K-means

Gustavo Aparecido de Souza Viana

***Abstract*— A área de machine learning vem crescendo exponencialmente, com intuíto de desenvolver algoritmos ou aplicações para analisar grande bases de dados e assim tirar conclusões. Partindo desses principio, existem alguns algoritmos de machine learning que tem como intuito de classificar dados, o K-means utiliza esse conceito mas sem observações com o resultado delas. O objetivo deste trabalho é utilizar o K-means como classificador de três bases citada na introdução. Os resultados mostraram que o é possível aplicar o K-means como classificador de N dimensões.**

I. INTRODUÇÃO

A necessidade de classificações está sendo bem comum na atualidade, podemos dar o exemplo de uma empresa que necessita predizer qual estado da máquina (bom, normal, ou ruim) e assim com base no histórico dela e com base nos dados atuais, conseguimos classificar a mesma.

Conseguimos predizer algum informação baseada em uma base conhecida contendo resultados utilizando machine learning ou com métodos estatísticos. É de extrema importância conhecer o ramo de atividade da aplicação pois métodos de machine learning geralmente estão relacionados a rede neural, consquentemente necessitam de mais dados para que as predições tenham uma maior acurácia. De contra partida, métodos estatísticos não necessitam.

Neste trabalho foi abordado a implementação do *K-means* com objetivo de clusteirizar a base de dados "Iris" mas aplicando o PCA previamente com intuito de deixar somente as dimensões necessárias e nas bases *Alps Water*, *Books x Grades* e *US Census Dataset*.

II. CONCEITOS FUNDAMENTAIS

Nesta seção, será apresentado os conceitos básicos utilizados para o *K-means* e para o *Principal Components Analysis*.

A. K-means

Primeiramente, *centroid* é o o ponto central de um conjunto de pontos, podendo ser de N dimensões. A ideia do *K-means* é clusteirizar pontos com N dimensões baseado em uma quantidade fixa de *centroids* que o usuário necessita, ou seja, o usuário necessita previamente saber quatas classes deverão ser encontradas pelo *K-means* [2].

O processo de clusteirização inicia pela geração dos centroids aleatórios ou com alguma entropia. Após isso, é calculado a distância de cada ponto para cada *centroid* e assim é relacionado o ponto com o centroid onde há a menor distância. No seguinte passo, recalculamos os *centroids* com a média dos pontos pertencentes e classificados por esse *centroid*. E assim, repetimos o processo de classificação para cada *centroid* e o processo de atualização dos *centroids* até que um critério de parada é verdadeiro. Importante ressaltar que o cáculo inicial dos *centroids*, o cálculo de distância de pontos e o critério de parada são definidos pela implementação, ou seja, pode variar de acordo com a necessidade. A Figura 1 representa o processo simples do *K-means*.

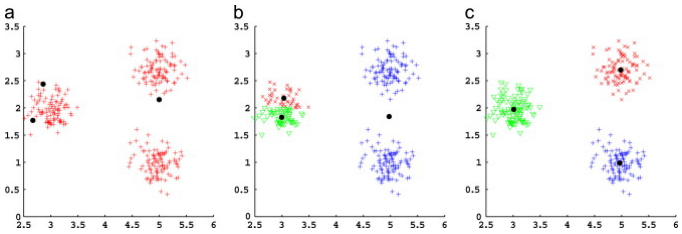

Fig. 1: Simples exemplo do *K-means*

B. Principal Component Analysis

A ideia do *PCA* é reduzir a dimensionalidade dos dados para assim representar o mesmo dado com menos dimensões que são novas variáveis linearmente independentes chamadas de componentes [1].

O processo de conversão inicia pelo cáculo da covariância, e após isso conseguimos calcular os autovetores e autovalores. Os autovalores demonstram o quanto aquela componente representa ou explica do dado em si, quanto maior o autovalor mais importante e consequentemente, explica majotariamente o dado. O autovetor é a componente em si, uma matriz de transformação, ou seja, se multiplicarmos os dados pelo autovetor teremos uma nova base reduzida. Imporante ressaltar que cada componente é ortogonal a anterior e assim sucessivamente.

III. METODOLOGIA

Nesta seção será apresentado a metodologia utilizada para implementar o *K-means* juntamente, implementado em Python. O código fonte pode ser encontrado em https://github.com/apparecidoo/master-special-learning-topic.

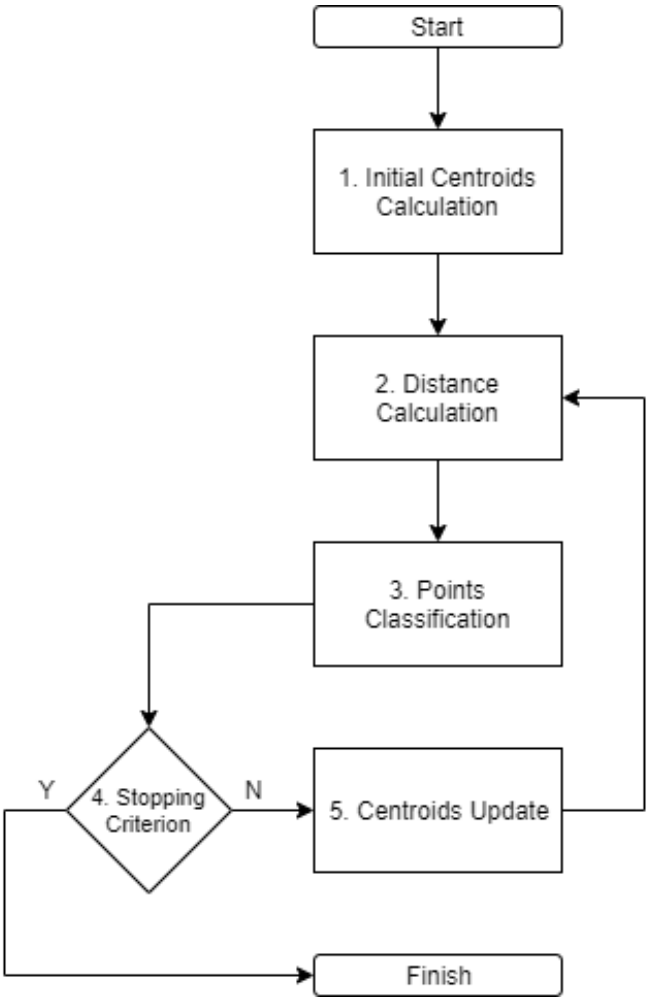

Fig. 2: Fluxograma da implementação do *K-means*

A metodologia consiste em 5 passos. Com a base definida, é iniciado a partir do cálculo inicial dos *centroids* onde a quantidade é definida previamente; no caso deste trabalho foi utilizado a escolha randomica, onde é escolhido pontos aleatórios da base definida. O segundo passo, é calculado a distância de cada ponto para cada *centroid*, no caso desta implementação foi utilizado distância Euclidiana, formando uma matriz $Rows$ x $Centroids$. O terceiro passo é definido pela classificação destes pontos, é escolhida a menor distância baseada na distância calculada de cada ponto até o *centroids*, assim é definido a que *centroid* o ponto pertence. No passo seguinte, é verificado o critério de parada, no caso desta implementação é validado se há alguma alteração na classificação, caso não há é finalizado o processo e retornado a base de dados classificada juntamente com os *centroids*, caso contrário inicia a etapa 5. Na etapa 5, os *centroids* são atualizados com o cálculo da média dos pontos pertencentes a esse *centroid*, e assim retornamos a etapa 2 para reiniciarmos o processo de classificação até que o critério de parada seja verdadeiro.

IV. EXPERIMENTOS E RESULTADOS

Nesta seção será apresentado os experimentos e seus respectivos resultados.

O experimento consiste na aplicação do *Principal Component Analysis* para a base de dados Iris e em seguida a aplicação do *K-means*. Para as demais bases de dados foi aplicado somente o *K-means*, sendo elas a *Alps Water*, *Books x Grades* e *US Census Dataset*.

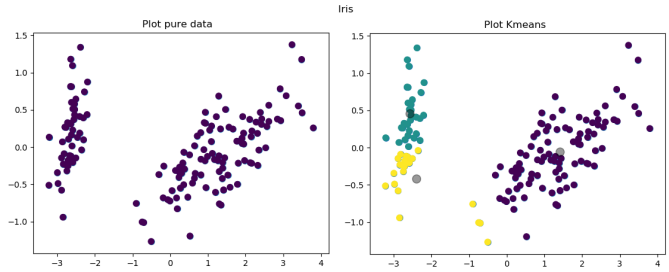

Fig. 3: Resultado do *K-means* na base de dados Iris

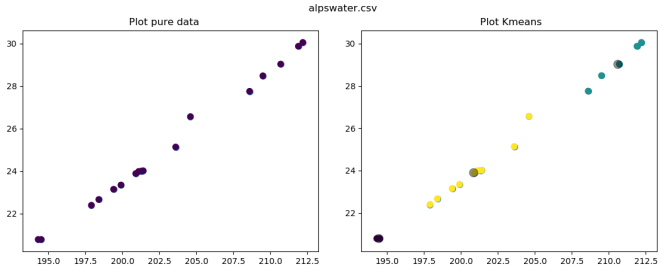

Fig. 4: Resultado do *K-means* na base de dados *Alps Water*

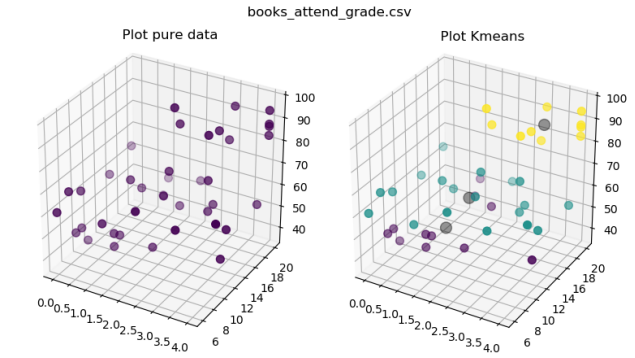

Fig. 5: Resultado do *K-means* na base de dados *Books Attend Grade*

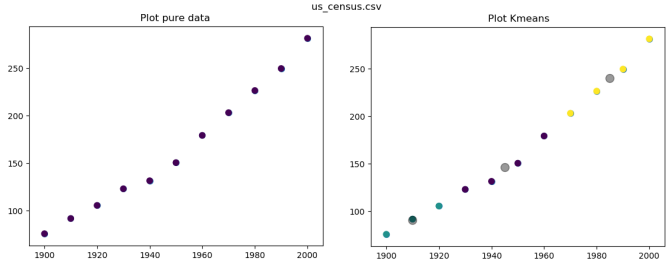

Fig. 6: Resultado do *K-means* na base de dados *Us Census*

V. CONCLUSÃO

Após a análise dos conceitos e resultados obtidos após a aplicação dos métodos *K-means*, podemos concluir que o *K-means* tem capacidade de clusteirizar dados númericos mas *outliers* podem prejudicar o cálculo dos centroids. Um ponto negativo é que para bases de dados que tenham muitos pontos, o cálculo para a classificação fica custoso, dependendo do caso ser inviável a utilização. Os próximos passos seria fazer a predição de alguns dados e validar a precisão e acurácia.

REFERENCES

[1] M. Baxter and M. Heyworth. Principal components analysis of compositional data in archaeology. 1989.

[2] A. Likas, N. A. Vlassis, and J. J. Verbeek. The global k-means clustering algorithm. *Pattern Recognition*, 36:451–461, 2003.