Proceeding Series of the Brazilian Society of Computational and Applied Mathematics, v. 8, n. 1, 2021.

Trabalho apresentado no XL CNMAC, Evento Virtual - Co-organizado pela Universidade do Mato Grosso do Sul (UFMS).

Proceeding Series of the Brazilian Society of Computational and Applied Mathematics

AdditiveClusterKDE: Um algoritmo para clusterização baseado no estimador de densidade kernel multivariado

Dirceu Scaldelai[1]
Depto. de Matemática/Unespar, Campo Mourão, PR
Luiz Carlos Matioli[2]
Depto. de Matemática/UFPR, Curitiba, PR
Solange Regina dos Santos[3]
Depto. de Matemática/Unespar, Campo Mourão, PR

Resumo. Nesse artigo apresentamos um algoritmo para clusterização baseado na estimativa de densidade kernel multivariada. O algoritmo denominado AdditiveClusterKDE é centrado na múltipla otimização da função do Estimador de Densidade Kernel com kernel Gaussiano multivariado. O algoritmo proposto tem a vantagem de não exigir a priori o número de clusters, além disso, ele é simples, fácil de implementar, bem definido e para em um número finito de etapas, ou seja, sempre converge independentemente do conjunto de dados a ser agrupado. Implementamos o algoritmo no *software R* e fizemos experimentos numéricos com dados de problemas práticos obtidos na literatura e dados produzidos de forma sintética para uma melhor abrangência das potencialidades do algoritmo. Resultados preliminares indicam que o algoritmo AdditiveClusterKDE é competitivo quando comparado com os algoritmos K-means e PdfCluster.

Palavras-chave. Estimador de Densidade Kernel, Clusterização, Otimização.

1 Introdução

Com o crescimento e o uso massivo de tecnologias, a quantidade de dados gerados e disponibilizados tem crescido exponencialmente. No entanto, para que essa grande quantidade de dados forneça informações que possam ser transformadas em conhecimentos úteis e relevantes, é preciso a realização de análises essenciais para correta interpretação desses dados. O processo de explorar grandes quantidades de dados à procura de padrões, regras ou sequências de informações, para detectar correlações entre as variáveis faz parte do campo da ciência denominada mineração de dados, que pode ser subdividida em 5 principais linhas de pesquisa, a saber: a regressão, a análise sequencial, a classificação, a clusterização e a análise de *outliers*.

Essa pesquisa está focada na clusterização, que de acordo com [5], é um dos métodos de mineração de dados que visa identificar padrões ou grupos de objetos semelhantes em um conjunto de dados de interesse. A clusterização pode ser classificada em métodos particionados, hierárquicos, baseados em densidade, em grade e em modelos, sendo utilizada nos mais variados campos da pesquisa científica, de forma que seus resultados são fundamentais para prever, analisar e replicar fenômenos, além de fomentar novas pesquisas.

Sendo assim, propomos nesse trabalho o algoritmo AdditiveClusterKDE, utilizado para agrupar dados n-dimensionais, baseado na otimização da função do estimador de densidade kernel, do inglês

[1]dirceuscaldelai@gmail.com
[2]lcmatioli@gmail.com
[3]solaregina@gmail.com

Kernel Density Estimation (KDE). A denominação do algoritmo é oriunda da sua característica de adicionar um novo cluster a cada nova iteração executada. A ideia central do algoritmo é trabalhar com a otimização da função KDE sujeita a restrição de caixa, determinada pelos limitantes das observações pertencentes ao cluster com maior heterogeneidade das observações a ele designado. O AdditiveClusterKDE não necessita da especificação a priori de um número de clusters, sendo esse determinado pela construção da sequência $Nc = \{n + 1\}$, com $n = 1, \cdots, k$, onde k é o número de iterações. Para cada elemento da sequência de clusters, o algoritmo determina o coeficiente de silhueta da clusterização, interrompendo seu processo quando um número pré-determinado de iterações não apresenta melhora na solução. O coeficiente de silhueta foi proposto por [9] e, segundo [6], é um dos índices para avaliar a estrutura de clusterização.

Na sequência, apresentamos uma breve introdução sobre o estimador de densidade kernel em seguida, o algoritmo AdditiveClusterKDE e, também, os resultados numéricos que evidenciam o desempenho do algoritmo proposto. Para finalizar o artigo, as considerações finais com os principais apontamentos acerca da pesquisa.

2 Estimador de Densidade Kernel

Seja $x \in \mathbb{R}^n$ uma variável aleatória e f uma função de densidade de probabilidade (f.d.p.) associada a tal variável. O conhecimento de f fornece uma descrição natural do comportamento da variável x e permite que as características associadas a ela sejam estudadas e replicadas. No entanto, nem sempre a função de densidade de um dado conjunto de observações é conhecida e, em geral, no caso de problemas provenientes de fenômenos reais, isso faz com que não seja simples identificar uma f.d.p. capaz de explicar suas características. Diante disso, estimadores não-paramétricos são utilizados para contornar tal dificuldade. Segundo [10], o foco da estimação não-paramétrica é diferente daquele da estimação paramétrica. Nos estimadores paramétricos a ênfase está em obter o melhor estimador $\widehat{\theta}$ para um dado parâmetro θ, enquanto no caso não-paramétrico, o objetivo está diretamente ligado à obtenção de uma boa estimativa da função f.

A estimativa de densidade kernel é uma técnica não-paramétrica para suavização de dados com base em amostras finitas, o qual fornece uma maneira simples de encontrar estruturas em conjuntos de dados sem a imposição de um modelo paramétrico. Segundo [12] o KDE é uma das técnicas de suavização de dados mais importantes e amplamente usadas. Por suavização de dados, [4] versa que esta consiste em encontrar soluções aproximadas de uma função que captura padrões importantes dos dados, considerando cada um dos elementos, desprezando ruídos, removendo variações aleatórias e mostrando componentes de tendência do conjunto de dados sendo a suavização um conceito fundamental na análise de dados.

De acordo com [4, 12], uma forma geral do KDE multivariado é dada por

$$\widehat{f}(x, H) \quad = \quad m^{-1} \sum_{i=1}^{m} |H|^{-\frac{1}{2}} K(H^{-\frac{1}{2}}(x - X_i)) \tag{1}$$

em que H é a matriz largura de banda ou matriz de suavização, de ordem n, não-aleatória, simétrica e definida positiva; $|H|$ é o determinante de H; $x = (x_1, x_2, \cdots, x_n)^T \in \mathbb{R}^n$ é o vetor das variáveis do espaço n-dimensional; $X_i = (X_{i1}, X_{i2}, \cdots, X_{in})^T$, com $i = 1, 2, \cdots, m$, o conjunto das observações com função de densidade desconhecida; K é a função Kernel, tal que a variável $x \in \mathbb{R}^n$, satisfaz $\int_{-\infty}^{\infty} K(x)dx = 1$. Segundo [11], K é normalmente escolhida dentre as f.d.p. que tem como características o fato de serem contínuas, suave, unimodal e radialmente simétrica. [4, 10, 11] relatam que as funções normal padrão, Epanechnikov, Biweight, Triangular, Uniforme e Triweight são utilizadas na estimativa de Densidade Kernel, sendo a densidade normal padrão multivariada a utilizada no algoritmo AdditiveClusterKDE.

DOI: 10.5540/03.2021.008.01.0350

Considerando a relação (1) e a f.d.p normal padrão multivariada, o Estimador de Densidade Kernel Gaussiano é dado por

$$\widehat{f}(x,H) = \frac{1}{m}(2\pi)^{\frac{-n}{2}}\,|H|^{-\frac{1}{2}} \sum_{i=1}^{m} \exp\left(-\frac{1}{2}(x-X_i)^T H^{-1}(x-X_i)\right). \tag{2}$$

A função dada em (2) depende apenas da variável $x \in \mathbb{R}^n$, uma vez que a matriz H estará fixada. Porém, escolher a matriz largura de banda ideal é um problema extremamente delicado, uma vez que, pequenas variações em seus elementos afetam significativamente a forma e a orientação axial do KDE. Segundo [4, 12], a matriz largura de banda é definida em três níveis de complexidade. O caso mais simples é quando um escalar $h^2 \in \mathbb{R}_+^*$ multiplica a matriz identidade de ordem n, isto é, $H \in S$, onde $S = \{h^2 I_n : h > 0\}$. O segundo nível de complexidade da matriz largura de banda se dá por H ser uma matriz diagonal definida positiva, isto é, $H \in D$, com $D = diag(h_1^2, h_2^2, \cdots, h_n^2)$. Por fim, o terceiro nível e o mais complexo, tem H como uma matriz simétrica e definida positiva. Nesta pesquisa, optamos por considerar a matriz H como sendo a definida pelo segundo nível de complexidade,ou seja, uma matriz de suavização diagonal. Para uma análise mais detalhada ver [4, 6, 7, 10–12].

De acordo com [10, 12], sobre a hipótese de que a função densidade probabilidade é uma f.d.p. normal e KDE possui kernel Gaussiano, a escolha ótima das componentes da matriz diagonal H é dada por

$$h_i = \left(\frac{4}{n+2}\right)^{\frac{1}{n+4}} \sigma_i m^{-\frac{1}{n+4}}, \tag{3}$$

em que n é a dimensão do espaço, σ_i o desvio padrão das componentes e m o número de observações.

Segundo [7] cada valor de h_i, da relação (3), deve ser multiplicado por um fator de encolhimento definido pelo escalar $\frac{3}{4}$, a fim de aliviar o excesso de suavização determinado pelo cálculo das larguras de banda sob a premissa de normalidade multivariada. Nesse trabalho será introduzido um escalar α, de forma semelhante a proposta de [6] para o caso univariado. Com isso, temos que

$$h_i = \alpha\left(\frac{4}{n+2}\right)^{\frac{1}{n+4}} \sigma_i m^{-\frac{1}{n+4}}. \tag{4}$$

A matriz H, cujas componentes são definidas em (4) e a função definida em (2), são as bases para o desenvolvimento do algoritmo AdditiveClusterKDE, o qual apresentaremos a seguir.

3 Algoritmo AdditiveClusterKDE

O AdditiveClusterKDE é um algoritmo construtivo em que, a cada iteração, novos clusters são obtidos por meio da partição do conjunto de dados. Dessa forma, inicialmente todo o conjunto de dados constitui um único cluster e, a cada iteração subsequente, um novo cluster é determinado por meio do particionamento do cluster que apresenta maior heterogeneidade (variância) entre os elementos da sua matriz de distâncias. Após essa partição, todos os elementos são realocados ao cluster considerando o critério do centroide mais próximo. O algoritmo para quando não há melhoria no valor do coeficiente de silhueta. Os passos do AdditiveClusterKDE são apresentados no algoritmo 1.

No algoritmo AddiveClusterKDE, primeiramente são inicializadas as variáveis, as constantes e a função KDE. O Conjunto $X \in \mathbb{R}^{m\times n}$ contém os dados do problema, sendo m o número de observações e n o número de variáveis; o parâmetro $\alpha \in \mathbb{R}$ é usado, na relação (4), para determinar a matriz diagonal H que por sua vez é utilizada para determinar a função KDE; $RepMax$ é um

Algoritmo 1: AdditiveClusterKDE

Entrada: $X \in \mathbb{R}^{m\times n}$,$\alpha \in \mathbb{R}_+^*$, $RepMax \in \mathbb{Z}_+^*$;

Determine: A matriz H, a função KDE ($\widehat{f}(x)$), os limitantes superior e inferior de X, um ponto inicial;

enquanto *O critério de parada não for atendido* **faça**

  Determine $x^* \in argmin\left\{-\widehat{f}(x) : l \leq x \leq u\right\}$;

  se x^* *já é um centroide* **então**

    Fim da busca por centroides

  senão

    Acrescente x^* ao conjunto dos centroides;

    Realize um novo agrupamento para o conjunto de centroides atual;

    Determine o coeficiente de Silhueta do agrupamento atual ;

    se *O agrupamento atual é o melhor* **então**

      armazene as informações e zere o contador de iterações improdutivas,

    senão

      incremente o contador de iterações improdutivas em uma unidade.

    fim

    Verifique qual grupo é o mais heterogêneo;

    Determine os limitantes inferior e superior desse grupo (restrição de caixa para função KDE);

    Determine um novo ponto inicial, pertencente ao grupo mais heterogêneo, para otimização da função KDE;

  fim

fim

retorna *Centroides e o agrupamento que proporcionaram a melhor métrica*

número inteiro positivo fornecido pelo usuário, o qual define o número máximo de iterações que o algoritmo irá realizar após a não ocorrência de melhorá no valor do coeficiente de silhueta.

O algoritmo AdditiveClusterKDE executa um processo iterativo finito, onde a cada iteração este resolve um problema de otimização da função do estimador de densidade kernel $\hat{f}$ sujeita a restrição de caixa com o objetivo de determinar um novo centroide da clusterização. Optamos por utilizar para o processo de otimização da função KDE o método L-BFGS-B proposto por [3], implementado via *software R*. A escolha pelo L-BFGS-B se dá pelo fato desse ser um método quase-Newton com memória limitada e com restrição de caixa, o que assegura uma boa velocidade de convergência e se adapta de forma natural a metodologia empregada pelo algoritmo AdditiveClusterKDE.

Uma vez determinado o minimizador de $-\hat{f}$, o algoritmo avalia se este já foi definido em iterações anteriores como centroide da clusterização, visto que a repetição do ponto de mínimo da função KDE é um dos critérios de parada do AdditiveClusterKDE. Supondo que o minimizador da iteração atual seja distinto aos pontos definidos em iterações anteriores, logo ele é designado para compor o conjunto dos centroides e uma nova clusterização é realizada, pelo critério de mínima distancia euclidiana das observações aos centroides, com posterior avaliação da qualidade dessa clusterização por meio do coeficiente de silhueta.

Com base no coeficiente de silhueta, o algoritmo avalia se a clusterização proveniente da inserção do novo centroide produziu uma melhora da solução ou não. Em caso afirmativo, o algoritmo armazena as informações dos centroides e da clusterização e zera o contador de iterações improdutivas. Caso contrário, incrementa o contador em uma unidade e continua o processo. O contador de iterações improdutivas é o segundo critério de parada do algoritmo, isto é, caso este atinja o

DOI: 10.5540/03.2021.008.01.0350

valor de $RepMax$ iterações improdutivas o algoritmo é encerrado.

Na sequência, o AdditiveClusterKDE verifica qual dos clusters formados na iteração possui a maior distância média entre seus elementos, ou seja, o cluster com observações mais heterogêneas. Este cluster defini os novos limitantes e a nova restrição de caixa para a iteração subsequente. Para finalizar a iteração, o algoritmo seleciona um novo ponto inicial com a condição ser interno a nova restrição de caixa e apresentar a máxima distância ao centroide do cluster selecionado.

O processo é então repetido até que se atenda a um dos 2 critérios de de parada. Ao final o algoritmo retorna com a clusterização que propiciou o melhor coeficiente de silhueta. A Figura 1 ilustra o AdditiveClusterKDE para clusterização de um exemplo simples tendo uma configuração final de 3 clusters.

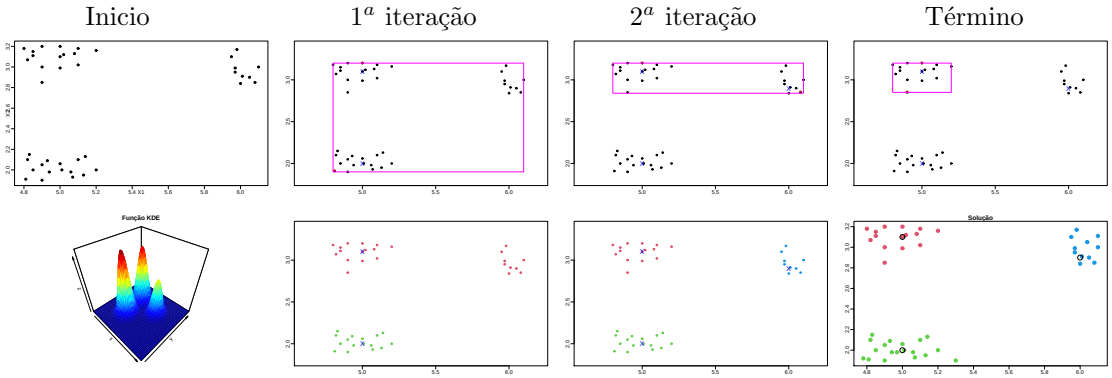

Figura 1: Ilustração do AdditiveClusterKDE.

4 Resultados numéricos

Nessa seção, apresentamos resultados que evidenciam e qualificam o algoritmo AdditiveClusterKDE, utilizando para isso dois grupos de problemas. O primeiro grupo é proveniente de trabalhos presentes na literatura, são eles: conjuntos de dados "USArrests" e "iris" disponibilizado no pacote *datasets*, [8]; "wine" e "Oliveoil" disponível no pacote PdfCluster, [2]; "TripAdvisor" disponível em [1]. O segundo grupo de problemas é composto por 4 conjuntos de dados, gerados randomicamente por múltiplas amostras normalmente distribuídas, sendo a quantidade de amostras e a dimensão de cada uma definida de forma aleatória.

A fim de verificar o desempenho do algoritmo AdditiveClusterKDE, o qual foi implementado no *sofware* R, cada um dos 8 problemas dessa seção foi submetido a 3 algoritmos, sendo eles o K-means, o PdfCluster e o AdditiveClusterKDE. Para a execução dos experimentos foi utilizado um notebook com processador Intel(R) Core(TM) i5-7200U CPU @ 2.50GHz 2.71GHz com sistema operacional Windows 10 home 64 bits. A versão do *software R* utilizado foi a 3.5.2 (2018-12-20) executado via Rstudio Version 1.1.463. Os resultados estão dispostos na Tabela 1.

A Tabela 1 apresenta o desempenho dos algoritmos para a resolução dos problemas propostos. Os critérios considerados para a comparação são: número de clusters, coeficiente de silhueta e tempo computacional. Com relação ao número de clusters, observamos que o PdfCluster apresentou pior resultado, uma vez que, não conseguiu determinar uma solução para dois problemas, principalmente para dimensões maiores. Os algoritmos AdditiveCluster e K-means divergiram apenas em um dos problemas analisados. Com relação ao coeficiente de silhueta, o AdditiveClusterKDE apresentou bons resultados, uma vez que, mesmo não determinando o melhor resultado

DOI: 10.5540/03.2021.008.01.0350

Tabela 1: Comparação dos algoritmos

problema	Dimensão	Algoritmo	n^o clusters	coef. Silhueta	tempo (s)
USArrests	50	AdditiveClusterKDE[1]	2	0,59	0,65
	×	PdfCluster	2	0,55	0,13
	4	K-means[2]	2	0,59	0,01
wine	178	AdditiveClusterKDE	3	0,60	4,03
	×	PdfCluster	10	-0,29	2,36
	13	K-means	2	0,65	0,007
Oliveoil	572	AdditiveClusterKDE	2	0,65	24,37
	×	PdfCluster	5	0,52	75,52
	8	K-means	2	0,54	0,034
TripAdvisor	980	AdditiveClusterKDE	2	0,44	13,25
	×	PdfCluster	2	0,39	314,4
	10	K-means	2	0,30	0,054
Multinormal 1	4902	AdditiveClusterKDE	3	0,77	38,57
	×	PdfCluster	7	0,46	50,79
	2	K-means	3	0,77	0,46
Multinormal 2	6543	AdditiveClusterKDE	9	0,71	151,2
	×	PdfCluster	-	-	-
	12	K-means	9	0,46	0,04
Multinormal 3	1884	AdditiveClusterKDE	3	0,84	29,51
	×	PdfCluster	-	-	-
	9	K-means	3	0,35	0,37
Multinormal 4	560	AdditiveClusterKDE	5	0,80	9,54
	×	PdfCluster	5	0,80	121,2
	15	K-means	5	0,45	0,05

[1]No algoritmo AdditiveClusterKDE foram utilizados os parâmetros $\alpha = 0,75$ e $RepMax = 2$. [2]No algoritmo K-means, executamos uma varredura para $k = 2, \ldots, 20$, selecionando k que produzisse o melhor coeficiente de silhueta.

para todos os problemas, sempre esteve muito próximo do melhor resultado, de modo que podemos considerar seus resultados consistentes.

Já com relação o tempo computacional, o algoritmo com melhor resultado foi com certeza o K-means. Porém esse fato deve ser observado com ressalvas, já que para sua aplicação é preciso conhecer a priori o número de clusters. Assim, levando em consideração apenas os algoritmos com características mais semelhantes, isto é, PdfCluster e AdditiveClusterKDE, é evidente a superioridade do algoritmo proposto nesse trabalho.

5 Considerações Finais

Nesse artigo, propomos um algoritmo empregado na clusterização de dados multivariados, o AdditiveClusterKDE. O algoritmo é baseado na múltipla otimização da função KDE multivariada com kernel Gaussiano sujeita a restrição de caixa.

Os resultados numéricos indicam que o AdditiveClusterKDE é eficiente e competitivo comparado aos algoritmos K-means e PdfCluster. A eficácia do AdditiveClusterKDE foi avaliada com base no coeficiente de silhueta que, com base nos problemas investigados, foi inferior em apenas

1 caso e, essa diferença no valor observado foi de 0,05. Com relação ao tempo computacional, o algoritmo proposto apresentou resultados competitivos, uma vez que superou o PdfCluster em todos os casos. Não cabe, para esse critério, compararmos algoritmo K-means aos algoritmos AdditiveClusterKDE e PdfCluster, já que sua metodologia exige a inserção do número de clusters a priori, o que não torna adequada essa comparação.

Para trabalhos futuros, almejamos minimizar as influências da escolha da matriz H no processo de obtenção dos centroides dos clusters. Outro ponto que merece destaque é a utilização de uma ferramenta alternativa para avaliação da qualidade da clusterização que não seja o coeficiente de silhueta. Por fim, almejamos ainda testar metodologias diferenciadas no processo de designação dos elementos aos clusters, que substitui adequadamente a mínima distância euclidiana.

Referências

[1] Asuncion, A. e Newman, D. UCI machine learning repository, *url: "http://archive.ics.uci.edu/ml", University of California, Irvine, School of Information and Computer Sciences*, 2007.

[2] Azzalini, A., Menardi, G., Clustering via Nonparametric Density Estimation: The R Package pdfCluster. *Journal of Statistical Software*, 57(11): 1-26, 2014 (to appear), DOI: 10.18637/jss.v057.i11.

[3] Byrd, R.H., Lu, P., Nocedal,J., Zhu, C. A limited memory algorithm for bound constrained optimization, *SIAM Journal on Scientific Computing*, 16(5):1190–1208, 1995. DOI: 10.1137/0916069.

[4] Gramacki, A. *Nonparametric kernel density estimation and its computational aspects*. Springer International Publishing, 2018, DOI: 10.1007/978-3-319-71688-6.

[5] Kassambara, A. *Practical Guide To Cluster Analysis in R:unsupervised machine learning, 1a. edição*. STHDA, 2017, ISBN:978-1542462709.

[6] Matioli, L. C., Santos, S. R., Kleina, M., Leite, E. A. A new algorithm for clustering based on kernel density estimation, *Journal of Applied Statistics*, 45(2): 347-366, 2017. DOI: 10.1080/02664763.2016.1277191.

[7] Menardi, G. e Azzalini, A. An advancement in clustering via nonparametric density estimation, *Statistics and Computing*, 24(5):753–767, 2014. DOI 10.1007/s11222-013-9400-x.

[8] R Core Team. R: A Language and Environment for Statistical Computing. *R Foundation for Statistical Computing*. Vienna, Austria, 2019. Disponível em https:www.R-project.org/

[9] Rousseeuw, P. J. Silhouettes: a graphical aid to the interpretation and validation of cluster analysis, *Journal of computational and applied mathematics*, 20: 53–65, 1987 (to appear), DOI: 10.1016/0377-0427(87)90125-7.

[10] Scott, D.W. *Multivariate density estimation: theory, practice, and visualization, 2a. edição*. John Wiley & Sons. Rice University Houston, Texas, 2015, ISBN:978-0-471-69755-8.

[11] Silverman, B.W. *Density Estimation for Statistics and data Analysis*, Chapman and Hall, London, New York, 1986, ISBN:0-412-24620-1.

[12] Wand, M.P., Jones, M.C. *Kernel smoothing, 1a. edição*. Chapman and Hall/CRC, New York, 1995, ISBN:978-0-412-55270-0.

