1997年度日本オペレーションズ・リサーチ学会
春季研究発表会

正決定木によるデータ解析

02601514　京都大学　牧野 和久　MAKINO Kazuhisa
02202454　京都大学　須田 高史　SUDA Takashi
(株) 日立製作所　矢野 浩仁　YANO Kojin
01001374　京都大学　茨木 俊秀　IBARAKI Toshihide

1　序論

近年のコンピュータ技術進歩によって, 大量のデータが簡単にしかも安く蓄えられるようになり, ともすれば情報の海に飲み込まれてしまうという状況が生じている. そのため, その大量のデータから意味のある知識を抽出するための科学的手法についての研究が, 知識獲得 (knowledge acqusition) あるいはデータ発掘 (data mining) という名称の下で盛んに成りつつある.

本研究では, 正例の集合 P と, 負例の集合 N の対で表されるデータ集合の組が与えられたとき (ただし $P, N \subseteq \mathbf{R}^n$ と仮定する), P と N とを識別する判別関数 $f : \mathbf{R}^n \mapsto \{0,1\}$ を求める問題を考える. より正確に言うと, 判別関数 f とは, 任意の $v \in P$ に対し $f(v) = 1$ となり, 任意の $w \in N$ に対し $f(w) = 0$ をみたす全域関数のことである.

例えば, データのベクトル $x = (x_1, x_2, \ldots, x_n)$ はある病気を診断するための症状を表している. 具体的には, x_1 は体温を意味し, x_2 は血圧を意味するなどである. 判別関数 f を構成するということは, 与えられたデータ集合の組 (P, N) (病気の例とそうでない例を区別している) に対する診断上の説明を見つけることになる. 新しい患者を診断するために f を利用したいので, f の性能は次の 2 つの観点から評価される:

(i) 表現の簡潔さ,
(ii) 新しいデータ集合の組 (P', N') に対する分類の正確さ.

一般に, 判別関数 f を構成する過程において, f について何らかの知識あるいは仮定があらかじめ手にはいることがよくある. そのような知識は通常これまでの経験, あるいは考慮する現象を引き起こす (あるいは引き起こさない) 仕組みを分析することによって得られ. 上述の例においては, 病気を発現させる傾向にある方向性を, 各属性ごとに何らかの方法で知っていると考えるのが自然であろう. したがって, 必要ならば属性の極性を変えることによって, 判別関数 f はすべての属性について正 (あるいは単調) であると考えて一般性を失わない. 同様に, 生命保険会社は, 高齢で不健康な申込者には, 若くて健康な申込者よりも高い保険料を見積もるような判別関数を望むだろう. これらの他にも消費者の選択, 学校と輸送機関の選択そして従業員の選択など, 正判別関数によって表現されるべきデータが現実には多数存在する.

$v \leq w$ (すなわち, すべての i について $v_i \leq w_i$) となるようなデータの対 $v \in P$ と $w \in N$ が存在しないとき, 与えられたデータ集合 (P, N) は正 (positive, or monotone) であるという. $w \leq v$ ならば $f(w) \leq f(v)$ であるとき, 関数 f は正であるという. このとき, 与えられた正データ集合 (P, N) に対する正判別関数 f を構成することが我々の目的である.

本研究では, 判別関数の表現として決定木を用いる. 決定木は有向根付き木であり, 根から有向路をたどることによってベクトルが分類される. これまで ID3 [2] など決定木を構成するさまざまな方法が提案されているが, これら既存の方法は, データ集合が正であっても, 得られる判別関数の正性を保証しない [3]. 従って, 本研究では正データ集合 (P, N) が与えられたとき, それを正しく分類する正決定木の構成法を提案する.

2　正決定木

ベクトル集合 $S \subseteq \mathbf{R}^n$ に対し, $S^+ = \{w \mid w \geq v \text{ for some } v \in S\}$ と $S^- = \{w \mid w \leq v \text{ for some } v \in S\}$ を定義する. データ集合 (P, N) が正であることは $P^+ \cap N^- = \emptyset$ であることと同値である. 正データ集合 (P, N) に対して, すべての $v \in P^+$ について $f(v) = 1$ であり, すべての $w \in N^-$ について $f(w) = 0$ であるとき, 判別関数 f は準正

(quasi-positive) であると言う.

ここでは決定木を2分有向根付き木とし, 葉は0あるいは1のラベルを持ち, その他の中間の節点は, ある $i \in \{1,2,\ldots,n\}$ と定数 $c \in \mathbb{R}$ によって定まる条件 $A_{(i,c)}$ (つまり条件の対 $x_i \geq c$ と $x_i < c$) のラベルを持つ. 決定木 T が表現する判別関数が正 (準正) であるとき, T は正 (準正) であると言う.

まず正データ集合 (P,N) に対する準正決定木を構成する我々の方法について述べる. 条件 $A_{(i,c)}$ に対して,

$$P_0 = \{x \in P \mid x_i < c\},\ N_0 = \{x \in N \mid x_i < c\},$$
$$P_1 = \{x \in P \mid x_i \geq c\},\ N_1 = \{x \in N \mid x_i \geq c\}.$$
$$P_1' = P_1 \cup \{(x_1,\ldots,x_{i-1},c_{i1},x_{i+1},\ldots,x_n) | x \in P_0\}$$
$$N_0' = N_0 \cup \{(x_1,\ldots,x_{i-1},c_{i0},x_{i+1},\ldots,x_n) | x \in N_1\},$$

とする. ただし, $c_{i1} = \min\{v_i \mid v \in P_1 \cup N_1\}$ そして $c_{i0} = \max\{v_i \mid v \in P_0 \cup N_0\}$ である. ここで,

$$E^+(A_{(i,c)}) = \frac{|P_0|+|N_0'|}{|P|+|N|} I(|P_0|,|N_0'|) + \frac{|P_1'|+|N_1|}{|P|+|N|} I(|P_1'|,|N_1|)$$
$$gain^+(A_{(i,c)}) = I(|P|,|N|) - E^+(A_{(i,c)})$$

とする. 提案するアルゴリズムの各再帰的ステップでは, $gain^+$ が最大になるような条件 $A_{(i,c)}$ を選び, (P,N) は (P_0,N_0') と (P_1',N_1) に分割される. この変更によって, 得られる決定木の準正性が保証される.

アルゴリズム QP-DT
入力: 正データ集合 (P,N).
出力: (P,N) に対する準正決定木.

ステップ 1. QP-DT-AUX(P,N) を呼び, 得られた結果を出力する. 停止. □

計算手順 QP-DT-AUX(P,N)
返値: (P,N) に対する準正決定木.

ステップ 1. P (N) が空ならば, 根がラベル 0 (1) を持つ決定木を返し, 終了.

ステップ 2. 決定木の根として, $gain^+(A_{(i,c)})$ を最大とする条件 $A_{(i,c)}$ を無作為に選ぶ. (P_0,N_0') と (P_1',N_1) に対して QP-DT-AUX を呼び, それぞれ決定木 T_0 と T_1 を得る. 部分木 T_0 と T_1 を, このステップで選ばれた根 $A_{(i,c)}$ の2つの子としてつなぎ, 決定木 T を構成する. T を返し, 終了. □

上述のアルゴリズムは必ず準正決定木を構成するが, その正性は保証していない. 従って, 以下では上述のアルゴリズムで得られた準正決定木を変形し, 正決定木にする方法を述べる. T を, 与えられたデータ集合 (P,N) を正しく分類する決定木であるとし, 各葉はあるベクトル $v \in \mathbb{D}^n$ を分類するとする. ただし, 与えられたデータ集合 (P,N) に対し, $D_i = \{v_i \mid v \in P \cup N\}$, $i = 1,2,\ldots,n$ とし, $P,N \subseteq \mathbb{D}^n$ $(= D_1 \times D_2 \times \ldots \times D_n)$ が成立する. T の葉 t に対して, t によって分類されるベクトル $v \in \mathbb{D}^n$ の集合は $C^{(t)} = C_1^{(t)} \times C_2^{(t)} \times \ldots \times C_n^{(t)}$ と表される. ただし, すべての i について $C_i^{(t)} \subseteq \mathbb{D}_i$ である. $\alpha^{(t)}$ $(\beta^{(t)})$ を $C^{(t)}$ の中の最大ベクトル (最小ベクトル) とする.

計算手順 P-DT
入力: 正データ集合 (P,N).
出力: (P,N) を正しく分類する (P,N) に対する正決定木.

ステップ 1. (P,N) に対して QP-DT を呼び, 準正決定木 T を得る (ラベル 0 (1) を持つ各葉 t は例の集合 N_t (P_t) を分類するとする). 初期状態として, T の葉はどれもマークされていない.

ステップ 2. T のすべての葉 t がマークされているならば, T を出力し停止する. その他の場合, T のマークされていない葉 t を無作為に選び, マークする. t がラベル 0 を持つならばステップ 3 へ; その他の場合ステップ 5 へ.

ステップ 3. T の各葉 t' に対し, $Q = \{v \in \mathbb{D}^n \mid v \leq \alpha^{(t)}\} \cap C^{(t')}$ とする. $Q \neq \emptyset$ ならば, Q の中の最大ベクトル v^* を見つける; t' がラベル 0 を持つならば, $N_{t'} := N_{t'} \cup \{v^*\}$ とする; その他の場合, $(P_{t'},\{v^*\})$ に対してアルゴリズム QP-DT を呼んで決定木 $T_{t'}$ を得, 現在の決定木の葉 t' を $T_{t'}$ で置き換えて変形する.

ステップ 4. ステップ 2 へ戻る.

ステップ 5. T の各葉 t' に対し, $Q = \{v \in \mathbb{D}^n \mid v \geq \beta^{(t)}\} \cap C^{(t')}$ とする. $Q \neq \emptyset$ ならば, Q の中の最小ベクトル v^* を見つける; t' がラベル 1 を持つならば, $P_{t'} := P_{t'} \cup \{v^*\}$ とする; その他の場合, $(\{v^*\},N_{t'})$ に対してアルゴリズム QP-DT を呼んで決定木 $T_{t'}$ を得, 現在の決定木の葉 t' を $T_{t'}$ で置き換えて変形する.

ステップ 6. ステップ 2 へ戻る. □

Theorem 1 正データ集合 (P,N) が与えられると, アルゴリズム *P-DT* は常に, (P,N) を正しく分類する正決定木を構成する. □

なお発表当日, 実験結果を報告する.

References

[1] K. Makino, T. Suda, K. Yano, T. Ibaraki, Data analysis by positive decision trees, to appear in (*CODAS*'96).

[2] J. R. Quinlan, Induction of decision trees, *Machine Learning* 1 (1986) 81-106.

[3] 矢野浩仁, 牧野和久, 茨木俊秀, 正論理関数の部分データに基づく正決定木の構成法について, 秋季 OR 学会研究発表会アブストラクト集, 2-B-9, 1994, pp 122-123.