PROPOSTA DE UM MÉTODO BASEADO EM DENSIDADE E GRADE PARA O PROBLEMA DE AGRUPAMENTO AUTOMÁTICO

Gustavo Silva Semaan[1], Raphael Borges Vasconcelos[2], José André de Moura Brito[3], Luiz Satoru Ochi[4]

[1] Instituto do Noroeste Fluminense de Educação Superior - Universidade Federal Fluminense (INFES - UFF)
[2] Departamento de Ciência da Computação - Centro Universitário Plínio Leite (DCC - UNIPLI)
[3] Escola Nacional de Ciências Estatísticas - Instituto Brasileiro de Geografia e Estatística (ENCE - IBGE)
[4] Instituto de Computação - Universidade Federal Fluminense (IC - UFF)

{*gustavosemaan@id.uff.br, raphael.vasconcelos01@gmail.com, jose.m.brito@ibge.gov.br, satoru@ic.uff.br*}

Resumo

A área de "*Cluster Analysis*" agrega diversos métodos que têm como objetivo a identificação de grupos dentro de um conjunto de dados. O novo método proposto neste trabalho foi desenvolvido a partir do estudo de uma técnica baseada em grade e densidade. Ele tem como objetivo identificar o número de grupos em um problema de agrupamento automático com base na maximização do índice Silhueta. Os resultados computacionais apresentados neste estudo indicam que o método proposto é promissor no diz respeito à qualidade das soluções produzidas.

Palavras-chave: Agrupamento Automático; Densidade; Grade; Índice Silhueta.

Abstract

The cluster analysis has several methods that aim to identify groups within a dataset. This paper presents a new method for the automatic clustering problem based on both Density and Grid methodologies. The goal of the new method is identify the ideal number of clusters given a dataset by the maximization of Silhouette Index. According the computational experiments, the use of this method is a new promising way to solve the problem.

Keywords: *Automatic Clustering; Density; Grid; Silhouette Index.*

1. Introdução

A análise de agrupamentos é uma técnica de análise multivariada [Hair et al, 2009] que agrega um conjunto de métodos que são aplicados à determinação de grupos a partir de um conjunto de objetos definidos por certas características (atributos). Basicamente, ao aplicar-se uma análise de agrupamentos, o objetivo é obter grupos que apresentem objetos (características) semelhantes e que possam refletir a forma como os dados são estruturados. Para isso, deve-se maximizar a similaridade (homogeneidade) entre os objetos de um mesmo grupo e minimizar a similaridade entre objetos de grupos distintos [Han & Kamber, 2006] [Larose, 2005] [Goldschimidt & Passos, 2005].

Formalmente, o problema clássico de agrupamento pode ser definido da seguinte maneira: dado um conjunto formado por n objetos $X=\{x_1,x_2,...,x_i,...,x_n\}$, com cada objeto $x_i \in X$ possuindo f atributos (dimensões ou características), ou seja, $x_i=\{x_i^1, x_i^2,..., x_i^f\}$, deve-se construir k grupos C_j $(j=1,...,k)$ a partir de X, sendo tais grupos correspondentes a uma solução. Ao realizar a construção dos grupos, deve-se garantir que os objetos de cada grupo sejam homogêneos entre si, segundo alguma medida de similaridade. Uma solução (ou partição) pode ser representada como $\pi = \{ C_1, C_2,.., C_k \}$. Além disso, devem ser respeitadas as restrições concernentes a cada problema particular abordado [Han & Kamber, 2006] [Ester *et al.*, 1995] [Baum, 1986] [Hruschka & Ebecken, 2001] [Dias & Ochi, 2003].

As Equações 1, 2 e 3 representam o conjunto de restrições que definem o problema clássico de agrupamento, e determinam, respectivamente: que o conjunto *X* corresponde à união dos objetos dos grupos, cada objeto pertence a exatamente um grupo e todos os grupos possuem pelo menos um objeto. No problema de agrupamento automático, objeto de estudo das pesquisas relacionadas ao presente trabalho, a identificação da quantidade de grupos (*k*) faz parte do problema.

$$\bigcup_{j=1}^{k} C_j = X \quad (1)$$

$$C_i \bigcap C_l = \emptyset \text{ i,l=1, .., k e i} \neq \text{l} \quad (2)$$

$$C_j \neq \emptyset \text{ i=1, .., k} \quad (3)$$

Conforme [Kumar *et al.*, 2009], as últimas décadas, e em particular os últimos anos, têm sido marcados pelo desenvolvimento de diversos algoritmos de agrupamento. Por sua vez, esses algoritmos têm sido aplicados em diversos domínios, quais sejam: inteligência artificial, reconhecimento de padrões, marketing, economia, ecologia, estatística, pesquisas médicas, ciências políticas etc. Porém, nenhum desses algoritmos é apropriado para todos os tipos de dados, formatos de grupos e aplicações [Kumar *et al.*, 2009].

Após a apresentação das definições do Problema de Agrupamento e do Problema de Agrupamento Automático, devem ser apresentadas as especificidades existentes no método proposto. Esse trabalho está dividido em cinco seções, incluindo a introdução. A Seção 2 apresenta uma revisão da literatura com uma breve descrição dos algoritmos que tratam o problema de agrupamento automático. Ainda nessa seção é apresentado o índice Silhueta para avaliação e validação das soluções. Já a Seção 3 apresenta o método baseado em densidade e grade proposto nesse trabalho. A Seção 4 apresenta os resultados computacionais obtidos, enquanto a Seção 5 apresenta as conclusões do trabalho e sugere trabalhos futuros.

Os métodos baseados em densidade permitem a identificação de grupos de formatos arbitrários. Mais especificamente, esses métodos classificam como grupos as regiões onde há o maior número de elementos (objetos) no espaço de dados que são, naturalmente, separados pelas áreas de baixa densidade, conhecidas como ruídos [Han & Kamber, 2006]. Como exemplo de algoritmo baseado em densidade, pode-se citar o algoritmo DBSCAN (*Density-Based Spatial Clustering of Applications with Noise*) [Ester *et al.*, 1996]. A aplicação desse algoritmo permite a identificação grupos de objetos em regiões de alta densidade e permite, também, a descoberta grupos de formatos arbitrários em bases de dados espaciais e com ruídos.

Diferentemente dos métodos baseados em densidade, o método de grade cria uma grade no espaço de dados, e depois identifica o número de objetos dentro de cada célula da grade, formando assim um mapa multidimensional. Com base nas coordenadas dos objetos, e das células da grade, os grupos são identificados. A seção 3 apresenta uma ilustração da construção da grade e da formação de uma solução com base no método proposto.

Um exemplo de algoritmo baseado em grade (e em densidade) é o algoritmo CLIQUE [Rakesh et. al., 1999]. Esse algoritmo particiona o conjunto de dados em subespaços (grade de células) para encontrar agrupamentos suficientemente densos, ou seja, com grande concentração de objetos. Cada grade contém um conjunto de dados, separando os valores contínuos de cada atributo em um número de intervalos discretos. Por fim, cada objeto é atribuído a uma célula a qual seu intervalo contém o valor original do objeto. Os agrupamentos são formados a partir da junção de células densas adjacentes [Oliveira, 2007].

2. Revisão da Literatura

Conforme [Kumar *et al.*, 2009], talvez um dos problemas de seleção de parâmetros mais conhecido seja o de determinar o número ideal de grupos em um problema de

agrupamento. Neste sentido, diversas técnicas não supervisionadas de avaliação de soluções podem ser utilizadas.

Em [Naldi, 2011] são apresentadas duas abordagens sistemáticas que atuam na maximização do índice silhueta e que consistem em múltiplas execuções do algoritmo *k-means*. São elas: o *MRk-means* (do inglês *Multiple Runs of k-means*) e o *OMRk-means* (do inglês *Ordered Multiple Runs of k-means*). Já em [Semaan et. al., 2012] é proposto um método sistemático (MS) baseado em densidade MRDBSCAN (do inglês *Multiple Runs of DBSCAN)*, que utiliza um algoritmo baseado no DBSCAN (*Density Based Spatial Clustering of Applications with Noise*) [Ester et. al., 1996].

Ainda no contexto do problema de agrupamento automático, são vários os trabalhos na literatura que trazem a proposta de algoritmos baseados em meta-heurísticas. Estes algoritmos têm por objetivo encontrar um número ideal de grupos e a sua solução correspondente. [Soares and Ochi, 2004] [Cruz, 2010] [Hruschka and Ebecken, 2003] [Hruschka et. al., 2004a][Hruschka et. al., 2004b] [Hruschka *et al.*, 2006] [Ma *et al.*, 2006] [Alves *et al*. 2006] [Tseng & Yang, 2001] [Naldi & Carvalho, 2007] [Pan and Cheng, 2007].

Os índices relativos, como próprio nome sugere, têm como finalidade avaliar a qualidade relativa das soluções (partições) produzidas por diferentes métodos de agrupamento. Esses índices não têm a propriedade de monotonicidade, ou seja, não são afetados pelo aumento ou pela redução do número de grupos da solução. Dessa forma, podem ser utilizados na avaliação de diversas soluções, provenientes de diversos algoritmos [Naldi, 2011].

Em [Maulik & Bandyopadhyay, 2000], assim como em diversos trabalhos da literatura como [Cruz, 2010] e [Semaan, 2013], o algoritmo de clusterização *k-means* [MacQueen, 1967] foi utilizado como um procedimento em um Algoritmo Genético (AG) proposto. Na fase de avaliação dos cromossomos, o cálculo do *fitness* no AG corresponde ao cálculo das distâncias entre os centroides codificados no cromossomo, e cada ponto da base. Caso a distância entre um ponto e um centroide seja menor do que a distância do ponto para qualquer outro centro, então o ponto é adicionado a este grupo de menor distância, assim como no *k-means*.

Em [Garai & Chaudhuri, 2004] os dados são decompostos em alguns grupos fragmentados. O Algoritmo Genético é aplicado sobre esses fragmentos a fim de procurar a melhor formação dos grupos. Esses grupos encontrados serão submetidos a uma fase de união, considerando a aplicação de procedimentos que realizam uniões entre grupos adjacentes com base em uma análise de vizinhanças.

No trabalho de [Oliveira, 2007] é proposto um novo Algoritmo Evolutivo para a tarefa de análise de agrupamento chamado EDACluster, baseado no algoritmo EDA (Algoritmo de Estimativa de Distribuição - *Estimation of Distribution Algorithms*) [Larrañaga and Lozano, 2002]. Esse algoritmo não utiliza os operadores de cruzamento e mutação, mas sim uma amostragem da distribuição de probabilidade da população. Em seu trabalho, aplica uma metodologia híbrida para formação de grupos baseada nos métodos de densidade e grade.

2.1. Índice Silhueta

O Índice Silhueta foi proposto por [Rousseeuw, 1987] e é capaz de determinar a qualidade das soluções com base na proximidade entre os objetos de determinado grupo e na distância desses objetos ao seu grupo mais próximo. O índice silhueta é calculado para cada objeto, sendo possível identificar se o objeto está alocado ao grupo mais adequado. Esse índice combina as ideias de coesão e de separação. Os quatro passos a seguir explicam, sucintamente, como calculá-lo:

1. Nesse trabalho d_{ij} (Equação 5) corresponde à distância euclidiana entre os objetos x_i e x_j, e f é a quantidade de atributos dos objetos. Para cada objeto x_i calcula-se a sua distância média $a(x_i)$ (Equação 6) em relação aos demais objetos do mesmo grupo. Na

Equação 6, $|C_w|$ representa a quantidade de objetos do grupo C_w ao qual o objeto x_i pertence.

$$\max Silhueta(\Pi) = \frac{1}{n}\sum_{i=1}^{n} s(x_i) \tag{4}$$

$$d_{ij} = \sqrt{\sum_{q=1}^{f} (x_i^q - x_j^q)^2} \tag{5}$$

$$a(x_i) = \frac{1}{|C_w| - 1}\sum d_{ij} \quad \forall x_j \neq x_i, \quad x_j \in C_w \tag{6}$$

2. A Equação 7 apresenta a distância entre o objeto x_i e os objetos de outro grupo C_t, sendo $|C_t|$ correspondente à quantidade de objetos do grupo C_t. Para cada objeto x_i calcula-se a sua distância média em relação a todos os objetos dos demais grupos ($b(x_i)$) (Equação 8).

$$d(x_i, C_t) = \frac{1}{|C_t|}\sum d_{ij} \quad \forall x_j \in C_t \tag{7}$$

$$b(x_i) = \min d(x_i, C_t) \quad C_t \neq C_w \quad C_t \in C \tag{8}$$

3. O coeficiente silhueta de cada objeto x_i ($s(x_i)$) é obtido mediante a aplicação da Equação 9.

$$s(x_i) = \frac{b(x_i) - a(x_i)}{\max\{b(x_i), a(x_i)\}} \tag{9}$$

4. O cálculo da silhueta de uma solução $\exists\{C_1,...,C_k\}$ é a média das silhuetas dos objetos, conforme apresenta a Equação 4, em que *n* é a quantidade de objetos da solução. Essa função deve ser maximizada.

Os objetos que têm os valores da silhueta correspondentes a valores positivos e próximos de 1 estão bem localizados em seu grupo, enquanto valores negativos indicam que o objeto está mais próximo de outro(s) grupo(s). Ou seja, este índice permite avaliar a pertinência dos objetos em relação aos seus respectivos grupos.

3. Método Sistemático Baseado em Densidade e Grade

O algoritmo descrito na presente seção utiliza uma técnica baseada em densidade e grade para formação de soluções para o problema de agrupamento automático. As soluções obtidas são avaliadas pelo índice relativo de silhueta. A solução (partições) com o maior valor para esse índice é retornada pelo método e a quantidade de grupos dessa solução é considerada a ideal.

Em um primeiro passo, para a identificação do espaço em que os objetos estão distribuídos, o algoritmo busca pelo maior e menor valor para cada um dos atributos. A Figura 1 ilustra um exemplo de uma instância cujos objetos têm dois atributos (*f*=2). Neste exemplo é apresentado um objeto em que o atributo *y* possui como maior e menor valor,

respectivamente, 130 e 80. Já o atributo *x* possui como maior e menor valor, respectivamente, 230 e 120. O espaço bidimensional identificado, então, possui as dimensões 50 e 110.

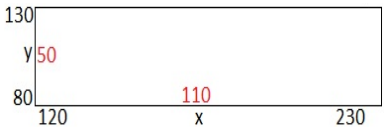

Figura 1. Espaço identificado em uma instância exemplo.

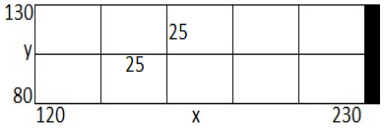

Figura 2. Formação das células da grade.

Uma vez identificado o espaço em que os objetos estão distribuídos, devem ser formadas as células da grade. Para isso, com base no parâmetro *fator* submetido ao método, definem-se os valores das dimensões de cada célula (e consequentemente as configurações da grade). As equações utilizadas para a formação da grade estão relatadas na seção 4. Destaca-se, que no presente trabalho, todas as instâncias consideradas à aplicação do método têm duas dimensões. A Figura 2 apresenta o espaço da Figura 1 após a divisão em células. A grade obtida possui duas linhas e cinco colunas. É possível verificar que as células da última coluna possuem um espaço extra, em que nenhum objeto da instância está inserido. Isso se deve ao fato de a divisão do valor do atributo *x* (110) pelo valor da dimensão da célula (25) não resultar em um valor inteiro (110/25 = 4,4), ou seja, 5 células (colunas).

O próximo passo do método consiste em identificar a qual célula cada objeto está alocado. Por exemplo, os objetos a=(168,103) e b=(200,110) estão localizados nas células [2, 1] e [4, 2], respectivamente, conforme apresenta a Figura 3. Na Figura 4 os valores exibidos correspondem aos quantitativos de objetos em cada célula. As células que possuem ao menos um objeto formam grupos iniciais que têm os objetos contidos em seu espaço.

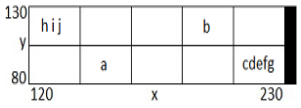

Figura 3. Células e objetos.

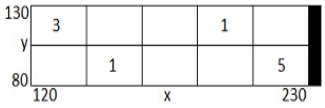

Figura 4. Quantidade de objetos por célula (Figura 3).

Após a formação dos grupos iniciais, deve ocorrer a formação da solução final. Uma vez que o método proposto utiliza conceitos de agrupamento baseado em densidade, grupos de formatos arbitrários devem ser obtidos. Nesse sentido, para a obtenção da solução deve ser realizada uma busca local em cada grupo, que deve ser unido ao(s) grupo(s) adjacente(s) existentes. A Figura 5 ilustra o alcance da busca local da célula *x*, em que as células em destaque (coloridas) são visitadas.

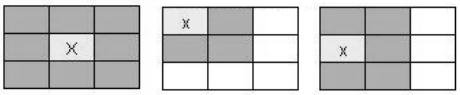

Figura 5. Alcance da busca local.

Com o objetivo de visualizar a formação da solução final, as Figuras 6, 7 e 8 apresentam um estudo de caso em que os valores indicados em cada célula representam a quantidade de objetos a ela alocados. A grade apresentada na Figura 6 possui 228 células (19 linhas e 12 colunas) e 33 grupos inicias, formados pelas células que possuem ao menos um objeto.

A célula em destaque na Figura 6 (célula [1, 1]) representa o *grupo 1* e é selecionada para a aplicação da busca local. Como foi apresentado anteriormente, a busca local consiste em realizar uniões entre os grupos iniciais adjacentes ao grupo selecionado. Com base na Figura 7, os grupos que serão considerados pela busca local são indicados pelas células [2,1](com um objeto) e [2,2] (com três objetos). As três células são unidas e formam o novo *grupo 1* (com sete objetos), conforme ilustra a Figura 8. No estudo de caso apresentado a solução formada possui seis grupos e deve ser avaliada por meio da aplicação do índice silhueta. É importante destacar que diferentes soluções podem ser obtidas conforme as dimensões das células da grade.

É importante destacar que a ordem em que as células fora selecionada para a aplicação da busca local não interfere no resultado obtido. Além disso, diferentes soluções podem ser obtidas conforme as dimensões das células da grade. Neste sentido, o método proposto considera diferentes grades, e a solução com maior valor para o índice silhueta é retornada, e indica a quantidade ideal de grupos. A seção 4 apresenta as fórmulas consideradas para a formação das grades consideradas nos experimentos computacionais realizados.

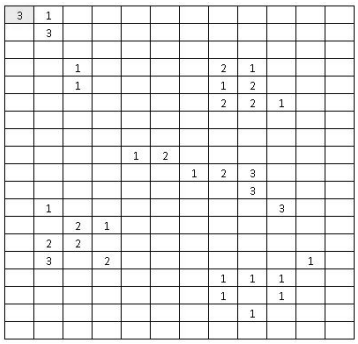

Figura 6. Instância exemplo.

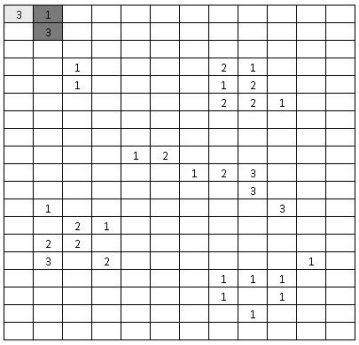

Figura 7. Busca local célula [1, 1].

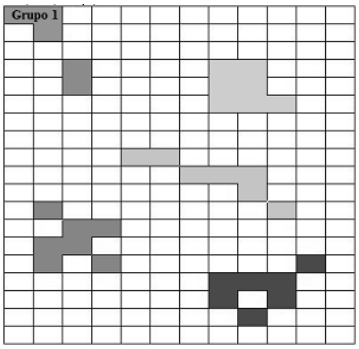

Figura 8. Solução obtida (k = 6).

4. Experimentos Computacionais

Para a realização dos experimentos computacionais, o método proposto foi implementado em Linguagem Python, utilizando o paradigma de orientação a objetos. Foram consideradas válidas apenas soluções com a quantidade de grupos no intervalo [2, $n^{1/2}$] em que *n* é a quantidade de objetos da instância, conforme [Naldi, 2011][Semaan et. al., 2012]. Todas as 14 instâncias utilizadas nos experimentos foram consideradas em outros trabalhos da literatura e possuem entre 100 e 1000 objetos com duas dimensões e quantidade de grupos entre 3 e 23.

Os experimentos computacionais foram realizados em um computador dotado de processador I7 de 1.9GHz, 4 GB de memória RAM e com sistema operacional Windows 8. Foram realizados comparativos com alguns algoritmos da literatura, sejam eles: com o algoritmo CLUES (*CLUstEring based on local Shirinking*) proposto por [Wang et. al., 2007]; Os melhores resultados dos algoritmos baseados nas metaheurísticas Algoritmos Genéticos, Busca Local Iterada (*Iterated Local Search*) e GRASP (*Greedy Randomized Adaptive Search Procedure*) propostos por [Cruz, 2010]; e com o método sistemático baseado em densidade MRDBSCAN (*Multiple Runs of DBSCAN*). relatado em [Semaan, et. al., 2012] e [Semaan, 2013]

O experimento computacional realizado para a obtenção dos resultados relatados na Tabela 1 consistiu em executar o método proposto com 10 iterações, em que cada iteração considerou um diferente valor para o parâmetro *fator*. Neste sentido, a Equação 10 é utilizada para calcular o valor da célula base para cada iteração, em que:

- $\min_{gap}$: menor gap entre os valores extremos de todos os atributos da instância.
- *n*: é a quantidade de objetos da instância.
- *fator*: em experimentos preliminares foi considerado *fator* = 1 para obter o valor de $m_{\text{cell.}}$ A variável *fator* é obtida de maneira aleatória e foi calibrada em experimentos preliminares para pertencer ao intervalo [0,5 , 2,0] conforme apresenta a Equação 11.

$$m_{cell} = \frac{\min_{gap}}{\sqrt{n}} * fator \tag{10}$$

$$fator = \frac{(500 + Rand(1501))}{1000} \tag{11}$$

A Tabela 1 apresenta os resultados obtidos com o método proposto em um subconjunto de instâncias classificadas como *comportadas* por [Cruz, 2010] e também utilizadas por [Semaan, G.S, 2013]. A utilização desta nomenclatura deve-se ao fato de os grupos estarem bem definidos, coesos em bem separados. Além disso, suas nomenclaturas estão no formato quantidade de pontos e quantidade de grupos. Por exemplo, a instância 100p3c é artificial e foi construída como uma instância comportada, com 100 objetos (‘*p’* de pontos) e 3 grupos (‘c’ de *cluster*).

Os dados apresentados na Tabela 1 representam a melhor solução (maior silhueta) produzida, considerando as dez iterações do algoritmo. Destaca-se que, a cada iteração, um valor diferente para a variável *fator* foi considerado. As colunas *Silhueta* e *Tempo* apresentam, respectivamente, o maior valor de Silhueta obtido e o tempo total de processamento (em segundos) considerando as 10 iterações do algoritmo. A coluna *Grupos* possui a quantidade de grupos identificada como ideal pelo método proposto. É possível observar que apenas para a instância 100p3c a diferença entre quantidade de grupos identificada como a ideal, e a relatada pela literatura foi superior a uma unidade.

Ainda com base na Tabela 1, os tempos de processamento relatados refletem as características do algoritmo utilizado no método proposto. Mais especificamente, além da quantidade de objetos, o valor da m_{cell} e a quantidade de grupos da instância têm impacto direto no tempo de processamento. Por exemplo, as instâncias 1000p6c e 1000p14c possuem a mesma quantidade de objetos (1000 unidades), mas as quantidades de grupos identificadas pelo método foram de 6 e 15 unidades, respectivamente. Neste caso, o tempo de processamento da instância 1000p6c foi de cerca de 80 segundos, enquanto a instância 1000p14c demandou mais de 160 segundos.

A Tabela 2 apresenta os comparativos com as quantidades de grupos obtidas a partir da aplicação de outros algoritmos da literatura. Entre as 14 instâncias utilizadas no trabalho, para 11 instâncias a quantidade ideal de grupos obtida corresponde ao valor relatado nos trabalhos de [Cruz, 2010] e [Wang, X. et. al., 2007]. Além disso, para duas instâncias a diferença foi de apenas uma unidade. Em relação aos resultados apresentados por [Semaan, et. al., 2012], para 11 instâncias a quantidade ideal de grupos obtida corresponde ao valor relatado, e para uma instância a diferença foi de apenas uma unidade.

Tabela 1. Resultados dos experimentos realizados.

Instâncias	Grupos (k)	Silhueta	Tempo (s)
100p3c	5	0,062	0,6
100p7c	7	0,833	0,8
100p10c	10	0,833	1,1
200p4c	4	0,02	2,3
300p3c	3	0,026	4,5
400p3c	3	0,301	8,2
500p3c	3	0,824	12,7
600p15c	15	0,781	57,0
700p4c	4	0,333	28,5
800p23c	23	0,787	151,3
900p5c	5	0,447	53,4
900p12c	13	0,651	112,0
1000p6c	6	0,399	76,8
1000p14c	15	0,605	160,9

Tabela 2. Comparativos com a quantidade de grupos de outros trabalhos da literatura:
k*: [Semaan, et. al., 2012]; **k**:** [Cruz, 2010] e [Wang, X. et al., 2007].

		Literatura	
Instâncias	**k**	**k***	**k****
100p3c	5	***3***	***3***
100p7c	***7***	***7***	***7***
100p10c	***10***	***10***	8
200p4c	***4***	***4***	***4***
300p3c	***3***	***3***	***3***
400p3c	***3***	***3***	***3***
500p3c	***3***	***3***	***3***
600p15c	***15***	***15***	***15***
700p4c	***4***	***4***	***4***
800p23c	***23***	***23***	***23***
900p5c	***5***	***5***	***5***
900p12c	13	***12***	***12***
1000p6c	***6***	***6***	***6***
1000p14c	15	***14***	15

5. Conclusões e Trabalhos Futuros

O presente trabalho propôs um método baseado em grade e densidade para a identificação do número ideal de grupos em problemas de agrupamento automático. De forma a avaliar a qualidade das soluções obtidas foi utilizado o índice silhueta, que combina características como coesão e separação. Os resultados apresentados neste estudo sugerem que o método proposto se constitui como uma boa alternativa ao problema de agrupamento automático. O novo método proposto foi capaz de identificar a quantidade ideal de grupos para 11 das 14 instâncias. Além disso, em 13 das 14 instâncias a diferença entre o *k* identificado e o *k* da literatura [Cruz, 2010] foi de apenas uma unidade e para a instância 100p3c a diferença foi de duas unidades.

Durante as pesquisas relacionadas ao Problema de Agrupamento Automático alguns caminhos apresentaram-se como promissores. Seguem algumas sugestões para trabalhos futuros:

- ☐ Realizar mais experimentos computacionais com novos conjuntos de instâncias com mais atributos e diferentes características.
- ☐ Investigar novas formas de obter a variável m_{cell} de maneira automática.
- ☐ É fato conhecido da literatura que o índice Silhueta não produz bons resultados para grupos com formatos arbitrários. Por exemplo, em grupos muito alongados ou grupos internos a outros grupos [Naldi, 2011]. O estudo de novas medidas relativas ou mesmo uma pesquisa para propor uma medida que seja mais adequada às soluções com características especificas de grupos baseados em densidade.

Referências

Alves, V., R. Campello, & E. Hruschka (2006). *Towards a fast evolutionary algorithm for clustering. In IEEE Congress on Evolutionary Computation*, 2006, Vancouver, Canada, pp. 1776–1783.

Baum, E.B. *Iterated descent: A better algorithm for local search in combinatorial optimization problems. Technical report Caltech*, Pasadena, CA. Manuscript, 1986.

Cruz, M. D. O Problema de Clusterização Automática. Tese de Doutorado, UFRJ, Rio de Janeiro, 2010.

Dias, C.R.; & Ochi, L.S.. *Efficient Evolutionary Algorithms for the Clustering Problems in Directed Graphs*. Proc. of the IEEE Congress on Evolutionary Computation (IEEE-CEC), 983-988. Canberra, Austrália, 2003.

Ester, M., Kriegel, H.-P., and Xu, X., *A Database Interface for Clustering in Large Spatial Databases*, In: Proceedings of the 1st International Conference on Knowledge Discovery in Databases and Data Mining (KDD-95), pp. 94- 99, Montreal, Canada, August, 1995.

Ester, M., H.-P. Kriegel, J. Sander, & X. Xu (1996). *A density-based algorithm for discovering clusters in large spatial databases with noise. In Proceedings of 2nd International Conference on Knowledge Discovery and Data Mining* (KDD'96), pp. 226–231.

Garai, G. & Chaudhuri, B. (2004), A novel genetic algorithm for automatic clustering, Pattern Recognition Letters, Ed. 25, pg. 173–187.

Goldschimidt R.; Passos, E. Data Mining: um guia prático. Editora Campus, Rio de Janeiro: Elsevier, 2005.

Hair, J.F, Black, W.C, Babin, B.J., Anderson, R.E. e Tatham, R.L. Análise Multivariada de Dados, Bookman, Sexta Edição, 2009.

Han, J., e Kamber, M., *Cluster Analysis*. In: Morgan Kaufmann. Publishers (eds.), *Data Mining: Concepts and Techniques*, 2 ed., chapter 8, New York, USA, Academic Press, 2006.

Hruschka, E. R., Ebecken, N. F. F. *A Genetic algorithm for cluster analysis. IEEE Transactions on Evolutionary Computation* , 2001.

Hruschka, E. R. & Ebecken, N. F. F. (2003). *A genetic algorithm for cluster analysis.* Intelligent Data Analysis 7 (1), 15–25.

Hruschka, E. R., R. J. G. B. Campello, & L. N. de Castro (2004a). *Evolutionary algorithms for clustering gene-expression data*. In Proc. IEEE Int. *Conf. on Data Mining*, Brighton/England, pp. 403–406.

Hruschka, E. R., R. J. G. B. Campello, & L. N. de Castro (2004b). *Improving the efficiency of a clustering genetic algorithm. In Advances in Artificial Intelligence* - IBERAMIA 2004: 9th Ibero-American Conference on AI, Puebla, Mexico, November 22-25. Proceedings, Volume 3315, pp. 861–870. Springer-Verlag GmbH, Lecture Notes in Computer Science.

Hruschka, E. R., R. J. G. B. Campello, & L. N. de Castro (2006). *Evolving clusters in gene-expression data. Information Sciences* 176 (13), 1898–1927.

Larose, D. T. *Discovering Knowledge in Data, An Introduction to Data Mining*. John Wiley & Sons, 2005.

Larrañaga, Pedro; & Lozano, Jose A. *Estimation of distribution algorithms: A new tool for evolutionary computation*. Kluwer Academic Publishers, Boston, 2002.

Maulik, U. & Bandyopadhyay, S. (2000), Genetic Algorithm-based Clustering Technique, Pattern Recognition p.33,1455-1465.

Macqueen, J. B. (1967). Some Methods for Classification and Analysis of MultiVariate Observations. Proc. of the fifth Berkeley Symposium on Mathematical Statistics and Probability. University of California Press. P. 281-297, V. 1.

Naldi, M. C. & A. C. P. L. F. Carvalho (2007). *Clustering using genetic algorithm combining validation criteria*. In Proceedings of the 15th European Symposium on Artificial Neural Networks, ESANN 2007, Volume 1. 2007.

Naldi, C. N. *Técnicas de Combinação para Agrupamento Centralizado e Distribuído de Dados*. Tese de Doutorado, USP - São Carlos, 2011.

Oliveira, C. EDACLUSTER: Um Algoritmo Evolucionário para Análise de Agrupamentos Baseados em Densidade e Grade, Dissertação (Mestrado em Engenharia Elétrica), Universidade Federal do Pará, 2007.

Pan, S. & K. Cheng (2007). *Evolution-based tabu search approach to automatic clustering. IEEE Transactions on Systems*, Man, and Cybernetics, Part C - Applications and Reviews 37 (5), 827–838.

Rakesh, A., Johanners, G., Dimitrios, G. & Prabhakar, R. (1999). Automatic subspace clustering of high dimensional data for data mining applications. In: Proc. of the ACM SIGMOD, p.94-105.

Rousseeuw, P. J. (1987). *Silhouettes: a graphical aid to the interpretation and validation of cluster analysis*. Journal of Computational and Applied Mathematics 20, 53–65.

Semaan, G. S., Cruz, M.D., Brito, J. A. M., and Ochi, L. S. "Proposta de um método de classificação baseado em densidade para a determinação do número ideal de grupos em problemas de clusterização", *Learning & Nonlinear Models* v.10 n4, 2012.

Semaan, G.S. Algoritmos para o Problema de Agrupamento Automático. Tese de Doutorado, Instituto de Computação, Universidade Federal Fluminense, 2013.

Soares, S. S. R. F., Ochi, L. S. *Um Algoritmo Evolutivo com Reconexão de Caminhos para o Problema de Clusterização Automática*. in *XII Latin Ibero American Congress on Operations Research*, Proc. of the XII CLAIO, 2004.

Tseng, L. & . Yang, S.B. *A genetic approach to the automatic clustering problem*. Pattern Recognition 34, 2001.

Wang et. al., Wang, X., Qiu, W., Zamar, R. H. (2007). CLUES: *A non-parametric clustering method based on local shrinking. Computational Statistics & Data Analysis* 52, 2007.