UNIVERSIDADE FEDERAL DE LAVRAS
DEPARTAMENTO DE CIÊNCIA DA COMPUTAÇÃO
GCC 128 - INTELIGÊNCIA ARTIFICIAL
Prof. Ahmed Ali Abdalla Esmin

Érika Mara de Morais Machado **Matrícula: 202310058**
Maria Luiza Bernardo Madeira **Matrícula: 202310981**

RELATÓRIO TÉCNICO - ALGORITMO K-Means

Foram desenvolvidos dois algoritmos para a implementação do modelo KMeans. Um deles foi construído do zero, utilizando apenas a linguagem Python e bibliotecas básicas como NumPy e Matplotlib, enquanto o outro utilizou a biblioteca scikit-learn, que já disponibiliza a classe KMeans pronta e otimizada para uso. O objetivo principal foi comparar os dois métodos na tarefa de agrupamento utilizando a base de dados Iris, avaliando a qualidade da clusterização, a distribuição dos grupos formados e o tempo de execução.

Na versão manual, o algoritmo foi implementado com inicialização dos centróides pelo método KMeans++, cálculo de distâncias euclidianas e realocação iterativa dos pontos até a convergência. Foram testados dois valores de k (3 e 5), sendo que os resultados mostraram que k=3 apresentou melhor desempenho, com Silhouette Score de 0.5512. O valor de k=5 apresentou uma queda no índice, atingindo 0.4899, o que indica que a divisão em cinco clusters não representou bem a estrutura dos dados. Além disso, foram gerados gráficos para a visualização dos agrupamentos.

Na versão utilizando scikit-learn, o modelo foi aplicado com k=3, considerado o número ideal de clusters para a base Iris. O algoritmo alcançou um Silhouette Score de 0.5528, muito próximo do resultado obtido na implementação manual, validando a consistência dos resultados. A distribuição dos dados entre clusters também foi semelhante, o que reforça que o valor de k=3 é o mais adequado para ser utilizado nessa base de dados.

Em relação ao tempo de execução, foi possível notar que a versão manual apresentou valores ligeiramente menores do que a versão com scikit-learn. Porém, esse resultado deve ser interpretado com cautela, pois a biblioteca scikit-learn, por padrão, realiza múltiplas inicializações (n_init=10) e possui verificações adicionais que aumentam a robustez e a precisão do algoritmo, o que justifica o tempo um pouco maior. Já a implementação manual, por ser mais simples, executa apenas uma inicialização e menos verificações internas, o que a torna aparentemente mais rápida em bases pequenas, mas menos estável e escalável em bases maiores.

A implementação manual proporcionou um melhor entendimento do funcionamento interno do KMeans e possibilitou explorar conceitos como inicialização de centróides, convergência e métricas de qualidade. Entretanto, ela não é a mais indicada para aplicações práticas em cenários reais, principalmente quando se trabalha com grandes volumes de dados. Já a versão com scikit-learn se mostrou mais prática, estável e eficiente, apresentando recursos adicionais de configuração e otimizações internas que garantem maior confiabilidade nos resultados.

Link para o vídeo de apresentação no YouTube: https://youtu.be/0ga0rmU02Wo