Planejamento da construção de aeroportos na cidade de Belém utilizando o método de clusterização K-*Means*

Eduardo Gil S. Cardoso[1], **Gabriela S. Maximino**[1], **Igor Matheus S. Moreira**[1]

[1] Instituto de Ciências Exatas e Naturais – Faculdade de Computação
Universidade Federal do Pará – Belém, PA – Brasil
{eduardo.gil.s.cardoso,gabriela.maximino,igor.moreira}@icen.ufpa.br

Abstract. *This article demonstrates the application of the K-Means algorithm to the clustering problem involving the construction of three airports in the city of Belém, Pará. This report is part of the deliverable associated to the task proposed by professor Reginaldo Cordeiro dos Santos Filho for the Artificial Intelligence course, taught under the Computer Science Bachelor's degree program at the Federal University of Pará.*

Resumo. *Este artigo demonstra a aplicação do algoritmo K-Means para o problema de clusterização envolvendo a construção de três aeroportos na cidade de Belém, Pará. Este trabalho é parte do entregável relativo à tarefa proposta pelo Prof. Dr. Reginaldo Cordeiro dos Santos Filho para a disciplina de Inteligência Artificial, ministrada sob o curso de Bacharelado em Ciência da Computação na Universidade Federal do Pará.*

1. Introdução

A cidade de Belém, capital paraense, é o segundo munícipio mais populoso da região Norte do Brasil [IBGE 2019]. Possuindo 71 bairros divididos em distritos administrativos [Wikipédia 2020], a cidade possui apenas um aeroporto, o Aeroporto Internacional Val-de-Cans. Nesse contexto, o segundo trabalho da disciplina Inteligência Articifial desenha um cenário hipotético em que, devido à grande demanda de viagens no estado, faz-se necessária a construção de três novos aeroportos para atender de forma igualitária aos bairros da cidade, considerando – devido à maior demografia – apenas cinco dos oito distritos: de Belém; do Entroncamento; do Guamá; do Benguí; e da Sacramenta.

Para esse problema, alguns critérios foram considerados: cada novo aeroporto deve atender a pelo menos um bairro e ao máximo de bairros próximos a ele; a distância total dos bairros para os aeroportos deve ser mínima no intuito de evitar transtornos; e a localização do Aeroporto Internacional Val-de-Cans deve ser considerada. Tendo em vista o especificado, tem-se que o prolema pode ser resolvido por meio da clusterização – uma técnica de aprendizado não-supervisionado que realiza o agrupamento de dados de acordo com características em comum. Nesse caso, os bairros que possuem proximidade geográfica serão agrupados em *clusters*, cujos centroides indicarão a localização dos novo aeroportos, sempre lembrando que o já existente Aeroporto Internacional Val-de-Cans representa um centroide fixo.

Considerando as informações levantadas, escolheu-se o algoritmo K-*Means* para realizar a clusterização. Esse método foi considerado ideal devido à utilização de centroides cujos valores são atualizados no decorrer da execução, e ao redor dos quais as amostras

serão agrupadas baseadas na sua distância entre todos os centroides presentes. Além disso, o fato de K ser conhecido e os dados serem do tipo real também fizeram com que a escolha convergisse para esse método.

Diante do exposto, as seções subsequentes estão divididas da seguinte forma: a Seção 2 apresenta uma descrição da base de dados construída; a Seção 3 descreve o processo de realização do trabalho; a Seção 4 apresenta os resultados da aplicação do algoritmo; por fim, a Seção 5 sintetiza o trabalho e apresenta as considerações finais.

2. Descrição da base de dados

A base de dados construída é composta pelas coordenadas geográficas do ponto central de cada bairro dos cinco distritos administrativos de Belém considerados para esse trabalho, já citados anteriormente. Como resultado, obteve-se um conjunto de dados contendo 40 instâncias e 2 *features*, que são 'Bairro' e 'Coordenadas'. As coordenadas estão no formato (latitude, longitude). A Figura 1 apresenta o gráfico construído a partir do conjunto de dados produzido, no qual a marcação em preto indica a localização do Aeroporto Internacional Val-de-Cans – nosso centroide fixo.

3. Metodologia do trabalho

O trabalho foi dividido em três etapas: construção da base de dados, implementação do algoritmo e realização de experimentos, e escrita do artigo.

Para a construção da base de dados, inicialmente verificou-se a lista de bairros pertencentes aos cinco distritos administrativos de Belém [Wikipédia 2020]. Em seguida, buscou-se a coordenada de cada bairro utilizando a ferramenta Nominatin do projeto *OpenStreetMap*, a qual fornece a delimitação e o ponto central dos bairros. Ao todo, 40 pontos centrais foram coletados e armazenados em uma planilha no formato `.csv`.

A implementação do algoritmo foi feita na linguagem `python`. Uma série de módulos foram utilizados na implementação, entre os quais podem-se nomear `dask`, `numba`, e `numpy`. `dask` foi utilizado para a paralelização dos experimentos ao criar um *cluster* de *workers*; `numba` foi utilizado para a compilação *just-in-time* de várias funções em código de máquina, bem como a paralelização de laços `for` dentro das funções compiladas; e `numpy` foi utilizado como base para as operações realizadas sobre os dados.

Em termos do algoritmo utilizado, o algoritmo K*-Means* foi implementado. A implementação desenvolvida pelo autores possui a particularidade de permitir a especificação de centroides fixos, i.e., centroides que não devem ser atualizados a despeito de serem considerados como parte da solução. Dois critérios de parada foram adotados: número máximo de iterações (definido por padrão como 200) e não-alteração de valores de centroides entre iterações.

Os experimentos foram realizados da seguinte forma: uma função foi redigida para realizar n execuções do algoritmo em paralelo. Esta função executa n vezes uma segunda função de forma paralelizada. Esta segunda função instancia um objeto da classe `KMedias`, passa a ele o conjunto de dados criado a fim de que ele seja clusterizado e retorna uma tupla contendo a solução encontrada e a função-objetivo (denominada na *codebase* como erro, uma vez que uma solução ótima minimiza esse valor). As especificações do trabalho requisitaram no mínimo $n = 20$; contudo, considerando o

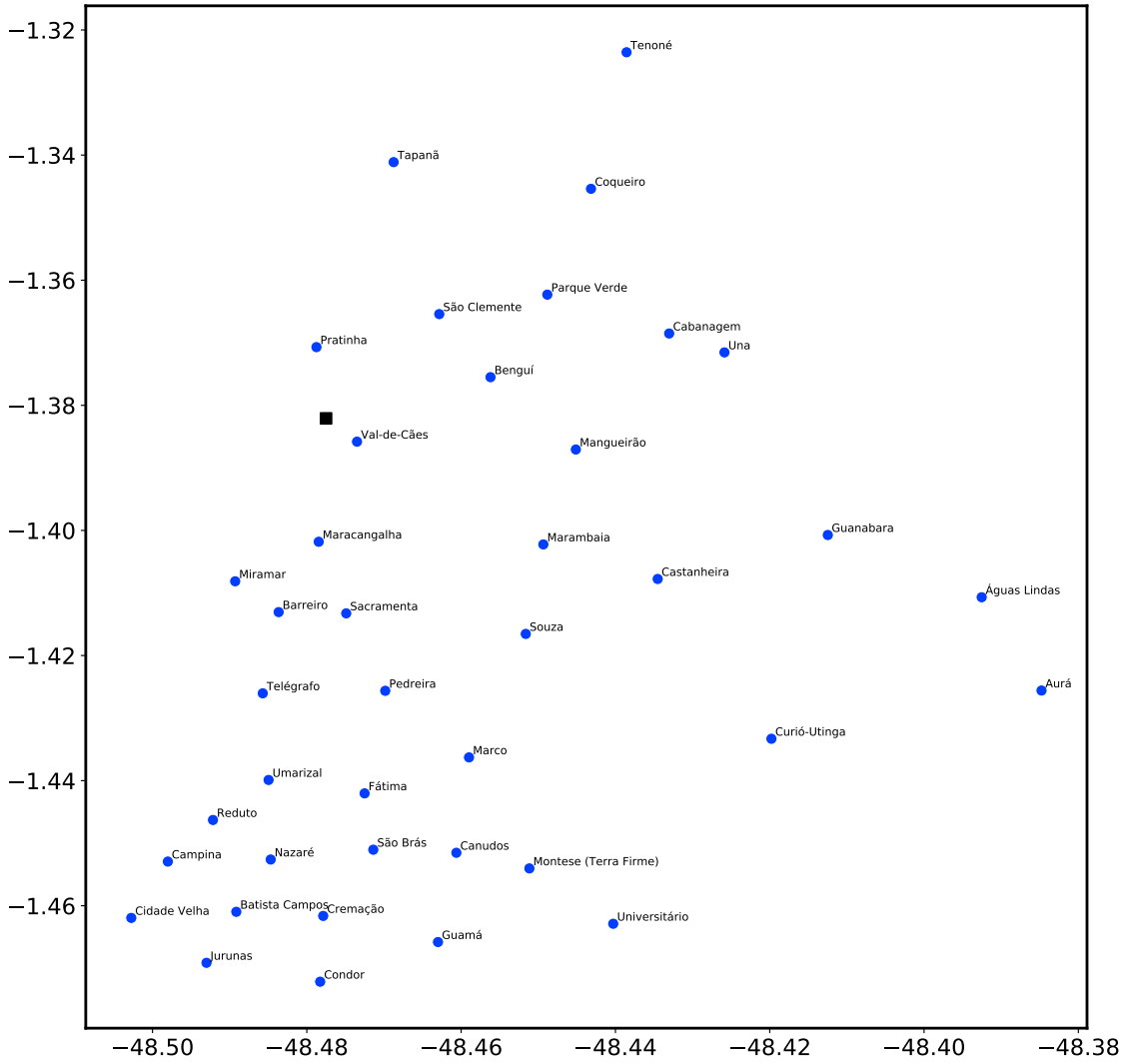

Figura 1. Conjunto de dados construído. Fonte: acervo próprio.

tamanho diminuto da base de dados e a rapidez da implementação realizada, optou-se por definir $n = 1000$.

4. Resultados

O algoritmo K-*Means* foi implementado de forma bem-sucedida, e a implementação desenvolvida foi utilizada para a resolução do problema proposto. A Figura 2 expõe a melhor e a pior solução encontrada nas 1000 execuções da implementação desenvolvida do algoritmo K-*Means*. Uma análise dessas soluções expõe uma significativa discrepância na qualidade dos resultados em favor da melhor solução, uma vez que as distâncias entre as observações e os centroides foram minimizadas.

É importante observar como a localização do Aeroporto Internacional Val-de-Cans, centroide do *cluster* azul, permaneceu inalterada entre as duas soluções e o visto na Figura 1, o que comprova que o algoritmo não a adulterou.

A *codebase* desenvolvida registra todas as soluções encontradas, bem como seus erros, em um objeto `DataFrame`. Dessa forma, o usuário final pode verificar as demais

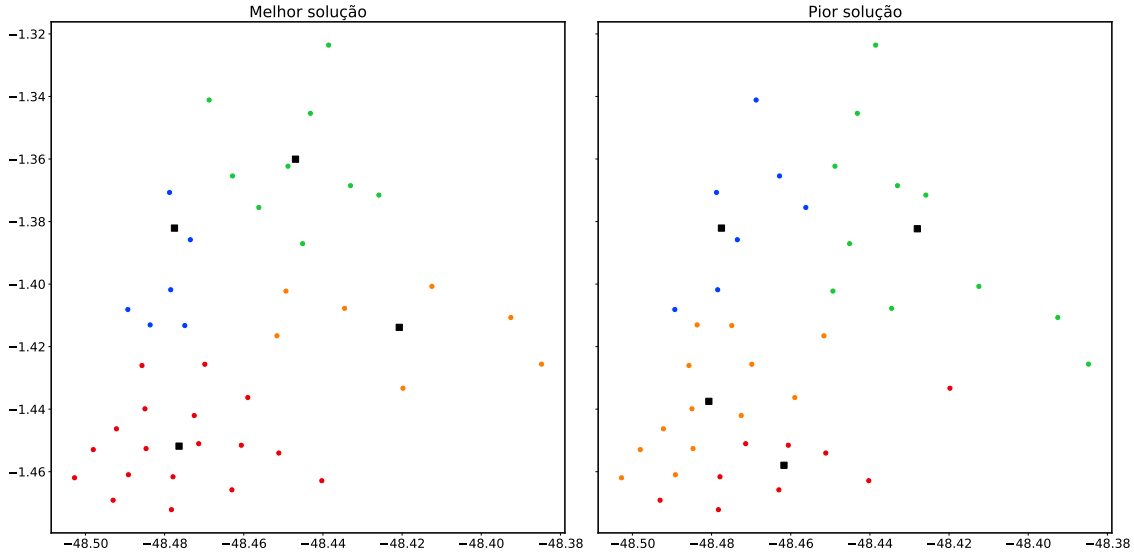

Figura 2. Melhor e pior solução encontrada. Fonte: acervo próprio.

soluções encontradas variando das melhores até as piores, a fim de auxiliar o processo de tomada de decisão em relação à melhor localização para os aeroportos.

Como produto adicional deste trabalho, tem-se que todas as funções desenvolvidas, bem como o conjunto de dados, seu pré-processamento, e seu processamento por meio da implementação desenvolvida do algoritmo K-*Means*, foram consolidados em um arquivo `.ipynb`, i.e., em um Jupyter Notebook. Este arquivo encontra-se no repositório do GitHub **@ygarasab/kmeans**. Neste repositório, pode-se também consultar as contribuições realizadas por cada integrante desta equipe e auditar a implementação desenvolvida. Para além disso, os rótulos dos bairros dentro de cada solução (i.e., a qual cluster cada bairro pertence em cada solução) podem ser consultados, caso seja pertinente.

5. Conclusão

Com base no exposto acima, tem-se que o problema do aeroporto foi resolvido de forma satisfatória pelo algoritmo K-*Means*. A implementação desenvolvida se vale do módulo `numba` para compilação *just-in-time* e paralelização de laços `for` a fim de tornar a execução do algoritmo tão rápida quanto possível e tirar máximo proveito da máquina em que é executado. Cumpre mencionar que todas as implementações produzidas foram desenvolvidas almejando total concordância com o estipulado no documento que descreve a atividade.

Em aderência ao requisitado nas especificações do trabalho e em reforço ao mencionado na Seção 4, cumpre ressaltar que a ferramenta GitHub foi utilizada como sistema de versionamento no decorrer do desenvolvimento deste trabalho, de forma que as contribuições dos integrantes desta equipe possam ser registradas e vistas. Neste repositório, para além da *codebase* desenvolvida para implementar o algoritmo K-*Means*, foi criado um arquivo `.ipynb` (i.e., um Jupyter Notebook) consolidando todas as peças de código relevantes, executando os experimentos aqui descritos e expondo os resultados alcançados em mais detalhes. Todas essas contribuições adicionais podem ser encontradas em **@ygarasab/kmeans**.

Referências

IBGE (2019). https://agenciadenoticias.ibge.gov.br/media/com_mediaibge/arquivos/7d410669a4ae85faf4e8c3a0a0c649c7.pdf. (Acessada em 23/12/2020).

Wikipédia (2020). Lista dos bairros de Belém (Pará). https://pt.wikipedia.org/wiki/Lista_de_bairros_de_Bel%C3%A9m_(Par%C3%A1). (Acessada em 23/12/2020).