

Algoritmos 2

Relatório do Trabalho Prático 1

Eduardo Assis Tomich
Raphael Alves dos Reis
Victor Yuji Yano

Junho 2025

1 Introdução

Este relatório descreve a implementação do Trabalho Prático 1 da disciplina DCC207 – Algoritmos 2, ministrada pelo Prof. Renato Vimieiro, no Departamento de Ciência da Computação da Universidade Federal de Minas Gerais. O objetivo principal do trabalho é desenvolver um sistema interativo para consulta ortogonal de estabelecimentos comerciais (bares e restaurantes) em Belo Horizonte, utilizando uma árvore k-dimensional (k-d tree) para busca espacial eficiente. O sistema permite visualizar os estabelecimentos em um mapa interativo, filtrá-los por uma região retangular e exibir informações em uma tabela sincronizada. Este documento detalha o problema abordado, as decisões de implementação, os resultados obtidos e exemplos de funcionamento do sistema.

2 Descrição do Problema

O trabalho consiste em criar uma aplicação web interativa que exibe bares e restaurantes de Belo Horizonte, registrados na base de dados da Prefeitura (PBH), disponível em <https://dados.pbh.gov.br/dataset/atividades-economicas1>, com dados fixados em 01/04/2025. A aplicação deve:

- Exibir estabelecimentos como pinos em um mapa interativo da cidade.
- Permitir consultas ortogonais via seleção retangular no mapa, utilizando uma árvore k-dimensional.
- Mostrar informações dos estabelecimentos (nome, endereço, data de início, alvará) em uma tabela.
- Suportar filtros por nome, endereço e alvará, com opção de reset.
- Hospedar a aplicação no Render.
- Coletar os dados do Comida di Buteco para marcação no mapa

Os dados requerem pré-processamento para filtrar bares e restaurantes (baseado na descrição CNAE) e converter endereços em coordenadas geográficas (latitude e longitude).

A biblioteca `dash-leaflet` foi utilizada para o mapa interativo, e a implementação foi feita em Python.

3 Implementação

A implementação foi dividida em quatro etapas principais: pré-processamento dos dados, construção da árvore k-dimensional, reprojeção do GeoJSON dos bairros e configuração da interface interativa com Dash.

3.1 Pré-processamento dos Dados

Os dados foram carregados de um arquivo CSV (`20250401_bares_e_restaurantes.csv`) utilizando a biblioteca `pandas`. O pré-processamento envolveu:

- **Filtragem:** Estabelecimentos foram filtrados com base na descrição CNAE, mantendo aqueles com termos relacionados a bares e restaurantes.
- **Conversão de Coordenadas:** As geometrias no formato UTM (EPSG:31983) foram convertidas para latitude e longitude (EPSG:4326) usando a biblioteca `pyproj`. A função `utm_to_latlon` extrai coordenadas do formato POINT (x y) e realiza a transformação.
- **Criação de Endereço Completo:** Combinamos os campos `DESC_LOGRADOURO`, `NOME_LOGRADOURO`, `NUMERO_IMOVEL`, `COMPLEMENTO` e `NOME_BAIRRO` para formar um endereço legível.
- **Formatação de Data:** A coluna `DATA_INICIO_ATIVIDADE` foi convertida para o formato YYYY-MM-DD para exibição na tabela.
- **Seleção de Atributos:** Mantivemos apenas os atributos necessários: `NOME_FANTASIA`, `latitude`, `longitude`, `FULL_ADDRESS`, `DATA_INICIO_STR`, `IND_POSSUI_ALVARA`.
- **Webscrapping:** Extraímos dados dos bares e restaurantes participantes do “Comida di Buteco” e seus respectivos endereços. Fizemos merge com os dados baixados da prefeitura PBH utilizando os endereços e CNPJ.

3.2 Árvore K-Dimensional

A árvore k-dimensional foi implementada para realizar buscas ortogonais eficientes. A classe `KDTreeNode` representa um nó com um ponto (latitude, longitude), dados associados, eixo de divisão e filhos. As funções principais são:

- `build_kd_tree`: Constrói a árvore recursivamente, alternando os eixos (latitude e longitude) e dividindo os pontos pela mediana.
- `range_search`: Realiza a busca ortogonal em um retângulo definido por `[lat_min, lon_min, lat_max, lon_max]`, retornando índices de pontos dentro da região.

“A árvore k-dimensional foi construída a partir das coordenadas geográficas dos estabelecimentos (longitude, latitude). Durante a execução do sistema, as buscas ortogonais são realizadas ao desenhar um retângulo no mapa. A árvore é então percorrida para identificar rapidamente quais estabelecimentos estão contidos nesse retângulo de seleção.”

3.3 Reprojeção do GeoJSON

Os contornos dos bairros de Belo Horizonte, construídas utilizando o seguinte arquivo:

- `bairro_popular_lei1069814012014.geojson`

estavam no sistema UTM (EPSG:32723). Utilizamos `pyproj` para reprojeter as coordenadas para WGS84 (EPSG:4326). A função `reproject_geometry` processa geometrias `MultiPolygon`, transformando cada ponto com `transformer_bairros`. O GeoJSON reprojetoado é usado para exibir os contornos dos bairros no mapa, controlado por um toggle.

3.4 Interface Interativa

A interface foi construída com `dash` e `dash-leaflet`, dividida em duas seções:

- **Mapa:** Exibe o mapa de Belo Horizonte com:
 - Camada de tiles do OpenStreetMap.
 - Marcadores para estabelecimentos, com tooltips e popups (nome e endereço).
 - Ferramenta de seleção retangular (`d1.EditControl`).
- **Tabela e Filtros:** Inclui:
 - Tabela com nome, endereço, data de início e alvará, com suporte a ordenação.
 - Filtros
 - * nome
 - * endereço, alvará e participação no Comida di Buteco.
 - * Botão de reset para limpar filtros e seleções.

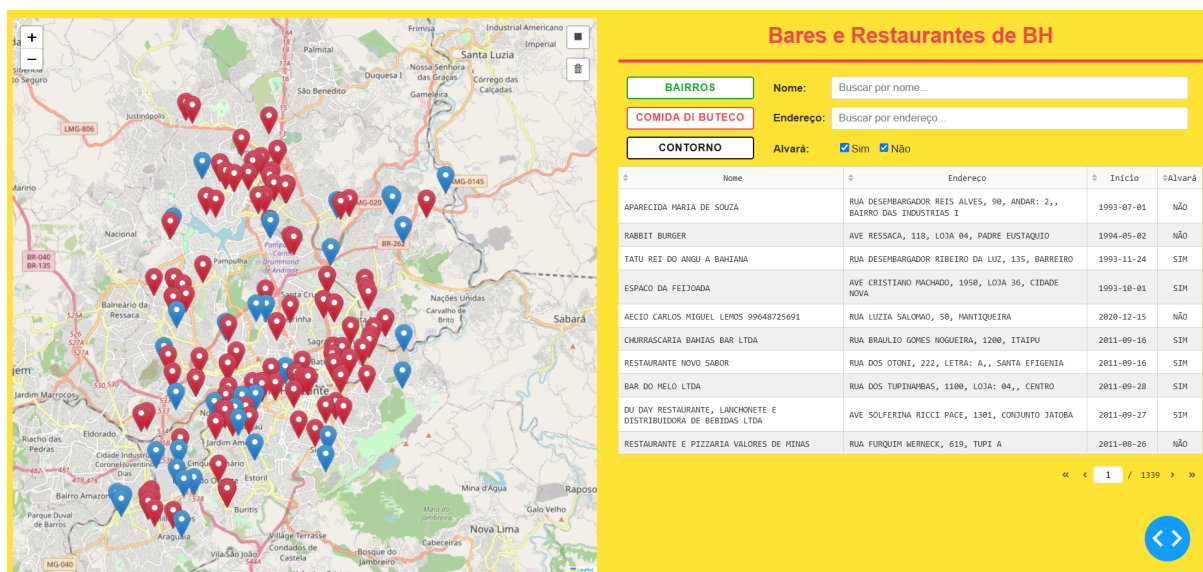


Figure 1: Tela Inicial.

Os callbacks sincronizam o mapa e a tabela:

- `update_markers`: Atualiza marcadores com base nos limites do mapa ou retângulo desenhado, usando a k-d tree. Limita a 200 marcadores para desempenho, exceto quando há seleção.
- `toggle_bairros_layer`: Exibe ou oculta os contornos dos bairros.
- `update_table_and_selection`: Atualiza a tabela com base em filtros (nome, endereço, alvará) e seleção espacial, retornando índices selecionados para destacar linhas.

4 Exemplos de Funcionamento

A seguir, descrevemos exemplos que ilustram o funcionamento do sistema:

4.1 Consulta Ortogonal

O usuário desenha um retângulo no mapa centrado no bairro Savassi (aproximadamente $[-19.938, -43.933, -19.928, -43.923]$). A k-d tree retorna os estabelecimentos dentro da região, e a tabela exhibe:

- Nome: “Bar do João”
- Endereço: “Rua Antônio de Albuquerque, 123, Savassi”
- Início: “2020-05-10”
- Alvará: “Sim”
- Comida de Boteco: “Sim”

Os marcadores correspondentes aparecem no mapa, com tooltips mostrando o nome ao passar o mouse.

4.2 Filtros

O usuário filtra por nome (“Pizza”) e alvará (“Sim”). A tabela exhibe apenas estabelecimentos com “Pizza” no nome e alvará válido, como:

- Nome: “Pizza Sur Savassi”
- Endereço: “Avenida Getúlio Vargas, 456, Savassi”
- Início: “2018-03-15”
- Alvará: “Sim”

O mapa atualiza os marcadores para refletir o filtro.

4.3 Reset

Ao clicar no botão “Reset Filter”, a tabela retorna todos os estabelecimentos visíveis nos limites iniciais do mapa (aproximadamente $[-20, -44, -19, -43]$), e os marcadores são atualizados sem filtros aplicados.

5 Decisões de Projeto

Algumas decisões foram tomadas para otimizar a implementação:

- **Limitação de Marcadores:** Para evitar sobrecarga no navegador, limitamos a exibição a 200 marcadores quando não há seleção, usando amostragem aleatória.
- **Reprojeção de Coordenadas:** Optamos por realizar a conversão UTM para WGS84 no pré-processamento, armazenando latitude e longitude no DataFrame para evitar cálculos repetitivos.
- **Estilo Visual:** Utilizamos ícones azuis e vermelhos do Leaflet para marcadores e contornos pretos tracejados para bairros, garantindo clareza visual.
- **Sincronização:** A tabela e o mapa são atualizados dinamicamente via callbacks, garantindo consistência entre a seleção espacial e os filtros textuais.

6 Conclusão

O sistema implementado atende aos requisitos do trabalho, proporcionando uma interface interativa para consulta de bares e restaurantes em Belo Horizonte. A árvore k-dimensional garante buscas ortogonais eficientes, enquanto a integração com `dash-leaflet` oferece uma experiência visual fluida. O pré-processamento dos dados e a reprojeção do GeoJSON foram realizados com sucesso, e a aplicação está hospedada no Render, tendo em vista se tratar de um sistema dinâmico.

7 Referências

- Dados da PBH: <https://dados.pbh.gov.br/dataset/atividades-economicas1>
- Dados da Comida di Buteco: <https://comidadibuteco.com.br/butecos/belo-horizonte/>
- Discussão sobre GeoJSON: <https://medium.com/starschema-blog/draw-a-map-of-the-distr>
- Fórum de Jornalismo de Dados: <https://forum.jornalismodedados.org/t/geojson-nao-funciona/467>