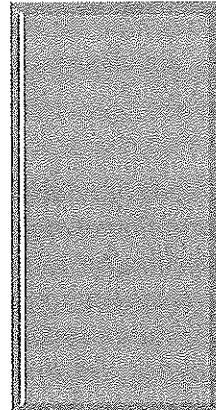


Carlos G. González (*)
Edson Campos Maia (**)

Aplicações de álgebras de Boole produto em reconhecimento de padrões

(*) Doutor em Lógica e Filosofia da Ciência pela Unicamp. Professor do Centro Universitário Nossa Senhora do Patrocínio – CEUNSP, Itu, SP.
(*) Professor de Lógica da Universidade de Sorocaba – Uniso.



RESUMO

Exceder o limite dos conceitos básicos das álgebras de Boole tem sido um dos desafios da ciência da computação — objetivando oferecer novas técnicas e algoritmos para esta ciência. Com este intuito, desenvolvemos aspectos teóricos das álgebras de Boole produto, direcionados para a produção de técnicas na área de reconhecimento de padrões, assim como alguns algoritmos e aplicações concretas destes.

ABSTRACT

A structural analysis of product Boolean algebras is used as an intuitive ground for the formulation of algorithms in pattern recognition. In this analysis, we show mathematical results to elucidate the properties of product Boolean algebras relevant to understand how its structure can be used for classification in the feature space. In this sense, we discuss two algorithms for clustering selection in feature spaces where the clusters are not clearly delimited.

Introdução

As álgebras de Boole têm sido usadas desde o início da ciência da computação como uma de suas principais bases teóricas. Entretanto, na quase totalidade dos casos, não é ultrapassado o limite dos conceitos básicos dessas álgebras, embora essa área de pesquisa receba hoje conceitos e técnicas matemáticas de considerável sofisticação. Em [1] foi desenvolvido um enfoque no qual são utilizadas álgebras de Boole produto para codificar seqüências de bits, constituindo esse trabalho um novo enfoque e técnicas de como a teoria das álgebras de Boole pode ser usada no tratamento maciço de dados. Consideraremos que esse enfoque ainda tem muito por oferecer em várias áreas da informática, e aqui continuaremos nesse caminho.

A finalidade deste artigo é desenvolver aspectos teóricos das álgebras de Boole produto direcionados para o desenvolvimento de técnicas na área de reconhecimento de padrões e assinalar alguns algoritmos e aplicações concretas. Nesse sentido, definimos uma nova noção de distância, demonstramos alguns resultados matemáticos nos quais essa noção intervém, e consideramos alguns desenvolvimentos que mostram como ela pode ser aplicada para uma melhor compreensão de possíveis métodos para processar uma informação determinada. Por último, discutimos alguns algoritmos novos embasados nessa noção e indicamos campos de aplicação.

1. Similaridade

O reconhecimento de padrões trata em geral de se um determinado objeto corresponde ou não com um determinado padrão. No caso mais comum, temos um conjunto de objetos e devemos classificá-los segundo um padrão ou um conjunto de padrões. Neste contexto, classificar significa subdividir o conjunto de objetos em vários outros conjuntos que reúnem os objetos similares quando consideramos um padrão específico. Em outras palavras, o padrão determina a similaridade entre objetos e os objetos similares são agrupados no mesmo conjunto. O estudo dos agrupamentos (clustering) constitui uma das principais áreas de pesquisa em reconhecimento de padrões.

Vejamos um exemplo. Temos uma imagem em preto e branco e o padrão que vai determinar a classificação são os tons de cinza. Os agrupamentos são os pontos (pixels) que têm cor similar. A maneira mais radical de

entender a noção de similaridade seria que dois pontos são similares se têm exatamente o mesmo tom de cinza. Nesse caso teríamos pontos iguais (ou seja, do mesmo tom de cinza) e pontos diferentes. O mínimo necessário para uma classificação é uma relação que seja satisfeita pelos elementos de cada agrupamento, ou entre os elementos de cada agrupamento e algum outro. Às vezes, essa relação é definida a partir de uma relação mais simples e um procedimento (um algoritmo). Nos casos nos quais a informação é completa em relação ao problema tratado, dados dois objetos quaisquer temos de poder dizer se essa relação simples é ou não satisfeita por eles.

Na maioria dos casos, a classificação por igual ou diferente não resulta a mais apropriada, pois existem freqüentemente nuances que requerem uma classificação mais sofisticada. A maneira mais natural de pensar isto é recorrer a algum tipo de medida da similaridade (Theodoridis, 357). A forma imediata de pensar essa medida de similaridade é como uma mensurabilidade, ou seja, uma função que estabeleça uma quantificação. Detalhadamente, se A é o conjunto dos objetos a serem classificados, então uma mensurabilidade é uma função $\mu: A \times A \rightarrow M$, sendo M o conjunto imagem da mensurabilidade. Nem todo conjunto é apropriado para ser usado como base para uma mensurabilidade. Se um conjunto tem uma ordem total e as operações aritméticas definidas nele, então a imagem da função de mensurabilidade permite um tratamento simples e intuitivo, e por isso é que são amplamente utilizadas mensurabilidades definidas sobre o conjunto dos números reais, e também mensurabilidades discretas, em graus inteiros, definidas sobre o conjunto dos números inteiros. Esta mensurabilidade formaliza uma noção de proximidade, que significa que os objetos mais similares estão, num sentido abstrato, mais próximos. O contrário da noção de proximidade é a noção de distância, que representa o grau de dissimilaridade entre dois objetos.

Para uma discussão mais geral observamos que poderiam ser usadas noções diferentes de uma mensurabilidade sobre um conjunto de números. O que aparece como realmente indispensável é que no conjunto imagem da função de mensurabilidade possa ser definida uma relação de ordem, pois com auxílio dessa relação podemos colocar limites e considerar o conjunto de elementos dentro desses limites. Se não existir tal relação, pode ser difícil ou impossível usar o conjunto da imagem para gerar uma classificação. Por exemplo, a utilização de relações que não são ordens pode necessitar de critérios adicionais para tomar conta de subgrafos cícli-

cos da relação. Em geral, precisamos estabelecer de uma maneira mais exata a noção de similaridade e, nesse sentido, a topologia dos espaços métricos fornece uma noção abstrata de distância que pode ser usada com sucesso para representar o grau de similaridade entre objetos.

2. Noções de distância

A noção de distância da geometria foi generalizada em topologia dos espaços métricos. Para isso foram abstraídas as características fundamentais, que foram resumidas na Definição 2.1.

Definição 2.1 Dado um espaço E , uma função de E em \mathbb{R} é uma distância se:

- a) $d(x, x) = 0, \forall x \in E;$
- b) $d(x, y) = d(y, x), \forall xy \in E;$
- c) $d(x, y) = 0 \Rightarrow x = y, \forall xy \in E;$
- d) $d(x, z) \leq d(x, y) + d(y, z), \forall xyz \in E.$

Em reconhecimento de padrões têm-se definido diversas noções de distância. Uma noção abstrata de distância parece uma maneira natural de representar as diferenças. De uma maneira mais geral, uma representação geométrica abstrata faz uso de um espaço de características (*feature space*). Os objetos passam a ser vetores nesses espaços e a proximidade representa a similaridade entre eles. Nesse sentido, a condição c) da Definição 2.1 enuncia a identidade dos indiscerníveis nesse espaço, ou seja, que dois objetos sem nenhuma diferença (representada pela distância 0), são iguais. Isto pode não ser necessariamente verdadeiro no uso da noção de distância em reconhecimento de padrões, pois pode ocorrer que nem todas as características estejam representadas no espaço em questão, de modo que dois objetos diferentes poderiam ser indistinguíveis nesse espaço. De qualquer maneira, isto não representa nenhum problema sério, pois podemos pensar os objetos do espaço de características como classes de equivalência que reunam os objetos diferentes, mas de distância 0 entre eles.

A quase totalidade dos casos que alguma noção de distância tem sido utilizada em reconhecimento de padrões, segue a Definição 2.1. Em alguns poucos casos, se vê como mais pertinente uma função com imagem nos inteiros, devido a que o espaço de características é discreto, implicando que a noção mais apropriada também seja discreta. Várias noções de proximi-

dade foram definidas, assim como também alguns critérios gerais. Uma noção de proximidade é, em algum sentido, a noção contrária à de distância. Mas as noções de distância definidas ou implícitas nesses casos ou são funções com imagem nos reais ou nos inteiros, ou não satisfazem as condições da Definição 2.1 (não seriam distâncias de espaços métricos).

Um dos usos comuns dos espaços de características, tem como objetos seqüências de bits. Nesses casos é usada majoritariamente uma função com imagem nos inteiros¹. Por exemplo, uma definição geral de uma função de distância simples pode estar baseada na quantidade de bits que diferem duas seqüências. Em tal caso, a função passa a estar definida com imagem nos inteiros (mais especificamente, nos naturais). As condições a) a d) da Definição 2.1 podem ser verificadas. Para a condição a) verificamos que uma seqüência não difere em nenhum bit com ela própria. A condição b) também é imediata. Para c), se duas seqüências não diferem em nenhum bit, então são a mesma seqüência. A condição d) precisa de um argumento: sejam x e z duas seqüências e seja $d(x,z) = \varepsilon$. Seja y uma seqüência de bits. Suponhamos, sem perda de generalidade, que $d(x,y) = \sigma$, com $\sigma < \varepsilon$. Consideremos agora os bits nos quais x e y coincidem. Vemos que y e z devem diferir no mínimo em $\varepsilon - \sigma$ bits. Caso contrário, se y e z diferem em κ bits, com $\kappa < \varepsilon - \sigma$, como esses são os mesmos bits em x e em y , então os bits considerados de x e de z também difeririam só em κ , sendo $d(x,z) = \kappa + \sigma < \varepsilon$, o que é uma contradição.

Esta noção de distância é, sem dúvida, interessante e de múltiplas aplicações. Mas o problema que surge é que não diferencia quais são os bits diferentes. Se pensarmos, por exemplo, numa aplicação típica de agrupamento (clustering) estaremos mais interessados em encontrar um conjunto que seja diferente do representante em determinados bits e não em n bits quaisquer. Com esta motivação foi que pensamos numa modificação drástica do conjunto imagem da função de distância. Para representar a nossa noção de distância, em lugar de tomar valores em conjuntos como os reais ou os inteiros, usamos uma álgebra de Boole produto. Este enfoque de utilizar álgebras de Boole produto para representar seqüências de bits já foi utilizado por um dos autores em [1].

Uma álgebra de Boole pode ser definida a partir de 3 operadores ou de uma relação de ordem. Segundo a primeira definição, uma álgebra de Boole

¹ Na verdade, às vezes a função é definida com imagem nos reais, mas todos pontos que são imagem de algum elemento são inteiros.

é uma séxtupla $\langle A, \wedge, \vee, \sim, 0, 1 \rangle$ onde $\langle A, \wedge, \vee \rangle$ é um reticulado distributivo complementado com mínimo 0 e máximo 1. Na segunda definição uma álgebra de Boole é um conjunto ordenado $\langle A, \leq, 0, 1 \rangle$ no qual $\langle A, \leq \rangle$ constitui novamente um reticulado distributivo complementado com mínimo 0 e máximo 1.

A equivalência das duas definições pode ser demonstrada definindo $x \wedge y = x \Leftrightarrow x \leq y$. Dadas álgebras de Boole $\langle A_0, A_1, \dots, A_{n-1} \rangle$ pode ser definida uma nova álgebra de Boole, a álgebra produto, que tem como domínio o produto cartesiano $A_0 \times A_1 \times \dots \times A_{n-1}$ e as operações definidas ponto a ponto. Se entendermos um bit como uma álgebra de Boole binária, então a maneira natural de conceber uma seqüência de bits é a álgebra produto, onde o mínimo está representado pelo elemento que possui todos os bits zerados e o máximo com aquele que tem todos os bits com valor 1.

Se olharmos às operações e relações necessárias na Definição 2.1, vemos que precisamos de uma operação + e de uma relação ρ . A relação habitual em álgebras de Boole produto é definida ponto a ponto, e foi a utilizada em [1]. O candidato mais comum a representar a soma, é a união (join) que em álgebras de Boole é simbolizado por " \vee ". O problema é agora encontrar uma noção de distância definível nessas álgebras. A operação booleana que tem como resultado uma representação dos bits diferentes é o XOR, pois dá como resultado 1 quando os bits são diferentes e 0 no caso contrário². Desta maneira, sendo x e y duas seqüências de bits, definimos $d(x, y) = x \text{ XOR } y$. Devemos, então, verificar as condições similares à Definição 2.1. Para isso precisamos da definição geral de distância em álgebras produto.

Definição 2.2. Seja B uma álgebra de Boole produto. $d : B \times B \rightarrow B$ é uma distância se:

- a) $d(x, x) = 0, \forall x \in B$;
- b) $d(x, y) = d(y, x), \forall xy \in E$;
- c) $d(x, y) = 0 \Rightarrow x = y, \forall xy \in E$;
- d) $d(x, z) \leq d(x, y) \vee d(y, z), \forall xyz \in E$.

Como no caso anterior, a), b) e c) são de verificação imediata. Por exemplo, para b) temos que $x \text{ XOR } y = y \text{ XOR } x$. Para d) precisamos de um argumento. Como as operações foram todas definidas ponto a ponto (ou seja,

² A operação é definida para um bit, mas nos processadores atuais essa operação é aplicada bit a bit, de modo que processa seqüências de bits.

bit a bit), é suficiente argumentar para um bit. Denominemos x_n, y_n e z_n o n -ésimo bit de x, y e z , respectivamente. Suponhamos, sem perda de generalidade, que $x_n \text{ XOR } z_n = 1$, ou seja esse bit é diferente em x e em z . Temos de provar que $x_n \text{ XOR } y_n \vee y_n \text{ XOR } z_n = 1$. Há somente duas possibilidades: ou $x_n = y_n$ ou $y_n = z_n$. Portanto, $x_n \text{ XOR } y_n \vee y_n \text{ XOR } z_n = 1$.

3. Distâncias e agrupamentos

A maneira mais simples de formar agrupamentos é considerar os elementos do espaço de características mais próximos, no sentido que a função de distância definida nesse espaço seja menor que um certo para elementos do mesmo agrupamento. Uma outra maneira, consiste em fixar também uma distância máxima, mas, em lugar de estabelecer que para dois elementos a e b quaisquer do mesmo agrupamento vale $d(a,b)$, determinar que existe um caminho $a = a_0, a_1, \dots, a_n = b$ tal que $d(a_i, a_{i+1})$ para elementos dessa seqüência. Com o primeiro enfoque, o melhor que pode acontecer é que exista um conjunto de representantes. Usamos propositadamente o termo “conjunto de representantes”, porque os agrupamentos particionam o espaço, pelo qual existe uma relação de equivalência implícita que define os agrupamentos. Um conjunto de representantes pode ser pensado como um elemento central para cada agrupamento, de modo que para realizar a partição do espaço é suficiente reunir todos os elementos que se encontram a uma distância do representante. Um tal conjunto de representantes é chamado completo e definido como segue:

Definição 3.1. Dado um espaço E , e uma distância ε , $D \subseteq E$ é um conjunto completo de representantes se:

- $d(x,y) > 2\varepsilon; \forall x, y \in D$;
- $\forall x \in E; \exists y \in D \quad d(x,y) \leq \varepsilon$.

A expressão 2ε deve ser entendida somente como uma abreviatura de $\varepsilon + \varepsilon$, pois uma noção de distância que não tenha imagem nos conjuntos habituais, como os reais e inteiros, pode não ter ou não ficar claro o que seria o produto. A existência de um tal conjunto de representantes é suficiente para induzir uma partição do espaço em agrupamentos de objetos próximos.

Proposição 3.1. Seja D um conjunto de representantes satisfazendo as condições da Definição 3.1. Então D induz uma partição do espaço.

Demonstração: Para cada $x \in D$, definimos $\bar{x} = \{y \in E : d(x, y) \leq \varepsilon\}$. Devemos provar que $P = \{x : x \in D\}$ é uma partição do espaço E . Em primeiro lugar, observamos que cada $x \neq \emptyset$, pois $x \in x$. Para ver que $\cup P = E$, só devemos argumentar no caso $E \subseteq \cup P$, pois a outra inclusão segue-se da definição das classes x . Seja $x \in E$. Então $\exists y \in D$, por b) da Definição 3.1, tal que $d(x, y) \leq \varepsilon$, e portanto $x \in y \in P$. Para $x \neq y$, devemos ver que $x \cap y = \emptyset$. Para um raciocínio pelo absurdo, suponha que $\exists z \in \bar{x} \wedge z \in \bar{y}$. Então, temos que $d(x, z) < \varepsilon$ e $d(y, z) < \varepsilon$. Somando essas duas desigualdades, temos $d(x, z) + d(y, z) < 2\varepsilon$. Usando b) e d) da Definição 2.1, temos que $2\varepsilon < d(x, z) + d(y, z) = d(x, z) + d(y, z) < 2\varepsilon$, o que é uma contradição.

Dado um espaço de características, esse conjunto completo de representantes pode estar entre os vetores que devemos classificar ou não. Neste último caso, esse conjunto completo de representantes é simplesmente uma ferramenta para definir o agrupamento.

Se o espaço de características é isomorfo a $R^n : n \in N^+$, então a existência de um conjunto completo de representantes não será freqüente. Para esclarecer mais este problema, vejamos as situações possíveis no agrupamento:

1) O que determina o agrupamento tem importância por si próprio. Por exemplo, numa imagem de radiotelescópio as diferentes radiofreqüências têm relevância por si mesmas;

2) no processamento de imagem denominado posterização a situação é intermediária. Não é tão rígida como a anterior, mas também não seria desejável que existam duas áreas diferentes de tons muito próximos para o olho humano;

3) numa aplicação para marketing onde se pretenda obter perfis de clientes na base de suas compras, a função de distância utilizada para o agrupamento não tem valor por si mesma. Poderia, por exemplo, priorizar-se a obtenção de um número pré-determinado de agrupamentos, ou seja, de perfis de clientes.

Se voltarmos à nossa definição de conjunto completo de representantes, observamos que a distância que caracteriza os agrupamentos tem sido considerada de uma maneira absoluta. Uma maneira de relativizar isto é considerar um ε para cada agrupamento A .

Definição 3.2. Dado um espaço E, seja D um conjunto de duplas $\langle x, ex \rangle$ com $x \in E$ e ex uma distância mínima dependendo de x. Então dizemos que D é um conjunto relativo de representantes se:

- $d(x, y) > \varepsilon_x + \varepsilon_y; \forall \langle x, \varepsilon_x \rangle, \langle y, \varepsilon_y \rangle \in D;$
- $\forall x \in E; \exists \langle y, \varepsilon_y \rangle \in D \quad d(x, y) \leq \varepsilon_y.$

O primeiro problema que encontramos na aplicação dessas noções é que freqüentemente não existem conjuntos de representantes satisfazendo algumas das definições acima. Um outro problema é que se o conjunto imagem da função de distância for R^n , então estamos particionando um conjunto de pontos em hiper-esferas de raio ε ou de raio ε_x , de modo que em alguns casos o tal particionamento (finito) pode não existir, independentemente da existência ou não do conjunto de representantes. Particionar usando hiper-esferas parece uma maneira mais problemática que usando hipercubos, mas os hipercubos em R^n podem não agrupar os objetos mais próximos, por exemplo, se estiver perto de um vértice. Ou seja, dado um particionamento de R^n em hipercubos um elemento de um agrupamento pode estar mais distante do representante (centro) do agrupamento que do representante de um outro agrupamento. Isto não acontece se tomarmos medidas sobre uma álgebra de Boole, porque as medidas que satisfazem à Definição 2.2 definem agrupamentos que correspondem aos hipercubos booleanos.

Na realidade, dada uma noção de distância numa álgebra de Boole, é muito fácil definir uma partição do espaço gerada pela função de distância e um predeterminado.

Para isso, fixamos uma função de distância d (na realidade, o), e uma distância máxima ε . Então

$$H_{x,\varepsilon} = \{y \in B : d(x, y) \leq \varepsilon\}$$

é um hipercubo booleano, e também uma álgebra de Boole³ com as operações restritas. Representamos o mínimo dessa álgebra $\wedge_{H_x} = x_0 \wedge x_1 \wedge \dots \wedge x_n$, com $x_i \in H_{x,\varepsilon}$. Quando o ε fique claro pelo contexto, falaremos de H_x . Para o máximo, usamos $\vee_{H_x} = x_0 \vee x_1 \vee \dots \vee x_n$. Um conceito interessante é o de máscara, que anotamos \equiv_{H_x} e que é definido da seguinte maneira. Seja x_i o i -ésimo bit de x . Então

³ Não é sub-álgebra no sentido que o mínimo e o máximo de H_x podem não coincidir com os de B , mas as operações booleanas de H_x são a restrição das de B .

$$(\equiv_{H_x})_i = 1 \text{ se e somente se } \forall a, b \in H_x; a_i = b_i$$

Em outras palavras, \equiv_{H_x} tem o i -ésimo bit setado se e somente se esse bit é igual para todos os elementos de H_x . É fácil ver que, para $a \in H_x$, vale $a \wedge \equiv_{H_x} = \wedge_{H_x}$. Se definirmos, para $A \subseteq B$, $\neg A = \{\neg a : a \in A\}$, então temos $a \vee \neg \equiv_{H_x} \vee_{H_x}$. Além disso, para $a, b \in H_x$, temos que $a \wedge \equiv_{H_x} = b \wedge \equiv_{H_x}$. É fácil ver que qualquer elemento de B , quando é considerado uma máscara, partitiona B em classes de equivalência. Cada uma dessas classes é uma álgebra de Boole, com as operações restritas e o mínimo e o máximo definidos a partir de um elemento qualquer da álgebra e a máscara. Para cada classe pode ser encontrada uma distância $\varepsilon \in B$ tal que, se a e b pertencem à mesma classe, então $d(a, b) = \varepsilon$. Em síntese, qualquer objeto da álgebra é um candidato para gerar uma partição da álgebra em objetos similares.

Quando analisamos as noções de distância, falamos que podia ser definida uma distância baseada na quantidade de bits diferentes entre duas seqüências. Primeiro estabelecemos:

Definição 3.3. Denominamos $q(x)$ à quantidade de bits setados de $x \in B$.

Agora podemos definir a distância de $Q(x, y)$, como:

Definição 3.4. $Q : B \times B \rightarrow N \quad Q(x, y) = q(\equiv_{\{x, y\}})$

4. Algoritmos

As idéias analisadas geram uma nova perspectiva para aplicações, pois as maneiras puramente pragmáticas ou parciais de trabalhar com seqüências de bits dão lugar a uma teoria mais geral que se apresenta como um marco teórico poderoso. Consideremos, por exemplo, algoritmos do tipo de morfologia binária (BMCA), (Theodoridis, 516). O primeiro passo, a discretização do espaço de características pode não ser necessário, pois o espaço já é discreto. Pode ser necessário considerar a “resolução” (Theodoridis, 516) ou granulosidade, pois pode acontecer que não seja possível analisar todos os casos: se tivermos 32 bits e quisermos analisar todas as possibilidades, teríamos que analisar $2^{32} \approx 4,29 \cdot 10^9$ casos. Uma solução simples para esse problema consiste em considerar blocos de bits. Em nosso exemplo, poderíamos dividir um elemento de 32 bits em 16 elementos de

2 bits, e considerar somente esses blocos de 2 bits para todo tratamento posterior. Na realidade, estamos definindo um homorfismo numa álgebra de 16 bits, de modo que podemos utilizar os conceitos acima esquecendo a álgebra original.

Como exemplo de aplicação destas idéias queremos apresentar um algoritmo para agrupamento. Seja B uma álgebra de Boole produto. Definimos primeiro:

Definição 4.1. Para cada $n \in N$, seja F_n a família de distâncias de grau n , definida por:

$$F_n = \{e \in B : q(e) = n\}$$

Podemos enunciar agora o seguinte algoritmo:

Algoritmo 4.1. Detecção de agrupamentos por criação de vales (Cluster detect algorithm by valley creation, CDAVC).

Inicialmente considere todos os elementos de E como não processados.

Fixe $b, n \in N$.

Repita

Escolha um elemento não processado x .

Considere no que se segue somente os elementos não processados.

Se houver alguma classe $H_{x,e}$ para cada $e \in F_b$ com mais de n elementos:

a) Escolha as $H_{x,e}$ com a quantidade máxima de elementos.

b) Para cada $H_{x,e}$, defina $H'_{x,e} = H_{x,e} - \cap_{\delta \in F_b} H_{x,\delta}$.

c) Para cada $y \in H'_{x,e}$, escolha um dos $H'_{y,e} = H_{y,e} - \cap_{\delta \in F_b} H_{y,\delta}$ com maior quantidade de elementos, forme um novo agrupamento e considere x e os elementos de $H_{y,e}$ como processados.

Se não,

considere x como processado

Fim se.

Até não haver elementos não processados.

A vantagem deste algoritmo fica em evidência quando o espaço de características apresenta diferenças de densidade de elementos em diferentes áreas, mas não existem "vales" ou áreas sem elementos que facilitem a separação dos agrupamentos, mas também não queremos demasiadamente poucos agrupamentos, mas separar aqueles que representam as áreas de maior densidade. Com efeito, o significado de tirar a interseção das classes

selecionadas é, justamente, criar essas espécies de vales artificiais para tentar dividir regiões grandes demais. Nesse sentido, tem uma certa analogia com outros algoritmos, como os baseados em k -vizinhos-mais-próximos (Velaid, 88, Teorodidis, 485). Resulta interessante ver como a estrutura da álgebra de Boole permite um procedimento sofisticado que em outros espaços, com o R^n , ficariam triviais. As diferentes distâncias em F_b formalizam as tentativas de definir as classes do elemento x em diferentes direções do espaço booleano. Se pensarmos o espaço de características, então essas distâncias representam a tentativa de encontrar conjuntos de características compartilhadas entre vários objetos. Isso dá uma idéia de que pode ser interessante em certos casos colocar um peso a cada elemento da álgebra de Boole, de modo que no passo a) do Algoritmo 4.1, em lugar de selecionar as classes com maior número de elementos, somaríamos os pesos dos elementos de cada classe e selecionaríamos as classes de maior peso.

Um outro algoritmo que queremos apresentar lembra o jogo do caçador de minas:

Algoritmo 4.2 Caçador de minas (Minesweeper).

Determine um $k \in N$ e uma distância ε .

Divida de maneira uniforme o espaço booleano em hipercubos (álgebras com a mesma quantidade de elementos).

Para cada hipercubo, calcule a quantidade de objetos.

Para cada hipercubo, adicione seus elementos aos de todos os hipercubos vizinhos, o que resulta num número n para cada hipercubo.

Considere todos os hipercubos com $n < k$ como processados e os demais como não processados.

Considere todos os objetos como não processados

Repita

Escolha um c entre os hipercubos não processados de maior n .

Se existir elementos não processados nesse hipercubo

Escolha um elemento x não processado desse hipercubo.

Forme um agrupamento $A = \{y \in B : d(x, y) \leq \varepsilon\}$, usando somente y não processados.

Considere c e todos os elementos de A como processados.

Se não

Considere c como processado.

Fim se.

Até todos os hipercubos ser processados.

Uma maneira de encontrar os hipercubos vizinhos consiste em usar o mínimo de cada um dos hipercubos (que são, como vimos, álgebras de Boole com as operações restritas). Dois hipercubos são vizinhos se os respectivos mínimos x e y são diferentes e contíguos: ou $x \leq y$ e $\neg \exists z (x \leq z \wedge z \leq y)$ ou $y \leq x$ e $\neg \exists z (y \leq z \wedge z \leq x)$. Novamente aqui podemos considerar pesos, da maneira discutida no algoritmo anterior.

4. Conclusão

O uso das álgebras de Boole produto para representar seqüências de bits parece ser um terreno de amplas perspectivas e que possibilita novas pesquisas e desenvolvimentos. A definição de uma noção de distância mostrada neste artigo provou ser frutífera proporcionando novos resultados e mostrando tanto como podem ser reformuladas as aplicações conhecidas como produzir outras fundamentalmente novas. Os algoritmos mostrados pretendem fundamentar as últimas. A estrutura das álgebras de Boole produto pode parecer complexa numa primeira visão, mas logo se mostra como altamente motivadora e oferece um novo enfoque no tratamento de problemas conhecidos.

Um caminho que merece ser percorrido é a reformulação de algoritmos conhecidos para identificação de agrupamentos levando em consideração a estrutura mais sutil que resulta da utilização de uma álgebra de Boole como conjunto imagem da função de medida. Uma promissora área está se abrindo para novas pesquisas.

REFERÊNCIAS

1. BELAÏD, A., BELAÏD, Y. *Reconnaissance des formes, méthodes et applications*. Paris: Inter, 1992.
2. GONZÁLEZ, Carlos G. Aplicação de produtos de álgebras de Boole para codificar seqüências de bits In: Anais do Congresso de Lógica Aplicada à Tecnologia Laptec'2000. Faculdades SENAC de Ciências Exatas e Tecnológicas São Paulo (SP). São Paulo: Pléiade, 2000 p. 679-689.
3. THEODORIDIS, S.; KOUTROUMBAS, K. *Pattern Recognition*. San Diego: Academic Press, 1999.