Os algoritmos Fuzzy C-Means, Robust C-Prototypes e Unsupervised Robust C-Prototypes aplicados à uma base de dados das bacias hidrográficas da Região de Criciúma.

Ademar Crotti Junior[1], Maicon Bastos Palhano[1], Gabriel Felippe[1], Carlyle T. B. de Menezes[2], Priscyla Waleska T. de Azevedo Simões[1], Merisandra Cortês de Mattos Garcia[1]

[1]Grupo de Pesquisa em Inteligência Computacional Aplicada do Curso de Ciência da Computação – Universidade do Extremo Sul Catarinense (UNESC) – Criciúma – Santa Catarina – Brasil

[2]Grupo de Pesquisa em Gestão de Recursos Hídricos e Restauração de ambientes Alterados do Curso de Engenharia Ambiental – Universidade do Extremo Sul Catarinense (UNESC) – Criciúma – Santa Catarina - Brasil

jcrotti@gmail.com, {maiconpalhano, cbm, pri, mem}@unesc.net, gabrielheavy@hotmail.com

Abstract: *Currently the databases are large and robust, enabling the application of tools to somehow discover something new in these databases with automatically form, therewith arise the process of Knowledge Discovery in Databases, have with principal step the data mining, wich search patters and relations in databases. Therefore, this article present the data clustering through fuzzy logic method by application of Fuzzy C-Means algorithms, Robust C-Prototypes and unsupervised Robust C-Prototypes, on a database of hydrographic basins of the region's coal Criciúma.*

Keywords: *Data Mining, Fuzzy Logic, RCP, URCP, FCM.*

Resumo: *Atualmente as bases de dados são grandes e robustas, possibilitando a aplicação de ferramentas para de alguma forma descobrir algo novo nessas bases de dados de forma automática, com isso surge o processo de Descoberta de Conhecimento em Bases de Dados, tendo como principal etapa o Data Mining, que busca padrões e relações nas bases de dados. Sendo assim, este artigo apresenta a clusterização de dados por meio do método de lógica fuzzy pela aplicação dos algoritmos Fuzzy C-Means, Robust C-Prototypes e Unsupervised Robust C-Prototypes, em uma base de dados das bacias Hidrográficas da região Carbonífera de Criciúma.*

Palavras-chave*: Data Mining, Lógica Fuzzy, RCP, URCP, FCM.*

1. Introdução

A capacidade de armazenamento digital possibilitou que as organizações pudessem ter bases de dados grandes e robustas, crescendo cada vez mais, porém, geralmente, pouco é feito com esse aglomerado de dados, consequentemente perde-se informações úteis

nesse meio. Desta forma, a fim de se extrair algum tipo de conhecimento nessas bases de dados, tem-se o Processo de Descoberta de Conhecimento em Bases de Dados (DCBD) ou Knowledge Discovery in Database (KDD).

Segundo Fayyad, Piatetsky-Shapiro e Smith (1996, tradução nossa) DCBD é um processo automático de identificação de características e relacionamentos novos e válidos nos dados, e que possam ser transformados em conhecimento úteis e compreensíveis, composto de diversas etapas sendo mostrada de forma gráfica pela figura 1.

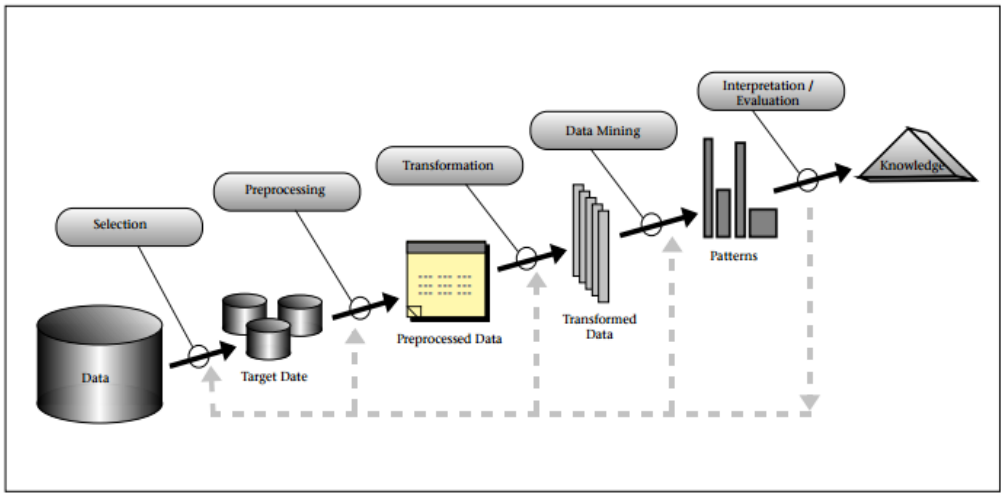

Figura 1. Etapas do processo de DCBD definidas por FAYYAD.

Fonte: FAYYAD, U. M.; PIATETSKY-SHAPIRO, G.; SMYTH P. (1996)

O *Data Mining* (DM) é a principal etapa do processo DCBD, pois é nela que ocorre a busca por padrões nos dados. O DM consiste na aplicação de diferentes conhecimentos, tais como de Inteligência Computacional, Banco de dados, Estatística, Aprendizado de Máquina e Reconhecimento de padrões, empregando-se tarefas e métodos com características distintas, que são aplicados de acordo com o objetivo da descoberta de conhecimento (GOLDSCHMIDT; PASSOS, 2005). Dentre essas tarefas, tem-se a clusterização.

A tarefa de Clusterização, segundo Han e Kamber (2006, tradução nossa), reúne um conjunto de dados em grupos de objetos similares, formando *clusters*. Na clusterização busca-se maximizar a semelhança entre objetos do mesmo *cluster* e minimizar a semelhança entre *clusters* (LAROSE, 2005, tradução nossa).

As metodologias de clusterização têm sido largamente utilizadas em numerosas aplicações, incluindo reconhecimento de padrões, análise de dados, processamento de imagens, e pesquisa de mercado (CARLANTONIO, 2001). Serra (2002) e Goldschmidt e Passos (2005) mostram, que os resultados provenientes dessa tarefa podem ser utilizados como etapas de pré-processamento para a aplicação de outras tarefas de DM, tais como a classificação e sumarização.

Dentre os métodos tradicionais aplicados à tarefa de clusterização, a lógica *fuzzy* se destaca, pois faz com que a clusterização fique mais próxima da realidade, onde é possível um dado pertencer a mais de um *cluster* ao mesmo tempo, ou seja, a lógica *fuzzy* ao contrário de outros métodos, não faz com que um dado seja forçado a pertencer

a um único *cluster* e sim que um determinado dado tenha graus de pertinência a determinados *clusters*.

Existem diversos algoritmos que implementam o modelo *fuzzy* para a tarefa de Clusterização, como: Gustafson-Kessel, Gath-Geva, *Fuzzy* C-Means, *Robust C-Prototypes e Unsupervised Robust C-Prototypes*, sendo nesta pesquisa utilizado o FCM, RCP e URCP.

2. Algoritmo *Fuzzy* C-Means

O *Fuzzy* C-Means (FCM) é uma adaptação do algoritmo *K-means*, sendo que sua versão final foi proposta por James C. Bezdek em 1973 (BEZDEK et al, 2005, tradução nossa). Este algoritmo foi o primeiro a utilizar o elemento *fuzzificador*, que determina o grau de *fuzzyficação* entre os elementos, considerando que quanto maior o seu valor mais os elementos serão relacionados. A fim de obter *clusters* de qualidade foram realizados testes que definiram o valor 2 como recomendado (COX, 2005, tradução nossa).

O FCM utiliza a distância Euclidiana, gerando *clusters* esféricos, assim como o *K-means*. A Figura 2 mostra o resultado gerado pelo FCM, onde os dados foram divididos em dois *clusters*, e os elementos entre os grupos pertencem a estes dois *clusters* com diferentes pertinências.

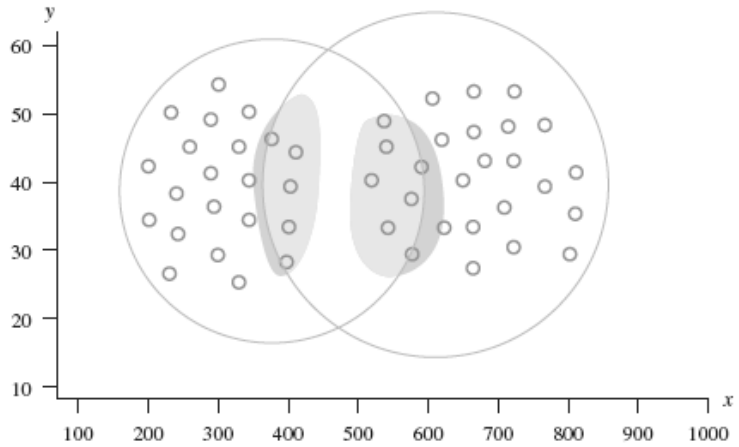

Figura 2. Clusterização pelo algoritmo FCM
Fonte: Adaptado de COX, E. (2005)

O FCM é um algoritmo tradicional na tarefa de clusterização, sendo que muitos algoritmos foram criados baseando-se nele, com o objetivo de obter melhores resultados, geralmente considerando sua dificuldade em lidar com dados ruidosos e suas limitações na identificação de *clusters* de diferentes formas (BEZDEK et al, 2005, tradução nossa).

3. Os Algoritmos Robust C-Prototypes e Unsupervised Robust C-Prototypes

Os algoritmos RCP e URCP, ambos foram propostos por Hichem Frigui e Raghu Krishnapuram, em 1996 no artigo denominado *A Robust Algorithm for Automatic Extraction of an Unknown Number of Clusters from Noisy Data*, na *Pattern Recognition Letters*, sendo uma adaptação do algoritmo FCM.

A distância Euclidiana utilizada pelo FCM é bastante sensível a ruídos, o que em uma base de dados sem presença de ruídos ou *outliers* não fazem diferença, porém em uma base de dados contaminada, pode interferir no resultado final do algoritmo (DÖRING; LESOT; KRUSE, 2006, tradução nossa).

Dessa forma, o RCP incorpora estimadores robustos ao FCM, para diminuir a interferência de ruídos no processo de execução do algoritmo e consequentemente não interferir no resultado, porém para a execução do RCP é preciso ter conhecimento prévio e bem específico da base de dados a ser utilizada, pois deve-se informar a quantidade exata de *clusters* presentes na base como parâmetro (OLIVEIRA; PEDRYCZ, 2007, tradução nossa).

Quando não se tem esse conhecimento específico, recomenda-se o URCP, pois também utiliza estimadores robustos para tratar os ruídos, porém não é preciso informar a quantidade exata de *clusters* presentes na base, e sim a quantidade máxima de *clusters* a serem considerados pelo algoritmo, que em sua execução determina o número ideal de *clusters* (FRIGUI; KRISHAPURAM, 1996, tradução nossa).

4. Validação dos algoritmos

A clusterização, na maioria das vezes, é efetuada sem conhecimento prévio da base de dados, sendo a quantidade de grupos, necessária para efetuar o processo, um parâmetro difícil de definir. Devido a isto, foram desenvolvidos índices que ajudam o usuário a definir o número ótimo de *clusters* existentes na base (KIM et al, 2004, tradução nossa).

Índices de validação são utilizados para encontrar o número ótimo de *clusters* em uma determinada base de dados. Eles ajudam a definir a quantidade de *clusters* que encontra partições estáveis, que melhor definem e explicam a estrutura da base de dados em questão (KIM et al, 2004, tradução nossa).

Bezdek et al (2005, tradução nossa) propôs os seguintes índices para validação *fuzzy* de *clusters*:

> **a) Índice de Coeficiente de Partição:** a medida de validação Coeficiente de Partição (Partition Coefficient) indica o número ótimo de categorias de um espaço amostral quando seu valor máximo é atingido (BEZDEK apud KOSANOVIC, 1995b). O valor deve se encontrar no intervalo de [1/*C*, 1], indicando inexistência de grupos bem definidos quando este valor estiver próximo de 1/*C* (BEZDEK et al, 2005, tradução nossa);
>
> **b) Índice de Partição Entrópica:** a medida de validação Participação Entrópica (Partition Entropy) também valida o número ideal de categorias para um espaço amostral particionado. Este número ideal é alcançado quando o valor mínimo é atingido (BEZDEK apud KOSANOVIC, 1995b).
>
> O valor tende para zero quando encontra grupos bem definidos, sendo que um valor próximo do limite superior do intervalo indica a ausência de grupos definidos no conjunto de dados ou a incapacidade do algoritmo de obtê-las. Seu valor fica no intervalo [0, log(*C*)];

Outro índice proposto foi desenvolvido por Xie e Beni, sendo chamado de Xie-Beni, devido a seus criadores. Este índice procura definir o número ótimo de *clusters* considerando a separação e compactação dos *clusters*. Quando o índice encontra um valor baixo significa que os grupos são bem separados e compactos (KIM et al, 2004, tradução nossa) .

5. Shell Orion Data Mining Engine

Esta ferramenta encontra-se em desenvolvimento pelos discentes e docentes do Grupo de Pesquisa em Inteligência Computacional Aplicada da Universidade do Extremo Sul Catarinense (UNESC).

Dentre as tarefas de *data mining* disponibilizadas na *Shell Orion*, tem-se a clusterização pelos algoritmos K-Means, Kohonen, Gustafson–Kessel, Gath-Geva, Density-based Spatial Clustering of Applications With Noise (DBSCAN), Fuzzy C-Means, Robust C-Prototypes (RCP) e Unsupervised Robust C-Prototypes (URCP).

6. Base de dados

A base de dados utilizada nesta pesquisa, para a aplicação dos três algoritmos de clusterização, foi proveniente do 4º Relatório de Monitoramento das Bacias Hidrográficas da Região Carbonífera de Criciúma, onde contém índices de amostras coletadas nas regiões hidrográficas afetadas por empresas carboníferas. Esta base de dados mostra vários índices medidos, como pH, Ferro, Manganês, Acidez, Alumínio entre outros, contando com 1723 registros de coletas realizadas do ano de 2002 a 2009.

Na aplicação do *data mining* empregam-se ferramentas computacionais, denominadas *shells*, as quais em sua maioria são comerciais, tendo-se algumas iniciativas gratuitas, como por exemplo, a *Shell Orion Data Mining Engine,* utilizada nesta pesquisa para a análise da base de dados mencionada anteriormente.

7. Resultados obtidos

Antes de aplicar uma base de dados em ferramentas de DM é necessário realizar o pré-processamento da base utilizada, sendo nesta etapa que ocorrem alterações de modo a tornar a base de dados de acordo com o algoritmo de DM à ser utilizado, retirando do conjunto de dados impurezas como dados faltantes, informações errôneas, dados duplicados, ou qualquer erro que possa interferir no processo de execução do algoritmo ou na interpretação do conhecimento. Após esta etapa é que se pode executar os algoritmos e a interpretação do conhecimento gerado.

A Figura 3 mostra a interface inicial para inserção dos parâmetros principais dos algoritmos FCM, RCP e URCP na Shell Orion.

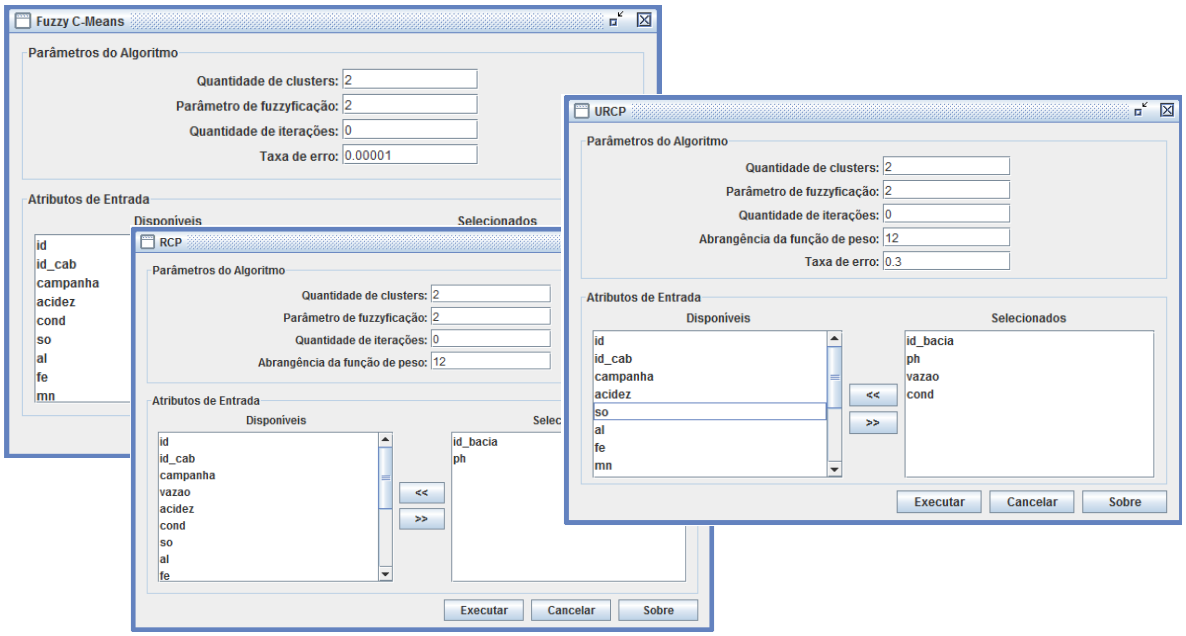

Figura 3. Telas de Parâmetros dos algoritmos FCM, RCP e URCP na *Shell Orion*.

Quanto aos atributos utilizados na execução, foram usados o id das bacias, pH, Ferro e Alumínio, depois da escolha dos atributos foram executados os algoritmos e posteriormente analisou-se e interpretou-se os resultados gerados.

Os resultados do FCM, RCP e URCP, foram muito parecidos, como se pode observar na Figura 4, pois a separação dos *clusters* pelos algoritmos ficou da mesma forma. Todos separaram em dois *clusters*, onde o *cluster* 1 (em azul) foi considerado como um grupo com dados ruins, com valores elevados, e o *cluster* 2 (em vermelho) com valores médios a bons.

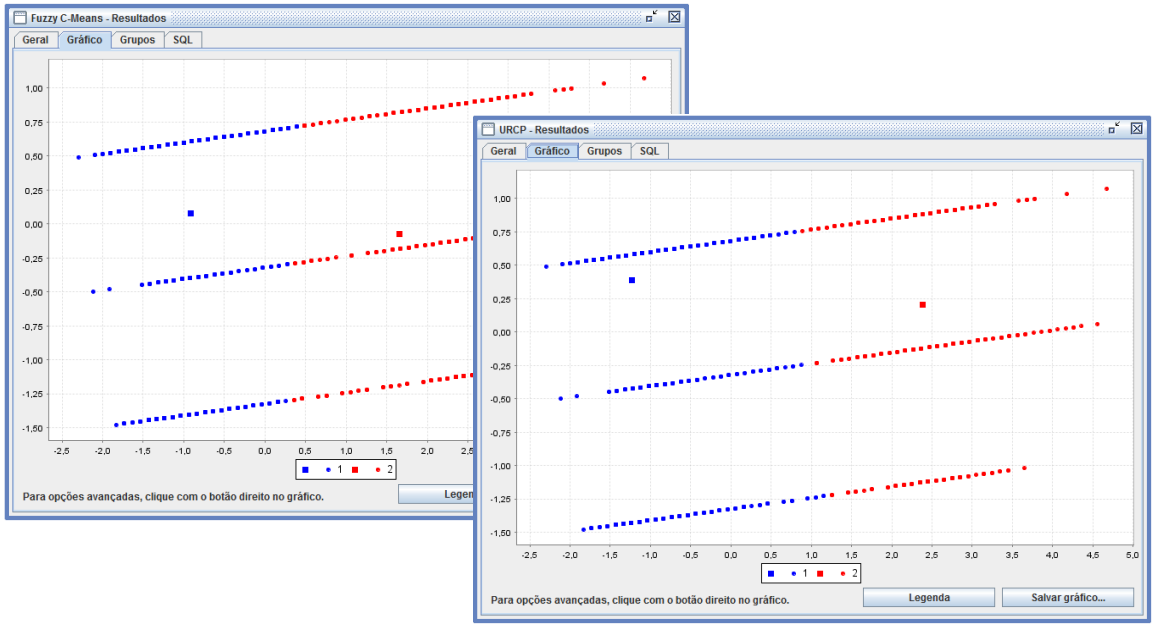

Figura 4. Resultado da Clusterização dos algoritmos URCP e FCM na *Shel Orion*.

Também foram considerados os resultados dos Índices de validação dos algoritmos, como podem ser visto na Tabela 1.

Tabela 1. Índices de validação encontrados pelos algoritmos FCM, RCP e URCP na *Shell Orion*

ÍNDICES	FCM	RCP	URCP
Índice de Coeficiente de Entropia	0,65222	0,27588	0,27588
Índice de Coeficiente de Partição	0.53975	0,8187	0,8187
Índice Xie-Beni	0.72997	5,66131	5,66131

Com relação aos resultados dos índices do FCM**,** pode-se dizer que teve bons resultados, mesmo não definindo os clusters gerados com exatidão, o RCP e URCP, chegou-se a conclusão de que os algoritmos foram capazes de extrair os grupos corretamente. Sendo que os coeficientes de partição e entropia apresentaram melhores resultados em relação à partição dos dados. O índice Xie-Beni no RCP e URCP, mesmo tendo um valor maior do que o encontrado pelo FCM indicou bons resultados, sendo que o valor deste índice deve ser o menor possível.

8. Considerações Finais

A aplicação dos algoritmos FCM, RCP e URCP na ferramenta *Shell Orion*, possibilitou observar que, dentre as três bacias hidrográficas da região carbonífera de Criciúma (Araranguá, Tubarão e Urussanga), a de Araranguá possui os piores valores referentes aos índices avaliados, pois possui os piores índices de pH, Ferro e Alumínio. Mesmo sendo a bacia com mais pontos de coleta avaliados, possui índices piores, percebendo-se também que quanto mais baixo o índice de pH, mais altos são os valores dos outros índices, percebendo-se a alta acidez da água em alguns pontos.

Como foi observado os resultados dos índices de validação, índices de Coeficiente de partição, índice de Coeficiente de entropia e índice de Xie-Beni, geraram resultados satisfatórios, indicando o correto particionamento dos dados, validando os resultados dos algoritmos.

Referências

CARLANTONIO, Lando. Mendonça. Novas Metodologias para Clusterização de Dados, Coordenação de Programas de Pós-Graduação em Engenharia, COPPE/UFRJ, 2001.

DÖRING, Christian; LESOT, Marie-Jeanne; KRUSE, Rudolf. Data Analysis with Fuzzy Clustering Methods. Computational Statistics & Data Analysis, Volume 51(1):192-214, 2006.

FAYYAD, Usama M.; PIATETSKY-SHAPIRO, Gregory; SMYTH, Padhraic. From data mining to knowledge discovery in databases. AI Magazine, Providence, v.17, n. 3, p. 37-54, autumn 1996.

FRIGUI, Hichem; KRISHNAPURAM, Raghu. A Robust Algorithm for Automatic Extraction of an Unknown Number of Clusters from Noisy Data. Pattern Recognition Letters 17. p. 1223-1232, 1996.

GOLDSCHMIDT, Ronaldo; PASSOS, Emmanuel. *Data Mining: um guia prático*. Rio de janeiro: Elsevier, 2005.

HAN, Jiawei; KAMBER, Micheline. *Data mining: concepts and techniques*. San Francisco:Morgan Kaufmann, 2001.

LAROSE, Daniel T. Discovering knowledge in data: an introduction to data mining. Hoboken: Wiley-Interscience, 2005.

KIM, Young-Il. et al. A cluster validation index for GK cluster analysis based on relative degree of sharing. Information Sciences, Vol. 168, p. 255-242, 2004.

OLIVEIRA JUNIOR, Hime Aguiar e; CALDEIRA, André Machado. Inteligência computacional aplicada à administração, economia e engenharia em Matlab. São Paulo: Thomson, 2007.

REZENDE, Solange Oliveira. *Sistemas Inteligentes: Fundamentos e Aplicação*. Editora Manole, 2002.

TAN, Pang-Ning; STEINBACH, Michael; KUMAR, Vipin; Introdução ao DATAMINING Mineração de dados. Rio de Janeiro: Editora Ciência Moderna Ltda., 2009.