Algoritmos de Agrupamento Divisivos com Pré-Determinação do Número de Grupos

Paulo Rogério Nietto, Maria do Carmo Nicoletti
Faculdade de Campo Limpo Paulista - FACCAMP
Campo Limpo Paulista - SP
{pnietto@outlook.com,carmo@cc.faccamp.br}

Abstract. *Hierarchical clustering algorithms (HC) construct a cluster hierarchy also known as dendrogram. They have, as main advantages, flexibility in regard the level of granularity as well as applicability to any attribute types. HC algorithms are categorized into agglomerative (bottom-up) and divisive (top-down). A hierarchical divisive clustering (HD) algorithm starts with a single cluster containing all patterns and recursively splits the most appropriate cluster. The main goal of this research is to identify data characteristics which promote a good performance of HD algorithms as well as to experiment possible data-preprocessing strategies for determining the number of clusters in a clustering, to be used as a stopping condition by HD algorithms. The paper describes the ongoing work and the main steps for achieving such goals.*

Resumo. *Algoritmos de agrupamento hierárquicos (AH) constroem uma hierarquia de agrupamentos também conhecida como dendrograma. Eles tem, como principais vantagens, flexibilidade em relação ao nível de granularidade bem como aplicabilidade a qualquer tipo de atributo. AH são categorizados em aglomerativos (bottom-up) e divisivos (top-down). Um algoritmo hierárquico divisivo (HD) começa com um único agrupamento contendo todos os padrões e, recursivamente, divide o grupo mais apropriado. O principal objetivo da pesquisa é identificar características de padrões de dados que promovem um bom desempenho de algoritmos HD e experimentar possíveis estratégias de pré-processamento de padrões, com o objetivo de determinar o número de grupos do agrupamento buscado, para ser usado como critério de parada de algoritmos HD. O artigo descreve o trabalho sendo conduzido e os principais passos para atingir os objetivos propostos.*

1. Introdução

Aprendizado de Máquina (AM) é uma subárea da Inteligência Artificial com foco em investigações e propostas de novos formalismos e algoritmos computacionais neles baseados, com vistas a dotar computadores com a habilidade de realizar aprendizado automático. Ao longo das últimas décadas inúmeras ideias de como viabilizar AM tem sido propostas e implementadas. O sucesso e a popularidade de métodos automáticos de aprendizado se devem, principalmente, aos chamados algoritmos de aprendizado indutivo de máquina (AIM). Inúmeras referências abordam revisões de algoritmos de AM e, particularmente, de AIM, tais como [Mitchell 1997] [Duda *et al.* 2001] [Murtagh & Contreras 2011] e [Witten *et al*. 2011].

Para que algoritmos de AIM aprendam a(s) expressão(ões) que representam conceito(s), é mandatório que esteja disponível a tais algoritmos um conjunto de padrões

(de dados), chamado *conjunto de treinamento*, que agrupa instâncias concretas do(s) conceito(s) a ser(em) aprendido(s). Via de regra cada padrão de um conjunto de treinamento é representado por um vetor de valores de atributos e de uma *classe* associada (*i.e.*, o conceito que o padrão representa). Algoritmos de AIM que fazem uso da informação da classe associada a cada padrão do conjunto de treinamento são caracterizados como algoritmos *de aprendizado supervisionado*. Nem sempre, entretanto, a classe participa da descrição dos padrões. Para que o aprendizado de máquina possa ainda assim ser realizado usando tais padrões, têm sido propostos, ao longo dos anos, um grupo de algoritmos, caracterizados como de *aprendizado não-supervisionado*, capazes de aprender sem supervisão i.e., sem a informação extra da classe à qual o padrão pertence (ver [Theodoridis & Koutroumbas 1999], [Duda *et al.* 2001] e [Bandyopadhyay & Saha 2013]). Via de regra algoritmos e técnicas de agrupamento são utilizados quando não existe classe associada aos padrões; o uso clássico de agrupamento é em situações em que o conjunto original de dados deve ser particionado no que se convenciona chamar de 'grupos naturais'. Esses grupos, presumivelmente, refletem alguma tendência inerente ao domínio de conhecimento (de onde provém os padrões de dados), tendência essa que causa alguns dos padrões serem mais similares entre si do que mais similares a alguns outros, do mesmo conjunto.

Em várias taxonomias propostas na literatura, com o objetivo de organizar os algoritmos de agrupamento (ver [Xu *et al.* 2007], [Jain 2010]), uma categoria sempre presente é a dos chamados *algoritmos hierárquicos* que, via de regra, produzem um conjunto de agrupamentos aninhados, organizados como uma árvore hierárquica, que pode ser visualizada como um dendrograma. Os algoritmos hierárquicos, por sua vez, se subdividem em dois grupos: (1) *hierárquicos aglomerativos*, que se caracterizam por iniciar o processo com um agrupamento em que cada um dos padrões é considerado um grupo do agrupamento e (2) *hierárquicos divisivos*, que se caracterizam por iniciar o processo com um agrupamento tendo apenas um grupo, aquele com todos os padrões fornecidos e que, a cada passo, divide um dos grupos do agrupamento, até que cada grupo contenha um padrão (ou então, existam apenas k grupos, em que k é fornecido como parâmetro), em uma abordagem caracterizada como *top-down*.

Este artigo descreve os passos já realizados relativos a um projeto de pesquisa em nível de mestrado voltado à investigação empírica de algoritmos de agrupamento caracterizados como hierárquicos divisivos, que tem por objetivos: (1) experimentar possíveis estratégias para serem incorporadas em um pré-processamento inicial dos dados, de maneira a determinar, de antemão, quantos grupos o agrupamento a ser construído deveria ter e (2) identificar as características de domínio de dados que promovem um bom desempenho de algoritmos hierárquicos divisivos, considerando a informação obtida em (1). Na Seção 2 é introduzida a notação formal empregada nas descrições e são apresentados alguns conceitos necessários para o entendimento do que segue, bem como suas respectivas formalizações. A Seção 3 inicialmente apresenta uma descrição geral do Esquema Divisivo Generalizado (EDG), algoritmo fundamental ao estudo de agrupamentos divisivos para, então, apresentar seu pseudocódigo. A Seção 4 descreve os próximos passos para a continuação e finalização do projeto de pesquisa.

2. Estabelecimento da Notação Empregada e Conceitos Relevantes

Seja $X = \{P_1, P_2, ..., P_N\}$ um conjunto com N padrões de dados, cada um deles, P_i, $1 \leq i \leq N$, descrito por M atributos, $A_1, A_2, ..., A_M$. Um K-agrupamento de X é uma partição de X em K conjuntos (grupos), $G_1, G_2, ..., G_K$. Uma vez que um K-

agrupamento é definido como uma partição do conjunto X, as três condições a seguir devem ser verificadas: (1) $G_i \neq \emptyset$, i = 1, ..., K (cada um dos grupos do agrupamento é não-vazio); (2) $\bigcup_{i=1}^{K} G_i = X$ (a união de todos os grupos produz o conjunto X); (3) $G_i \cap G_j = \emptyset$, i ≠ j e i, j = 1, ..., K (os grupos são dois-a-dois disjuntos).

Considere dois padrões de um espaço M-dimensional, $P_i = (P_{i_1}, P_{2_2}, ..., P_{i_M})$ e P_j, $P_j = (P_{j_1}, P_{j_2}, ..., P_{j_M})$. A Eq. (1) define a distância euclidiana entre eles.

$$d(P_i, P_j) = \sqrt{\sum_{k=1}^{M} \left(P_{i_K} - P_{j_K}\right)^2} \qquad (1)$$

Considere novamente o conjunto de N padrões $X = \{P_1, P_2, ..., P_N\}$ e considere dois agrupamentos dos padrões de X, identificados por AG_1 e AG_2, respectivamente. O agrupamento AG_1, contendo k grupos, está aninhado no agrupamento AG_2 que contém r (< K) grupos, (notado por $AG_1 \langle AG_2$) se cada grupo em AG_1 for subconjunto de um conjunto de AG_2 e, pelo menos um grupo de AG_1 for um subconjunto próprio de um elemento de AG_2. Seja $X = \{P_1, P_2, P_3, P_4, P_5, P_6, P_7\}$. O agrupamento $AG_1 = \{\{P_1, P_4, P_7\}, \{P_2, P_3, P_5\}, \{P_6\}\}$ está aninhado em $AG_2 = \{\{P_1, P_4, P_6, P_7\}, \{P_2, P_3, P_5\}\}$. Entretanto, AG_1 não está aninhado nem em $AG_3 = \{\{P_1, P_3, P_5, P_7\}, \{P_2, P_4, P_6\}\}$ ou tampouco em $AG_4 = \{\{P_1, P_2, P_4, P_7\}, \{P_3, P_5, P_6\}\}$.

3. O Esquema Divisivo Generalizado (EDG)

O Esquema Divisivo Generalizado (EDG) (*Generalized Divisive Scheme (GDS)*), como descrito em [Theodoridis & Koutroumbas 1999], é uma proposta de um procedimento geral para algoritmos hierárquicos divisivos, em que algumas das funções empregadas podem ser customizadas, na dependência da aplicação considerada. Devido à possibilidade de customização tal esquema pode dar origem a diferentes 'instanciações', muitas vezes consideradas novos algoritmos.

No processo iterativo conduzido pelo EDG, o t-ésimo agrupamento produzido tem t+1 grupos. No que segue, G_{tj} representa o j-ésimo grupo do agrupamento AG_t, para t = 0, ..., N–1, j = 1, ..., t+1. Seja $g(G_i,G_j)$ uma função de dissimilaridade, definida para todos os possíveis pares de grupos que pertencem a um agrupamento. O agrupamento inicial é estabelecido como $AG_0 = \{X\}$. Para determinar o próximo agrupamento, são considerados todos os possíveis pares de grupos que formam uma partição de X. Entre eles, é escolhido o par, denotado por (G_{11},G_{12}), que maximiza g. Esses grupos formam o próximo agrupamento $AG_1 = \{G_{11},G_{12}\}$. No próximo passo são considerados todos os grupos produzidos por G_{11} e, dentre eles, é escolhido aquele que maximiza g (note que uma função de similaridade pode também ser usada e, se esse for o caso, é escolhido o par de grupos que minimiza g). O mesmo procedimento é repetido para G_{12}. Assuma agora que dos dois pares de grupos resultantes, aquele originário de G_{11} obtém o maior valor de g; seja esse par notado por (G_{11}^1, G_{11}^2). O novo agrupamento então consiste de G_{11}^1, G_{11}^2 e G_{12}. Renomeando esses grupos como G_{21}, G_{22} e G_{23}, respectivamente, o segundo agrupamento produzido é $AG_2 = \{G_{21}, G_{22}, G_{23}\}$. Continuando com o mesmo processo, são formados todos os agrupamentos subsequentes. A Figura 1 mostra um pseudocódigo alto nível do EDG (GDS).

Como descrito no pseudocódigo da Figura 1, algoritmos hierárquicos divisivos inicializam o processo de construção do agrupamento considerando um único agrupamento (que é o próprio conjunto original de padrões (treinamento)) e, subsequentemente, vão subdividindo o(s) grupos existente(s) com o objetivo de encontrar a melhor subdivisão. Tipicamente um grupo é subdividido em dois subgrupos (quando a estratégia de bisecção for usada), o que induz a construção de uma árvore binária de grupos (a hierarquia).

procedure EDG (X, AG_t)
Input: X = {P_1, P_2,...,P_N}
Output: AG_t
1. **begin**
2. $AG_0 \leftarrow \{X\}$ % agrupamento inicial
3. Nro_G $\leftarrow$ 1
4. t $\leftarrow$ 0
5. **repeat**
6. t $\leftarrow$ t + 1
7. **for** i $\leftarrow$ 1 to t **do**
8. entre todos os possíveis pares de grupos (G_r,G_s) que formam uma partição de $AG_{t-1,i}$, encontrar o par ($G_{t-1,i}^1$,$G_{t-1,i}^2$) que produza o maior valor de g.
9. Considerando os t pares definidos no comando anterior, escolher aquele que maximiza g. Suponha que esse par seja o ($G_{t-1,j}^1$,$G_{t-1,j}^2$).
10. $AG_t \leftarrow (AG_{t-1} - \{G_{t-1,j}\}) \cup \{G_{t-1,j}^1, G_{t-1,j}^2\}$.
11. renomear os grupos de AG_t
until cada padrão esteja em um grupo unitário.
return AG_t
end procedure

Figura 1. Pseudocódigo do Esquema Divisivo de Agrupamento (EDG).

Como comentado em [Berthold *et al.* 2010], a cada iteração de um algoritmo divisivo duas perguntas devem ser consideradas: (1) qual dos grupos deve ser subdividido? e (2) como dividir o grupo escolhido em dois subgrupos? A resposta à questão (1) geralmente implica o uso de medidas de validação para avaliar a qualidade de um dado grupo e, talvez aquele com pior qualidade possa ser o melhor candidato à uma próxima subdivisão. Se o número de subgrupos a ser obtido a partir de um grupo for conhecido ou, então, assumido ser fixo (como por exemplo 2, na estratégia da bisecção, como na Figura 1), então algoritmos de agrupamentos que particionam grupos em um número fixo de subgrupos podem ser usados.

Como discutido em [Berthold *et al.* 2010], na abordagem *bottom-up* adotada por algoritmos hierárquicos aglomerativos as decisões são baseadas em informação local (distância entre vizinhos), a qual não é muito conveniente em situações em que padrões de dados têm fronteiras difusas e/ou têm ruídos. Em tais situações a abordagem *top-down* implementada por algoritmos divisivos pode fornecer melhores resultados, uma vez que a distribuição global dos padrões é considerada desde o início do processo de agrupamento. Por outro lado, entretanto, o esforço computacional realizado por algoritmos divisivos é maior; esse inconveniente, entretanto, pode ser minimizado por meio da interrupção do processo de construção da hierarquia após os primeiros passos uma vez que, usualmente, o número desejado de grupos é pequeno.

Para a investigação empírica dos algoritmos hierárquicos divisivos selecionados e de métodos de determinação do número de grupos, é parte do projeto o desenvolvimento de um ambiente computacional para as experimentações com os algoritmos a serem

implementados. Pretende-se usar uma versão do K-Means como *baseline* para comparações. A tentativa de identificação das características de domínio de dados que promovem bom desempenho de um algoritmo HD será feita empiricamente, com foco principalmente com variações em forma e densidade de grupos.

4. Comentários Finais e Próximas Etapas

O trabalho até então desenvolvido, que envolveu levantamento bibliográfico e estudo de algoritmos HD, continuará com o desenvolvimento do ambiente computacional e a pesquisa e estudo de métodos que, com base no conjunto de dados fornecido, podem fornecer uma indicação aproximada do número de grupos em um agrupamento. Tal indicação será então usada nos experimentos com o HD, como critério de parada do algoritmo. Particularmente, dois métodos discutidos em [Theodoridis & Koutroumbas 1999] serão considerados. Também, uma das propostas de refinamento de algoritmos HD, evidenciada durante o levantamento bibliográfico será estudada (a descrita em [Guénoche *et al.* 1991]) com o objetivo de identificar sua real contribuição e, eventualmente, incorporá-la ao sistema computacional pretendido.

Referências

Bandyopadhyay, S.; Saha, S. (2013) "Unsupervised Classification", Heidelberg: Springer-Verlag.

Berthold, M. R.; Borgelt, C.; Höppner, F.; Klawonn, F. (2010) "Guide to Intelligent Data Analysis", Texts in Computer Science, D. Gries & F. B. Schneider (Eds.), Springer-Verlag.

Duda, R. O.; Hart, P. F.; Stork, D. G. (2001) "Pattern Classification", USA: John Wiley & Sons, Inc.

Guénoche, A.; Hansen, P.; Jaumard, B. (1991) "Efficient algorithms for divisive hierarchical clustering with the diameter criterion", Journal of Classification, v. 8, pp. 5-30.

Jain, A.K. (2010) Data clustering: 50 years beyond K-Means, Pattern Recognition Letters, v. 31, no. 8, pp. 651–666.

Mitchell, T. M (1997) *Machine Learning*, USA: McGraw-Hill.

Murtagh, F. & Contreras, P. (2011) "Methods of hierarchical clustering", arXiv:1105.0121 [cs.IR], Cornell University Library.

Theodoridis, S.; Koutroumbas, K. (1999) "Pattern Recognition", USA: Academic Press.

Witten, I. H., Frank E. Hall M. A. (2011) Data mining: practical machine learning tools and techniques, *Morgan Kaufmann*.

Xu, H.; Xu, D, Lin, E. (2007) "An applicable hierarchical clustering algorithm for content-based image retrieval", In: Proc. of The International Conference on Computer Vision/Computer Graphics Collaboration Techniques and Applications, MIRAGE'07, pp.82–92.