

**Universidade Federal de Pernambuco**

**Centro de Informática - CIn**

**Disciplina:** IF969

**Docente:** Sérgio Ricardo de Melo Queiroz

**Discente:** Liosvaldo Mariano Santiago de Abreu

**Relatório do projeto: Grupo #3.6**

Recife, 18 de fevereiro de 2024

**Contexto do problema**

Nesse presente relatório, utilizaremos a base de dados aberta da rede atenção à saúde no Recife, disponibilizada pela prefeitura do Recife, para realizar um grafo de menor caminho utilizando os 120 primeiros itens da tabela de dados de pontos de saúde (PSFs, Academias da cidade, Policlínicas). A base de dados contém, além de informações de latitude e longitude dos pontos de saúde, informações como endereço, distrito sanitário e especialidade.

**Implementação**

**Algoritmo utilizado.** Algoritmo de PRIM

**Desenvolvimento.** Inicialmente, realizamos uma matriz de distâncias adjacentes com base na latitude e longitude de cada um dos pontos citados. Na sequência, utilizamos o algoritmo de Prim, que é um algoritmo guloso que encontra uma árvore geradora mínima de um grafo G dado um vértice inicial. para encontrar uma árvore com a menor distância entre as localidades

O algoritmo de prim inicia o processo a partir de um vértice inicial e visita os vértices vizinhos que estão dentro de uma sequência prioritária não visitados, atualizando o seu custo e seus predecessores. Feito isso, é escolhido um vértice com menor custo até que não exista vértice na fila.

Na sequência, utilizamos bibliotecas de visualização e plot para imprimir um grafo contendo todos os vértices utilizados e suas respectivas arestas.

**Bibliotecas utilizadas.** Inicialmente utilizamos a biblioteca de pandas para ler o CSV e realizar tratamento das informações. Para gerar a matriz adjacência, utilizamos a biblioteca Distance para calcular a distância entre os pontos solicitados. Na sequência, utilizamos a biblioteca networkx para geração de uma estrutura grafo que utilizamos durante o processo. Após isso, aplicamos o algoritmo de prim realizado de maneira autoral, utilizando a extensão da biblioteca networkx para geração de gráficos, baseada no matplotlib.pyplot.

**Conclusão**

Para facilitar o desenvolvimento, realizamos o download do arquivo disponível na página da prefeitura do recife Recife, Dados criamos o arquivo.

Figura 1 - Página dataset rede de atenção à saúde no Recife



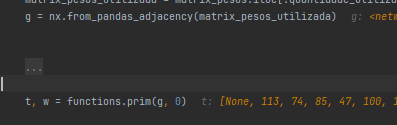
Após o download, executamos um algoritmo para obtermos a matriz adjacente dos 120 primeiros pontos da rede de atenção à saúde utilizando Pandas e a biblioteca Distance.

Figura 2 - Algoritmo de geração da matriz adjacente com base na latitude e longitude.



Feito isso, criamos um arquivo um grafo networkx para iniciar os demais passos do algoritmo de prim

Figura 3 - inicialização do grafo e execução do algoritmo de PRIM



No algoritmo, carregamos todos os pesos inicialmente com o valor infinito da lista de predecessores com o valor None. Na sequência, inicializamos o peso do vértice de início em 0 e adicionamos ele na lista de vértices a serem avaliados.

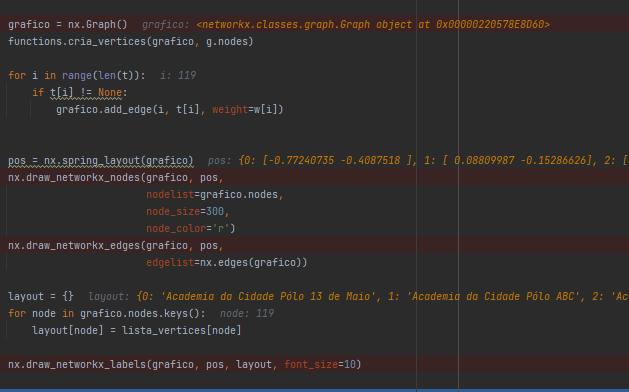
Na sequência, fazemos um laço que, enquanto a lista de vértices a serem avaliados não for vazia, ele deverá carregar a lista uma sequência de vizinhos e pesos associados a cada uma de suas ligações.

Figura 4 - Algoritmo de Prim utilizado



Após obtermos os predecessores e seus respectivos pesos, criamos um grafo para realizarmos a plotagem de seus resultados. Para isso, utilizamos novamente a biblioteca networkx.

Figura 5 - Algoritmo de criação da plotagem para o usuário final.



Por fim, salvamos o arquivo na pasta diretório com o caminho “Figure/grafo”

**Referências**

* Recife, Dados; Rede de atenção à saúde no Recife; Dados recife;. julho 2022; Disponível em: <http://dados.recife.pe.gov.br/dataset/rede-de-atencao-a-saude-no-recife/resource/d05f6ffa-304b-4a28-bd03-1ffb26cbf866>. Visualizado em: 18 de fevereiro de 2024
* Abreu, Liosvaldo M. S; IF969\_Algoritmos\_e\_estruturas\_de\_dados\_projetinho\_6; Github.com, fevereiro 2024; Disponivel em: https://github.com/liosvaldo/IF969\_Algoritmos\_e\_Estruturas\_de\_Dados. Vizualizado em: 18 de fevereiro de 2024.