Algoritmo de Gustafson-Kessel na Clusterização de Dados de uma Célula a Combustível de Hidrogênio

Ana Maria A. Bertone*(1) Jefferson Beethoven Martins **(2) Márcio José da Cunha ***(2)

(1) FAMAT, (2) FEELT, Universidade Federal de Uberlândia, Av. João Naves de Ávila, 2121
Uberlândia, MG 38.408-902, Brazil

Resumo O algoritmo de clusterização fuzzy de Gustafson-Kessel é aplicado a um banco de dados numérico, simulado a partir de uma célula real de combustível a hidrogênio. Este estudo constitui a etapa inicial de um projeto mais amplo, que envolve a identificação, também fuzzy, de sistemas dinâmicos não lineares. Nesta parte do estudo é feita a implementação do algoritmo de clusterização no software livre Python (Spyder). O objetivo é testar a eficiência do procedimento, através da interface de desenvolvimento, diante de um grande número de dados presentes no experimento. Como validação da escolha do número de clusters são utilizados simultaneamente os índices de partição e entropia.

Palavras-chave Fuzzy clustering; Gustafson-Kessel; Célula a combustível de Hidrogênio

1 Introdução

A clusterização é uma técnica matemática utilizada para agrupar dados segundo um critério, ou seja, similaridade matemática. Esta técnica é utilizada, principalmente, em mineração de dados (data mining). Os dados utilizados em um processo de clusterização podem pertencer a diferentes contextos como: histórico faturamento de uma empresa, números sobre a evolução de uma doença em determinado local ou, até mesmo, valores de pressão e temperatura utilizados em um equipamento industrial controlado. A diversidade das aplicações da técnica fuzzy de clustering pode ser vista em [1], em que Costa et al. caracterizam morfologicamente e estimam a diversidade genética de pimentas presentes no estado do Amazonas. O método utilizado é o hierárquico das médias das distâncias UPGMA (Unweighted Pair-Group Method Using an Arithmetic Average). Em [2] Rizzo et al., mostram um mapeamento digital de solos, utilizando o método

* anamaria@famat.ufu.br
** jefferson@iftm.edu.br
*** mjcunha@eletrica.ufu.br

fuzzy de C-Means [3]. Em [4], Alves et al., os cento e sessenta bairros da cidade de Rio de Janeiro são agrupados pelo algoritmo fuzzy C-means, para obter importantes informações para a gestão municipal. Finalmente, citando apenas trabalhos recentes feitos a nível nacional, [5] Veras et al. propõem um algoritmo que detecta os exsudatos (depósitos de gordura) através do tratamento de imagens. O estudo utiliza fuzzy clustering e técnicas de morfologia matemática.

Em 1979 Donald E. Gustafson e William C. Kessel publicaram o artigo [6] apresentando um algoritmo modificado do tradicional fuzzy C-Means [7]. Foi a partir dessa data que o algoritmo passou a ser conhecido como Gustafson-Kessel (GK). A principal diferença em relação ao fuzzy C-Means é a substituição da distância euclidiana, de forma que a matriz de covariância detecta as formas geométricas com mais precisão, independentes de cada cluster, isto a cada passo do algoritmo.

Em [8], Avelar apresenta o projeto de uma fonte alternativa de energia, com o funcionamento de uma célula a combustível de hidrogênio, através de um conversor elevador full-brigde e um inversor monofásico PWM senoidal. O modelo proposto em [8] foi implementado para uma célula de 1,2 KW e o processo de identificação da dinâmica de sistemas é o método dos mínimos quadrados ordinários. Uma das principais contribuições do trabalho de Avelar foi representar a dinâmica da temperatura interna da célula em função da corrente pois, outros simuladores, normalmente, não o fazem. Devido sua alta confiabilidade, o trabalho de [8] foi empregado para gerar os dados utilizados neste estudo. Por meio deste método de clusterização fuzzy, um grande número de dados numéricos gerados são agrupados [3]. Como ponto de partida, é desenvolvido o algoritmo de GK na plataforma Anaconda (Continuum Analytics) que usa a linguagem Python, a qual permite o uso de inúmeras bibliotecas por meio da importação de rotinas. Esta linguagem de desenvolvimento é livre, de fácil obtenção e com grande suporte de fóruns e sites da área.

O Spyder (Python) é um ambiente que facilita a programação devido a sua interface e quantidade de bibliotecas. De alta inteligibilidade, ele possui robusto debugger que torna mais rápido o trabalho do programador, testando, em tempo de execução, de forma muito parecida com o Matlab. Junto ao Anaconda Python, o Spyder se torna muito útil para programação científica. O cálculo de matrizes transpostas, multiplicação entre matrizes e geração de figuras são apenas algumas das facilidades apresentadas por essa linguagem de programação e ambiente de desenvolvimento integrados. O Spyder é uma IDE compatível com diferentes plataformas como MS Windows e Linux, possuindo simples instalação e grande número de informações a seu respeito (suporte amplo e facilitado).

Um importante aspecto a ser ressaltado é a escolha do número de clusters para o uso do algoritmo GK. Neste trabalho foram considerados simultaneamente os índices de partição e de entropia, que calculam a nebulosidade da clusterização.

Este trabalho é organizado em cinco seções. A Seção 1 é a introdução a este estudo. A seção 2 explica a origem dos dados utilizados neste estudo (particularidades da célula a combustível de hidrogênio). Na seção 3 é explicado o

conceito de clusterização fuzzy e suas propriedades. A seção 4 mostra os passos do algoritmo de Gustafson-Kessel no ambiente Python, as simulações realizadas, elucidando os resultados. A seção 5 são feitas as conclusões e são traçadas futuras ideias a ser desenvolvidas, ligadas a este estudo.

2 Simulador de células a combustível de hidrogênio

Existem várias fontes de energia, todavia as alternativas são buscadas devido ao impacto que as tradicionais (carvão e petróleo) geram no meio ambiente. Uma das possibilidades em estudo a nível mundial é o hidrogênio; já existem aplicações em funcionamento, entretanto os principais problemas de implantação se relacionam ao seu caráter explosivo e alto custo de pesquisas. O funcionamento de uma célula a combustível de hidrogênio com um conversor elevador "full-brigde" e um inversor monofásico PWM senoidal é mostrado no trabalho de Avelar [8] que, além de criar um software, simula o sistema real com fidelidade. O modelo proposto foi desenvolvido para uma célula de 1,2 KW e o processo de identificação de sistemas utilizado é dos mínimos quadrado ordinários.

O supracitado trabalho [8] desenvolveu um modelo para simular a célula a combustível de hidrogênio ligada à rede elétrica, levando em consideração o efeito da temperatura gerado internamente. O software gerado por esta pesquisa foi utilizado no nosso trabalhado, simulando a planta real. Ao citar algumas das principais características do software simulador, observa-se que a célula é capaz de gerar tensão entre 20 V e 50 V (contínua) e a tensão varia com a corrente drenada em seus terminais, sendo elevada a 380V antes de ser aplicada no módulo inversor. Todos os dados coletados são armazenados em um microcomputador. Salienta-se que as características da planta são diferentes para subida e descida de temperatura, pois o resfriamento é mais lento que o aquecimento. A célula a combustível de membrana polimérica (PEMFC Nexa, fabricada pela Ballard) foi testada em laboratório para levantamento estático e dinâmico, sendo toda a estrutura da célula a combustível, simulada através do software PSIM. Um esquema do processo aplicado por Avelar [8] é mostrado na Figura 1

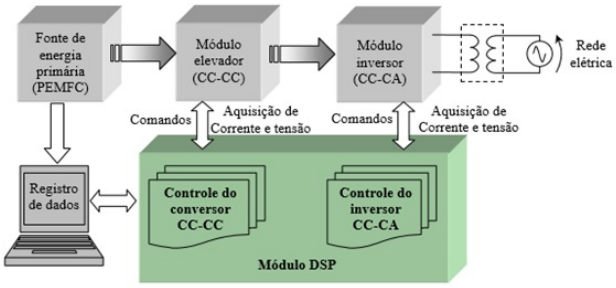


Figura 1. Fonte alternativa de energia baseada em célula a combustível (hidrogênio) [8].

Destaca-se que o modelo encontrado representa, de forma excelente, a dinâmica da planta (corrente-temperatura). Uma foto da montagem do laboratório[8] onde é feito o recolhimento dos dados é mostrada na Figura 2.

Figura 2. Laboratório para levantamento do comportamento estático e dinâmico da célula a combustível [8].

Os dados empregados na nossa proposta são gerados através do simulador descrito [8]. A estrutura de dados é do tipo SISO (single input - single output em inglês), ou seja, apenas uma entrada e uma saída. São 38.300 dados de entrada (corrente) com seus 38.300 dados de saída (temperatura) correspondentes. Os dados de entrada são gerados com o intuito de percorrer o maior espectro de frequência possível, sendo a taxa de amostragem de 0,2 segundos, os dados de saída são gerados através do modelo de Avelar[8].

3 Clusterização Fuzzy

Clusterização é a classificação não-supervisionada de dados, formando agrupamentos ou clusters. Esta metodologia representa uma das principais etapas de processos de análise de dados. A análise de clusters tem como objetivo a organização de padrões, representados matematicamente na forma de vetores ou pontos em um espaço multidimensional em clusters, de acordo com alguma medida de similaridade. A principal característica da classificação não-supervisionada, é agrupar um conjunto de padrões não-rotulados em clusters que possuam alguma propriedade em comum.

Distinguimos em um processo de clusterização as seguintes componentes:

1. representação dos valores que envolvem as definições do número, tipo e modo de apresentação das características de cada padrão;

2. definição de uma medida de similaridade, que é uma função de distância definida entre pares de dados ou padrões.
3. clusterização de acordo com a medida escolhida; se a função é determinística ou binária, ou seja, um padrão pertence ou não-pertence a um dado grupo, sendo uma clusterização crisp ou hard. Caso a medida seja uma função de pertinência fuzzy, então um padrão pode apresentar graus de pertinência em relação aos grupos.
4. validação do resultado que geralmente se recorre a critérios de otimalidade, muitas vezes definidos de forma subjetiva [10].

O método de clusterização fuzzy clustering permite que os elementos do banco de dados pertençam a vários grupos simultaneamente com diferentes graus de pertinências. Assim, o conjunto de dados é particionado em um número c de conjuntos fuzzy, onde c é o número total de clusters.

As principais características da clusterização fuzzy são:

- Cada elemento deve ter um conjunto de pertinências que expressa o "quanto ele pertence" a cada cluster.
- O somatório de todas as pertinências de um elemento é igual a um.

Concretamente, a formulação do fuzzy clustering a partir de um banco de dados de uma única entrada com n elementos é

$$z = (z_1, z_2, \ldots, z_n) \in \mathbb{R}^n,$$

sendo necessário encontrar uma matriz de pertinência fuzzy

$$U = \begin{bmatrix} \mu_{11} \cdots \mu_{1k} \cdots \mu_{1n} \\ \mu_{21} \cdots \mu_{1k} \cdots \mu_{2n} \\ \vdots \quad \vdots \quad \ddots \quad \vdots \\ \mu_{1c} \cdots \mu_{1k} \cdots \mu_{cn} \end{bmatrix}$$

e os centros de cada cluster

$$V = \{v_1, v_2, ..., v_c\}, \ v_i \in \mathbb{R},$$

usando um método de otimização.

O algoritmo de GK estende a clássica clusterização fuzzy C-Means utilizando uma atualização da distância, com o objetivo de detetar clusters com diferentes estruturas geométricas [6]. Cada cluster possui sua própria geometria induzida por uma matriz M^i, determinando a distância ao centro do cluster como sendo

$$d^2_{i,j,M_i} = (z_j - v_i)M^i(z_j - v_i)^T, \ i = 1, \ldots, c, \ j = 1, \ldots, n. \tag{1}$$

Sendo $M = (M^1, \ldots, M^c)$ a $c-$upla de matrizes correspondente a cada cluster, o funcional objetivo do algoritmo de GK [6] é dado por

$$J(X, U, V, M) = \sum_{i=1}^{c} \sum_{j=1}^{n} \mu^m_{i,j} d^2(z_j, v_i),$$

sujeito às restrições

$$\begin{aligned}
&0 \leq \mu_{i,j} \leq 1, i = 1, \ldots, c,\ j = 1, \ldots, n \text{ (restrição do grau de pertinência);} \\
&0 < \sum_{j=1}^{n} \mu_{i,j} < 1,\ i = 1, \ldots, c \text{ (restrição do cluster não vazio);} \\
&\sum_{i=1}^{c} \mu_{i,j} = 1,\ j = 1, ..., n \text{ (restrição do grau de pertinência total),}
\end{aligned} \tag{2}$$

em que m é conhecido como o parâmetro de fuzzificação e, em geral, é tomado igual a 2 [3], como é o caso do presente estudo. Além disso, cada matriz M_i tem como restrição

$$\det(M_i) = \text{determinate de } A_i < \rho_i,\ \rho_i > 0,\ \text{para todo } i.$$

Assim, M_i é obtida usando o método dos multiplicadores de Lagrange dado pela seguinte equação

$$M_i = \rho_i \det(F_i)^{\frac{1}{N}} F_i^{-1}, \tag{3}$$

em que N é a dimensão do espaço do cluster e F_i é a matriz de covariância do i-ésimo cluster, definida por

$$F_i = \frac{\sum_{j=1}^{N} \mu_{ij}^m (z_j - v_i)(z_j - v_i)^T}{\sum_{j=1}^{N} \mu_{ij}^m}. \tag{4}$$

Finalmente, os centros dos clusters são calculados através da forma [3]

$$v_i = \frac{\sum_{j=1}^{N} \mu_{ij}^m z_j}{\sum_{j=1}^{N} \mu_{ij}^m}. \tag{5}$$

A validação da escolha do número de clusters é equivalente a encontrar o número ideal de clusters a partir de determinado conjunto de dados. Vários índices de validade têm sido estudados para determinar o número ótimo de clusters. Bezdek [7] propõe dois índices:

1. Coeficiente de Partição (CP) que mede o número de superposições entre clusters e definido por

$$CP(c) = \frac{1}{N} \sum_{i=c}^{c} \sum_{j=1}^{n} \mu_{ij}^2.$$

Temos que $\frac{1}{c} \leq CP(c) \leq 1$ e o valor ótimo c_0 encontrado em um intervalo de escolha $I = [c_{min}, c_{max}] \cap \mathbb{Z}$, $\mathbb{Z}$ conjunto dos número inteiros positivos, se calcula como

$$c_0 = \max_{c \in I} CP(c).$$

2. Índice de Classificação de Entropia (CE) que mede a "nebulosidade" (fuzzyness em inglês) do cluster pela fórmula

$$CE(c) = -\frac{1}{N} \sum_{i=c}^{c} \sum_{j=1}^{n} \mu_{ij} \log(\mu_{ij}).$$

O valor ótimo c_0 é encontrado em um conjunto de escolha $I = [c_{min}, c_{max}]$ como sendo

$$c_P = \max_{c \in I} PC(c).$$

O número de clusters, c, é muito importante devido aos efeitos, por exemplo, da identificação fuzzy dos sistemas não-lineares. No nosso estudo é escolhido o número ótimo combinando os índices CP e CE simultaneamente em dois casos: para valores de c no de escolha $I = [2, 6] \cap \mathbb{Z}$, para mostrar em forma clara, no gráfico obtido,a estrutura geométrica dos clusters. O outro valor foi escolhido em $I = [20, 25] \cap \mathbb{Z}$, com o objetivo de ser utilizado na identificação do sistema dinâmico que relaciona os dados da célula de oxigênio.

4 O algoritmo de Gustafson-Kessel na interface Python Spyder

Descrevemos nesta seção o passo-a-passo do algoritmo de GK, mostrando em cada passo os resultados da programação no Python para o caso de $c = 2$.

Passo 1: Toma-se como valor de fuzzificação $m = 2$. Seja Z o conjunto de dados e $\{c_{ini}\ i = 1, \ldots, c\} \subset Z$. A forma de escolher os centros não foi aleatória, com o objetivo de comparar a perfomance do algoritmo em termos dos índices PC e CE. Assim, foram escolhidos dois elementos dos dados de forma "espalhada", como mostra a tabela da Figura 3

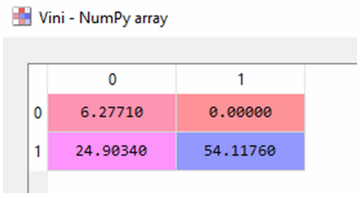
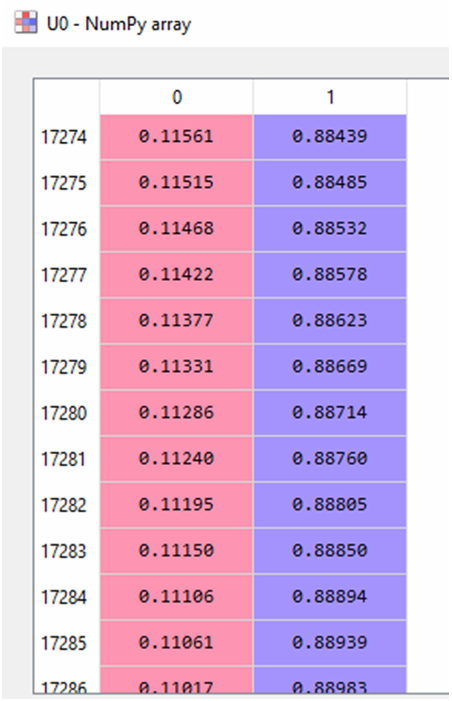


Figura 3. Coordenadas dos centros de cluster escolhidos: a primeira coluna são as abscissas.

Passo 2. Calcula-se a matriz de partição $U0 = (\mu_{ij})$, que contém as pertinências de cada dado ao clusters. Parte da matriz é mostrada na tabela da Figura 4



	0	1
17274	0.11561	0.88439
17275	0.11515	0.88485
17276	0.11468	0.88532
17277	0.11422	0.88578
17278	0.11377	0.88623
17279	0.11331	0.88669
17280	0.11286	0.88714
17281	0.11240	0.88760
17282	0.11195	0.88805
17283	0.11150	0.88850
17284	0.11106	0.88894
17285	0.11061	0.88939
17286	0.11017	0.88983

Figura 4. Matriz de partição $U0$.

Passo 3. Calcula-se a matriz de covariança Cov_i de cada cluster, dado pela fórmula (4). As primeiras matrizes de covariança obtidas pelo algoritmo é mostrada na Figura 5.

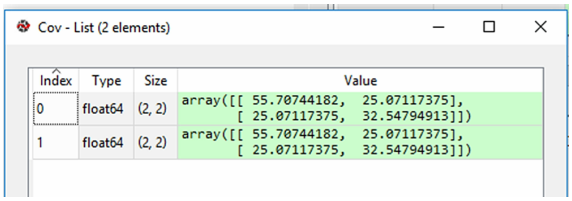


Figura 5. As duas matrizes de covariância de cada cluster. Cada "array" corresponde à matriz de um cluster, 0 ou 1, ou seja, primeiro e segundo cluster da liguagem Python.

Passo 4. Calcula-se o centro, v_i, de cada cluster, através do cálculo da fórmula (5).

Obtém-se dois primeiros centros de cluster cujas coordenadas são mostradas na tabela da Figura 6

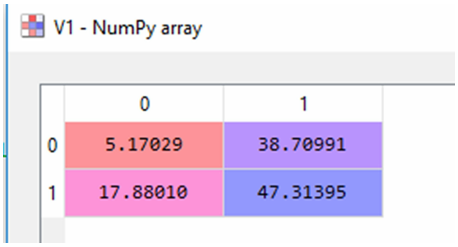


Figura 6. Coordenadas dos novos centros de cluster: a primeira coluna são as abscissas.

Passo 5. Calcula-se a matriz M_i que induz a distância e a geometria de cada cluster, usando a fórmula (3). No primeiro loop do algoritmo obtém-se a matriz M_i mostrado na Figura 7.

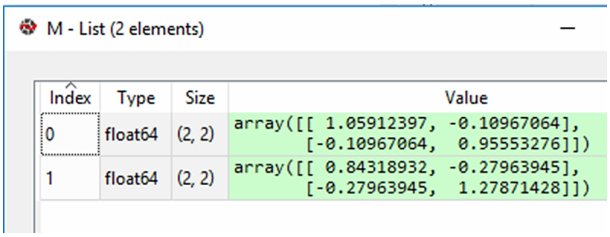


Figura 7. As matrizes M_i que induzem a norma de cada cluster. Cada "array" corresponde a um cluster, 0 ou 1.

Passo 6. Calcula-se a distância de cada centro de cluster em relação a cada elemento, utilizando a distância induzida pela matriz M_i, obtendo a matriz das distâncias, $d1$, algumas de cujas entradas são ilustradas na tabela da Figura 8

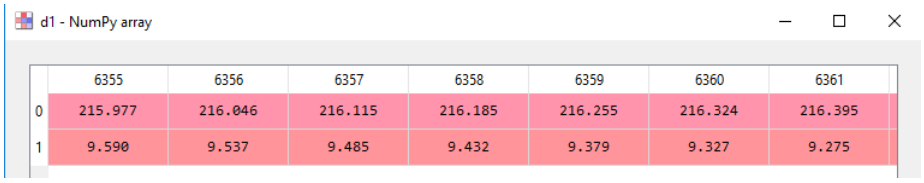


Figura 8. Algumas das entradas da nova matriz das distâncias $d1$.

Passo 7. Actualiza-se a matriz de pertinência $U0$ pela matriz U, calculando

$$\mu_{ij}^2 = \frac{1}{\sum_{k=1}^{c} \left(\frac{\left(d_{ij}^2\right)^2}{\left(d_{kj}^2\right)^2} \right)^{\frac{1}{m-1}}}, \; i = 1, \ldots, c, \; j = 1 \ldots n.$$

Passo 8. Calcula-se o erro e o primeiro ciclo é terminado. No nosso estudo calculamos o erro por meio da norma do máximo, ou seja,

$$\text{erro} = \max \left\| \mu_{ij}^1 - \mu_{ij}^2 \right\|,$$

em que μ_{ij}^k representam as entradas das matrizes de partição em voltas de loop de ordem k e seguinte. Neste passo 8 do primeiro loop do algoritmo, o erro obtido é de 0,639188829.

Passo 9. Caso o erro seja maior que o pré-estabelecido, se deve voltar ao Passo 2. Neste caso a meta pre-estabelecida, como teste, é do erro ser menor que 0,05.

Uma vez terminado o algoritmo, foram calculados os índices de validação CP e CE, que tornaram os valores descritos na Tabela 1.

	2	3	4	5	6
CP	0.8078	0.7433	0.6822	0.6380	0.6385
CE	0.3201	0.4659	0.6057	0.7314	0.7447

	20	21	22	23	24	25
CP	0.6520	0.7226	0.66331	0.6798	0.7407	0.6721
CE	0.8264	0.6715	0.8194	0.7877	0.6225	0.8124

Tabela 1. Tabela de valores dos índices CP e CE retornados após a finalização de cada algoritmo com erro de saída 0,01 e mesmo centros inicias de clusters da tabela da Figura 3. Em destaques os maiores valores de CP e os menores valores de CE simultâneos.

Com a finalidade de construir o gráfico das duas clusterizações escolhidas, são calculados os autovalores e autovetores das matrizes de covariança, que informam sobre a estrutura geométrica do cluster. De fato, o eixo principal da elipse de nível 1 do cluster, tem a direção do autovetor da matriz de covariança do cluster, correspondente ao maior autovalor, sendo a raíz quadrada desse autovalor a medida do semieixo maior. A direção perpendicular é dado pelo outro autovetor e a medida do eixo menor é a raíz quadrada do autovalor correspondente.

Na Figura 9 é mostrado a clusterização de dois clusters e na Figura 10 a de 24 clusters.

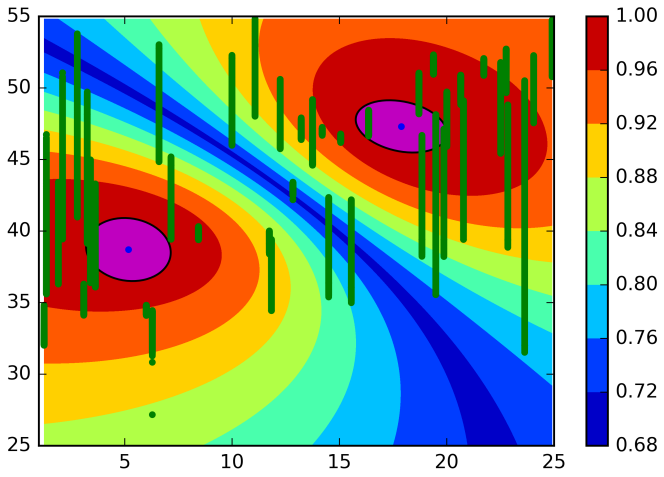


Figura 9. Partição fuzzy dos dados da célula de oxigênio. Os pontos em cor verde representam os pares de atributos (entrada, saída). As elipses preenchidas de cor magenta representam a geometria do cluster de, exatamente, nível 1. As outras regiões representam níveis de pertinência menor a 1, como indica a escala ao lado.

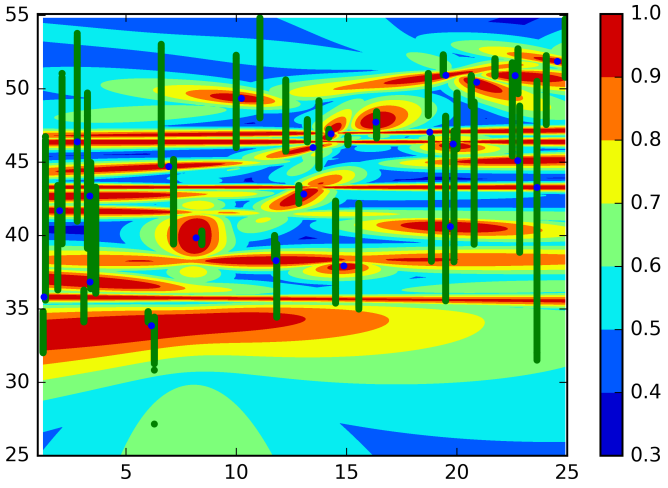


Figura 10. A partição fuzzy dos dados da célula de oxigênio, em cor verde, por 24 clusters com as regiões de nível coloridas de acordo com a escala ao lado.

5 Conclusões

O algoritmo de clusterização de Gustafson e Kessel funciona com sucesso para o grupo de dados utilizados na presente pesquisa. Isto motiva essa primeira etapa de um projeto maior, cujo objetivo é construir, utilizando um software livre, um aplicativo de propósito geral. A complexidade desse aplicativo iria abranger identificação de sistemas não lineares do tipo single input - single output (só uma saída e uma entrada), até multiple input - multiple output, ou seja, múltiplas entradas e múltiplas saídas. Para testar este algoritmo Gustafsson Kessel são utilizados, neste estudo, apenas uma entrada, com 38300 valores, e seus correspondentes dados de saída, obtidos em laboratório em períodos amostrais de 0,2 segundos.

Os desafios são grandes, mas possibilidades também são. Clusterizar, identificar e validar modelos através de métodos caixa-preta, como identificação de sistemas através de lógica fuzzy, é um promissor campo de pesquisa, sendo este o foco dos autores do presente artigo.

Referências

1. Costa, L.V., Bentes, J.L.S., Lopes, M.T.G., Alves, S.R.M., Viana Júnior, J.M.: Caracterização de acessos de pimentas do Amazonas. Horticultura Brasileira, 33, 290 – 298 (2015)
2. Rizzo, R., Demattê, J.A.M., Lacerda, M.P.C.: Espectros VIS-NIR do Solo e Fuzzy K-Médias Aplicados na Delimitação de Unidades de Mapeamento de Solos em Topossequências. R. Bras. Ci. Solo, 39, 1533 – 1543 (2015)
3. Abonyi, J., Babuska, R., Szeifert, F.: Modified Gath-Geva fuzzy clustering for identification of Takagi-Sugeno fuzzy models. IEEE Trans. on Systems, Man, and Cybernetics, 32(5), 612 – 621 (2002)
4. Alves, B.D.S., Barbosa, M.T.S., Caffarena, E.R., da Silva, A.S.: Caracterização do envelhecimento populacional no município do Rio de Janeiro: contribuições para políticas públicas sustentáveis. Saúde Colet, Rio de Janeiro, 24(1), 63 – 69 (2016)
5. Veras, R.M.S., Medeiros, F.N.S., Araújo, F.H.D., Santana, A.M., Silva, R.R.V.: Detecção de exsudatos em imagens de retina por técnicas de morfologia matemática e agrupamento nebuloso. Rev. Bras. Eng. Biom., 29(1), 45 – 56 (2013)
6. Gustafson, D.E., Kessel, W.C.: Fuzzy clustering with fuzzy covariance matrix. Proceedings of the IEEE Control and Decision Conference, San Diego, .761 – 766 (1979)
7. Bezdek, J., Keller, J., Krisnapuram, R., Pal, N.: Fuzzy models and algorithms for pattern recognition and image processing, Springer (2005).
8. Avelar, H.J.: Estudo e desenvolvimento de um sistema de energia baseado em célula a combustível para injeção de potência na rede elétrica.: Tese de Doutorado Faculdade de Engenharia Elétrica, Universidade Federal de Uberlândia (2012).
9. Eclipse Foundation
   http://www.eclipse.org Último acesso Maio (2016)
10. Jain, A.K., Dubes, R.C.: Algorithms for Clustering Data, Prentice Hall, (1988)