Learning and Nonlinear Models (L&NLM) – Journal of the Brazilian Computational Intelligence Society, Vol. 13, Iss. 1, pp. 52-66, 2015

MÉTODO BASEADO EM COMBINAÇÃO DE SOLUÇÕES COM PARTICIONAMENTO DE GRAFOS PARA O PROBLEMA DE AGRUPAMENTO AUTOMÁTICO

Gustavo Silva Semaan[1], Walace Rodrigues[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, walacerodrigues56@gmail.com, jose.m.brito@ibge.gov.br, satoru@ic.uff.br*}

Resumo. Os métodos de classificação podem ser aplicados com duas finalidades, quais sejam: identificar grupos dentro de um conjunto de dados, supondo fixado o número de grupos e uma função objetivo, ou identificar o número ideal de grupos mediante avaliação de algum índice de validação. Neste sentido, o presente trabalho traz a proposta de um método de combinação de agrupamentos baseada no particionamento de grafos para a obtenção de padrões em soluções para o problema de agrupamento automático. A qualidade das soluções obtidas com a utilização do método proposto é avaliada mediante a aplicação do índice silhueta, que combina coesão e separação. Foram realizados experimentos preliminares com o objetivo de selecionar instâncias que possuem tendência à formação de agrupamentos por meio da utilização da Estatística de Hopkins. Os resultados apresentados neste estudo indicam que o método proposto foi capaz de identificar padrões nas soluções do conjunto base, obtidas com a utilização do algoritmo da literatura baseado em densidade, DBSCAN. Além disso, as soluções obtidas com o método proposto foram equivalentes ou superiores às soluções do conjunto base.

Palavras Chave: Problema de Agrupamento Automático, Comitê de Agrupamentos, Estatística de Hopkins, Índice Silhueta.

1 - INTRODUÇÃO

A análise de agrupamentos agrega um conjunto de métodos que são aplicados à determinação de grupos a partir de um conjunto de objetos definidos por certas características (atributos). O objetivo é obter grupos que apresentem padrões (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 and Kamber, 2012] [Larose, 2005] [Goldschimidt and 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 *p* atributos (dimensões ou características), ou seja, $x_i=\{x_i^1,x_i^2,....,x_i^p\}$, deve-se construir *k* grupos C_j $(j=1,...,k)$ a partir de *X*, de forma a garantir que os objetos de cada grupo sejam homogêneos 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 and Kamber, 2012] [Ester et al., 1995] [Baum, 1986] [Hruschka and Ebecken, 2001] [Dias and Ochi, 2003]. Apresenta-se, a seguir, o conjunto de restrições que definem o problema clássico de agrupamento, que 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.

$$\bigcup_{j=1}^{k} C_j = X \tag{1}$$

$$C_i \bigcap C_l = \varnothing \text{ i,l=1, .., k e i} \neq \text{l} \tag{2}$$

$$C_j \neq \varnothing \text{ i=1, .., k} \tag{3}$$

Para este problema, o número de soluções possíveis, ou seja, o total de maneiras em que os *n* objetos podem ser agrupados, considerando um número fixo de *k* grupos, é dado pelo número de *Stirling* (*NS*) de segundo tipo [Johnson e Wichern, 2001], e podem ser obtidas pela Equação 4. Para problemas de agrupamento em que o valor de *k* é desconhecido

(agrupamento automático), o número de soluções possíveis aumenta ainda mais. E, neste caso, o número é dado pela Equação 5, que corresponde ao somatório da Equação 4 para o número de grupos variando no intervalo [1,k_{max}], sendo k_{max} o número máximo de grupos.

Para que se tenha uma ideia da ordem de grandeza deste número, no caso de n=10 objetos a serem alocados em k=3 grupos, o número de soluções a serem consideradas é de 9.330. Mas considerando apenas dobro de objetos, ou seja, n=20 e k=3, o número de soluções possíveis (Equação 4) sobe para 580.606.446. No problema de agrupamento automático estes valores crescem exponencialmente com o aumento da quantidade de objetos (n). Esta característica torna proibitiva a obtenção da solução ótima mediante a aplicação de um procedimento de enumeração exaustiva. Esta questão é comentada em vários trabalhos da literatura, como por exemplo, no trabalho de [Naldi, 2011].

$$NS(n,k) = \frac{1}{k!}\sum_{j=0}^{k}(-1)^{j-i}\binom{k}{j}j^n \tag{4}$$

$$NS(n) = \sum_{j=1}^{k_{\max}} NS(n, j) \tag{5}$$

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, estes algoritmos encontram aplicação em diversos domínios, quais sejam: inteligência artificial, reconhecimento de padrões, marketing, economia, ecologia, estatística, pesquisas médicas, ciências políticas, etc. Não obstante, nenhum desses algoritmos é apropriado para todos os tipos de dados, formatos de grupos e aplicações. Segundo [Naldi, 2011] estes algoritmos podem ser classificados em duas categorias principais, sejam elas particionais que, como sua nomenclatura sugere, geram partições de dados e os algoritmos hierárquicos, que também geram partições, mas em uma seqüência aninhada hierarquicamente.

Com base nas definições do problema clássico de agrupamento, define-se a combinação de agrupamentos (ou comitê de agrupamento, do inglês "*cluster ensemble*") como: Dado um conjunto de soluções do problema de agrupamento de tamanho q, $\prod = \{\pi_1, \pi_2, .., \pi_q\}$, deve-se encontrar uma única solução consenso. O conjunto $\prod$, denotado por *conjunto base*, é formado por soluções resultantes da aplicação de um algoritmo de agrupamento várias vezes (considerando variação de seus parâmetros) ou da aplicação de alguns algoritmos de agrupamento em determinado conjunto de dados X [Naldi 2011] [Naldi 2007]. A combinação de agrupamentos pode ter diferentes objetivos, como:

- **Robustez:** obter uma solução consenso de melhor qualidade que a maioria das soluções do conjunto base ou mesmo uma solução com menor sensibilidade a ruídos e *outliers*.
- **Novidade:** obter uma solução consenso inédita, que não poderia ser formada com a utilização dos algoritmos de agrupamentos utilizados no processo individualmente. Destaca-se que tais algoritmos foram responsáveis pela formação do conjunto base.
- **Reaproveitamento de conhecimento:** utilizar o conhecimento obtido para a formação das soluções base para construir a solução consenso.
- **Consistência:** obter uma solução consenso tal que, de alguma forma, esteja em concordância com as partições base.
- **Desempenho e custo computacional:** para reduzir a complexidade (custo) computacional podem ser utilizados algoritmos que utilizem diferentes técnicas e diferentes objetivos para que seus resultados sejam combinados, de forma a produzir uma solução consenso mais robusta que as soluções do conjunto base.

O objetivo relacionado ao Desempenho e Custo Computacional sugere também que há espaço para o estudo e o desenvolvimento de novos algoritmos de agrupamento que sejam mais eficientes ou mais apropriados, levando em conta as características específicas de conjuntos de dados. Em muitos casos, inclusive, a análise de "*o que é uma boa solução*" é subjetiva, tendo em vista as especificidades do problema estudado.

Não obstante, para aumentar a chance de obter sucesso com a utilização das técnicas de comitês de agrupamento, é necessário considerar dois aspectos importantes, sejam eles: a diversidade, relacionada às soluções que compõem o conjunto base e a função consenso, que realiza efetivamente a combinação das soluções. Em relação ao primeiro aspecto apresentado, é necessário que as soluções do conjunto base possuam um grau de diversidade mínimo, de forma a justificar tanto o custo computacional do algoritmo de combinação, quanto formar soluções que atendam a um dos objetivos almejados na utilização da combinação de agrupamentos. Já sobre a função consenso, destacam-se as técnicas: baseadas em Coassociação, em votação e em particionamento de grafos [Naldi et. al., 2009].

Após a apresentação das definições do problema de agrupamento, de agrupamento automático e combinação de agrupamentos, 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 concernente aos algoritmos que tratam o problema de agrupamento automático e aos comitês de agrupamentos. Ainda nessa seção é apresentado o índice Silhueta para avaliação das soluções. Já a seção 3 apresenta o algoritmo DBSCAN (*Density-Based Spatial Clustering of Application with Noise*), utilizado para a obtenção das soluções que compõem o conjunto base, que será submetido ao algoritmo de combinação

de agrupamentos. Ainda na seção 3 é apresentado o método proposto nesse trabalho. A seção 4 apresenta a *Estatística de Hopkins* (EH), utilizada para identificar se existe tendência à formação de agrupamentos. Nessa mesma seção são apresentados experimentos relacionados à EH. A seção 5 traz os resultados computacionais obtidos considerando, inclusive, os comparativos realizados com algoritmos mais sofisticados da literatura, propostos por [Cruz, 2010]. Por fim, a seção 6 apresenta as conclusões do trabalho e sugere trabalhos futuros.

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.

Com base no algoritmo *k-Means*, que utiliza a ideia de protótipos, [Pelleg and Moore, 2000] propuseram o algoritmo *X-Means* para a resolução do problema de agrupamento automático. Este algoritmo recebe como parâmetros a instância a ser processada e um intervalo com a quantidade de grupos [k_{min}, k_{max}]. A partir destes dados, o algoritmo utiliza o índice BIC (*Bayesian Information Criterion*) para identificar e retornar qual o melhor número de grupos. Em [Zalik, 2008] é apresentado um algoritmo que também adapta o *k-Means* para resolver um problema de agrupamento automático.

Ainda no contexto do problema agrupamento automático, vários trabalhos na literatura propõem algoritmos baseados em metaheurísticas que têm por objetivo encontrar um número ideal de grupos e a sua solução correspondente. Dentre estes, destacam-se os seguintes trabalhos: [Soares, 2004] [Cruz, 2010] [Cole, 1998] [Cowgill, 1999] [Bandyopadhyay and Maulik, 2001] [Bandyopadhyay and Maulik, 2002b] [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 and Yang, 2001] [Naldi and Carvalho, 2007] [Pan and Cheng, 2007]

Existem, também, as heurísticas que utilizam alguns procedimentos de busca local baseados no algoritmo *k-Means*. Em um primeiro momento, essas heurísticas utilizam algoritmos para construção de grupos, denominados grupos parciais (temporários, componentes conexos) com o objetivo de unir os objetos mais homogêneos. Em seguida, são aplicados algoritmos de busca local e de perturbação sobre esses grupos produzindo soluções de boa qualidade, ou seja, os grupos parciais são unidos e formam grupos finais [Cruz, 2010] [Tseng and Yang, 2001] [Hruschka et. al., 2004b] [Alves et. al. 2006] [Hruschka et. al., 2006] [Naldi and Carvalho, 2007].

Em [Semaan, et. al., 2012] foi proposto um *Método de Classificação Baseado em Densidade*, que utiliza o conhecido algoritmo baseado em densidade DBSCAN para obter soluções para o problema de agrupamento automático. Esse algoritmo foi adaptado para que os objetos identificados como *outliers* não fossem ignorados, ou seja, eles devem ser considerados nas soluções, não violando assim a restrição apresentada na Equação 1 (todos os objetos precisam estar associados a algum grupo).

Os índices relativos, como próprio nome sugere, têm como finalidade avaliar a qualidade relativa das soluções produzidas por diferentes métodos de agrupamento. Estes 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. Conforme [Naldi, 2011], os índices de validação relativos têm sido utilizados e investigados extensivamente, tendo apresentado resultados satisfatórios para diversas aplicações.

Segundo [Naldi, 2011], de uma forma prática, pode-se definir o procedimento de determinar o número ideal *k* de grupos em um problema de agrupamento automático em dois passos. O primeiro passo consiste em executar diversas vezes algoritmos de agrupamento, considerando que o número *k* de grupos (parâmetro de entrada) irá variar em um intervalo pré-determinado. Já o segundo passo consiste na utilização de índices de validação para verificar a qualidade das soluções obtidas.

Os algoritmos de agrupamento baseados em densidade têm como objetivo a determinação de grupos (regiões) de alta densidade de objetos separados por regiões de baixa densidade. Nesse contexto, as soluções do conjunto base, que serão submetidas ao método proposto no presente trabalho, foram obtidas com a aplicação do algoritmo baseado em densidade apresentado DBSCAN [Ester et. al., 1996].

2.1 - A Silhueta

O Índice Silhueta foi proposto por [Rousseeuw, 1987]. Esta medida determina 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, brevemente, como calculá-lo:

1. Neste trabalho d_{ij} (Equação 7) corresponde à distância euclidiana entre os objetos *i* e *j*, e *p* é a quantidade de atributos dos objetos. Para cada objeto x_i calcula-se a sua distância média $a(x_i)$ (equação 8) em relação a todos os demais objetos do mesmo grupo. Na Equação 8, $|C_w|$ representa a quantidade de objetos do grupo C_w , ao qual o objeto x_i pertence.

$$\max Silhueta(S) = \frac{1}{n}\sum_{i=1}^{n} s(x_i) \tag{6}$$

$$dist_{i,j} = \sqrt{\sum_{q=1}^{p}(x_i^q - x_j^q)^2} \tag{7}$$

$$a(x_i) = \frac{1}{|C_w|-1}\sum d_{ij} \qquad \forall x_j \neq x_i, \quad x_j \in C_w \tag{8}$$

2. A Equação 9 apresenta a distância entre o objeto x_i e os objetos do grupo C_t, em que $|C_t|$ é a 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 10).

$$d(x_i, C_t) = \frac{1}{|C_t|}\sum d_{ij} \qquad \forall x_j \in C_t \tag{9}$$

$$b(x_i) = \min d(x_i, C_t) \qquad C_t \neq C_w \qquad C_t \in C \tag{10}$$

3. O índice silhueta do objeto x_i ($s(x_i)$) pode ser obtido pela equação 11.

$$s(x_i) = \frac{b(x_i) - a(x_i)}{\max\{b(x_i), a(x_i)\}} \tag{11}$$

4. O cálculo da silhueta de uma solução S é a média das silhuetas de cada objeto, conforme apresenta a equação 6, em que *n* é a quantidade de objetos da solução. Essa função deve ser maximizada. Valores positivos de silhueta indicam que o objeto está bem localizado em seu grupo, enquanto valores negativos indicam que o objeto deveria ser alocado a outro grupo.

3 – MÉTODO PARA COMBINAÇÃO DE SOLUÇÕES BASEADO PARTICIONAMENTO DE GRAFOS

Na introdução deste trabalho foram apresentadas as definições dos problemas de agrupamento, agrupamento automático, da combinação de agrupamentos, bem como alguns dos objetivos almejados em sua utilização. Foram apresentados, também, os aspectos necessários para o sucesso da utilização da combinação, que são: função consenso e diversidade.

O método proposto neste trabalho consiste em utilizar a função consenso baseado em particionamento de grafos para obtenção de soluções para o problema de agrupamento automático. A presente seção apresenta o método proposto bem como o algoritmo DBSCAN, utilizado na formação das soluções do conjunto base.

3.1 – DBSCAN

Os algoritmos de agrupamento baseados em densidade têm como objetivo a determinação de grupos (regiões) de alta densidade de objetos separados por regiões de baixa densidade. Nesse contexto, o algoritmo DBSCAN [Ester et. al., 1996] é

um dos mais conhecidos da literatura e possui uma complexidade computacional $O(n^2)$. Trata-se de um algoritmo simples, eficiente, e que contempla conceitos importantes, que servem de base para qualquer abordagem baseada em densidade.

O DBSCAN utiliza-se de um conceito de densidade tradicional baseada em centro. Ou seja, a densidade de um objeto x_i é a quantidade de objetos em um determinado raio de alcance de x_i, incluindo o próprio objeto. Este algoritmo possui como parâmetros de entrada o raio (*raioDBSCAN*) e a quantidade mínima de objetos em um determinado raio (*qtdeObjetos*). Assim, a densidade de um objeto depende do raio especificado.

Deve-se, então, calibrar o parâmetro *raioDBSCAN* para que o seu valor não seja tão alto de forma que todos os objetos tenham densidade *n* (solução com apenas um grupo), e nem tão baixo em que todos os objetos terão densidade 1 (solução com *n* grupos denominados *singletons*). A abordagem da densidade baseada em centro realiza a classificação dos objetos em:

- **Interiores ou Centrais**: objetos que pertencem ao interior de um grupo baseado em densidade. Deve possuir uma quantidade de objetos em seu raio *raioDBSCAN* igual ou superior ao parâmetro *qtdeObjetos - 1*.
- **Limítrofes**: não é um objeto central, mas é alcançável por ao menos um objeto central, ou seja, está dentro do raio de vizinhança de algum objeto central.
- **Ruídos**: demais objetos que não são Centrais e nem estão na vizinhança de um objeto central.

Para a aplicação do algoritmo DBSCAN são considerados os seguintes passos:

1. Classificar os objetos como Objetos *Centrais*, *Limítrofes* ou *Ruídos*.
2. Eliminar os objetos que sejam classificados como *Ruídos*.
3. Adicionar arestas entre todos os Objetos Centrais que estejam dentro do *raioDBSCAN*.
4. Tornar cada grupo de Objetos de centro um grupo separado.
5. Atribuir cada Objeto limítrofe a um dos grupos dos seus objetos centrais associados.

Como base nestas informações, a Figura 1 ilustra a classificação de cada um dos objetos em *Ruído*, *Limítrofe* ou *Interior*. Essa mesma figura apresenta também uma solução obtida com a execução do DBSCAN, em que é possível observar que objetos identificados como dos tipos *Interior* ou *Borda* formam grupos, enquanto objetos do tipo *Ruído* permanecem isolados e não fazem parte de nenhum grupo.

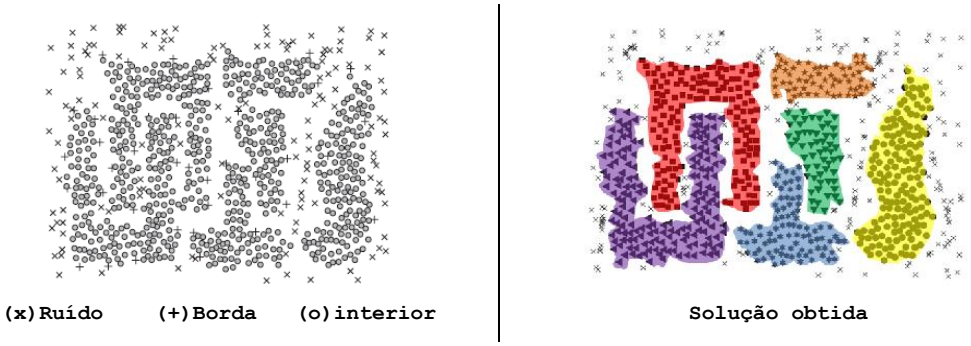
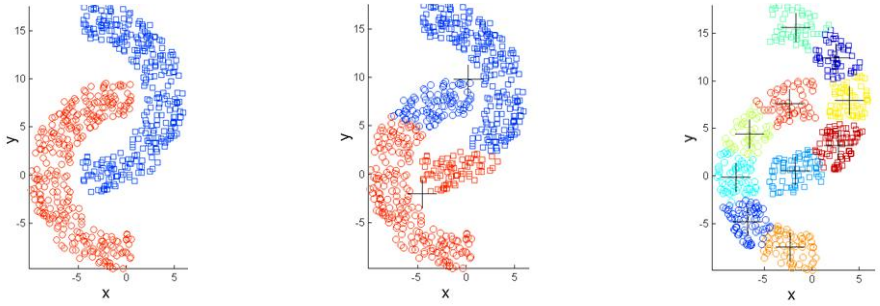


Figura 1: classificação de 3000 objetos de duas dimensões pelo DBSCAN [Kumar et. al., 2009]

Tendo em vista que o DBSCAN é um algoritmo baseado em densidade, o mesmo é imune a ruídos, uma vez que esses objetos são identificados e ignorados (não pertencem a grupos). Além disso, o algoritmo pode trabalhar com grupos de tamanhos (número de objetos) e formas arbitrárias. Dessa forma, ele é capaz de identificar grupos que não poderiam ser encontrados mediante a aplicação de outros algoritmos, como por exemplo, o *k*-means. A Figura 2 (a) apresenta os objetos iniciais de uma instância em que é possível observar dois grupos, que poderiam ser obtidos com o algoritmo DBSCAN. Já as Figuras 2 (b) e (c) apresentam resultados obtidos com a utilização do *k*-means. Esse algoritmo tem como característica a obtenção de grupos esféricos e de tamanhos semelhantes em relação aos raios dos centróides. É possível observar que mesmo para k=2 os grupos da Figura 2 (a) não foram obtidos.



(a) Objetos Originais **(b)** Solução do *k-Means* para k=2 **(c)** Solução do *k-Means* para k=10

Figura 2: (a) Objetos Originais; (b) e (c) soluções do *k*-Means [Kumar et. al., 2009].

3.2 – Método Proposto

O método proposto consiste em utilizar uma modelagem baseada em grafos para a identificação de padrões em soluções do Problema de Agrupamento Automático. Os padrões identificados são considerados na formação de novas soluções, obtidas por meio da aplicação de um algoritmo de particionamento de grafos, e as soluções consenso obtidas são avaliadas com a utilização do *Índice Silhueta*.

Com o objetivo de ilustrar a modelagem considerada, a Figura 3 apresenta um Conjunto Base exemplo utilizado também em [Naldi, 2011] e [Strehl and Ghosh, 2002]. Esse conjunto possui quatro soluções, em que as soluções π_1, π_2 e π_3 possuem três grupos e a solução π_4 possui apenas dois grupos. Nesses trabalhos da literatura, os objetos dos grupos *singletons* foram declarados como objetos não agrupados. Porém, com base nas definições do problema abordado, todos os objetos devem pertencer a um e somente um grupo. Dessa forma, a solução π_4 possui cinco grupos, em que três são *singletons.*

Partição	*Clusters*
π_1	$C_1^1 = \{\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3\}, C_2^1 = \{\mathbf{x}_4, \mathbf{x}_5\}, C_3^1 = \{\mathbf{x}_6, \mathbf{x}_7\}$
π_2	$C_1^2 = \{\mathbf{x}_6, \mathbf{x}_7\}, C_2^2 = \{\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3\}, C_3^2 = \{\mathbf{x}_4, \mathbf{x}_5\}$
π_3	$C_1^3 = \{\mathbf{x}_1, \mathbf{x}_2\}, C_2^3 = \{\mathbf{x}_3, \mathbf{x}_4\}, C_3^3 = \{\mathbf{x}_5, \mathbf{x}_6, \mathbf{x}_7\}$
π_4	$C_1^4 = \{\mathbf{x}_1, \mathbf{x}_4\}, C_2^4 = \{\mathbf{x}_2, \mathbf{x}_5\}$, objetos $\mathbf{x}_3$, $\mathbf{x}_6$ e $\mathbf{x}_7$ não agrupados

Figura 3: conjunto base para ilustração conforme [Naldi, 2011] [Strehl and Ghosh, 2002].
Grupos *singletons* foram desconsiderados.

Com base nas soluções apresentadas na Figura 3, na Figura 4 (a) cada coluna π_i representa uma solução, em que o índice referencia o objeto e o valor no vetor indica o grupo em que esse objeto está alocado. Por exemplo, na solução π_2 os objetos os objetos x_6 e x_7 estão no grupo 1, x_1, x_2 e x_3 estão no grupo 2 e os objetos x_4 e x_5 estão no grupo 3.

O próximo passo da modelagem é construir uma matriz de adjacência em que deve-se associar cada solução π_i a uma partição H^i. Além disso, cada grupo não *singleton* de cada solução deve ser associado a uma *hiperaresta* h_j. A utilização de hiperarestas deve-se ao fato de arestas em grafo simples conectarem exatamente dois vértices, enquanto hiperarestas podem ser utilizadas para conectar uma quantidade ilimitada de vértices.

Na Figura 4 (b) a solução π_2, por exemplo, está associada a partição H^2. Com o objetivo de facilitar a visualização, com base na solução π_2 são formadas três hiperarestas, uma para cada grupo. A hiperarestas h_4, h_5 e h_6 estão associadas aos grupos 1, 2 e 3, respectivamente. Esta figura também apresenta a estrutura utilizada para indicar quais objetos estão associados a quais hiperarestas. Na hiperaresta h_6, por exemplo, estão associados os objetos x_4 e x_5 (grupo 3 da solução π_2).

Ainda em relação à modelagem, cujo as estruturas de dados são apresentadas pelas Figuras 4 (a) e (b), destaca-se que os grupos *singletons* são desconsiderados. Para a solução π_4, por exemplo, apenas os grupos 1 e 2 (que possuem mais de um objeto) estão relacionados a hiperarestas. Outra observação importante é que as hiperarestas h_1, h_5 e h_9 possuem mais de dois objetos.

	π_1	π_2	π_3	π_4
x_1	1	2	1	1
x_2	1	2	1	2
x_3	1	2	2	3
x_4	2	3	2	1
x_5	2	3	3	2
x_6	3	1	3	4
x_7	3	1	3	5

(a)

	H^1			H^2			H^3			H^4	
	h_1	h_2	h_3	h_4	h_5	h_6	h_7	h_8	h_9	h_{10}	h_{11}
x_1	1	0	0	0	1	0	1	0	0	1	0
x_2	1	0	0	0	1	0	1	0	0	0	1
x_3	1	0	0	0	1	0	0	1	0	0	0
x_4	0	1	0	0	0	1	0	1	0	1	0
x_5	0	1	0	0	0	1	0	0	1	0	1
x_6	0	0	1	1	0	0	0	0	1	0	0
x_7	0	0	1	1	0	0	0	0	1	0	0

(b)

Figura 4: Conjunto Base para ilustração. (a) vetores com quatro soluções.
(b) as hiperarestas formadas [Naldi, 2011] [Strehl and Ghosh, 2002].

Uma vez que a função consenso considerada no presente trabalho atua no particionamento de grafos, é necessário definir os vértices e as arestas do mesmo. Para simplificar o desenvolvimento e o entendimento, cada hiperaresta h_j será manipulada como um vértice h_i em um grafo *G*. Em relação às arestas de G, seus pesos $w(h_i,h_j)$ devem ser calculados.

Segundo [Naldi, 2009], seja C_j^i o *j*-ésimo grupo da *i*-ésima solução, o peso da aresta entre os vértices correspondentes aos grupos C_j^i e C_t^s é dado pela medida *Jaccard Estendida*, também conhecida como *Coeficiente de Tanimoto* [Tanimoto, 1958]. Quando utilizada em vetores binários, como ocorre na modelagem do presente trabalho, essa medida é equivalente a razão entre a cardinalidade da intersecção e a cardinalidade da união dos objetos pertencentes aos dois vértices (h_i e h_j), conforme apresenta a Equação 12. A seguir são apresentados exemplos dos cálculos realizados: $w(h_1,h_4)$=0/5=0 (não existe aresta); $w(h_1,h_8)$=1/4=0.25; $w(h_7,h_{10})$=1/3=0.33; $w(h_1,h_7)$=2/3=0.66 e $w(h_1,h_5)$=3/3=1. A Figura 5 apresenta o grafo G obtido após o cálculo das arestas.

$$w(C_j^i, C_t^s) = \frac{|C_j^i \cap C_t^s|}{|C_j^i \cup C_t^s|} \tag{12}$$

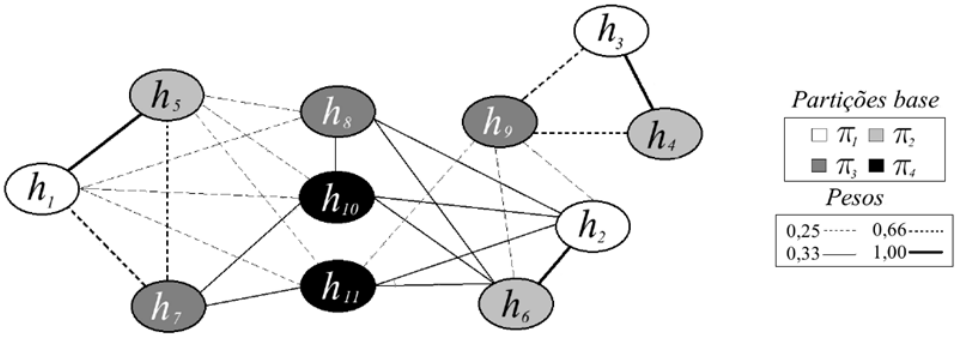


Figura 5: grafo construído com base nas soluções do conjunto base [Naldi, 2011].

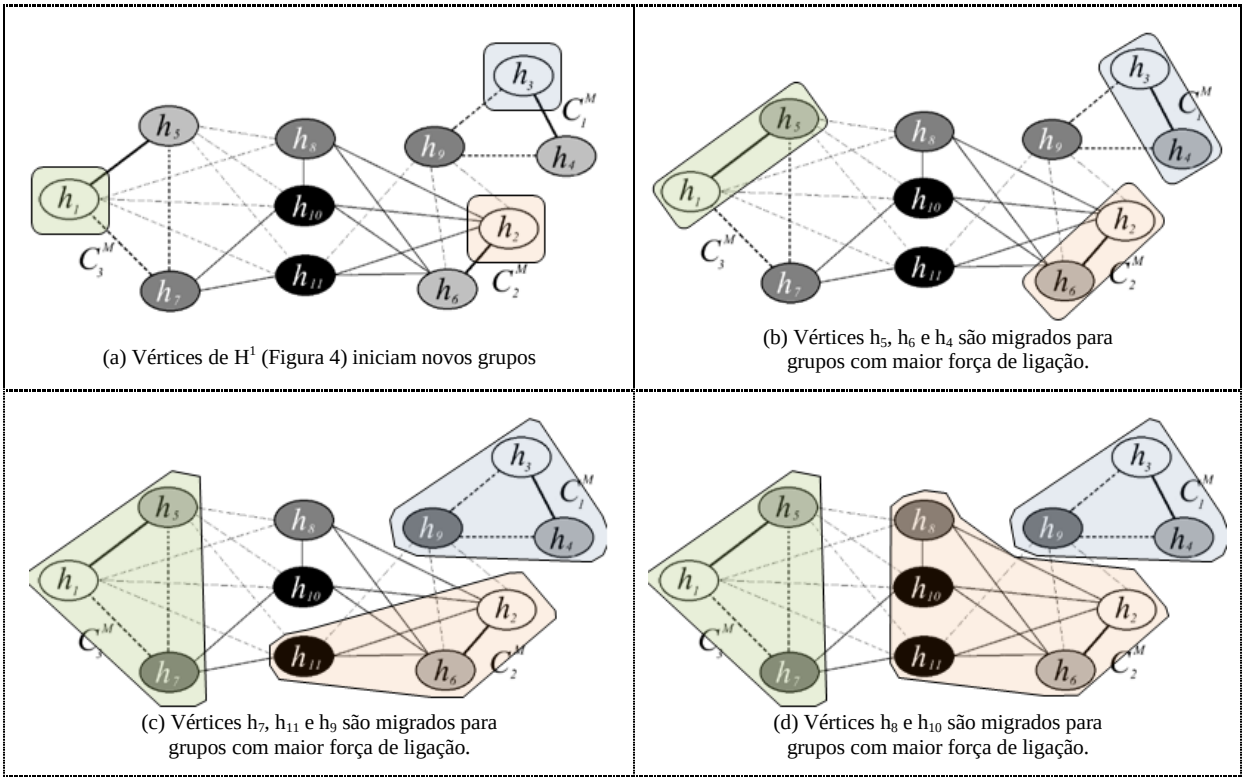
Em [Strehl and Ghosh, 2002] são citados três algoritmos para a etapa de particionamento de grafos no contexto de comitê de agrupamento: CSPA (*Cluster based Similarity Partitioning Algorithm*), HGPA (*HiperGraph-Partitioning Algorithm*) e MCLA (*MetaCLustering Algorithm*). Neste trabalho é apresentado um algoritmo simples, que utiliza também conceitos de particionamento de grafos, e que consiste nos seguintes passos:

- Para cada partição H^i, cada um de seus vértices (hiperarestas) h_j devem iniciar um novo grupo C_l^M. Dessa forma a solução consenso possuirá $|H^i|$ grupos.
    - Na Figura 6 (a) a partição H^1 foi selecionada, e os vértices h_1, h_2 e h_3 formam grupos *singletons*.
- Em seguida, de maneira aleatória, cada um dos vértices restantes h_l deve ser migrado para o grupo em que possui maior força de ligação (peso da aresta). Em caso de empate uma das opções é selecionada de maneira aleatória.
    - Na Figura 6 (b) vértices h_4, h_6 e h_3 foram inseridos nos grupos C_1^M, C_2^M e C_3^M, respectivamente.
    - Em seguida, na Figura 6 (c) os vértices h_9, h_{11} e h_7 foram inseridos nos grupos C_1^M, C_2^M e C_3^M, respectivamente. Com base na Figura 6 (b) é possível Observar que os vértices h_{11} e h_7 só possuíam vértices com os grupos C_2^M e C_3^M, respectivamente. Dessa forma, não foi necessário verificar o peso das arestas, uma vez que só existia uma opção de destino. Já o vértice h_9 possui arestas com os grupos C_1^M (

$w(h_9,h_3)$= 0,66 e $w(h_9,h_4)$=0,66) e C_2^M ($w(h_9,h_6)$=0,25). Este vértice deve ser migrado para o grupo C_1^M (aresta de maior peso).

- Por fim, na Figura 6 (d), os vértices h_8 e h_{10} inseridos no grupo C_2^M.

- Os vértices que eventualmente não possuem arestas (grau *0*) formam grupos *singletons*.



(a) Vértices de H^1 (Figura 4) iniciam novos grupos

(b) Vértices h_5, h_6 e h_4 são migrados para grupos com maior força de ligação.

(c) Vértices h_7, h_{11} e h_9 são migrados para grupos com maior força de ligação.

(d) Vértices h_8 e h_{10} são migrados para grupos com maior força de ligação.

Figura 6: ilustração, passo a passo, do algoritmo para o particionamento do grafo.

Para facilitar a visualização a formação de uma solução consenso, a Figura 7 apresenta o grafo da Figura 5 após a realização de um particionamento, em que C_1^M ={h_3,h_4,h_9}, C_2^M ={h_2,h_6,h_8,h_{10}} e C_3^M ={h_1,h_5,h_7,h_{11}}. Com base nesse particionamento deve-se formar uma solução consenso. Para isso, a tabela apresentada na Figura 8 apresenta a estrutura de dados utilizada para a formação da solução consenso. Nessa tabela, as linhas indicam os objetos e cada grupo (C_j^M) possui uma coluna. A coluna C_1^M possui duas colunas, h_i^M e $\alpha(h_j^M)$. A coluna h_1^M , por exemplo, indica que os objetos x_5, x_6 e x_7 são utilizados em C_1^M .

Ainda com base na Figura 8, a coluna $\alpha(h_1^M)$, por exemplo, apresenta o vetor de associação dos objetos nos vértices de C_1^M (h_3, h_4 e h_9). Nesse sentido, com base na Figura 4 (b), o objeto x_6 é utilizado apenas em um dos três vértices de C_1^M. Dessa forma, para esse objeto, $\alpha(h_1^M)$=1/3=0.33. Já os objetos x_6 e x_7 são utilizados em todos os vértices ($\alpha(h_1^M)$=3/3=1 para ambos). Esses cálculos devem ser realizados para todos os objetos para todas as colunas.

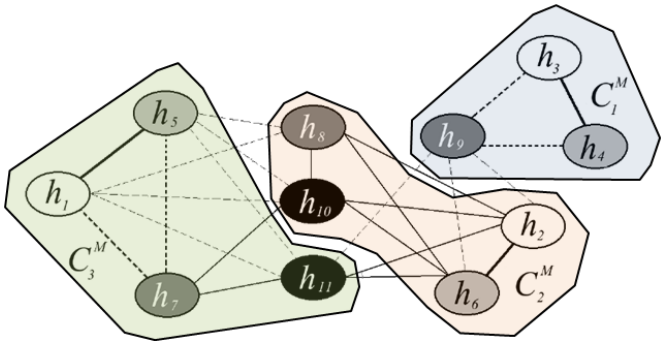


Figura 7: exemplo de metagrafo particionado conforme [Naldi, 2011].

	C_1^M		C_2^M		C_3^M	
	$\{h_3, h_4, h_9\}$		$\{h_2, h_6, h_8, h_{10}\}$		$\{h_1, h_5, h_7, h_{11}\}$	
Objetos	h_1^M	$a(h_1^M)$	h_2^M	$a(h_2^M)$	h_3^M	$a(h_3^M)$
x_1	0	0	1	0,25	1	**0,75**
x_2	0	0	0	0	1	**1**
x_3	0	0	1	0,25	1	**0,5**
x_4	0	0	1	**1**	0	0
x_5	1	0,33	1	**0,5**	1	0,25
x_6	1	**1**	0	0	0	0
x_7	1	**1**	0	0	0	0

Figura 8: tabela com vetores de associação para a identificação de padrões e formação da solução consenso [Naldi, 2011].

A tabela apresentada pela Figura 8 possui o cálculo de $\alpha(h_i^M)$ para todos os objetos. O próximo passo consiste em verificar em qual grupo cada objeto deve pertencer. Para isso, deve-se verificar o grupo C_j^M em que cada objeto possui o maior valor de associação. Por exemplo, o objeto x_5 possui valor de associação 0.33 com o grupo C_1^M, 0.5 com o grupo C_2^M e 0.25 com o grupo C_3^M. Dessa forma esse objeto irá pertencer ao grupo final de C_2^M. Na tabela, as células com os maiores valores para cada objeto estão em destaque, em cor cinza. Ainda com base na tabela, a solução consenso obtida possui três grupos, sejam eles: $C_1 = \{x_6, x_7\}$, $C_2 = \{x_4, x_5\}$, $C_3 = \{x_1, x_2, x_3\}$.

O próximo e último passo é calcular o índice silhueta para a solução consenso. Uma vez que foram apresentadas somente as soluções de entrada (Figura 3), para o cálculo da silhueta é necessário detalhar os valores para cada atributo de cada objeto. A Figura 9 possui os valores dos atributos para os objetos da instância exemplo. Além disso, são relatados também os índices silhuetas de cada objeto e a silhueta média da solução.

	Objetos						
	x_1	**x_2**	**x_3**	**x_4**	**x_5**	**x_6**	**x_7**
Solução	3	3	3	2	2	1	1
Atributo 1	-1,08	-0,66	-1,08	0,18	0,18	1,45	1,02
Atributo 2	-1,31	-0,76	-0,76	-0,20	0,91	0,91	1,19
Silhuetas	0,70	0,61	0,72	0,18	-0,03	0,66	0,60
Silhueta da solução				**0,49**			

Figura 9: instância exemplo, silhuetas dos objetos conforme a solução consenso e silhueta média da solução.

A solução consenso obtida é equivalente à solução π_1 do Conjunto Base submetido ao método proposto. Conforme foi apresentado na seção 1 (Introdução), a utilização de comitê de agrupamentos pode possuir diferentes objetivos. Uma vez que a solução consenso foi equivalente a uma solução base, destaca-se que objetivos *Reaproveitamento de Conhecimento* e *Consistência* foram alcançados, uma vez que os padrões foram identificados (objetos que ocorrem juntos com frequência permanecem juntos na solução consenso) e que a solução está em concordância com as soluções do conjunto base.

4 – ESTATÍSTICA DE HOPKINS

O *Teste de Tendência de Agrupamento*, também descrito como um *Teste de Aleatoriedade Espacial*, como o próprio nome sugere, consiste em verificar se em uma instância existe uma tendência à formação de agrupamentos de um dado conjunto de objetos *X* com *n* unidades $X = \{x_1, x_2, .., x_n\}$, em um espaço *p*-dimensional, $x_i = \{ x_i^1, x_i^2, .., x_i^p \}$ [Banerjee 2004] [Han and Kamber, 2012]. Nesse sentido, a Estatística de Hopkins (EH) utiliza um critério interno, em que nenhuma informação a priori é necessária para a realização das análises. Além do conjunto X, dois outros conjuntos são considerados nessa abordagem, sejam eles:

- **X*:** trata-se de uma amostra do conjunto X ($X^* \subset X$) com *m* objetos que são selecionados de maneira aleatória.
- **A:** possui *m* objetos construídos artificialmente com valores aleatórios no espaço de cada uma das *p*-dimensões.

Após a apresentação dos conjuntos de objetos utilizados, devem ser apresentadas as distâncias utilizadas:

- $\mathbf{w_j}$**:** distância entre um objeto $x^* \in X^*$ até o objeto de $X - \{x^*\}$ mais próximo.
- $\mathbf{u_j}$**:** distância entre um objeto $a \in A$ até o objeto mais próximo em X .

Na Equação 13, para cada objeto, são consideradas as distâncias w_j e u_j. Busca-se a maximização de *H*, cujo valor pertence ao intervalo [0,1]. Em uma instância em que objetos estão em grupos bem definidos, coesos e bem separados, a distância média entre os objetos é pequena. Nesse caso o somatório de w_j tende a ser próximo de 0 e, consequentemente, *H* é

próximo de 1. Já em instâncias em que os objetos estão dispersos no espaço, os somatórios de w_j e u_j são próximos, ou seja, o valor de *H* é próximo a 0.5. Conforme [Banerjee 2004], existem três classes em que a instância pode ser classificada:

- ***Objetos são regularmente espaçados***: instância sem tendência a formação de agrupamentos. Em resultados da literatura, para essa classe, o valor de *H* variou no intervalo (0,0.3].
- ***Objetos distribuídos de maneira aleatória no espaço:***indica que o conjunto de objetos não tem uma estrutura propícia para o agrupamento (*H próximo a 0.5)*.
- ***Existe uma tendência a formação de agrupamentos*:** existem grupos bem definidos. Em resultados da literatura, para essa classe, o valor de *H* variou no intervalo [0.7,1).

$$\max H = \frac{\sum_{j=1}^{m} u_j}{\sum_{j=1}^{m} u_j + \sum_{j=1}^{m} w_j} \tag{13}$$

4.1 – Experimentos Computacionais: Estatística de Hopkins

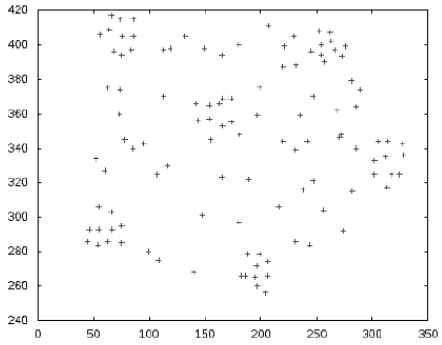
Como foi apresentado no início da seção, a EH utiliza uma amostra do conjunto de objetos da instância ($X^* \subset X$) e um conjunto de objetos artificiais *A*, cujo os atributos possuem valores aleatórios no espaço de cada uma das *p*-dimensões. Uma vez que fatores aleatórios foram considerados (tanto em X^* quanto em *A*), tornou-se necessário o desenvolvimento de um algoritmo que realizasse diferentes iterações com o objetivo de obter estatísticas para os valores de H. A cada iteração o algoritmo seleciona objetos para a formação do conjunto X^* e constrói objetos artificiais para o conjunto *A*.

As Figuras 10 (a) e (b) apresentam duas instâncias e as Medianas dos valores de *H* para a instância *100p7c1* classificada como "*não comportada*" que possui 112 objetos e para a instância 100p7c, em que é possível identificar 7 grupos bem definidos (coesos e bem separados), com 100 objetos. Nesse exemplo o algoritmo foi executado com 1000 iterações.

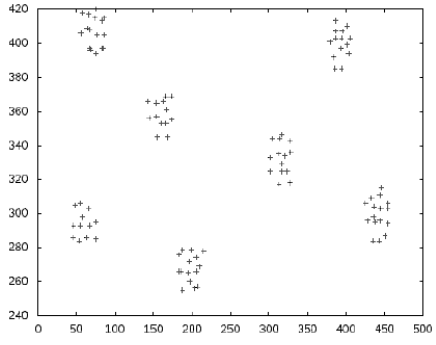
No experimento da presente subseção foram consideradas 51 instâncias propostas e utilizadas por [Cruz, 2010], que possuem entre 100 e 2000 objetos. Além disso, 63% das instâncias utilizadas possuem grupos bem definidos (denominadas "*comportadas*") e as demais 37% são as instâncias denominadas "*não comportadas*", conforme a classificação indicada por [Cruz, 2010]. É importante ressaltar que a classificação proposta pelo autor foi realizada em uma análise visual, durante a construção das instâncias. Dessa forma, a hipótese é que uma instância classificada como *comportada* também seja classificada como instância com *tendência à formação de agrupamentos*.

Embora possam surgir outras hipóteses relacionando instâncias *não comportadas* ou mesmo a instâncias *sem tendência à formação de agrupamentos*, o foco do presente trabalho está em utilizar o método baseado em coassociação apenas em instâncias consideradas *comportadas* e que possuam *tendência à formação de agrupamentos*.

Em experimentos preliminares foram utilizadas conjuntos de amostras (X^*) com tamanhos 1%, 3%, 5%, 10% e 15% em relação à quantidade de objetos da instância. Além disso, foram consideradas as quantidade de iterações: 10, 100, 500 e 1000. Após a análise dos resultados obtidos nos experimentos preliminares, foi selecionada uma configuração em que o tamanho da amostra é de 1% e o algoritmo realiza 10 iterações. Essa escolha deve-se ao fato da compatibilidade entre os resultados concernentes ao valor de H e, além disso, do reduzido custo computacional necessário. A Tabela 7 apresenta as estatísticas dos resultados obtidos com a configuração selecionada considerando, separadamente, apenas as instâncias "*comportadas*" e as instâncias "*não comportadas*".



(a) *100p7c1*: *H**=0,516



(b) *100p7c*: *H**=0,831

Figura 10: Instâncias 100p7c1 e 100p7c. H* é a Mediana dos valores H em 1000 iterações.

Estatística Hopkins (H)			
Instância	**Não Comportadas**	**Comportadas**	**Tempo (s)**
Maior	0,95	***0,96***	0,03
Menor	0,45	***0,77***	0
Média	0,66	***0,90***	0,01
Mediana	0,64	***0,90***	0

Figura 11: resultados da Estatística de Hopkins com 1% de amostra e 10 iterações.

Com base na tabela da Figura 11, a EH identificou a *Tendência de Agrupamentos* em todas as instâncias consideradas *comportadas*, em que a média e a mediana foram 0,9. Em relação ao conjunto de instâncias *não comportadas*, a média e a mediana foram inferiores a 0,7. É importante ressaltar que os valores extremos (menor e maior), embora sejam apresentados na tabela, não são considerados na análise. A explicação para isto é que, eventualmente, uma configurações dos objetos do conjunto de amostra e dos objetos artificiais podem resultar em um falsos positivos, ou seja, indicar tendência à formação onde não existe. Por exemplo, em um dos resultados para as instâncias *não comportadas* o resultado de *H* foi 0,95. É importante ressaltar que uma instância classificada como *não comportada* pelo autor (em [Cruz, 2010]) não necessariamente corresponde a uma instância *sem tendência à formação de agrupamentos*.

A partir dos resultados e análises realizadas na presente subseção foram selecionadas 19 instâncias para a realização dos experimentos com o método proposto, em que todas são classificadas como *comportadas*, possuem *tendência à formação de agrupamentos*. Essas instâncias possuem entre 100 e 2000 objetos e entre 3 e 26 grupos.

5 – EXPERIMENTOS COMPUTACIONAIS

As implementações dos algoritmos propostos foram feitas em Linguagem *C++*, utilizando o paradigma orientação à objetos. É uma prática comum em abordagens sistemáticas, para a identificação da quantidade ideal de grupos em problemas de agrupamento automático, utilizar k=2..k_{max} , sendo $k_{max} = \sqrt{n}$ ([Pal and Bezdek, 1995][Pakhira et. al., 2005][Campello et. al., 2009]. Em [Han and Kamber, 2012], entretanto, um método simples para a estimativa do número ideal de grupos consiste em utilizar valores inteiros de *k* próximos a $\sqrt{n/2}$, na expectativa que cada grupo possua cerca de $\sqrt{2n}$ objetos. Com o objetivo de atender a ambos os intervalos apresentados na literatura, neste trabalho foi considerado k=2.. $\sqrt{n}$.

O método proposto considera um conjunto S, composto por 28 soluções obtidas por meio da utilização do *Método de Classificação Baseado em Densidade* proposto por [Semaan, et. al., 2012]. Destaca-se que foram consideradas apenas soluções válidas (viáveis), em que a quantidade de grupos está no intervalo pré-determinado (k =2.. $\sqrt{n}$). Na tabela da Figura 12 a coluna |Base| indica a quantidade de soluções que compõe o conjunto *S.*

A Figura 12 apresenta comparativos entre as soluções do *Conjunto Base* (entrada) e os resultados obtidos com o método proposto. A coluna "Silhueta \ Conjunto Base" apresenta a média e a maior Silhueta para cada instância do *Conjunto Base*. Já a Coluna "Silhueta \ Comitê" apresenta a silhueta da melhor solução obtida com o método proposto. A coluna *gap* apresenta a diferença (Equação 14) entre a maior silhueta obtida com o método proposto (s_{obtido}) e a maior silhueta existente no Conjunto Base ($s_{referencia}$). São apresentados também os números de grupos das soluções com os maiores valores para o índice silhueta (coluna "Qtde Grupos"), o tempo de processamento e o tamanho do Conjunto Base (quantidade de soluções de entrada).

$$gap = s_{obtido} - s_{referencia} \quad (14)$$

Conforme os dados apresentados na Figura 12, o método proposto obteve as melhores soluções para todas as instâncias ($gap \geq 0$). Além disso, em cerca de 26% das instâncias os resultados obtidos foram superiores aos valores das soluções de entrada ($gap > 0$). Esses resultados indicam que o método foi capaz de formar soluções de alta qualidade com a utilização dos padrões identificados nas soluções do conjunto base.

Ainda com base nos resultados reportados na Figura 12, observa-se que os números de grupos das novas soluções obtidas ($gap > 0$) são diferentes em relação aos apresentados nas soluções do Conjunto Base. Com exceção dos resultados obtidos para a instância *100p10c*, todos os números de grupos identificados pelo método proposto correspondem aos valores relatados no trabalho em que essas instâncias foram propostas [Cruz, 2010].

	Silhueta				Quantidade de Grupos (k)		Comitê	
	Conjunto Base [Semaan et. al., 2012]		Comitê					
Instâncias	**Média**	**Maior**	**Maior**	***gap***	**Base**	**Comitê**	**Tempo (s)**	**|Base|**
100p3c	0,6677	**0,7858**	**0,7858**	***0***	3	3	0,03	24
100p7c	0,7483	**0,8339**	**0,8339**	***0***	7	7	0,06	20
100p10c	0,4596	**0,6917**	**0,6917**	***0***	8	8	0,00	***6***
200p4c	0,6796	**0,7725**	**0,7725**	***0***	4	4	0,05	23
300p3c	0,5486	**0,7664**	**0,7664**	***0***	3	3	0,10	28
400p3c	0,6246	**0,7986**	**0,7986**	***0***	3	3	0,08	27
500p3c	0,5848	**0,8249**	**0,8249**	***0***	3	3	0,13	28
600p15c	0,6805	**0,7812**	**0,7812**	***0***	15	15	3,59	23
700p4c	0,6225	**0,7970**	**0,7970**	***0***	4	4	0,10	26
800p23c	0,6216	**0,7874**	**0,7874**	***0***	23	23	7,42	21
900p5c	0,5326	**0,7160**	**0,7160**	***0***	5	5	0,47	27
900p12c	0,6936	**0,8409**	**0,8409**	***0***	12	12	3,47	26
1000p6c	0,5362	**0,7357**	**0,7357**	***0***	6	6	1,09	27
1000p14c	0,6711	0,8075	**0,8306**	***0,0232***	15	***14***	5,78	26
1300p17c	0,7041	0,8051	**0,8229**	***0,0178***	18	***17***	6,30	24
1800p22c	0,6563	0,7908	**0,8036**	***0,0129***	23	***22***	13,35	24
1900p24c	0,6386	0,7871	**0,7992**	***0,0121***	25	***24***	16,73	24
2000p11c	0,5355	**0,7130**	**0,7130**	***0***	11	11	3,33	26
2000p26c	0,6196	0,7890	**0,8005**	***0,0115***	27	***26***	17,30	23

Figura 12: comparativos com soluções do conjunto Base.

Para a instância *100p10c*, especificamente, o tamanho do conjunto de soluções utilizado pelo método proposto foi reduzido, com apenas seis soluções (coluna |Base| da Figura 12). Além disso, o número de grupos dessas soluções foram 2, 4, 8 e 10 (duas soluções com 2 e 10 grupos). Uma vez que a maioria das soluções do conjunto base possui quantidade de grupos inferior ao considerado ideal pela literatura (10 unidades conforme a tabela da Figura 13), o processo de identificação de padrões e o método proposto tendem a formar soluções também com quantidade inferior de grupos.

	Silhueta				Quantidade de Grupos (*k*)		
Instâncias	**[Semaan et. al., 2012]**	**[Cruz, 2010]**	**Comitê**	***gap***	**[Semaan et. al., 2012]**	**[Cruz, 2010]**	**Comitê**
100p3c	**0,7858**	***0,7858***	***0,7858***	0	3	3	3
100p7c	**0,8339**	***0,8338***	***0,8339***	0,0001	7	7	7
100p10c	**0,6917**	***0,8336***	0,6917	***-0,1419***	8	**10**	8
200p4c	**0,7725**	***0,7725***	***0,7725***	0	4	4	4
300p3c	**0,7664**	***0,7663***	***0,7664***	0,0001	3	3	3
400p3c	**0,7986**	***0,7985***	***0,7986***	0,0001	3	3	3
500p3c	**0,8249**	***0,8249***	***0,8249***	0	3	3	3
600p15c	**0,7812**	***0,7812***	***0,7812***	0	15	15	15
700p4c	**0,7970**	***0,7969***	***0,7970***	0,0001	4	4	4
800p23c	**0,7874**	***0,7873***	***0,7874***	0,0001	23	23	23
900p5c	**0,7160**	***0,7160***	***0,7160***	0	5	5	5
900p12c	**0,8409**	0,8404	***0,8409***	***0,0005***	12	12	12
1000p6c	**0,7357**	**0,7356**	***0,7357***	0,0001	6	6	6
1000p14c	0,8075	**0,8306**	***0,8306***	0	15	14	14
1300p17c	0,8051	**0,8229**	***0,8229***	0	18	17	17
1800p22c	0,7908	**0,8036**	***0,8036***	0	23	22	22
1900p24c	0,7871	0,7990	***0,7992***	***0,0002***	25	24	24
2000p11c	**0,7130**	***0,7129***	***0,7130***	0,0001	11	11	11
2000p26c	0,7890	0,7790	***0,8005***	***0,0215***	27	26	26

Figura 13: comparativos com resultados da literatura.

A Figura 13 apresenta comparativos entre os resultados do método proposto e alguns resultados da literatura. Foram considerados os melhores resultados obtidos pelos trabalhos [Cruz, 2010] e [Semaan, et. al., 2012]. Conforme relatado na seção 2 do presente trabalho, embora em [Cruz, 2010] sejam propostos vários algoritmos heurísticos, os *Algoritmos Evolutivos* ([Reeves 2010]) destacaram-se. Já em [Semaan et. al., 2012] foi proposto um *Método de Classificação Baseado em Densidade*.

Na tabela da Figura 13 foram apresentados os índices Silhueta e os números de grupos das melhores soluções do método proposto e dos resultados relatados na literatura.

Ainda com base na Figura 13, observa-se que o número de grupos identificado como *ideal* no método proposto é igual ao número relatado da melhor solução da literatura para todas as instâncias, novamente com exceção da *100p10c*. Além disso, embora os experimentos do presente trabalho tenham utilizados as soluções de [Semaan et. al., 2012], quando o número de grupos das soluções obtidas é diferente, o resultado obtido pelo método proposto é melhor.

6 –CONCLUSÕES E TRABALHOS FUTUROS

O presente trabalho trouxe a proposta de um método baseado em combinação de soluções que considera a técnica de Particionamento de Grafos para a identificação do número ideal de grupos em problemas de agrupamento automático. Para isso, foi utilizado o índice silhueta, que combina características como coesão e separação. A técnica utilizada calcula a similaridade entre dois objetos por meio do número de grupos compartilhados entre eles nas soluções do conjunto base. Em seguida, com base em uma heurística para o particionamento de grafos (hipergrafos) são formadas soluções, avaliadas por meio do índice silhueta.

Os resultados apresentados neste estudo indicam que o método proposto foi capaz de identificar padrões nas soluções do conjunto base, obtidas com a utilização do método de classificação baseado em Densidade da literatura. Em relação aos índices silhuetas do conjunto base, as soluções obtidas com a implementação do método proposto foram equivalentes ou superiores, com exceção dos resultados obtidos para a instância 100p10c, em que o conjunto base submetido ao método foi reduzido e as soluções base possuíam número de grupos igual ou inferior ao da melhor solução da literatura.

Não obstante, de forma a reforçar ainda mais esta análise, em trabalhos futuros serão efetuadas novas análises com mais instâncias da literatura. Seguem algumas propostas para trabalhos futuros:

- *Utilização de técnicas de processamento paralelo e distribuído:* no particionamento de grafos cada núcleo pode trabalhar com a formação da solução consenso resultante da seleção de uma partição H^i.
- *Implementação de Algoritmos baseados em metaheurísticas:* o algoritmo proposto para o particionamento do grafo consiste em um procedimento de construção de soluções. Submeter as soluções obtidas a procedimentos de busca local e perturbações pode resultar em soluções de melhor qualidade.
- *Utilizar Conjunto Base com Lista Restrita de Candidatos:* a utilização de subconjuntos do Conjunto Base considerado neste trabalho pode resultar em soluções diversificadas e de boa qualidade (alto valor do Índice Silhueta). De maneira aleatória, *m* soluções distintas são selecionadas do Conjunto Base e formam novos conjunto Base $\prod_i^*$.

REFERÊNCIAS

[Alves et. al. 2006] 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.

[Bandyopadhyay and Maulik, 2001] Bandyopadhyay, S. & U. Maulik (2001). *Nonparametric genetic clustering: Comparison of validity indices. IEEE Transactions on Systems, Man and Cybernetics*, Part C : Applications and Reviews. 31 (1), 120–125.

[Bandyopadhyay and Maulik, 2002b] Bandyopadhyay, S. & U. Maulik (2002b). *Genetic clustering for automatic evolution of clusters and application to image classification. Pattern Recognition* 35, 1197–1208.

[Banerjee 2004] Banerjee, A.. Validating clusters using the hopkins statistic. IEEE International Conference on Date of Conference, 2004.

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

[Campello et. al., 2009] Campello, R. J. G. B., E. R. Hruschka, & V. S. Alves (2009). *On the efficiency of evolutionary fuzzy clustering. Journal of Heuristics* 15 (1), 43–75.

[Campello et. al., 2009b] Campello, R.J.G.B., Hruschka, E.R., Alves, V.S. *On comparing two sequences of numbers and its applications to clustering analysis*. *Information Sciences* 129(8), 2009.

[Cole, 1998] Cole, R. M. (1998). *Clustering with genetic algorithms*. MSc Dissertation, Department of Computer Science, University of Western Australia.

[Cowgill, 1999] Cowgill, M. C., R. J. Harvey, & L. T. Watson (1999). *A genetic algorithm approach to cluster analysis. Computational Mathematics and its Applications* 37, 99–108.

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

[Dias and Ochi, 2003] 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 et al., 1995] 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 et. al., 1996] 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.

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

[Han and Kamber, 2012] Han, J., e Kamber, M., *Cluster Analysis*. In: Morgan Kaufmann. Publishers (eds.), *Data Mining: Concepts and Techniques*, 3 ed., chapter 8, New York, USA, Academic Press, 2012.

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

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

[Hruschka et. al., 2004a] 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 et. al., 2004b] 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 et. al., 2006] 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.

[Jain and Dubes, 1988] Jain, A. & R. Dubes (1988). *Algorithms for Clustering Data*. Prentice Hall.

[Johnson A.R. e Wichern D.W**.,** 2002]. *Applied Multivariate Statistical Analysis*. Prentice Hall. Fifth Edition.

[Kumar et. al., 2009] Kumar, V. ; Steinbach, M. ; Tan, P. N. Introdução ao Data Mining - Mineração De Dados. Ciência Moderna, 2009.

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

[Ma et. al., 2006] Ma, P. C. H., K. C. C. Chan, X. Yao, & D. K. Y. Chiu (2006). *An evolutionary clustering algorithm for gene expression microarray data analysis*. IEEE Trans. Evolutionary Computations 10 (3), 296–314.

[Maronna and Jacovkis, 1974] Maronna, R.; Jacovkis, P. M. (1974). *Multivariate clustering procedures with variable metrics*. Biometrics30, pp. 499-505.

[Matloff 2011] Matloff, N. *The Art of R Programming: A Tour of Statistical Software Design*. No Starch. Press, 2011.

[Naldi et. al., 2009] Naldi, M. C. ; Faceli, K. ; Carvalho, A. C. P. L. F. . Uma Revisão Sobre Combinação de Agrupamentos. Revista de Informática Teórica e Aplicada, v. 16, p. 25-51, 2009.

[Naldi and Carvalho, 2007] 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, pp. 139–144. Evere.

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

[Pakhira et. al., 2005] Pakhira, M., S. Bandyopadhyay, & U. Maulik (2005). *A study of some fuzzy cluster validity indices, genetic clustering and application to pixel classification*. Fuzzy Sets Systems 155 (2), 191–214.

[Pal and Bezdek, 1995] Pal, N. & J. Bezdek (1995). *On cluster validity for the fuzzy c-means model. IEEE Transactions of Fuzzy Systems* 3 (3), 370–379.

[Pan and Cheng, 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.

[Pelleg and Moore, 2000] Pelleg, D. & A. Moore (2000). *X-means: extending k-means with efficient estimation of the number of clusters*. In *Proceedings of the Seventeenth International Conference on Machine Learning*, pp. 727–734.

[Reeves 2010] Reeves, C. R. *Genetic algorithms*. In Glover, F. and Kochenberger, G., editors, *Handbook of Metaheuristics*, pages 109 139. Kluwer Academic Publishers, 2010.

[Rousseeuw, 1987] 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 et. al., 2012] Semaan, G. S., Cruz, M.D., Brito, J. A. M., and Ochi, L. S. (2012) "*Proposta de um método de classificação baseado em densidade para a determinação do número ideal de grupos em problemas de clusterização*", vol. 10 número 4.

[Soares and Ochi, 2004] 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*, 2004, Havana. Proc. of the XII CLAIO (em CD-ROM). ALIO, 2004. v.1, p. 7 -13.

[Soares, 2004] Soares, A. S. R. F. Metaheurísticas para o Problema de Clusterização Automática, Dissertação de Mestrado, UFF - Niterói, 2004.

[Strehl and Ghosh, 2002] A. Strehl and J. Ghosh, *Cluster ensembles: A knowledge reuse framework for combining multiple partitions*. *Journal of Machine Learning Research* - JMLR, vol. 3, pp. 583–617, 2002.

[Tanimoto, 1958] T. T. Tanimoto. *An elementary mathematical theory of classification and prediction*. Tech. Rep. 13, IBM Report, 1958.

[Tseng and Yang, 2001] Tseng, L. & . Yang, S.B.. *A genetic approach to the automatic clustering problem*. Pattern Recognition 34, 415–424, 2001.

[Zalik, 2008] An Efficient *K'-Means Clustering Algorithm*, Pattern Recognition Letters 29, 2008.