関数.

(1) $M \stackrel{\text{def}}{=} \emptyset$, $R \stackrel{\text{def}}{=} V$.
(2) $K(-M)$ を用いた SVM で識別関数 f を学習する.
(3) $v \in R$ に対して，$K(-M')$ $(M' = M \cup \{v\})$ を用いて，f の制限 f' と $D(f,f')$ を計算し，最も D が小さい変数を v^* とする.
(4) もしも，停止条件が成立すれば，R を出力して停止する.
(5) $M \stackrel{\text{def}}{=} M \cup \{v^*\}$, $R \stackrel{\text{def}}{=} R \setminus \{v^*\}$.
(6) 2 に戻る.

表 1 変数選択アルゴリズム FERK

表 1 中の停止条件は，重要な研究課題であるが，本論文では考察しない．5.3 節で示す実験においては，変数の数が，ある与

えられた数に到達した時に停止するように実装されている．また，表 1 のアルゴリズムは，計算コストを減少させるために，一度に複数の変数を除去するように拡張することも可能である．

4. 関連研究

変数選択に関しては，これまでに多くの手法が提案されてきたが [1], [2]，その中でも，最も単純な手法は，各変数とクラス変数との相関を，例えば相互情報量等を用いて個別に評価し，変数のランキングを作る手法である．

RELIEF [13] は，nearest-neighbor 法に基づく別のアプローチを取る．RELIEF は訓練データからランダムにサンプリングされたデータを，同じクラスに属する/属さない最近隣データと比較することで各変数の重みを更新する．

文献 [14] では，決定木学習アルゴリズムをサブルーチン的に用いて，ある与えられた変数の部分集合のみから分類器を学習させ，その分類精度に基づいて，部分集合を評価し，最適な変数の集合を貪欲法で探索する wrapper 法が提案されている．

学習アルゴリズムと一体化した変数選択手法アルゴリズムとしては，Recursive Feature Elimination (RFE) [12] が知られている．RFE は，FERK と同様に，学習アルゴリズムとして SVM を用いるが，各変数の評価尺度が異っている．RFE においては，各変数 x に対して，その変数を除去することによる目的関数 (2) の変化 $DJ(x)$ を計算し，最も変化を与えない変数を分類に寄与しない変数として取り除く．

$$DJ(x) = \frac{1}{2}\sum_{i\in SV}\sum_{j\in SV}\alpha_i\alpha_j y_i y_j K(\boldsymbol{x}_i,\boldsymbol{x}_j) - \frac{1}{2}\sum_{i\in SV}\sum_{j\in SV}\alpha_i\alpha_j y_i y_j K(\boldsymbol{x}_i(-x),\boldsymbol{x}_j(-x)).$$

線形の場合，すなわち $K(\boldsymbol{x}_i,\boldsymbol{x}_j)=\langle \boldsymbol{x}_i\cdot\boldsymbol{x}_j\rangle$ の時，$DJ(x_i)=\frac{1}{2}(w_i)^2$ となる．なぜならば，最適解 $\alpha_1,\ldots,\alpha_n$ に対して，$\boldsymbol{w}=\sum_{i\in SV}\alpha_i y_i\boldsymbol{x}_i$ であるからである．この場合，学習された識別関数の出力に与える影響が最も小さな変数が除去される．

5. 実　　験

本節では，計算機実験によって，FERK の性能を，前節で挙げた変数選択アルゴリズムと比較する．まず，最初に，様々なパラメータを変化させたときのアルゴリズムの性能の変化を見るために，人工的に合成したデータセットを用いた実験を行う．次に，実データへの適用可能性を見るために，テキスト分類の代表的データセットを用いた実験を行う．

5.1 ランダムに生成したブール関数の学習

この実験では，人工的に合成されたブール関数の入出力例の集合から，各変数選択アルゴリズムにより選択された変数のみを用いて学習された，ブール関数の分類誤差を測定することにより，次の 5 つの変数選択アルゴリズムの性能を比較する: (1)FERK, (2)RFE, (3)MINFO (相互情報量を用いた変数ランキング法), (4)RELIEF, (5)WCBE (C4.5 を用いた wrapper 法). WCBE は，変数の部分集合の生成に，backward elimination 法を用い，各部分集合を用いて学習した分類器の分類精度を 10 分割交差検定法で推定する．RFE と FERK は，K^k カーネルを用いた SVM を学習アルゴリズムとして用いる．RELIEF は，全ての訓練データと，各データに対する全ての最近隣データを用いた，決定的に動作するアルゴリズムを用いる．

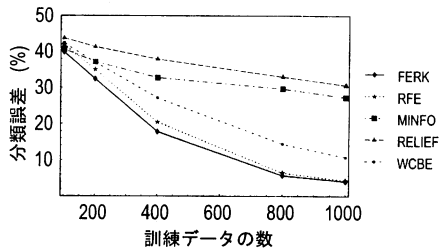

図 1　分類誤差 対 訓練データの数

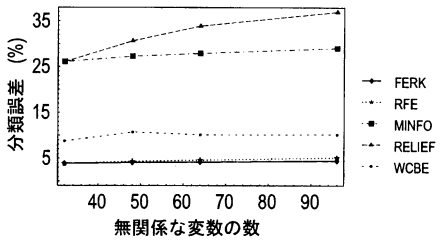

図 2　分類誤差 対 無関係な変数の数

データの生成は，3 つのパラメタ (1)DNF 式の真偽に無関係な変数の数 r, (2) 訓練データの数 n, (3) 論理積の長さ ℓ で定義される DNF 式の複雑さ，を制御して，以下のように行われる．まず，$16+r$ 個の変数の中で，ある固定された 16 個の変数のみを用いて DNF 式を生成する．DNF 式の各論理積は，ランダムに選ばれた ℓ 個の変数を $\frac{1}{2}$ の確率で負リテラルとすることで生成される．論理積の数は，ほぼ等しい数の正例と負例を有するように，$2^{\ell-1}$ 個とする．そして，この DNF 式に対して，n 個の訓練データと，2000 個のテストデータが，一様分布に従って独立に生成される．

このように生成された訓練データは，各変数選択アルゴリズムに与えられ，m 個の変数が選択される．次に，各アルゴリズムが選択した変数と，先に生成された訓練データを用いて，共通の学習アルゴリズム (ブーリアンカーネル K^ℓ を用いた SVM) により分類器を学習し，テストデータに対する分類誤差を測定する．このような測定を，160 個の DNF 式に対して行い，その平均値を用いて各変数選択アルゴリズムの性能を比較する．

図 1 は，訓練データ数 n の変化に対する分類誤差の変化を表わしている．ただし，この実験では，$\ell=4$, $r=48$ とし，m は，各 DNF 式に表われた変数の数とした．

図 2 は，無関係な変数の数 r の変化に対する分類誤差の変化を表わしている．ただし，この実験では，$\ell=4$, $n=1000$ とし，m は 各 DNF 式に表われた変数の数とした．

図 3 は，論理積の長さ ℓ の変化に対する分類誤差の変化を表わしている．ただし，この実験では，$r=48$, $n=1000$ とし，m は 各 DNF 式に表われた変数の数とした．

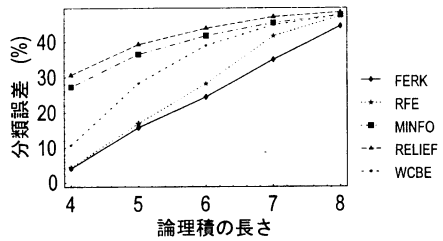

図 3　分類誤差 対 DNF 式の複雑さ

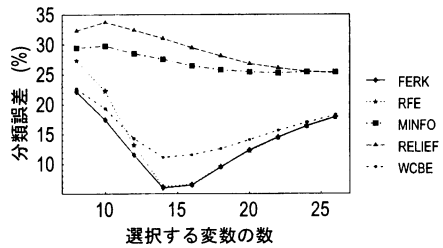

図 4　分類誤差 対 選択する変数の数

これまでの実験では, 選択する変数の数 m は, 各 DNF 式に表われる変数の数とした. これは, 他のパラメータの影響をより明確に分析するためであるが, 正しい変数の数があらかじめ分かっているという設定は理想的に過ぎる. そこで, m の値を変化させるときに, 分類誤差がどのように変化するのかを調べた. 図 4 は, $r = 48$, $n = 1000$, $\ell = 4$ とする時に, 選択する変数の数 m を変化させた実験の結果である. DNF 式に現われる変数の数は平均 14.5 個であったが, RFE と FERK は, この値の周辺で最も小さな分類誤差を示した後, 変数を絞り込むにつれて急激に性能が劣化している.

5.2 FERK と RFE の比較

これまでの実験結果は, FERK と RFE がほぼ同等の性能を有しているものの, 訓練データの数が少ない場合, 無関係な変数の数が多い場合, DNF 式が複雑な場合など, 過適応の危険性が高い場合に RFE は FERK よりも性能が劣化することがわかる. さらに, より顕著な性能劣化が見られるのは, 選択する変数の数を変化させる実験において, 本来必要な変数の数よりも少ない変数を選択せねばならない場合である. 図 4 を見ると, 全ての無関係な変数を除去した後で, DNF 式に含まれる変数を取り除き始めるやいなや RFE の性能が急激に劣化しているように見える. この性能劣化の原因は, RFE が, DNF 式で用いられている変数の重要度を適切に評価していないからではないかと考えられる. 例えば, DNF 式で用いられている変数の中でも, 多くのデータの真偽値に影響を与える変数とそうでない変数が存在するが, 影響の小さな変数よりも, 影響の大きな変数を先に除去している可能性がある. 以下で示す例は, このような不適切な変数のランキングが実際に起り得ることを示している.

その前に, より正確に, 変数の影響力を定義しよう. 任意の変数 x と任意のブール関数 y に対して,

$$I(x,y) = \frac{|\{u \in \{0,1\}^d \mid y(u) \cdot y(u(\bar{x})) = -1\}|}{2^d},$$

を変数 x の y における影響力とする. ここで, $u(\bar{x})$ は, ビット列 u において, x の値を反転して得られるビット列とする. このとき, $0 \leqq I(x,y) \leqq 1$ であり, y に現われない変数 x に対しては, $I(x,y) = 0$ であることに注意されたい.

例えば, DNF 式として $y = x_1x_2x_3 \vee \overline{x_1}x_2x_4$ ($\overline{y} = \overline{x_2} \vee x_1\ \overline{x_3} \vee \overline{x_1}\ \overline{x_4}$) を考えると, $I(x_2,y) = \frac{1}{2}$, $I(x_1,y) = \frac{1}{4}$ であるから, x_1 よりも x_2 の方が影響力が大きい. しかしながら, RFE を用いて, 長さが高々 3 の論理積が張る特徴空間上で学習された最適な識別関数 f を 分析すると, x_2 よりも x_1 の方が, 識別に寄与する変数と判断されてしまう. ブーリアンカーネルを用いるとき, RFE で用いられている変数の評価尺度 $DJ(x)$ は, x を含む論理積の重みの自乗和に相当する. 表 2 は, f を構成する幾つかの論理積の重みを示している. この表から, 論理

x_1	$\overline{x_1}$	x_2	$\overline{x_2}$
-8.5×10^{-2}	-8.5×10^{-2}	1.2×10^{-1}	-3.0×10^{-1}
$x_1\ \overline{x_3}$	$\overline{x_1}\ \overline{x_4}$	$x_2\ \overline{x_3}$	$\overline{x_2}\ \overline{x_4}$
-2.7×10^{-1}	-2.7×10^{-1}	-7.5×10^{-2}	-1.1×10^{-1}

表 2　最適分類器における各論理積の重み

積 x_2 や $\overline{x_2}$ の重みの絶対値は, x_1 や $\overline{x_1}$ のそれよりも大きいことがわかる. しかしながら, 表中の長さ 2 の論理積については, x_2 を含む論理積よりも, x_1 を含む論理積の方が, 重みの絶対値は大きい. これは, $x_1\ \overline{x_3}$ や $\overline{x_1}\ \overline{x_4}$ が $\overline{y}$ の十分条件になっているためである. このような x_1 を含む論理積の重みの自乗和を考えると, $DJ(x_1)$ は, およそ 1.05 となる. 一方, x_2 を含む論理積の重みの自乗和を考えると, $DJ(x_2)$ は, およそ 0.962 となり, x_2 は x_1 よりも識別に寄与しないと判断されてしまう.

以下では, 誤って影響力のある変数を除去してしまう危険性を評価するために, FERK と RFE の影響力の大きな変数を保存する能力について考察する. そのために, 変数選択によって失われた影響力を測る次のような評価尺度 DI を定義する. 任意の DNF 式 y と変数の集合 V に対して,

$$DI(V,y) \stackrel{\text{def}}{=} \sum_{v \in var(y)} I(v,y) - \sum_{v \in V} I(v,y),$$

ここで, $var(y)$ は y に現われる変数の集合を表わす. 上述したブール関数の学習実験において, ランダムに生成された DNF 式 y と, 各変数選択アルゴリズムにより選択された変数の集合 V に対して, 分類誤差の代わりに, 今度は $DI(V,y)$ を測定した. 図 5 は, その実験の結果を示している. グラフ (d) が示す通り, FERK に比べて RFE は, 無関係な変数を除去した後, 影響力の小さな変数よりも先に, 影響力の大きな変数を除去してしまう危険性が高いことがわかる. また, その他のグラフは, 過適応の危険性が高い場合, RFE が, 影響力の大きな変数を誤って除去してしまう危険性が高いことを示している. これは, 偶然にも大きな DJ を与えられてしまった無関係な変数の代わりに, 影響力が大きいけれども DJ の値が小さな変数が除去されていると説明できる.

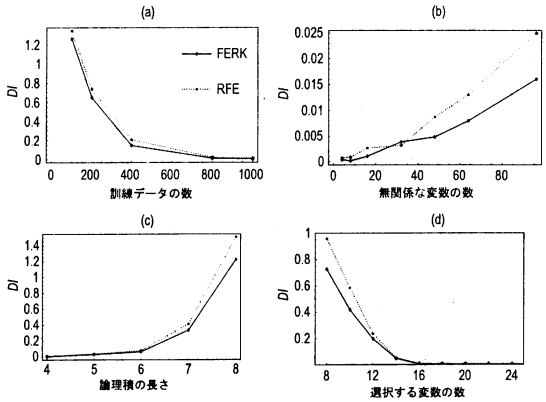

図 5　影響力の大きな変数を除去する危険性の評価

5.3　テキスト分類

実データにおける, 変数選択アルゴリズムの性能を比較するために, テキスト分類 [3] のためのデータセットである Reuter-21578 を用いた実験を行った. 実験には, 文献 [4] で用いられた, 前処理済みのデータセットのうち, "re0" と呼ばれるデータセットを用いた. このデータセットには, 1504 のニュース記事が含まれ, 各記事は, 分類カテゴリが付与された, 2886 次元の 2 値ベクトルで表現されている. 分類カテゴリは 13 個存在するが, 図 6 は, 変数選択が最も効果的だった, "trade" というカテゴリに関する分類学習の結果を示している.

この実験では, 3 つの変数選択アルゴリズム MINFO, RFE, FERK を比較した. RFE と FERK は, 共に, ブーリアンカーネル K_m^3 を用いた SVM を使用した. ただし, 計算コストの削減のために, d 個の変数が残っているときに, 一度に $10^{\lfloor \log_{10} d-1 \rfloor}$ 個の変数を除去するように修正したアルゴリズムを用いている. これら変数選択アルゴリズムを適用した後, K_m^3 を用いる SVM を使って, 分類器を学習させ, その precision/recall Break Even Point (BEP) を 8-分割交差検定により推定した.

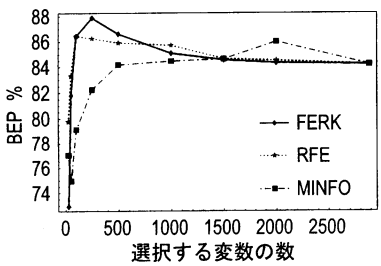

図 6　テキスト分類における特徴選択

実験の結果, FERK は, わずか 250 個の変数を用いて, およそ 87.9 という最も高い BEP を示した. それに対して, テキスト分類において, よく用いられる MINFO は, 分類に寄与する変数を絞り込めていないことがわかる. このことは, FERK のような変数間の依存関係を考慮した変数選択アルゴリズムを用いることで, 簡潔かつ高精度のテキスト分類器を構築できる可能性を示している.

6.　結　　論

本論文では, 離散データ分類器の学習のための, 変数間の依存関係を考慮した変数選択アルゴリズム FERK を提案した. FERK の特徴は, ある変数が分類に寄与するか否かを調べるために, 変数の組合せの重み付き線形和として学習された分類器から, その変数を含む全ての組合せを除去して, 分類能力の変化を分析する点である. 人工的に合成したデータセットを用いた実験により, このアルゴリズムが, 既存のアルゴリズムよりも優れていることが示された. 特に, RFE との比較して, FERK は, 影響力の大きな変数を誤って除去してしまう危険性が低く, 従って, より高い分類精度を達成可能であることが示された. さらに, FERK はテキスト分類問題に適用され, 分類に寄与する比較的小さな変数の集合を選択できる可能性が示されたが, テキスト分類における有用性を示すためには, さらなる研究が必要である.

謝辞　本研究は, 科研費 若手研究 (B)(No.14780315) の支援を一部受けている.

文　　献

[1] A.L.Blum and P.Langley: "Selection of relevant features and examples in machine learning", Artificial Intelligence, **97**, 1–2, pp. 245–271 (1997).

[2] I. Guyon and A. Elisseeff: "An introduction to variable and feature selection", JMLR, **3**, pp. 1157–1182 (2003).

[3] F. Sebastiani: "Machine learning in automated text categorization", ACM Computing Surveys, **34**, 1, pp. 1–47 (2002).

[4] G. Forman: "An extensive empirical study of feature selection metrics for text classification", JMLR, **3**, pp. 1289–1305 (2003).

[5] V. Vapnik: "The Nature of Statistical Learning Theory", Springer-Verlag (1995).

[6] N. Cristianini and J. Shawe-Taylor: "An Introduction to Support Vector Machines", Cambridge Press (2000).

[7] J. Platt: "Fast training of support vector machines using sequential minimal optimization", Advances in Kernel Methods – Support Vector Learning, MIT Press, pp. 185–208 (1998).

[8] B. Schölkopf and A. Smola: "Learning with kernels", MIT Press (2002).

[9] K. Sadohara: "Learning of Boolean functions using support vector machines", Proc. of ALT, pp. 106–118 (2001).

[10] R.Khardon, D.Roth and R.Servedio: "Efficiency versus convergence of Boolean kernels for on-line learning algorithms", NIPS, Vol. 14, pp. 423–430 (2002).

[11] K. Sadohara: "On a capacity control using Boolean kernels for the learning of Boolean functions", Proc. of ICDM, pp. 410–417 (2002).

[12] I. Guyon, J. Weston, S. Barnhill and V.Vapnik: "Gene selection for cancer classification using support vector machines", Machine Learning, **46**, pp. 389–422 (2002).

[13] K. Kira and L. Rendell: "A practical approach to feature selection", Proc. of ICML, pp. 249–256 (1992).

[14] G.H.John, R.Kohavi and K.Pfleger: "Irrelevant features and the subset selection problem", Proc. of ICML, pp. 121–129 (1994).

[15] J. Quinlan: "C4.5: Programs for Machine Learning", Morgan Kaufmann (1993).