PEL208 — Relatório Atividade 4
Implementação do *k-Means*

Miller Horvath
Mestrando em Engenharia Elétrica (Processamento de Sinais e Imagens)
Centro Universitário FEI, São Bernardo do Campo, SP, Brasil

18 de novembro de 2018

1 Introdução

Este trabalho apresenta o relatório do desenvolvimento da quarta atividade avaliativa referente à disciplina PEL208, intitulada Tópicos Especiais em Aprendizagem, apresentada pelo Prof. Dr. Reinaldo Augusto da Costa Bianchi.

O objetivo desta atividade é implementar o algoritmo de clusterização *k-Means*, conforme abordado em sala de aula. Para isso, a linguagem de programação *C++* foi adotada.

2 Conceitos Fundamentais

2.1 k-Means

A origem do *k-Means* não é precisamente definida na literatura, pois o método foi descoberto individualmente em diferentes campos de pesquisa (JAIN, 2010). O termo *k-Means* foi criado por James MacQueen, em (MACQUEEN et al., 1967), a idealização do método, por Hugo Steinhaus, figura em (STEINHAUS, 1956) e a proposta do algorítimo foi concebida por Stuart Lloyd em 1957, apesar de só ter sido publicada em 1982 (LLOYD, 1982). Em 1965, Forgy (FORGY, 1965) publicou basicamente o mesmo método que Lloyd. Por isso, frequentemente o método k-Means é referenciado por Lloyd-Forgy.

O *k-Means* é um algoritmo iterativo que separa um conjunto de dados em k clusters. Cada cluster é definido pelo seu ponto médio (centroide), onde o *k-Means* visa minimizar os resíduos entre as observações e as centroides. Sendo assim, dado um conjunto inicial de centroides, o *k-Menas* é basicamente dividido em duas etapas (MACKAY; KAY, 2003):

- Atribuição — Determina-se à qual cluster cada observação pertence, calculando qual a centroide mais próxima através de alguma métrica de distância, como distância euclidiana, distância de Manhattan, distância de Mahalanobis, entre outras. Sendo que cada observação só pode pertencer à um único cluster.
- Atualização — Determina-se um novo conjunto de centroides, através do cálculo do vetor médio de todas as observações pertencentes ao mesmo cluster.

Estas etapas se repetem até que haja convergência das centroides, ou seja, até que as centroides não sejam alteradas após a etapa de atualização, ou que o método atinja um limite pré-estabelecido de iterações sem que o modelo convergir.

A Figura 1 apresenta o resultado da aplicação do *k-Means* num conjunto de dados proposto em aula. Este conjunto de dados possuí 17 observações e 2 *features*, sendo separada em 3 clusters (utilizando $k = 3$).

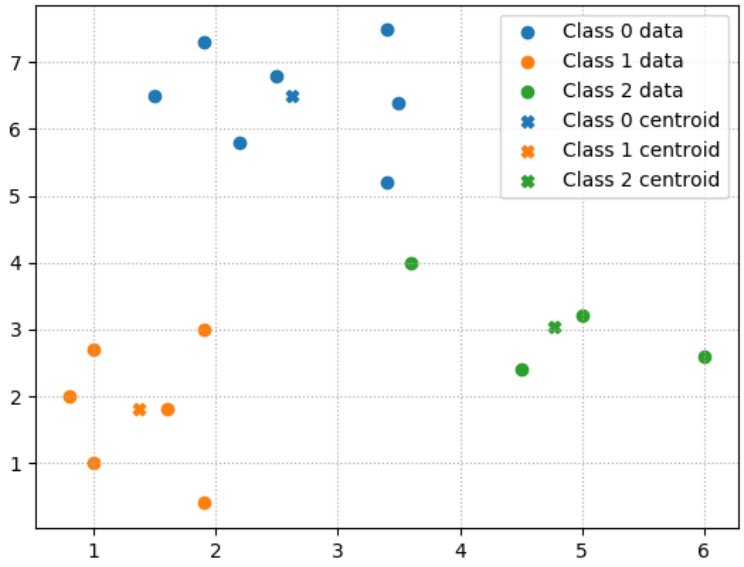

Figura 1 – Resultado da aplicação do *k-Means* no conjunto de dados proposto em aula utilizando $k = 3$.

3 Trabalhos Relacionados

O *k-Means* tem sido amplamente utilizado em algoritmos de otimização, onde métodos tradicionais apresentam uma alta complexidade computacional, inviabilizando a sua utilização em grandes massas de dados. Em (ZHONG et al., 2015), é proposto um algoritmo otimizado para o calculo da árvore de espalhamento mínimo de um conjunto grande de dados baseado no *k-Means*. Em (JIMÉNEZ et al., 2018), o *k-Means* é aplicado para

acelerar o processo de análise de vizinhança para o problema de seleção de características em dados multivariados. Em (DAS; DAS; DEY, 2018), é proposta uma variação do algoritmo de otimização por colônia de abelhas, chamado MBCO. Visando aumento de performance do modelo, os autores propõem um modelo híbrido que combina o MBCO com o k-Means.

Em (SOMMER; FOUSS; SAERENS, 2017), uma nova abordagem do k-Means baseada em *kernel* é avaliada na clusterização de nós em grafos, comparando-a com técnicas bem estabelecidas na literatura, como o método Louvain.

Em (HORVATH et al., 2018), o algoritmo k-Means foi utilizado para investigar o impacto da clusterização de usuários e de itens em modelos de sistemas de recomendação por filtragem colaborativa.

Em (ALASHRI et al., 2016), o *k-Means* é utilizado para classificar frases que tratam sobre mudanças climáticas. O objetivo deste trabalho é realizar 2 classificações: (1) identificar os textos em que ocorrem ou não enquadramento noticioso e (2) classificar as frases onde ocorre enquadramento noticioso em 4 categorias: solução, ameaça de problema, causa e motivação. O enquadramento noticioso é um conceito da teoria da comunicação onde o relato de acontecimentos é moldado para para defender um ponto de vista e enfraquecer o ponto de vista contrário, através do uso adequado de palavras, adjetivos, ideias e expressões.

4 Metodologia

A implementação da atividade foi desenvolvida na linguagem *C++*, utilizando o software Visual Studio 2017. Para apoiar os cálculos algébricos, foram utilizados os seguintes recursos da biblioteca *Eigen* (GUENNEBAUD; JACOB et al., 2010):

- Classe *MatrixXd*, sendo uma estrutura de dados para matrizes multidimensionais compostas por valores do tipo *double*;
- Classe *RowVectorXd*, sendo uma estrutura de dados para vetores multidimensionais compostos por valores do tipo *double*;
- Sobrecarga de operadores para soma, subtração; divisão e multiplicação de vetores e matrizes;
- Métodos de transposição de matrizes;

Foi desenvolvida uma classe chamada KMeans, que recebe a matriz de dados X, um valor inteiro k e um valor inteiro max_i como parâmetros de sua função construtora para que o k-Means seja aplicado em X, clusterizando os dados em k cluster, aplicando no máximo max_i iterações do k-Means. Após o cálculo do k-Means, a classe armazena os seguintes atributos privados:

- As centroides dos clusters resultantes;
- A quantidade de clusters k;

Além disso, a classe também oferece as seguintes funções:

- As funções *getters* para acessar os atributos privados;
- Função *classifyVector* — recebe como parâmetro um vetor de dados X e retorna à qual cluster esse vetor pertence;
- Função *classifyVector* — recebe como parâmetro uma matriz de dados X e retorna uma lista que representa à qual cluster cada vetor pertencente a X pertence;

5 Experimentos

Para testar e analisar a implementação desenvolvida, foram utilizadas três bases de dados de classificação, pertencentes ao repositório de aprendizado de máquina da University of California, Irvine (UCI) (DHEERU; TANISKIDOU, 2017), que serão descritas na sequência. Foi utilizado o k-Means, descrito na Seção 2.1, para encontrar as centroides dos clusters, sendo que o número de clusters utilizado para calcular o k-Means é equivalente ao número de classes contidas em cada base de dados consideradas. Devido à sua inicialização aleatória, o algoritmo k-Means foi aplicado dez mil vezes para cada base de dados. Cada uma das aplicações do k-Means foi avaliada levando em consideração o critério de otimização conhecido como relação *Within-Between*.

A relação *Within-Between* S é dada pela Equação 1, onde k representa o número de clusters, x é um vetor n-dimensional que representa uma observação do conjunto de dados X, $\bar{x}_i$ representa a centroide do cluster i, $\bar{x}$ representa o vetor médio de X e n_i representa o número de observações pertencentes ao cluster i.

Para avaliar o desempenho do k-Means, foi calculada a matriz de confusão para avaliar a taxa de verdadeiros positivos, falsos positivos, verdadeiros negativos e falsos negativos para cada uma das classes.

$$S_W = \sum_{i=1}^{k} \sum_{x \in X} (x - \bar{x}_i)(x - \bar{x}_i)^T$$

$$S_B = \sum_{i=1}^{k} n_i (\bar{x}_i - \bar{x})(\bar{x}_i - \bar{x})^T$$

$$S = \frac{S_B}{S_W} \tag{1}$$

5.1 Bases de Dados

5.1.1 Iris Data Set

Está é uma das bases de dados mais tradicionais em classificação. Esta base possuí informações sobre três diferentes classes de flores, tendo um total de 150 observações, sendo 50 de cada classe, e 5 atributos. Mais informações sobre esta base de dados podem ser encontradas em (FISHER, 1936).

5.1.2 UCI Wine Data Set

Esta base possuí informações sobre análises químicas de vinhos de três cultivares diferentes, todos de uma mesma região da Itália, possuindo um total de 178 observações e 13 atributos. Mais informações sobre esta base de dados podem ser encontradas em (FORINA, 1991).

5.1.3 UCI Seeds Data Set

Esta base possuí informações sobre três tipos diferentes de trigo, possuindo um total de 210 observações, sendo 70 de cada classe, e 7 atributos. Mais informações sobre esta base de dados podem ser encontradas em (CHARYTANOWICZ et al., 2010).

6 Resultados

Nesta seção, são apresentados os resultados obtidos através do desenvolvimento experimental descrito na Seção 5.

6.1 Iris Data Set

A Tabela 1 apresenta a matriz de confusão resultante da aplicação do algoritmo *k-Means*, descrito na Seção 2.1, na base de dados Iris Data Set, apresentada na Seção 5.1.1, seguindo a metodologia descrita na Seção 4.

		Prevista		
		Iris-setosa	Iris-versicolor	Iris-virginica
Observada	Iris-setosa	50	0	0
	Iris-versicolor	0	48	2
	Iris-virginica	0	15	35

Tabela 1 – Matriz de confusão resultante do k-Means aplicado a base de dados Iris Data Set, descrita na Seção 5.1.1.

A classe *Iris-setosa* é bem discriminada em relação às demais, pois todas as observações foram corretamente classificadas e nenhuma observação de outra classe foi mal classificada como *Iris-setosa*. Em contrapartida, algumas observações das classes

Iris-versicolor e *Iris-virginica* se confundem no *k-Means*. Para a classe *Iris-versicolor*, 96% (48 de 50) das observações foram corretamente. A classe *Iris-virginica* é a que apresenta a menor taxa de acerto,onde 70% (35 de 50) das observações foram classificadas corretamente. No total, o *k-Means* classificou 88.7% das observações corretamente.

6.2 Wine Data Set

A Tabela 2 apresenta a matriz de confusão resultante da aplicação do algoritmo *k-Means*, descrito na Seção 2.1, na base de dados Wine Data Set, apresentada na Seção 5.1.2, seguindo a metodologia descrita na Seção 4.

		Prevista		
		Classe 1	Classe 2	Classe 3
Observada	Classe 1	46	0	13
	Classe 2	1	50	20
	Classe 3	0	18	30

Tabela 2 – Matriz de confusão resultante do k-Means aplicado a base de dados Wine Data Set, descrita na Seção 5.1.2.

Nesta base de dados, $70,8\%$ das observações foram corretamente classificados pelo *k-Means*. Para a classe 1, 78% (46 de 59) das observações foram classificadas corretamente, sendo que todas observações restantes foram mal-classificadas como pertencentes a classe 3. Já para a classe 2, 70.4% (50 de 71) foram classificados corretamente, sendo que a grande maioria das observações mal-classificadas, 28.2% (20 de 71), foram confundidas com a classe 3 e apenas 1.4% (1 de 71) da observações foram confundidas com a classe 1. Por fim, para a classe 3, 62.5% (30 de 48) foram classificadas corretamente e as demais observações, 37.5% (18 de 48), foram confundidas com a classe 2.

6.3 Seeds Data Set

A Tabela 3 apresenta a matriz de confusão resultante da aplicação do algoritmo *k-Means*, descrito na Seção 2.1, na base de dados Seeds Data Set, apresentada na Seção 5.1.3, seguindo a metodologia descrita na Seção 4.

		Prevista		
		Classe 1	Classe 2	Classe 3
Observada	Classe 1	57	0	13
	Classe 2	10	60	0
	Classe 3	0	0	70

Tabela 3 – Matriz de confusão resultante do k-Means aplicado a base de dados Seeds Data Set, descrita na Seção 5.1.3.

Para a classe 1, 81.4% (57 de 70) das observações foram corretamente classificadas, sendo que as demais observações foram confundidas com a classe 3. 85.7% (60 de 70) das

observações pertencentes à classe 2 foram corretamente classificadas, sendo que as demais foram confundidas com a classe 1. Por fim, todas as 70 observações pertencentes a classe 3 foram corretamente classificadas. No total, 89% (187 de 210) das observações foram corretamente classificadas.

7 Conclusão

Este trabalho visa relatar a implementação do algoritmo *k-Means*, utilizando *C++* como linguagem de programação, como tarefa do curso PEL208 do programa de pós-graduação em engenharia elétrica do Centro Universitário FEI. O *k-Means* é um dos algoritmos de clusterização mais populares devido a sua simplicidade e facilidade de implementação.

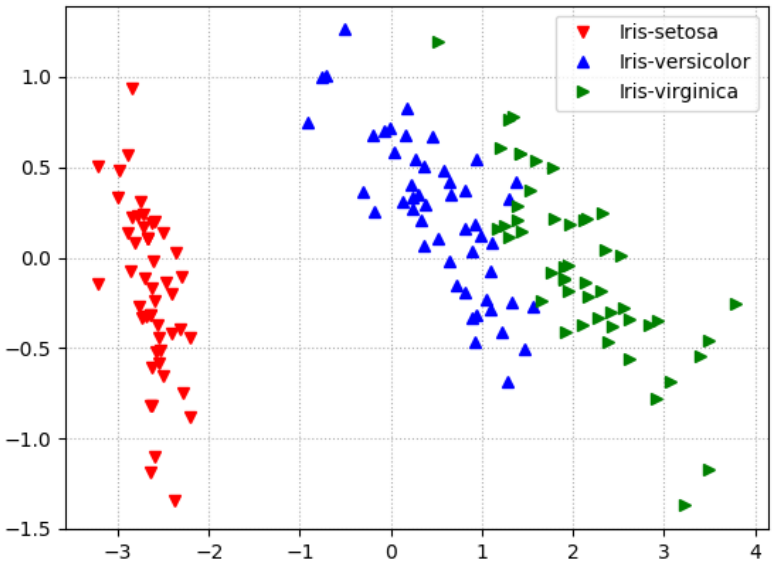

Figura 2 – Representação gráfica dos dados da base Iris Data Set, apresentados na Seção 5.1.1 após a aplicação da redução de dimensionalidade para duas através do PCA (PEARSON, 1901; HOTELLING, 1933).

O fato dos resultados, apresentados na Seção 6, serem condizentes com os apresentados em aula sugere que a implementação foi adequada. Ademais, estes resultados sugerem algumas reflexões sobre o funcionamento do *k-Means* e as características das bases de dados utilizadas para os experimentos.

A Figura 2 ilustra a base de dados Iris, apresentada na Seção 5.1.1, após ser aplicada uma redução de dimensionalidade de quatro para duas, através do PCA (PEARSON, 1901; HOTELLING, 1933). Nesta figura, podemos observar que, além dos grupos *Iris-versicolor* e *Iris-virginica* não serem esféricos, muitas das observações destes grupos são muito próximas.

Isso reforça a tendencia do *k-Means* apresentar uma qualidade inferior na precisão da clusterização quando os grupos não são esféricos. Além disso, o fato de nenhum dos elementos do grupo *Iris-setosa* ter sido mal-classificado sugere que o *k-Means* é efetivo na clusterização de grupos que são geometricamente discriminados.

Estas observações sobre os resultados da base de dados Iris sugerem algumas características sobre as demais bases. A base de dados Wine, descrita na Seção 5.1.2, foi a que apresentou menor precisão na clusterização. Isto sugere que os grupos não são esféricos e/ou são menos discriminados geometricamente a partir de seus atributos. Além disso, a alta taxa de de erros de classificação entre as classes 2 e 3 sugerem que elas apresentam alta intersecção entre si.

Por fim, a base de dados Seeds, descrita na Seção 5.1.3, foi a que apresentou maior precisão na clusterização. As observações da classe 3 foram todos classificados corretamente, porém, alguns elementos da classe 1 foram mal classificados como sendo pertencentes à classe 3. Ademais, as observações da classe 2 tiveram uma alta taxa de assertividade na classificação, mas algumas observações se confundirem com a classe 1. Isto sugere que, apesar dos grupos apresentarem uma pequena intersecção entre si, eles tendem a ser esféricos pois quase 90% das observações foram corretamente agrupadas pelo *k-Means*.

Referências

ALASHRI, S. et al. “climate change” frames detection and categorization based on generalized concepts. *International Journal of Semantic Computing*, World Scientific, v. 10, n. 02, p. 147–166, 2016. Citado na página 3.

CHARYTANOWICZ, M. et al. *seeds Data Set*. 2010. Acessado em: 2018-11-17. Disponível em: <https://archive.ics.uci.edu/ml/datasets/seeds>. Citado na página 5.

DAS, P.; DAS, D. K.; DEY, S. A modified bee colony optimization (mbco) and it’s hybridization with k-means for an application to data clustering. *Applied Soft Computing*, Elsevier, 2018. Citado na página 3.

DHEERU, D.; TANISKIDOU, E. K. *UCI Machine Learning Repository*. 2017. Disponível em: <http://archive.ics.uci.edu/ml>. Citado na página 4.

FISHER, R. *Iris Data Set*. 1936. Acessado em: 2018-11-17. Disponível em: <https://archive.ics.uci.edu/ml/datasets/Iris>. Citado na página 5.

FORGY, E. W. Cluster analysis of multivariate data: efficiency versus interpretability of classifications. *biometrics*, v. 21, p. 768–769, 1965. Citado na página 1.

FORINA, M. e. a. *Wine Data Set*. 1991. Acessado em: 2018-11-17. Disponível em: <https://archive.ics.uci.edu/ml/datasets/Wine>. Citado na página 5.

GUENNEBAUD, G.; JACOB, B. et al. *Eigen v3*. 2010. Http://eigen.tuxfamily.org. Citado na página 3.

HORVATH, M. et al. Exploiting cluster specialization into linear weighted hybrid recommender systems. In: *I Concurso Latino-americano de Trabalhos de Graduação*. [S.l.]: Sociedade Brasileira de Computação (SBC), 2018. cap. 4, p. 44–51. Citado na página 3.

HOTELLING, H. Analysis of a complex of statistical variables into principal components. *Journal of educational psychology*, Warwick & York, v. 24, n. 6, p. 417, 1933. Citado na página 7.

JAIN, A. K. Data clustering: 50 years beyond k-means. *Pattern recognition letters*, Elsevier, v. 31, n. 8, p. 651–666, 2010. Citado na página 1.

JIMÉNEZ, F. et al. Accelerating a multi-objective memetic algorithm for feature selection using hierarchical k-means indexes. In: ACM. *Proceedings of the Genetic and Evolutionary Computation Conference Companion*. [S.l.], 2018. p. 181–182. Citado na página 2.

LLOYD, S. Least squares quantization in pcm. *IEEE transactions on information theory*, IEEE, v. 28, n. 2, p. 129–137, 1982. Citado na página 1.

MACKAY, D. J.; KAY, D. J. M. *Information theory, inference and learning algorithms*. [S.l.]: Cambridge university press, 2003. Citado na página 1.

MACQUEEN, J. et al. Some methods for classification and analysis of multivariate observations. In: OAKLAND, CA, USA. *Proceedings of the fifth Berkeley symposium on mathematical statistics and probability*. [S.l.], 1967. v. 1, n. 14, p. 281–297. Citado na página 1.

PEARSON, K. Liii. on lines and planes of closest fit to systems of points in space. *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science*, Taylor & Francis, v. 2, n. 11, p. 559–572, 1901. Citado na página 7.

SOMMER, F.; FOUSS, F.; SAERENS, M. Modularity-driven kernel k-means for community detection. In: SPRINGER. *International Conference on Artificial Neural Networks*. [S.l.], 2017. p. 423–433. Citado na página 3.

STEINHAUS, H. Sur la division des corp materiels en parties. *Bull. Acad. Polon. Sci*, v. 1, n. 804, p. 801, 1956. Citado na página 1.

ZHONG, C. et al. A fast minimum spanning tree algorithm based on k-means. *Information Sciences*, Elsevier, v. 295, p. 1–17, 2015. Citado na página 2.