

LOCALIZAÇÃO ESTRATÉGICA PARA AÇÕES POLICIAIS EM RODOVIAS: UMA ABORDAGEM COM P-MEDIANAS E HISTÓRICO DE ACIDENTES

Francisco José de Oliveira Junior*
Diego Rocha Lima**

RESUMO

A frota de veículos e o número de pessoas no trânsito cresce a cada dia, contribuindo para o aumento de acidentes nas vias urbanas e nas rodovias. Apontados como principais causas se têm os fatores humanos. A fim de combater esse problema, a educação para o trânsito surge como ferramenta essencial, visando a formação de cidadãos mais conscientes e responsáveis, por meio de práticas preventivas. Assim, o presente trabalho utiliza-se do problema das p-medianas para obter localizações estratégicas onde possam ser realizadas práticas preventivas em períodos de maior ocorrência de acidentes, usando como base os dados de acidentes rodoviários dos últimos seis anos. Para isso, o problema foi modelado utilizando-se da ferramenta *Pyomo*, a fim de resolvê-lo pelo *solver Open Source GLPK*. Desse modo, obtendo os pontos estratégicos que mostram a capacidade resolutiva do modelo em diferentes instâncias e estados, nas quais foram geradas imagens ilustrativas para melhorar a compreensão e visualização de resultados.

Palavras-chave: Otimização. P-medianas. Acidentes Rodoviários. Pyomo.

ABSTRACT

The fleet of vehicles and the number of people in traffic are growing every day, contributing to an increase in accidents on urban roads and highways. The main causes are human factors. In order to combat this problem, traffic education has emerged as an essential tool, aimed at training more aware and responsible citizens through preventive practices. This paper uses

*Graduando em Bacharelado em Ciência da Computação, Instituto Federal de Educação, Ciência e Tecnologia do Ceará(IFCE), Aracati, Ceará, Brasil. E-mail: francisco.jose.oliveira10@aluno.ifce.edu.br.

**Doutor em Engenharia da Computação (UFRN), Mestre em Ciência da Computação (UFERSA), Graduado em Ciência da Computação (UERN) e Docente do Instituto Federal de Educação, Ciência e Tecnologia do Ceará (IFCE), Aracati, Ceará, Brasil. E-mail: diego.rocha@ifce.edu.br.

the p-median problem to obtain strategic locations where preventive practices can be carried out during periods of higher accident occurrence, based on road accident data from the last six years. To do this, we intend to model the problem using the Pyomo tool, in order to solve it using the GLPK Open Source solver. In this way, strategic points will be obtained that show the model's resolving capacity in different instances and states, in which illustrative images have been generated to improve understanding and visualization of the results.

Keywords: Optimization. P-medians. Road accidents. Pyomo.

1 INTRODUÇÃO

A frota de veículos e o número de pessoas no trânsito cresce a cada dia, contribuindo para o aumento de acidentes nas vias urbanas e nas rodovias. Nesse contexto, segundo a Polícia Rodoviária Federal (PRF), somente em 2024, foram registrados 73.156 sinistros de trânsito – que podem ser caracterizados como colisões frontais, laterais, traseiras, atropelamentos de animais e pessoas – resultando em 6.160 mortos e 84.526 feridos, apenas nas rodovias federais brasileiras (Peduzzi, 2025). Esses números só aumentam na maioria dos feriados prolongados, como o carnaval e a semana santa, que sozinhos apresentam um total de quase três mil acidentes espalhados por todo o Brasil no ano citado anteriormente (Brasil, 2025).

De acordo com a Polícia Rodoviária Federal (Brasil, 2025), cerca de 90% dos acidentes nas rodovias federais estão relacionados a fatores humanos, isto é, excesso de velocidade, ultrapassagem indevida, alcoolemia, uso de celular ao volante e desrespeito à sinalização. Todavia, também existem outros tipos de fatores causadores, não só os veiculares, que podem ser representados pela falta de manutenção do veículo e carga mal-acondicionada. Mas também, os de infraestrutura, tendo como exemplo as estradas em más condições e a falta de sinalização adequada. Além dos ambientais, sendo a pista escorregadia sua principal causa. Logo, todos esses fatores também influenciam em possíveis acidentes, mas representam uma parcela menor das ocorrências comparado aos causados pelas ações do homem.

Desse modo, percebe-se que diante dos dados apresentados a subjetividade humana ainda é o principal risco nas estradas brasileiras. Perante esse cenário, a PRF realiza operações

em períodos de maior movimento, como feriados prolongados, para tentar amortecer os índices de acidentes. Para exemplificar, pode-se citar a Operação Carnaval de 2025, que teve foco no combate à embriaguez ao volante. Em virtude disso, houve uma queda nas principais estatísticas relacionadas a acidentes viários em comparação ao ano anterior, obtendo uma diminuição de 7,5% de sinistros no trânsito (Brasil, 2025). No entanto, o índice de detidos por embriaguez ao volante aumentou, indicando maior fiscalização ou uma fiscalização mais efetiva, porém, o número de mortos e feridos nas rodovias continuam sendo altos.

A fim de combater esse problema, a educação para o trânsito surge como ferramenta essencial, visando a formação de cidadãos mais conscientes e responsáveis, seja como pedestre, ciclista, motociclista ou motorista. Essas práticas são respaldadas por bases legais, como o art. 74 do Código de Trânsito Brasileiro (Brasil, 1997), que estabelece que a educação para o trânsito é um direito de todos e dever prioritário para todos os componentes do Sistema Nacional de Trânsito (SNT). Em suma, isso significa que todos os órgãos e entidades do SNT têm a obrigação de promover a educação para o trânsito. Além disso, a Política Nacional de Trânsito (Brasil, 2014) inclui a educação como um dos pilares para a redução de acidentes. Essas ações educativas podem ser classificadas em práticas continuadas, que consistem em campanhas de conscientização e programas em escolas, ou práticas preventivas, ou seja, tipos de fiscalizações educativas, como também abordagens, sejam elas informativas ou punitivas.

Apesar do trabalho de fiscalização das autoridades competentes, os dados apresentados revelam um cenário alarmante, que, embora necessárias, as práticas preventivas mostram-se insuficientes para mudar o comportamento de risco causado pelo fator humano. Isto posto, surge o seguinte problema: Como a educação para o trânsito pode alcançar de maneira mais efetiva a população com políticas de fiscalização dentro das estradas, com o objetivo de reduzir acidentes, sendo que a maioria deles decorre das escolhas humanas? Em síntese, o problema reside na dificuldade de transformar a educação em prática, especialmente em uma sociedade onde a imprudência é tolerada. Se, por um lado, a legislação prevê práticas educativas, por outro, os dados expõem que seu alcance é insuficiente para resolver o problema. Como, então, melhorar a utilização de práticas preventivas para reduzir acidentes evitáveis?

Desse modo, o presente trabalho propõe sugerir pontos estratégicos para práticas preventivas em períodos de maior ocorrência de acidentes através de um modelo de otimização combinatória para determinar as melhores localizações para as fiscalizações

educativas, ou punitivas. Com esse propósito a metodologia baseia-se na análise e manipulação de dados abertos da PRF sobre acidentes, utilizando-se de ferramentas como a linguagem de programação *Python* e o pacote de código aberto *Pyomo*, que é capaz de suportar um conjunto diversificado de competências para formular, resolver e analisar modelos de otimização, a fim de realizar a aplicação de um modelo baseado no problema das p-medianas.

O problema das p-medianas é um problema de otimização combinatória e localização de instalações, estudado no campo da Pesquisa Operacional, foi formalizado por Hakimi (1964, 1965), que formalizou um modelo matemático para localizar p-centros em uma rede, representada por um grafo não orientado, de modo a minimizar a soma das distâncias entre os pontos de demanda e a instalação mais próxima. Além disso, Hakimi apresentou algoritmos eficientes para determinar as soluções ótimas de localização em grafos com pesos nos vértices e arestas. Ademais, deve-se destacar a contribuição de Revelle e Swain (1970) que introduziram uma formulação em programação inteira para o problema, tornando-o mais acessível para soluções computacionais da época. Estas abordagens matemáticas oferecem uma estrutura robusta para otimizar a alocação de recursos de fiscalização, potencializando seu impacto na redução de acidentes.

Desta forma, este trabalho se utiliza de dados geográficos reais para obtenção de localizações estratégicas nas estradas federais brasileiras, geradas a partir do modelo das p-medianas, implementados em tecnologias *Open Source* na plataforma *Google Colab*. Além disso, o modelo foi testado para diferentes instâncias de pontos e estados, mostrando a capacidade do mesmo. Como resultado foram geradas tabelas com as localizações encontradas e imagens que as ilustram.

Sendo assim, este trabalho está organizado da seguinte maneira: a Seção 2 apresenta os principais conceitos que embasam teoricamente esse trabalho; já, a Seção 3 apresenta alguns trabalhos relacionados a temática deste estudo; a Seção 4 descreve a metodologia usada para o desenvolvimento do presente trabalho; na Seção 5 são apresentados os resultados e na Seção 6 está contida a consideração final. E, por último, estão as referências bibliográficas.

2 FUNDAMENTAÇÃO TEÓRICA

Nesta seção serão apresentados os principais conceitos necessários para o entendimento da proposta do presente trabalho, bem como seus objetivos, sua organização e as características pertencentes a cada classe de problemas e algoritmos citados adiante.

2.1 Problemas de localização

Os problemas de localização são uma classe de problemas de Otimização Combinatória que tratam de decisões sobre onde instalar unidades de serviços, ou facilidades, supondo que as demandas, ou clientes, devem ser atendidos de forma a otimizar algum critério, segundo Lorena *et al* (2001 *apud* Drezner, 1995). O termo “facilidades”, no contexto desse problema, pode ser interpretado como um conjunto de instalações, que podem ser hospitais, fábricas ou escolas, enquanto o termo clientes se referem a pacientes, compradores, ou estudantes. No entanto, o problema de localização é mais abrangente do que apenas a localização de instalações, pois do ponto de vista da logística, outros aspectos costumam ser colocados como a quantidade, a capacidade, a funcionalidade, as fontes de abastecimento, as regiões de atendimento, entre outros, como é mostrado por Pizzolato, Raupp e Alzamora (2012).

Os primeiros trabalhos robustos foram produzidos em meados do século XX, cabendo destaque ao formulado por Hakimi (1964, 1965), onde é mostrado o problema dos p -centros que tem objetivo de minimizar a distância máxima entre clientes e um número fixo de instalações, hoje considerada uma variante do problema das p -medianas. Esse trabalho despertou o interesse pelos estudos de localização de pontos de serviços em redes, sendo literatura fundamental para a área. Outro estudo basilar nesse conjunto de problemas foi feito por Revelle e Swain (1970), que apresenta uma solução em programação inteira para o problema das p -medianas. A partir desses trabalhos, os estudos de localização foram modelados de diferentes maneiras, cada um com suas particularidades, mas tendo como objetivo minimizar a distância entre um conjunto de pontos de demanda e um, ou mais, pontos de instalações, podendo levar em consideração o custo, área de cobertura ou tempo.

Entretanto, os problemas de localização não estão restritos apenas à localização ótima, melhor localização possível ou factível, de uma fábrica para atender uma demanda de clientes e, tampouco, um centro de distribuição. Como também existem muitos outros contextos em

que os problemas de localização são modelados, tais como, a quantidade de ambulâncias para atender uma demanda populacional no menor tempo possível, a quantidade de poços de petróleo ligados a uma plataforma marítima de modo a minimizar os custos de extração durante a vida útil dos poços, problemas de localização de *hubs* de transporte aéreo, entre muitos outros.

Devido a sua natureza, os problemas de localização em uma rede frequentemente são modelados como problemas que ocorrem em árvores ou exigem formulação em grafos não orientados. Assim, uma árvore é uma rede na qual existe, no máximo, um caminho entre quaisquer dois nós dela. Geralmente, para esse tipo de situação é usada uma árvore geradora, que tem exatamente um caminho entre qualquer par de nós. Para Daskin (2011, p. 12), as árvores são escolhas mais interessantes por terem uma alta capacidade de representação de aplicações práticas, mas, principalmente, pela facilidade computacional de resolução desses problemas. No caso dos grafos, existem problemas de localização que são melhores representados quando modelados por eles, porém, como demonstrado por Kariv e Hakimi (1979), problemas de localização em grafos são NP-difíceis, enquanto para árvores são redutíveis para serem resolvidos em tempo polinomial, segundo Garey e Johnson (1979).

Dentro da teoria da complexidade, existem fatores mensuráveis, por exemplo o tempo, memória, ou outro recurso computacional, podendo ser investigados para definir uma classe ou conjunto de problemas. O termo NP-difícil é usado para problemas que podem, ou não, serem verificados em tempo polinomial, porém, caso não, podem ser demonstrados como sendo, ao menos, tão difícil quanto qualquer problema conhecido, por meio de uma redução de tempo polinomial. Os problemas que podem ser verificados em tempo polinomial são os NP e os que não são solucionáveis, mas podem ser redutíveis a tempo polinomial são os NP-completos, assim como dito por Sipser (2007, p. 261 – 300).

2.2 Problema das p-medianas

O modelo matemático das p-medianas é, sem dúvidas, um dos mais utilizados quando se trata de determinar a localização de atividades, instalação de fábricas, postos de serviço, centro de distribuição e muitos outros, de modo que o seu nome é, muitas vezes, sinônimo de problemas de localização como um todo. Sua modelagem tem como objetivo obter p medianas, ou pontos, para localizar uma ou mais instalações (facilidades) de modo a

minimizar as distâncias entre os clientes (pontos de demanda) às instalações disponíveis. Para Pizzolato, Raupp e Alzamora (2012), o uso deste modelo é apropriado nos casos em que cada cliente seguidamente percorre a distância entre a sua origem e o local que oferece o serviço, tornando-se conveniente minimizar o somatório das tais distâncias, seja para economia de combustível, tempo ou outro tipo de recurso.

Dada a sua importância, o problema das p-medianas serviu de base para implementação de novos modelos que acabaram se tornando variantes desse mesmo problema. Para exemplificar, podemos destacar o problema das p-medianas capacitados, no qual cada uma das possíveis instalações possuem uma capacidade de atendimento para os pontos de demanda. Outro exemplo é o problema de localização com cobertura completa, em que esse modelo tem como o objetivo de localizar o número mínimo de instalações necessárias para atender a todos os pontos de demanda. Ao contrário do último citado, outra variante é o problema de localização com cobertura máxima, que pretende minimizar o número de instalação de modo a atender toda a demanda, diminuindo o número de instalações, porém maximizando o atendimento por demanda de cada uma.

Outro modelo que merece destaque, devido a sua importância histórica é o problema de localização de uma planta simples, sendo o nome dado aos primeiros modelos de pesquisa operacional de localização e transportes de serviço. Esse modelo foi pensado para ser usado em um ambiente industrial, onde propõe-se localizar os melhores pontos de instalação de uma ou mais fábricas de modo a abastecer toda a demanda de um conjunto de centros de consumo.

2.2.1 Definição do modelo

De acordo com Pizzolato, Raupp e Alzamora (2012), sejam $N = \{1, \dots, n\}$ o conjunto de pontos de demanda, $i \in N$ um determinado cliente ou vértice; $j \in N$ uma facilidade em potencial ou mediana; p o número de instalações ou medianas a serem localizadas; w_i o peso ou importância do cliente i ; $[d_{ij}]_{n \times n}$ a matriz simétrica de distâncias de cada cliente i à instalação j , com $d_{ii} = 0, \forall i$; $[x_{ij}]_{n \times n}$ a matriz de alocação de cada cliente i , onde $x_{ij} = 1$ se o cliente i é alocado à instalação j e $x_{ij} = 0$, caso contrário; $x_{jj} = 1$ se a facilidade for aberta no local j , $x_{jj} = 0$, caso contrário.

Então o modelo da p-mediana é apresentado da seguinte maneira:

$$\text{minimize } \sum_{i \in N} \sum_{j \in N} w_i d_{ij} x_{ij} \quad (1)$$

$$\text{sujeito a } \sum_{j \in N} x_{ij} = 1, \forall i \in N \quad (2)$$

$$\sum_{j \in N} x_{ij} = p \quad (3)$$

$$x_{ij} \leq x_{ji}, \forall i, j \in N \quad (4)$$

$$x_{ij} \in \{0, 1\}, \forall i, j \in N \quad (5)$$

A função objetivo (1) minimiza as distâncias entre os clientes e as instalações, levando em consideração o peso de cada cliente. A restrição (2) indica que cada cliente i só pode ser alocado em uma facilidade j . A próxima restrição (3) garante que somente p instalações podem oferecer um determinado serviço. Já a restrição (4) implica que um cliente só é atendido em um local onde existe uma facilidade que oferece o serviço e, por fim, a restrição em (5) garante que a variável de decisão é binária.

Desse modo, assim como dito em Arenales(2007, p. 200), o modelo matemático das p -medianas pode ser formulado por uma linguagem de programação inteira binária, ou seja, ele é discreto, uma vez que trabalha com parâmetros finitos, e suas variáveis de decisão pertencem ao conjunto $\{0, 1\}$.

2.3 Métodos de Solução

Com o passar dos anos, o avanço dos estudos do problema das p -medianas geraram uma variedade de soluções para esse modelo. De acordo com Pizzolato, Raupp e Alzamora (2012, *apud* Reese, 2006) os métodos de solução para o problema de localização são classificados como algoritmos de aproximação, de enumeração, baseados em Teoria dos Grafos, heurísticas, baseados em formulações inteiras e reduções, em relaxação linear, relaxação *surrogate* e metaheurísticas, como busca em vizinhança variável (VNS), algoritmos genéticos, e redes neurais.

Consequentemente, devido à quantidade e variedade existente de métodos de solução para resolver esse problema, apenas alguns serão abordados adiante. Além disso, métodos heurísticos se baseiam em regras simples, tornando seu desenvolvimento mais fácil. Embora

não garantam uma solução ótima, porém factível, sendo que um método exato, consegue garantir a melhor solução possível, em troca de um tempo de execução presumidamente muito maior que um método heurístico.

2.3.1 Algoritmo guloso

Os algoritmos gulosos são heurísticas usadas para resolver problemas de otimização nos quais é interessante chegar a um melhor conjunto de soluções locais, com o objetivo de obter uma solução ótima global. De modo geral, são fáceis de implementar e rápidos como métodos de solução, pois vão sempre escolher a solução mais promissora por iteração, nunca reconsiderando a decisão, de modo que não há volta para camadas superiores, *backtracking*. Seu nome em inglês ilustra bem sua natureza, já que *greedy algorithm* pode ser traduzido como “algoritmo ganancioso”. Tratando-se do contexto de localização, sua resolução se dá pela escolha das medianas que minimizam a soma das distâncias entre os pontos de demanda e a mediana mais próxima, sempre de maneira iterativa. Contudo, existem algumas considerações a serem feitas. Apesar de serem simples em implementação, são sensíveis a ordem de seleção e podem não encontrar uma solução ótima global, além de sua performance piorar com o aumento de instâncias, sendo recomendado para problemas com instâncias pequenas e médias.

2.3.2 Heurística de Teitz e Bart

A heurística de Teitz e Bart é amplamente utilizada como método resolutivo para problemas de localização devido sua facilidade de implementação e eficiência computacional, tendo bom desempenho em muitas instâncias. Resumidamente, essa heurística se baseia na substituição de vértices tendo como objetivo o melhoramento do valor da função objetivo a cada iteração, a partir de uma solução inicial dada. Segundo Silva, Konrath e Bornia(2021), a heurística de Teitz e Bart para problemas de localização pode ser representada como um algoritmo de quatro passos que serão descritos a seguir.

Dado um conjunto de pontos de demanda e suas distâncias entre si, temos:

- Passo 1. É escolhido um conjunto de p instalações, para formar uma aproximação inicial de uma solução;

- Passo 2. Verifique se existe um ponto p que não pertence ao conjunto solução que ao substituir algum ponto pertencente melhore o valor da função objetivo, no caso, minimize o valor das distâncias;
- Passo 3. Caso seja encontrado um ponto p que minimize o valor da função objetivo, substitua-o;
- Passo 4. Se um ponto p for acrescentado ao conjunto solução minimizando o valor da função objetivo, volte ao Passo 2. Caso contrário, pare e apresente o conjunto solução encontrado como uma aproximação para a solução do problema das p -medianas.

É importante destacar que uma heurística gulosa pode ser usada para obtenção de um conjunto solução inicial e posteriormente refinada com o uso do algoritmo de Teitz e Bart para obtenção de uma solução ótima.

2.3.3 Branch-and-bound

Em relação aos métodos exatos, o *Branch-and-bound* (B&B) é um algoritmo de busca enumerativa mais usado para resolver problemas de otimização discreta e combinatória. Para Pizzolato, Raupp e Alzamora (2012), sua metodologia é baseada em enumerações implícitas, onde sucessivos limites para valores da função objetivo são testados, de modo a reduzir o número de enumerações, assim, garantindo uma solução ótima. Porém, devido à necessidade de enumerar as combinações possíveis, esse algoritmo acaba sendo viável apenas para instâncias pequenas e médias. Logo, sua estrutura de decisão é formada por uma árvore de busca que divide o problema em subproblemas menores e depois calcula os limites inferiores, no caso de problemas de minimização, como é o caso do problema das p -medianas, ou os limites superiores, no caso de maximização, de cada subproblema. À medida que avança, o subproblema que não contém a solução ótima é eliminando o “ramo”.

2.3.4 Método lagrangeano

O método lagrangeano, ou relaxação lagrangeana, é uma técnica de otimização que simplifica problemas difíceis com muitas restrições complexas. A princípio, são identificadas

as restrições complexas e “relaxadas” por meio de um multiplicador lagrangeano que é incorporado à função objetivo original. Desse modo o problema é dividido em outros subproblemas independentes, que apresentam, além das variáveis do problema original, novas variáveis, os multiplicadores. Para contornar isso, são utilizados algoritmos de otimização subgradientes para ajustar essas novas variáveis. De todo modo, os subproblemas gerados após a relaxação lagrangeana – nome do processo de associar as restrições complexas à função objetivo por meio de uma nova variável – tendem a ser mais simples que o original.

Para melhor solução de problemas, o método lagrangeano pode ser usado em conjunto com outros métodos exatos, como sendo uma sub-rotina de algoritmos B&B. Da mesma forma que pode ser usado para comparar heurísticas por meio dos limites inferiores de qualidade gerados, além de ser viável para instâncias médias e grandes. Entretanto, mesmo após dividir o problema em subproblemas, uma solução relaxada pode ser inviável.

2.4 Solver

O *solver*, resolvidor em português, é uma ferramenta matemática que pode ser apresentada na forma de um *software*, ou de uma biblioteca. É usada para resolver problemas de otimização realizando testes de hipóteses para garantir uma solução ótima, ou razoavelmente boa. Logo, um solver recebe uma série de parâmetros matemáticos, que estão divididos em três tipos, para realizar sua rotina de decisão. Assim, o primeiro tipo são as variáveis de decisão que consistem em incógnitas que determinarão a solução do problema. Bem como, a função objetivo é a função que será maximizada ou minimizada de acordo com a necessidade do problema, dependendo da variável de decisão. E, por fim, as restrições, que são expressões matemáticas que limitam os possíveis valores das variáveis de decisão.

Usando o modelo das p-medianas formulado anteriormente como exemplo, temos x_{ij} como a variável de decisão do modelo, enquanto a função objetivo é representada pela equação (1), já em (2), (3), (4) e (5) são apresentadas as restrições do problema.

Atualmente, existem diferentes *solvers* que têm capacidade de resolver diferentes tipos de problemas. Tais quais, problemas de otimização linear, não linear, problemas não convexos e, devido a praticidade e robustez, o próprio *software* tem a aptidão de escolher qual método usar mediante análise do problema. A maioria dos *solvers* que estão presentes no mercado

foram criados por empresas privadas e para serem utilizados necessitam de uma licença de uso, seja ela paga ou não. Os mais difundidos são o *IBM CPLEX*, *FICO Xpress* e o *Gurobi*. No entanto, existem aqueles que foram criados por uma comunidade, de modo *Open source*, e disponibilizado para o uso de maneira gratuita, como o *GLPK* e o *Open Solver*, que pode ser utilizado em conjunto com programas de planilhas eletrônicas, como o *Microsoft Excel* e o *Google Sheets*.

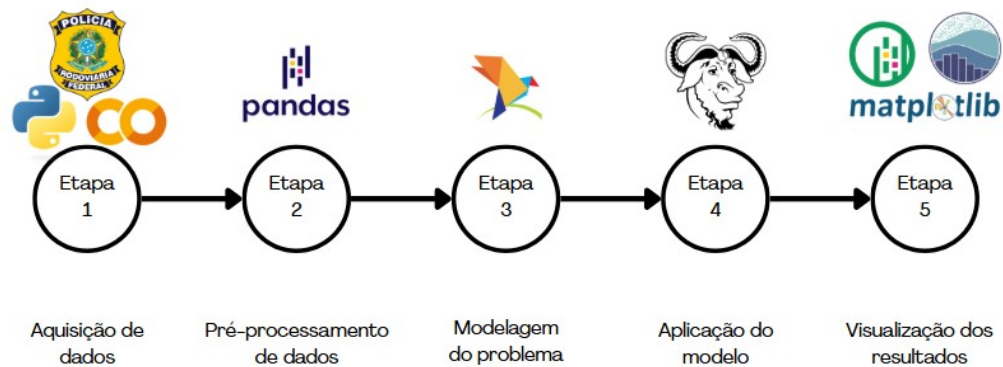
4 METODOLOGIA

Nesta seção são apresentadas todas as etapas da metodologia utilizada para a construção do presente trabalho. De modo geral, esse estudo utiliza da base teórica já apresentada para modelar um problema de localização com dados reais, usando as tecnologias disponíveis para condução do trabalho, objetivando encontrar os melhores pontos para realização de práticas preventivas nas estradas em período de maior ocorrência de acidentes, buscando ser uma ferramenta para uso da autoridade decisora. A Figura 1 apresenta o fluxo das principais etapas do processo de desenvolvimento e execução, bem como os meios usados em cada uma.

4.1 Aquisição de dados

Para a formulação deste trabalho foram utilizados dados institucionais da Polícia Rodoviária Federal, disponíveis em Brasil (2025), sendo classificados como dados abertos, sem restrições de licença e podendo ser utilizados de diferentes formas, seja pelo governo ou pela sociedade civil como um todo. Sua disponibilidade e acesso são garantidas pela Lei de Acesso à Informação (Brasil, 2011), Instrução Normativa SLTI nº 4, de 13 de abril de 2012 (Brasil, 2012), que institui a Infraestrutura Nacional de Dados Abertos, e pelo Decreto nº 8777, de 11 de maio de 2016 (Brasil, 2016), que institui a Política de Dados Abertos no Executivo Federal.

Figura 1 — Fluxo base da metodologia



Fonte: Elaborado pelos autores (2025)

Os dados estão organizados em arquivos *CSV* que são um tipo de arquivo de texto que armazena dados em formato tabular, tal qual uma planilha, onde cada linha representa um registro que possui um ou mais valores atrelados a ele, separados por algum tipo de delimitador, seja uma vírgula, ponto ou ponto e vírgula. Devido a sua simplicidade, esse formato de arquivos acaba por ser uma boa opção para armazenamento de um grande número de dados, sendo fácil de ler e criar. Para isso, podem ser utilizados editores de planilhas, banco de dados, linguagens de programação ou outros tipos de aplicações capazes de suportar esses dados estruturados.

Além disso, esses arquivos são fornecidos em formato *ZIP*, que é um método de compactação que reduz drasticamente o tamanho deles, facilitando seu armazenamento, de modo que é necessário descompactá-los para acessá-los. Já de posse dos *CSVs*, os registros contidos nesses arquivos descrevem todos os acidentes acontecidos nas rodovias federais no ano ao qual se refere, onde cada linha representa um acidente e cada coluna, das trinta existentes, apresentam um dado relativo ao acidente como horário, estado, rodovia, latitude, longitude ou tipo de acidente.

Cabe destacar que são fornecidos três tipos de arquivos *CSV* contendo acidentes por ano. Existem os agrupados por pessoas, os por ocorrências e os agrupados por pessoas e todas as causas e tipos de acidentes. Para o uso do presente trabalho foram usados os arquivos agrupados por ocorrências.

4.2 Pré-processamento de dados

Os dados utilizados foram trabalhados utilizando a linguagem de programação *Python 3* através da plataforma *Google Colab*. Visto que, *Python* é uma linguagem de programação de alto nível, interpretada e orientada a objeto. Bem como, é considerada versátil e amplamente utilizada na área de análise e ciência de dados. Por outro lado, a plataforma *Google Colab* é um ambiente de desenvolvimento integrado online, capaz de executar códigos em *Python*, *R* e *Julia*, diretamente do navegador, sem a necessidade de instalar o *software* no computador.

Antes de tudo, foi criado um código capaz de carregar os arquivos fornecidos pela PRF dos anos de 2020 a 2025, que contém todas as ocorrências acontecidas em rodovias federais brasileiras nesse período. Em seguida foram descompactados, gerando seis arquivos *CSV*, um para cada ano, e posteriormente convertidos em um único *Dataframe*, que consiste em uma estrutura de dados bidimensional, semelhante a uma tabela. Para tal fim, foi utilizada a biblioteca de *software* para a linguagem *Python* chamada *Pandas*, bastante utilizada para manipulação e análise de dados.

Desse modo, foi obtido uma estrutura de dados tabular com mais de 350 mil linhas no período entre janeiro de 2020 a abril de 2025, onde cada linha representa um acidente. Posteriormente, os dados foram filtrados em busca dos acidentes acontecidos em três feriados distintos, dada a sua importância cultural e por serem feriados prolongados fixos no ano, que são: Carnaval, Semana Santa e os feriados de Fim de Ano. Cabe destacar que os feriados de Fim de Ano agrupam o natal e o réveillon. Ademais, fins de semana também foram agrupados nesses feriados, caso pertencessem a eles. Em seguida, foram separados em três *Dataframes* menores, um para cada feriado, de modo a serem processados pelo modelo posteriormente.

4.2.1 Matriz de distâncias

Para a aplicação em um modelo de localização é necessário fornecer uma matriz de distâncias com o intuito de produzir pontos que as minimizem entre si. A matriz de distâncias consiste em uma matriz quadrada cuja as entradas estão na forma a_{ij} , onde a representa a distância entre os pontos i e j , de modo que, quando $i = j$, a distância entre eles é zero, pois não há distância entre um ponto e ele mesmo.

No contexto deste trabalho, foi utilizado a classe *DistanceMetric* da biblioteca *scikit-learn* para calcular as distâncias entre os pontos de acidentes, assim, fornecendo uma matriz

de distâncias. A Figura 2 apresenta o código usado para a construção da matriz.

A princípio, foi necessário definir o tipo de métrica utilizada para calcular as distâncias, que nesse caso foi definida como euclidiana usando o método *get_metric*. Em seguida, a latitude e longitude de cada ponto de acidente foram convertidas em unidades numéricas e, por fim, as distâncias foram calculadas usando o método *pairwise*. Após isso, os dados foram convertidos em um *Dataframe* para melhor manipulação posterior.

Figura 2 — Código de criação da matriz de distâncias.

```
d = DistanceMetric.get_metric('euclidean')
distancias = df_acidentes[['longitude', 'latitude']].to_numpy()
distancias = d.pairwise(distancias)
distancias = pd.DataFrame(distancias)
```

Fonte: Elaborado pelos autores (2025)

4.3 Construção do modelo

O modelo das p-medianas foi utilizado como base para resolução do problema proposto no presente trabalho. No entanto, para atender a necessidade do problema foram realizadas pequenas alterações no modelo apresentado previamente por Pizzolato, Raupp e Alzamora (2012).

Primeiramente, foram utilizados dois domínios distintos para a função, onde $I = \{1, 2, 3, \dots, m\}$ representa o conjunto de pontos de acidentes nas estradas e $J = \{1, 2, 3, \dots, n\}$ o conjunto de pontos candidatos a locais de realização de práticas preventivas, também chamado de conjunto de medianas, sendo $i \in I$ um ponto de acidente e $j \in J$ um ponto candidato a localização estratégica. Em seguida, a função objetiva foi minimamente alterada, como pode ser visto no modelo matemático a seguir.

$$\text{minimize } \sum_{i \in I} \sum_{j \in J} h_i d_{ij} x_{ij} \quad (6)$$

$$\text{sujeito a } \sum_{j \in J} x_{ij} = 1, \forall i \in I \quad (7)$$

$$\sum_{j \in J} y_j = p \quad (8)$$

$$x_{ij} \leq y_j, \forall i \in I, j \in J \quad (9)$$

$$x_{ij} \in \{0,1\}, \forall i \in I, j \in J \quad (10)$$

$$y_j \in \{0,1\}, \forall j \in J \quad (11)$$

É possível notar um novo parâmetro h_i em (6), que representa a quantidade de acidentes por ponto $i \in I$. Já, os demais parâmetros continuam tendo a mesma função que o modelo original, sendo d_{ij} as distâncias entre o ponto de acidente i e o ponto candidatos j e x_{ij} a variável de decisão binária, onde $x_{ij} = 1$, quando o ponto de acidente i é alocado a mediana j e $x_{ij} = 0$, caso contrário. Além disso, foi criada uma nova variável de decisão binária y_j , que será $y_j = 1$, quando o ponto candidato j for escolhido como mediana e, caso contrário, $y_j = 0$.

As restrições pouco mudam comparado ao modelo presente em Pizzolato, Raupp e Alzamora (2012). Além disso, apenas uma restrição de domínio é adicionada garantindo que a variável de decisão y_j é binária, como pode ser visto em (11). De modo geral, apenas em (8) e (9) ocorrem modificações mínimas.

A restrição (7) garante que cada acidente i só pode ser alocado a uma mediana j . Já, a restrição em (8) garante que a quantidade de pontos candidatos a locais de prática preventivas, ou simplesmente medianas, corresponde ao número p escolhido pelo decisor. Em seguida, a restrição em (9) implica que cada ponto de acidente i só pode estar alocado a uma mediana j que tenha sido escolhida, logo, exista. E, por fim, em (10) e (11) são definidos os domínios das variáveis de decisão x_{ij} e y_j , respectivamente.

4.3.1 Implementação do modelo

A implementação computacional do modelo das p -medianas foi realizada na plataforma *Google Colab* utilizando-se de uma biblioteca chamada *Pyomo* que, segundo Carvalho e Neto (2020), é uma biblioteca de software para a modelagem de problemas de otimização baseada em *Python* que tem capacidade de formular modelos grandes com sintaxe concisa e separar a declaração entre modelos e dados.

Para sua formulação optou-se pela implementação de um modelo abstrato, também chamado de simbólico, para que os valores das variáveis e parâmetros não precisam serem expressos diretamente no modelo. A Figura 3 apresenta o código necessário para declaração

do modelo abstrato.

Figura 3 — Código de declaração do modelo abstrato.

```
modelo = pyo.AbstractModel()
```

Fonte: Elaborado pelos autores (2025)

Na Figura 4 pode ser vista a simplicidade da sintaxe de implementação do modelo, sendo semelhante a linguagem matemática usada para descrevê-lo. A princípio, são definidos os conjuntos de domínio da função objetivo, chamados de “índices” no código, usando o método *RangeSet* para apresentar a quantidade de pontos m e n pertencentes aos conjuntos I e J , respectivamente. Logo em seguida, são definidos os parâmetros d , h e p , por onde passarão, respectivamente, a matriz de distâncias, o histórico de acidentes por ponto e a quantidade de pontos escolhidos pelo decisor. Além disso, faz-se necessário inicializar dentro do método *Param* todos os conjuntos e variáveis que serão usadas por aquele parâmetro, além dos atributos necessários que serão disponibilizados quando o modelo for instanciado. Por fim, as variáveis de decisão são implementadas usando o método *Var* para atribuir os conjuntos de domínio do problema e da própria variável de decisão.

Na Figura 5 é apresentado a implementação da função objetivo e das restrições do problema. Todas são definidas como uma função padrão da linguagem *Python*, porém, com sintaxe similar à expressão matemática. Por exemplo, a função *regra_fo* é a definição em código da equação presente em (7). Pode-se notar que sua formulação é, basicamente, uma transcrição literal da expressão matemática com a adição de pequenos parâmetros obrigatórios. Ademais, é necessário declarar cada uma das funções criadas, como em *modelo.z*, e utilizar o método adequado, seja para a função objetivo (*Objective*) ou restrição (*Constraint*). Além disso, os métodos precisam dos atributos que foram utilizados na definição da função. No entanto, destaca-se um atributo essencialmente necessário chamado *rule* que o conecta à declaração com a função definida previamente.

Figura 4 — Código de declaração dos principais parâmetros necessários

```

# Índices
modelo.I = pyo.RangeSet(modelo.m)
modelo.J = pyo.RangeSet(modelo.n)

# Parâmetros
modelo.d = pyo.Param(modelo.I,
                      modelo.J,
                      initialize=lambda modelo,
                      i,
                      j: modelo.distancias()[i-1][j-1],
                      mutable=True)
modelo.h = pyo.Param(modelo.I,
                      initialize=lambda modelo,
                      i: modelo.historico()[i-1])
modelo.p = pyo.Param()

# Variáveis de Decisão
modelo.y = pyo.Var(modelo.J, within=pyo.Binary)
modelo.x = pyo.Var(modelo.I, modelo.J, within=pyo.Binary)

```

Fonte: Elaborado pelos autores (2025)

Figura 5 — Código de declaração da função objetivo e das restrições

```

# Função objetivo:
def regra_fo(mod):
    return sum(mod.h[i] * mod.d[i,j] * mod.x[i,j] for i in mod.I for j in mod.J)

modelo.z = pyo.Objective(rule=regra_fo) # Default: minimize

# Restrições:
def regra_restr1(mod):
    return sum(mod.y[j] for j in mod.J) == mod.p

modelo.restr_1 = pyo.Constraint(rule=regra_restr1)

def regra_restr2(mod, i):
    return sum(mod.x[i,j] for j in mod.J) == 1

modelo.restr_2 = pyo.Constraint(modelo.I, rule=regra_restr2)

def regra_restr3(mod, i, j):
    return mod.x[i,j] <= mod.y[j]

modelo.restr_3 = pyo.Constraint(modelo.I, modelo.J, rule=regra_restr3)

```

Fonte: Elaborado pelos autores (2025)

4.4 Aplicação do modelo

Para que fosse possível utilizar o modelo citado em 4.3.1 foi necessário fazer com que os dados adquiridos se encaixassem em parâmetros idênticos aos utilizados nos métodos do modelo abstrato. A Figura 6 apresenta o dicionário utilizado para alocar os dados que posteriormente seriam usados para instanciar o modelo.

É importante salientar que o parâmetro *ponto_interesse* presente na Figura 6 é justamente a quantidade de pontos de prática preventivas que o modelo busca otimizar. Comparado a função matemática, esse parâmetro é o ponto p que cabe ao decisor instanciar dependendo de sua necessidade. Em seguida, foi necessário converter os dados do dicionário na classe *DataPortal* usada pelo *Pyomo*, conforme a Figura 7, para padronizar o processo de construção do modelo com o propósito de carregar diferentes dados de maneira uniforme.

Figura 6 — Dicionário de dados

```
params_instancia = {
    "m": m,
    "n": n,
    "distancias": distancias,
    "historico": historico,
    "p": ponto_interesse
}
```

Fonte: Elaborado pelos autores (2025)

A Figura 7 mostra a criação e atribuição de valores da classe *DataPortal* feito por um laço que atribui a cada uma das chaves da classe uma do dicionário presente na Figura 6.

Figura 7 — Conversão em *DataPortal*

```
data_portal_instancia = pyo.DataPortal()
for key in params_instancia:
    data_portal_instancia[key] = {None: params_instancia[key]}
```

Fonte: Elaborado pelos autores (2025)

Depois da criação da classe utilizada pelo *Pyomo* foi necessário criar uma instância do modelo abstrato utilizando o método *create_instance* passando como atributo os dados presentes na classe *DataPortal* declarada anteriormente. Esse processo está presente na Figura 8.

Figura 8 — Criação da instância do modelo

```
instancia = modelo_abs.create_instance(data_portal_instancia)
```

Fonte: Elaborado pelos autores (2025)

Após a criação da instância foi utilizado o método *SolverFactory* para escolher o resolvidor, ou *solver*, e o método *solve* para resolver o problema.

Figura 9 — Utilização do *solver*

```
result_instancia = pyo.SolverFactory('glpk').solve(instancia)
```

Fonte: Elaborado pelos autores (2025)

Como pode ser visto na Figura 9, o *solver* utilizado no presente trabalho foi o *GNU Linear Programming Kit*, ou simplesmente *GLPK*, que é um pacote de *software Open Source* e gratuito capaz de resolver problemas de programação linear, inteira e mista.

4.5 Visualização dos resultados

Para melhorar a visualização dos dados de saída do modelo optou-se por separar os pontos de acidentes por estados. Para isso, foi necessário utilizar malhas territoriais disponibilizadas pelo Instituto Brasileiro de Geografia e Estatística, IBGE, em arquivos do tipo *shapefile*. As malhas territoriais são representações cartográficas das estruturas territoriais, no caso do Brasil são estados, municípios, distritos e subdistritos. Enquanto isso, o formato de arquivos *shapefile* armazena dados geográficos em vetores e informações geométricas de uma determinada área, amplamente usado em Sistemas de Informações Geográficas (SIG).

Para que fosse possível visualizar os dados georreferenciados disponíveis no arquivo *shapefile* foi utilizada a biblioteca *Geopandas*. Ela permite tratar dados geográficos de maneira simples e eficiente. Desse modo, utilizou-se o método *read_file* para obter todos os dados necessários para conseguir gerar as imagens. Entre esses dados se destacam os polígonos que formam cada um dos municípios e estados escolhidos. Sendo os polígonos, nesse contexto, um conjunto de coordenadas geográficas que representam os limites de um

território, seja país, estado ou município.

Além da capacidade de trabalhar com dados georreferenciados, a biblioteca *Geopandas* possui uma boa integração com outra biblioteca chamada *Matplotlib* que, para este trabalho, foi utilizada para gerar as imagens dos mapas. Também foi usada, em conjunto com as demais citadas, a biblioteca *seaborn*, que é utilizada para visualização de dados estatísticos. No entanto, para o presente trabalho, foi utilizada para gerar os pontos de acidentes no mapa, assim como os pontos de práticas preventivas.

A seção 5 apresenta de maneira detalhada as imagens geradas para o estado do Ceará em três diferentes instâncias de pontos estratégicos. Além disso, no APÊNDICE C podem ser encontradas as demais imagens geradas.

5 RESULTADOS

Nessa seção são apresentados os resultados obtidos pela aplicação do problema das p -medianas nos dados de acidentes fornecidos pela PRF utilizando os pontos de acidentes nas rodovias federais brasileiras como pontos de demanda para o modelo criado.

Para demonstrar a capacidade resolutive do modelo foram escolhidas três diferentes instâncias de p . Também foram escolhidos três feriados fixos no ano com grande ocorrência de acidentes, que são: o Carnaval, a Semana Santa e os feriados de Fim de Ano. Ademais, foram utilizados dados de três estados turísticos que possuem um fluxo de veículos entre si como exemplo.

A Tabela 1 apresenta os dados de entrada para o estado do Ceará usados para alimentar o modelo implementado. É possível notar que os dados representam acidentes acontecidos em três janelas de tempo diferentes. Elas foram escolhidas por serem feriados fixos por ano, que possibilitam uma análise mais concisa, além de serem momentos de ocorrência acentuada de acidentes, principalmente nos feriados de final de ano. Ademais, os acidentes também foram divididos por rodovias federais de modo a melhorar a visualização.

Além dos dados do Ceará, outros dois estados foram escolhidos para serem analisados e utilizados pelo modelo, que são Rio Grande do Norte e Paraíba, dada a proximidade com o estado citado anteriormente e a capacidade turística elevada de ambos, gerando um razoável fluxo de veículos entre eles. Cabe destacar, que os dados de entrada desses dois estados encontram-se no APÊNDICE A.

Continuando a usar o estado do Ceará como exemplo, na implementação do modelo foram usadas três instâncias diferentes de p para cada feriado prolongado escolhido, cabendo destacar que a quantidade p é escolhida pelo decisor. Primeiramente, foi utilizado $p = 5$, onde foi possível, após aplicação dos dados de entrada no modelo, gerar cinco pontos otimizados de possíveis práticas preventivas. No Quadro 1 estão descritos os pontos obtidos, bem como suas coordenadas, a rodovia em que estão localizados e o município ao qual pertencem. Também é possível verificar o tempo de execução do modelo para cada um dos feriados escolhidos.

Tabela 1 — Dados de entrada para o estado do Ceará

Rodovia	Número de acidentes		
	Carnaval	Semana santa	Fim de Ano
BR-020	33	27	39
BR-116	69	39	76
BR-122	0	0	1
BR-222	58	38	70
BR-226	0	0	1
BR-230	3	1	1
BR-304	13	3	5
BR-402	2	3	6
BR-403	0	1	1
Total	178	112	200

Fonte: Elaborado pelos autores (2025)

A Figura 10 é a representação visual do Quadro 2 que consiste na diferentes instâncias de p escolhidas para o estado do Ceará nos feriados de Fim de ano. Os círculos em azul representam todos os 200 pontos de acidentes nos últimos cinco anos em estradas federais no estado citado, que serviram de dados de entrada para o modelo. Enquanto os círculos em laranjas representam os dados de saída do modelo, os mesmos presentes no Quadro 2. Também foram representados, em arcos vermelho, a ligação de cada ponto de demanda, ou pontos de acidentes, ao seu respectivo ponto de saída do modelo, que consiste na melhor mediana possível encontrada para uma quantidade p escolhida pelo usuário do modelo. À esquerda está representada a instância de $p = 5$, ao centro a instância de $p = 6$ e a direita a instância de $p = 7$.

As demais figuras geradas utilizando o modelo estão presentes no APÊNDICE C. Elas

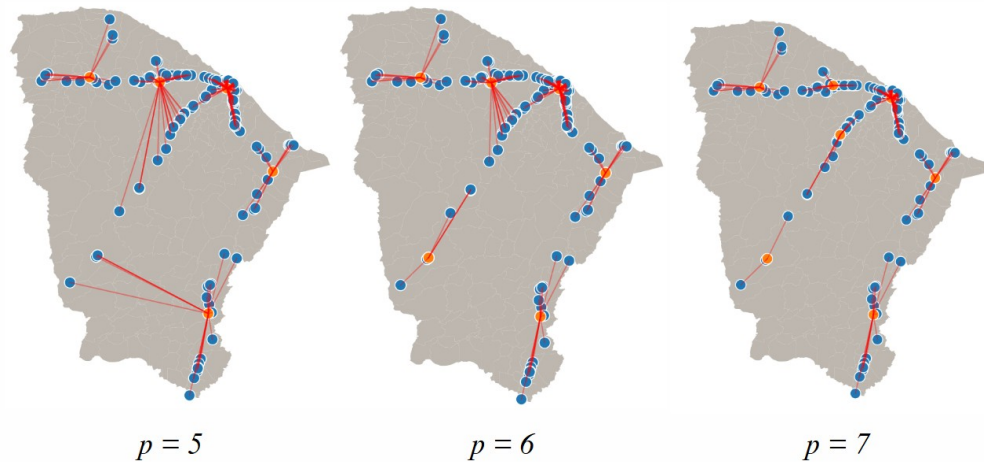
são a visualização de resultados dos estados do Ceará, Rio Grande do Norte e Paraíba, nos feriados de Carnaval, Semana santa e Fim de ano, para cada uma das instâncias de $p = 5$, $p = 6$ e $p = 7$.

Quadro 1 — Dados de saída do modelo para o estado do Ceará com $p = 5$

$p = 5$	Carnaval			Semana Santa			Fim de Ano		
	Lat., Long.	Município	Rodovia	Lat., Long.	Município	Rodovia	Lat., Long.	Município	Rodovia
1º ponto	-6.7923, -39.3079	Várzea Alegre	BR-230	-6.4055, -38.8462	Icó	BR-116	-6.7536, -38.8378	Lavras da Mangabeira	BR-230
2º ponto	-4.5790, -37.7707	Aracati	BR-304	-4.4230, -38.4032	Cascavel	BR-116	-4.9165, -37.9973	Russas	BR-116
3º ponto	-3.8774, -38.5002	Fortaleza	BR-116	-3.8286, -38.6242	Maracanaú	BR-222	-3.8408, -38.5970	Maracanaú	BR-222
4º ponto	-3.7700, -38.6701	Caucaia	BR-020	-3.7082, -40.3521	Sobral	BR-222	-3.7615, -39.4709	Itapajé	BR-222
5º ponto	-3.7069, -40.4083	Sobral	BR-403	-3.6723, -39.3117	Umirim	BR-222	-3.6969, -40.3842	Sobral	BR-222
—	Tempo de execução: 22,540s			Tempo de execução: 3,017s			Tempo de execução: 41,053s		

Fonte: Elaborado pelos autores (2025)

Figura 10 — Pontos de saída do modelo para o estado do Ceará nos feriados de Fim de ano



Fonte: Elaborado pelos autores (2025)

6 CONCLUSÃO

No presente trabalho utilizou-se de dados abertos da Polícia Rodoviária Federal, contendo informações georreferenciadas de acidentes acontecidos em rodovias federais brasileiras, para alimentar um modelo matemático de otimização com o objetivo de encontrar as melhores localizações para realização de práticas preventivas nas estradas federais brasileiras. Essas práticas podem ser campanhas educativas nas estradas ou práticas punitivas, como uma “blitz”. Para isso, foram utilizadas tecnologias *Open Source* como *Python* e *GLPK*, assim como diferentes bibliotecas para trabalhar com os dados, bem como a própria implementação do modelo. Como exemplo, foram utilizados três estados com alta capacidade turística e com fluxo interestadual de veículos para demonstrar a capacidade do trabalho.

O modelo citado utilizou como base o problema de localização das p -medianas que é um modelo clássico de obtenção de localidades da área de otimização combinatória que busca encontrar o melhor conjunto solução de pontos de localização, dado, previamente, um conjunto de pontos, ou localidades, de demanda. Para esse trabalho foram utilizados um conjunto de números $p = \{5, 6, 7\}$ de pontos a serem obtidos, para exemplificar a capacidade do modelo. Enquanto, o conjunto de pontos de demanda foi representado pela localização dos pontos de acidentes obtidos previamente. Desse modo, foi possível demonstrar a capacidade do modelo implementado de gerar diferentes soluções de acordo com a demanda de p , independente do estado da federação brasileira escolhido, para aquilo que o mesmo foi proposto.

Espera-se que esse trabalho possa auxiliar as autoridades competentes com poder decisório a escolher p localizações estratégicas para a realização de práticas preventivas nas estradas em períodos de maior ocorrência de acidentes de maneira otimizada, com o intuito de prevenção de acidentes provocados pelo fator humano, característica preponderante na causa dos mesmos, por meio de fiscalizações educativas, ou punitivas, nas melhores localizações geradas. Assim como, ajudar os estudos de Pesquisa Operacional relacionados a área de acidentes viários.

Cabe destacar, que houve limitações e dificuldades no decorrer do projeto. Apesar da facilidade de obtenção dos dados federais para acidentes em estradas, no âmbito estadual isso não se mostra parecido, existindo uma falta de periodicidade adequada, além de uma dificuldade de obtenção dos mesmos. Ademais, a utilização de métodos exatos, como o

utilizado pelo *solver GLPK*, mostra-se computacionalmente complexa para um modelo que utiliza uma alta quantidade de dados, sendo necessários diferentes abordagens para resolução do modelo.

Por fim, como trabalhos futuros, pretende-se aplicar diferentes tipos de modelo, como o problema de localização com cobertura completa para localizar a quantidade mínima necessária de pontos de práticas preventivas, assim como o algoritmo *k-means* para minimizar as distâncias entre os pontos de acidentes e os pontos p gerados pelo presente trabalho. Também pretende-se utilizar outros métodos de resolução desse modelo, como a heurística de Teitz e Bart, para possibilitar a utilização de instâncias grandes para obtenção de um conjunto de soluções viáveis.

REFERÊNCIAS

ARENALES, Marcos *et al.* **Pesquisa Operacional**. Rio de Janeiro: Elsevier, 2007.

BRASIL. Lei nº 9.503, de 23 de setembro de 1997. **Institui o Código de Trânsito Brasileiro**. Diário Oficial [da] República Federativa do Brasil, Brasília, DF, 23 set. 1997. Disponível em: https://www.planalto.gov.br/ccivil_03/leis/19503compilado.htm. Acesso em: 23 mai. 2025.

BRASIL. Lei nº 12.527, de 18 de novembro de 2011. **Regula o acesso a informações previsto no inciso XXXIII do art. 5º da Constituição Federal**. Diário Oficial [da] República Federativa do Brasil, Brasília, DF, 18 nov. 2011. Disponível em: https://www.planalto.gov.br/ccivil_03/_ato2011-2014/2011/lei/112527.htm. Acesso em: 27 mai. 2025.

BRASIL. Decreto nº 8.777, de 11 de maio de 2016. **Institui a Política de Dados Abertos do Poder Executivo Federal**. Diário Oficial da União, Poder Executivo, Brasília, DF, 12 mai. 2016. Disponível em: https://www.planalto.gov.br/ccivil_03/_ato2015-2018/2016/decreto/d8777.htm. Acesso em: 27 mai. 2025.

BRASIL. Polícia Rodoviária Federal. **Anuário 2024**. 2025. Disponível em: https://www.gov.br/prf/pt-br/acesso-a-informacao/dados-abertos/diest-arquivos/anuario-2024_final.html#S%C3%A9rie_hist%C3%B3rica. Acesso em: 24 mai. 2025.

BRASIL. Polícia Rodoviária Federal. **RODOVIDA: PRF encerra Operação Carnaval com redução de mortes, feridos e sinistros de trânsito**. 6 mar. 2025. Disponível em:

<https://www.gov.br/prf/pt-br/noticias/nacionais/rodovida-prf-encerra-operacao-carnaval-com-reducao-de-mortes-feridos-e-sinistros-de-transito>. Acesso em: 24 mai. 2025.

BRASIL. Polícia Rodoviária Federal. **Dados Abertos da PRF**. Brasília: PRF, 2025. Disponível em: <https://www.gov.br/prf/pt-br/aceso-a-informacao/dados-abertos/dados-abertos-da-prf>. Acesso em: 27 mai. 2025.

BRASIL. Resolução nº 514, de 18 de dezembro de 2014. **Dispõe sobre a Política Nacional de Trânsito**. Diário Oficial da União, Brasília, DF, 18 dez. 2014. Disponível em: <https://www.gov.br/transportes/pt-br/assuntos/transito/conteudo-contran/resolucoes/resolucao5142014.pdf>. Acesso em: 23 mai. 2025.

BRASIL. Secretaria de Logística e Tecnologia da Informação. Instrução Normativa nº 4, de 13 de abril de 2012. **Institui a Infraestrutura Nacional de Dados Abertos – INDA**. Diário Oficial da União, Poder Executivo, Brasília, DF, 13 abr. 2012. Disponível em: https://www.gov.br/agricultura/pt-br/aceso-a-informacao/dadosabertos/arquivos-raiz/in04_2012.pdf. Acesso em: 27 mai. 2025.

BRONDANI, A. Ebling *et al.* **Alocação de unidades urbanas de lazer por um modelo de p-medianas**. Revista Eletrônica Pesquisa Operacional para o Desenvolvimento, Rio de Janeiro, v. 5, n. 2, p. 209-223, ago. 2013.

CARVALHO, Claudemir W. V.; NETO, Anselmo R. P. **Manual de uso da biblioteca Pyomo para Programação Matemática**. Pesquisa Operacional em Produção e Logística, 2020.

DASKIN, Mark S. **Network and Discrete Location: Models, Algorithms, and Applications**. 2. ed. Hoboken: Wiley, 2011.

DREZNER, Z. **Facility Location: A Survey of Applications and Methods**. New York: Springer, 1995.

FREITAS, C. R. **Teoria Espectral Aplicada a Problemas de Localização**. 2012. Dissertação (Mestrado em Modelagem Computacional) – Universidade Federal do Rio Grande, Rio Grande, 2012.

GAREY, M. R.; JOHNSON, D. S. **Computers and Intractability: A Guide to the Theory of NP-Completeness**. San Francisco: W. H. Freeman, 1979.

HAKIMI, S. L. **Optimum Locations of Switching Centers and the Absolute Centers and**

Medians of a Graph. Operations Research, v. 12, n. 3, p. 450-459, 1964.

HAKIMI, S. L. **Optimum Distribution of Switching Centers in a Communication Network and Some Related Graph Theoretic Problems.** Operations Research, v. 13, n. 3, p. 462-475, 1965.

KARIV, O.; HAKIMI, S. L. **An Algorithmic Approach to Network Location Problems. Part I: The p-Center Problem.** SIAM Journal on Applied Mathematics, v. 37, n. 3, p. 513-538, 1979.

PEDUZZI, Pedro. **Acidentes nas rodovias federais matam 6,16 mil pessoas em 2024.** Agência Brasil, Brasília, 17 abr. 2025. Disponível em: <https://agenciabrasil.ebc.com.br/geral/noticia/2025-04/acidentes-em-rodovias-federais-matam-616-mil-pessoas-em-2024>. Acesso em: 24 mai. 2025.

PEREIRA, Elizangela Dias *et al.* **Localização de centros de coleta de EPS utilizando p-medianas:** uma alternativa para logística reversa do setor. In: XXVII Congresso Nacional de Pesquisa e Ensino em Transportes, 2013, Belém- PA. Anais... ANPET XXVII, 2013.

PIZZOLATO, Nelio D.; RAUPP, Fernanda M. P.; ALZAMORA, Guina S. **Revisão de desafios aplicados em localização com base em modelos da p-mediana e suas variantes.** Pesquisa Operacional para o Desenvolvimento, v. 4, n. 1, p. 13-42, dez. 2012.

RESSE, J. **Solutions Methods for the p-Median Problem:** An Annotated Bibliography. Networks, v. 48, n. 3, p. 125-142, 2006.

REVELLE, C. S.; SWAIN, R. W. **Central Facilities Location.** Geographical Analysis, v. 2, n. 1, p. 30-42, 1970.

SILVA, S. A. da; KONRATH, A. C.; BORNIA, A. C. **Otimização utilizando a heurística das p-medianas para identificar locais para instalação de institutos federais no estado de Santa Catarina.** Revista de Gestão e Projetos, v. 12, n. 1, p. 45-62, jan./mar. 2021.

SIPSER, Michael. **Introdução à Teoria da Computação.** 2. ed. São Paulo: Thomson Learning, 2007.

SOARES, L. M. Bento; SANTOS, Marcos dos. **Aplicação do método das p-medianas para otimização da localização dos hospitais de campanha na cidade do Rio de Janeiro em uma possível segunda onda de contágio do Covid-19.** In: Congresso de Administração,

Sociedade e Inovação, 20., 2021, On-line. Anais... Meio Digital: Even3, 2021.

APÊNDICE A – NÚMERO DE ACIDENTES NOS ESTADOS PESQUISADOS

Tabela 2 — Dados de entrada para o estado do Rio grande do Norte

Rodovia	Número de acidentes		
	Carnaval	Semana santa	Fim de Ano
BR-101	79	40	64
BR-110	18	4	12
BR-226	13	8	16
BR-304	45	28	28
BR-405	16	9	9
BR-406	11	10	16
BR-427	5	6	4
BR-437	1	0	0
Total	188	105	149

Fonte: Elaborado pelos autores (2025)

Tabela 3 — Dados de entrada para o estado da Paraíba

Rodovia	Número de acidentes		
	Carnaval	Semana santa	Fim de Ano
BR-101	62	27	39
BR-104	12	4	15
BR-110	0	1	0
BR-116	0	1	1
BR-230	129	47	122
BR-361	4	2	5
BR-405	0	2	3
BR-412	4	2	4
BR-427	1	0	4
BR-434	1	0	0
Total	213	86	193

Fonte: Elaborado pelos autores (2025)

APÊNDICE B – DESCRIÇÃO DOS DADOS DE SAÍDA DO MODELO

Quadro 3 — Dados de saída do modelo para o estado do Ceará para $p = 6$

$p = 6$	Carnaval			Semana Santa			Fim de Ano		
	Lat., Long.	Município	Rodovia	Lat., Long.	Município	Rodovia	Lat., Long.	Município	Rodovia

1º ponto	-7.2841, -38.862 2	Milagres	BR-116	-6.8886, -38.728 7	Ipaumirim	BR-116	-6.7536, -38.837 8	Lavras da Mangabeira	BR-230
2º ponto	-5.7284, -40.136 1	Tauá	BR-020	-5.1520, -39.748 8	Boa Viagem	BR-020	-6.0012, -40.280 3	Tauá	BR-020
3º ponto	-4.5790, -37.770 7	Aracati	BR-304	-4.4367, -38.391 4	Cascavel	BR-116	-4.9165, -37.997 3	Russas	BR-116
4º ponto	-3.8774, -38.500 2	Fortaleza	BR-116	-3.8286, -38.624 2	Maracanaú	BR-222	-3.8408, -38.597 0	Maracanaú	BR-222
5º ponto	-3.7700, -38.670 1	Caucaia	BR-020	-3.7082, -40.352 1	Sobral	BR-403	-3.7615, -39.470 9	Itapajé	BR-222
6º ponto	-3.7155, -40.444 3	Sobral	BR-222	-3.6723, -39.311 7	Umirim	BR-222	-3.6969, -40.384 2	Sobral	BR-403
—	Tempo de execução: 21,561s			Tempo de execução: 2,952s			Tempo de execução: 45,053s		

Fonte: Elaborado pelos autores (2025)

Quadro 4 — Dados de saída do modelo para o estado do Ceará para $p = 7$

$p = 7$	Carnaval			Semana Santa			Fim de Ano		
	Lat., Long.	Município	Rodovia	Lat., Long.	Município	Rodovia	Lat., Long.	Município	Rodovia
1º ponto	-7.3583, -38.9458	Milagres	BR-116	-6.8886, -38.7287	Ipaumirim	BR-116	-6.7536, -38.837 8	Lavras da Mangabeira	BR-230
2º ponto	-6.0411, -38.6459	Jaguaribe	BR-116	-5.1520, -39.7488	Boa Viagem	BR-020	-6.0012, -40.280 3	Tauá	BR-020
3º ponto	-5.7284, -40.1361	Tauá	BR-020	-4.4367, -38.3914	Cascavel	BR-116	-4.9165, -37.997 3	Russas	BR-116
4º ponto	-4.5790, -37.7707	Aracati	BR-304	-3.8873, -38.5019	Fortaleza	BR-116	-4.3353, -39.288 6	Canindé	BR-020
5º ponto	-3.8774, -38.5002	Fortaleza	BR-116	-3.7660, -38.6622	Caucaia	BR-020	-3.8408, -38.597 0	Maracanaú	BR-222
6º ponto	-3.7700, -38.6701	Caucaia	BR-020	-3.7082, -40.3521	Sobral	BR-403	-3.6969, -40.384	Sobral	BR-222

							2		
7º ponto	-3.7155, -40.4443	Sobral	BR-222	-3.6723, -39.3117	Umirim	BR-222	-3.6721, -39.3849	Umirim	BR-402
—	Tempo de execução: 18,766s			Tempo de execução: 2,326s			Tempo de execução: 39,659s		

Fonte: Elaborado pelos autores (2025)

Quadro 5 — Dados de saída do modelo para o estado do Rio Grande do Norte para $p = 5$

$p = 5$	Carnaval			Semana Santa			Fim de Ano		
	Lat., Long.	Município	Rodovia	Lat., Long.	Município	Rodovia	Lat., Long.	Município	Rodovia
1º ponto	-6.3457, -36.5872	Acari	BR-427	-6.4611, -37.1079	Caicó	BR-427	-6.0810, -37.5547	Messias Targino	BR-226
2º ponto	-5.9106, -35.2613	Parnamirim	BR-101	-5.9723, -38.1479	São Francisco do Oeste	BR-405	-5.9464, -35.2649	Parnamirim	BR-101
3º ponto	-5.7496, -35.2892	Natal	BR-101	-5.8841, -35.2829	Parnamirim	BR-304	-5.7505, -36.0511	Caiçara do Rio do Vento	BR-304
4º ponto	-5.6486, -37.7993	Apodi	BR-405	-5.6923, -36.2754	Angicos	BR-304	-5.7407, -35.2902	Natal	BR-101
5º ponto	-5.1980, -37.3184	Mossoró	BR-110	-5.2096, -37.3336	Mossoró	BR-110	-5.2034, -37.3256	Mossoró	BR-110
—	Tempo de execução: 25,511s			Tempo de execução: 6,625s			Tempo de execução: 8,689s		

Fonte: Elaborado pelos autores (2025)

Quadro 6 — Dados de saída do modelo para o estado do Rio Grande do Norte para $p = 6$

$p = 6$	Carnaval			Semana Santa			Fim de Ano		
	Lat., Long.	Município	Rodovia	Lat., Long.	Município	Rodovia	Lat., Long.	Município	Rodovia
1º ponto	-6.5703, -36.7545	Jardim do Seridó	BR-427	-6.4611, -37.1079	Caicó	BR-427	-6.0810, -37.5547	Messias Targino	BR-226
2º ponto	-5.9017, -35.2593	Parnamirim	BR-101	-6.1719, -35.2198	Arez	BR-101	-5.9374, -35.2648	Parnamirim	BR-101
3º ponto	-5.8336, -35.7362	Santa Maria	BR-304	-5.9723, -38.1479	São Francisco do Oeste	BR-405	-5.7480, -35.9418	Caiçara do Rio do Vento	BR-304
4º	-5.7280,	S. Gonçalo	BR-	-5.8738,	Macaíba	BR-304	-5.7407,	Natal	BR-10

ponto	-35.2881	do Amarante	101	-35.3150			-35.2902		1
5° ponto	-5.6486, -37.7993	Apodi	BR-405	-5.6923, -36.2754	Angicos	BR-304	-5.6018, -36.9212	Açu	BR-304
6° ponto	-5.1980, -37.3184	Mossoró	BR-110	-5.2096, -37.3336	Mossoró	BR-110	-5.1370, -37.3453	Mossoró	BR-304
—	Tempo de execução: 21,917s			Tempo de execução: 4,294s			Tempo de execução: 9,533s		

Fonte: Elaborado pelos autores (2025)

Quadro 7 — Dados de saída do modelo para o estado do Rio Grande do Norte para $p = 7$

$p = 7$	Carnaval			Semana Santa			Fim de Ano		
	Lat., Long.	Município	Rodovia	Lat., Long.	Município	Rodovia	Lat., Long.	Município	Rodovia
1° ponto	-6.5703, -36.7545	Jardim do Seridó	BR-427	-6.4611, -37.1079	Caicó	BR-427	-6.4593, -37.0985	Caicó	BR-427
2° ponto	-5.9017, -35.2593	Parnamirim	BR-101	-6.2313, -36.4380	Currais Novos	BR-226	-5.9374, -35.2648	Parnamirim	BR-101
3° ponto	-5.8503, -35.6564	São Pedro	BR-304	-6.1719, -35.2198	Arez	BR-101	-5.9232, -38.1015	Taboleiro Grande	BR-405
4° ponto	-5.7280, -35.2881	São Gonçalo do Amarante	BR-101	-5.9723, -38.1479	São Francisco do Oeste	BR-405	-5.7480, -35.9418	Caiçara do Rio do Vento	BR-304
5° ponto	-5.6486, -37.7993	Apodi	BR-405	-5.8738, -35.3150	Macaíba	BR-304	-5.7407, -35.2902	Natal	BR-101
6° ponto	-5.5856, -36.9897	Açu	BR-304	-5.2856, -36.2856	Pedro Avelino	BR-406	-5.5875, -36.9809	Açu	BR-304
7° ponto	-5.1980, -37.3184	Mossoró	BR-110	-5.2096, -37.3336	Mossoró	BR-110	-5.1370, -37.3453	Mossoró	BR-304
—	Tempo de execução: 20,780s			Tempo de execução: 3,258s			Tempo de execução: 7,779s		

Fonte: Elaborado pelos autores (2025)

Quadro 8 — Dados de saída do modelo para o estado da Paraíba para $p = 5$

$p = 5$	Carnaval			Semana Santa			Fim de Ano		
	Lat., Long.	Município	Rodovia	Lat., Long.	Município	Rodovia	Lat., Long.	Município	Rodovia
1° ponto	-7.2664, -35.9170	Campina Grande	BR-230	-7.2629, -35.9213	Campina Grande	BR-230	-7.2693, -35.9138	Campina Grande	BR-230
2°	-7.1881, -35.9170	Riachão	BR-230	-7.1646, -35.9170	João Pessoa	BR-230	-7.1657, -35.9170	João Pessoa	BR-230

ponto	-35.2810	do Poço		-34.8928	Pessoa		-34.8897	Pessoa	0
3º ponto	-7.1491, -34.9124	Bayeux	BR-230	-7.0071, -37.2885	Patos	BR-230	-6.9960, -37.1958	Patos	BR-230
4º ponto	-7.0182, -37.2265	Patos	BR-230	-6.9068, -35.1171	Capim	BR-101	-6.8745, -35.1323	Mamanguape	BR-101
5º ponto	-6.7840, -38.0949	Aparecida	BR-230	-6.8675, -38.4144	Umari	BR-230	-6.7752, -38.1982	Sousa	BR-230
—	Tempo de execução: 57,297s			Tempo de execução: 1,271s			Tempo de execução: 38,061s		

Fonte: Elaborado pelos autores (2025)

Quadro 9 — Dados de saída do modelo para o estado da Paraíba para $p = 6$

$p = 6$	Carnaval			Semana Santa			Fim de Ano		
	Lat., Long.	Município	Rodovia	Lat., Long.	Município	Rodovia	Lat., Long.	Município	Rodovia
1º ponto	-7.2581, -35.8908	Campina Grande	BR-104	-7.7937, -37.0184	Monteiro	BR-412	-7.2582, -35.9267	Campina Grande	BR-230
2º ponto	-7.1881, -35.2810	Riachão do Poço	BR-230	-7.2629, -35.9213	Campina Grande	BR-230	-7.2296, -35.5998	Ingá	BR-230
3º ponto	-7.1491, -34.9124	Bayeux	BR-230	-7.1646, -34.8928	João Pessoa	BR-230	-7.1651, -34.8941	João Pessoa	BR-230
4º ponto	-7.0705, -36.3480	Soledade	BR-230	-6.9353, -37.4067	Patos	BR-110	-6.9960, -37.1958	Patos	BR-230
5º ponto	-7.0182, -37.2265	Patos	BR-230	-6.9068, -35.1171	Capim	BR-101	-6.8141, -35.1394	Mamanguape	BR-101
6º ponto	-6.7840, -38.0949	Aparecida	BR-230	-6.8675, -38.4144	Umari	BR-230	-6.7752, -38.1982	Sousa	BR-230
—	Tempo de execução: 47,596s			Tempo de execução: 2,362s			Tempo de execução: 35,794s		

Fonte: Elaborado pelos autores (2025)

Quadro 10 — Dados de saída do modelo para o estado da Paraíba para $p = 7$

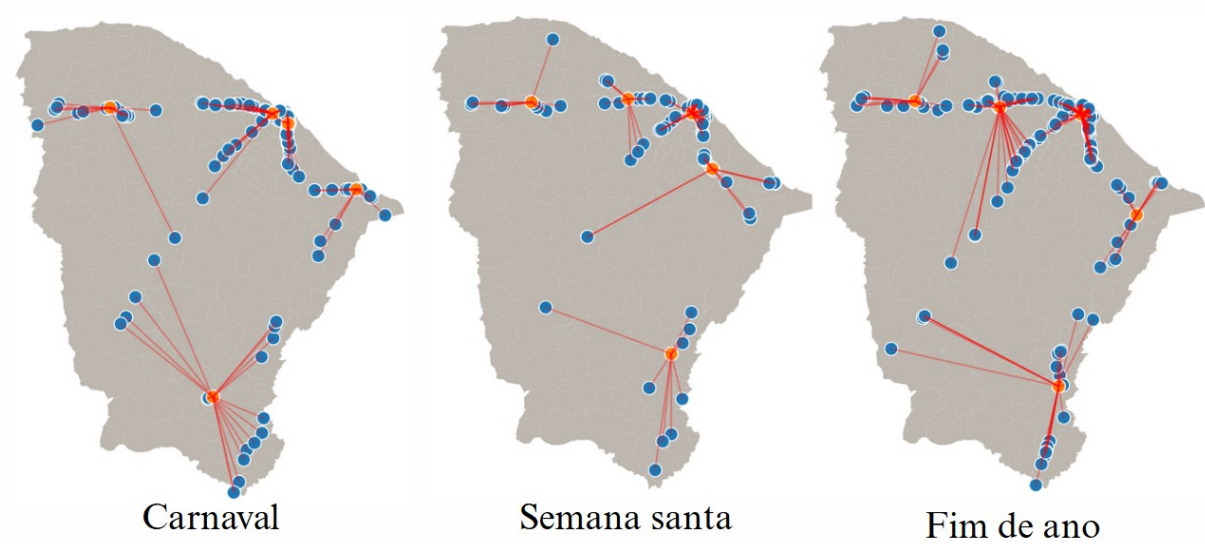
$p = 7$	Carnaval			Semana Santa			Fim de Ano		
	Lat., Long.	Município	Rodovia	Lat., Long.	Município	Rodovia	Lat., Long.	Município	Rodovia
1º	-7.2581,	Campina	BR-104	-7.7937,	Monteiro	BR-412	-7.2582,	Campina	BR-23

ponto	-35.8908	Grande		-37.0184			-35.9267	Grande	0
2° ponto	-7.1881, -35.2810	Riachão do Poço	BR-230	-7.3079, -35.8934	Queimadas	BR-104	-7.2296, -35.5998	Ingá	BR-230
3° ponto	-7.1512, -34.9101	João Pessoa	BR-230	-7.1646, -34.8928	João Pessoa	BR-230	-7.1651, -34.8941	João Pessoa	BR-230
4° ponto	-7.0705, -36.3480	Soledade	BR-230	-7.0692, -36.6002	Juazeirinho	BR-230	-6.9960, -37.1958	Patos	BR-230
5° ponto	-7.0182, -37.2265	Patos	BR-230	-6.9353, -37.4067	Patos	BR-110	-6.8141, -35.1394	Mamanguape	BR-101
6° ponto	-6.7840, -38.0949	Aparecida	BR-230	-6.9068, -35.1171	Capim	BR-101	-6.7817, -38.2695	Sousa	BR-230
7° ponto	-6.7403, -35.1315	Rio Tinto	BR-101	-6.8675, -38.4144	Umari	BR-230	-6.7641, -37.7891	Pombal	BR-427
—	Tempo de execução: 51,229s			Tempo de execução: 1,877s			Tempo de execução: 30,986s		

Fonte: Elaborado pelos autores (2025)

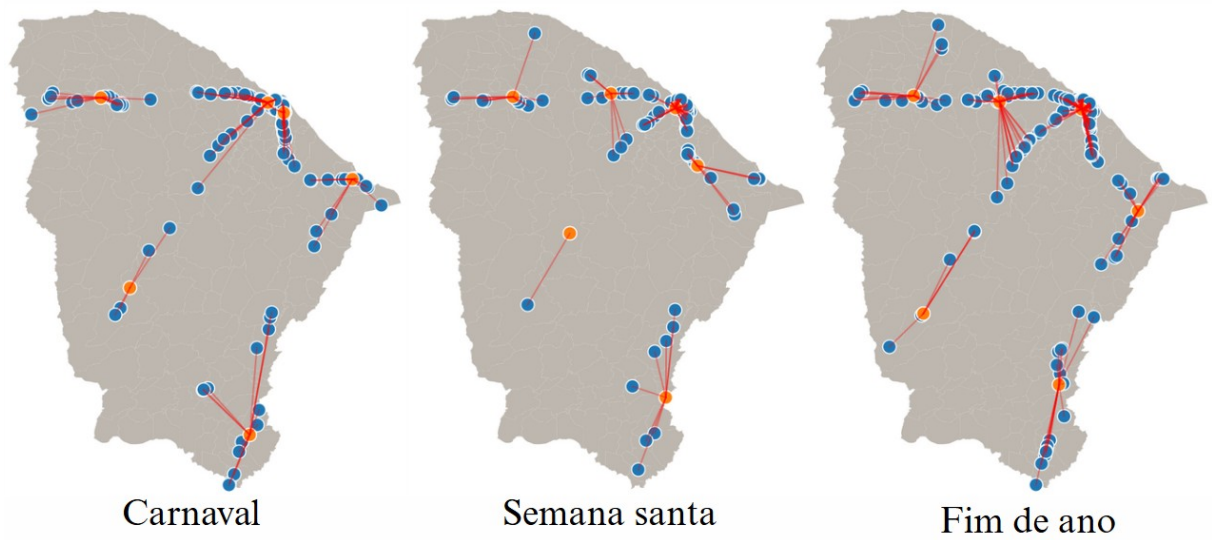
APÊNDICE C – IMAGENS GERADAS A PARTIR DOS DADOS DE SAÍDA DO MODELO

Figura 11 — Pontos de práticas preventivas no estado do Ceará para $p = 5$



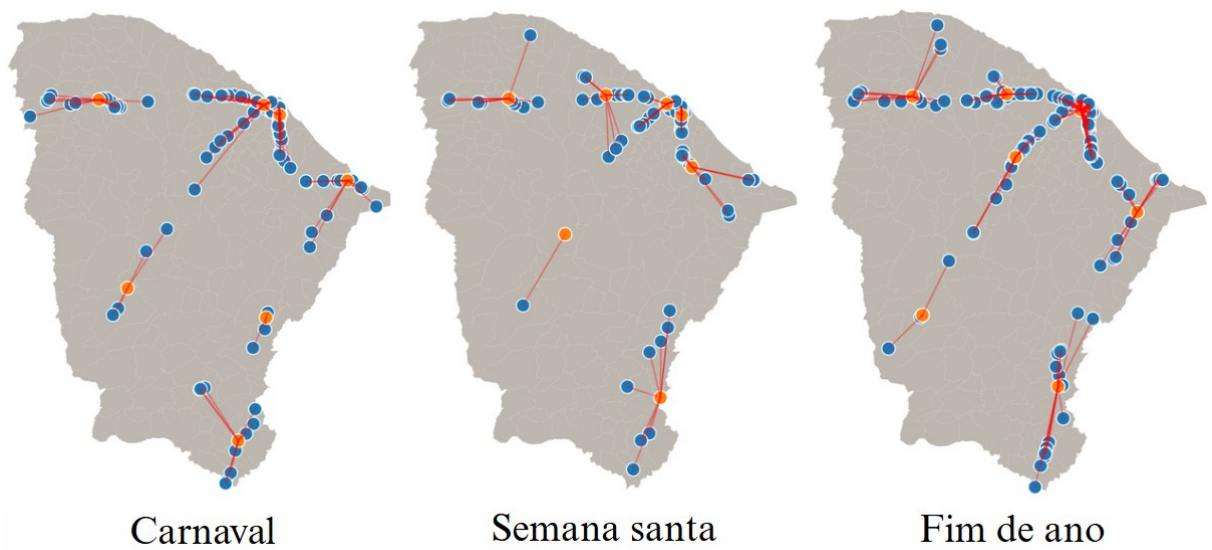
Fonte: Elaborado pelos autores (2025)

Figura 12 — Pontos de práticas preventivas no estado do Ceará para $p = 6$



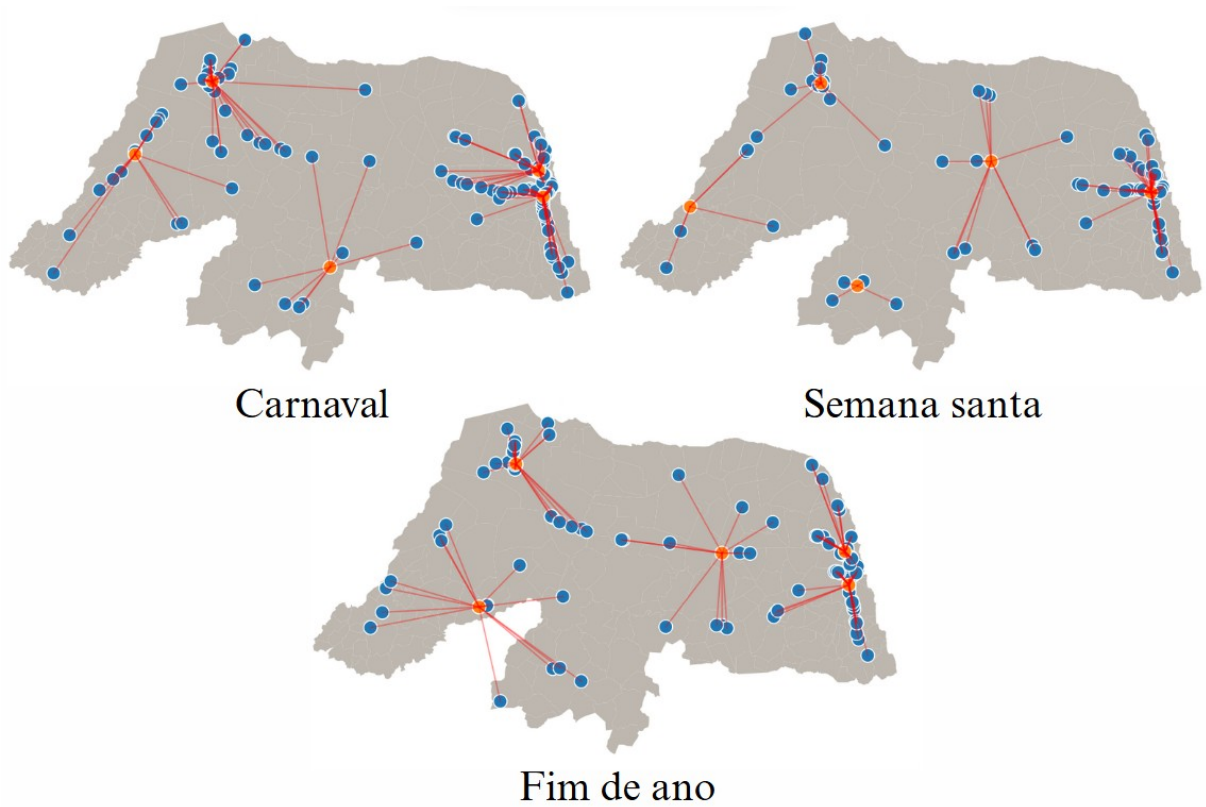
Fonte: Elaborado pelos autores (2025)

Figura 13 — Pontos de práticas preventivas no estado do Ceará para $p = 7$



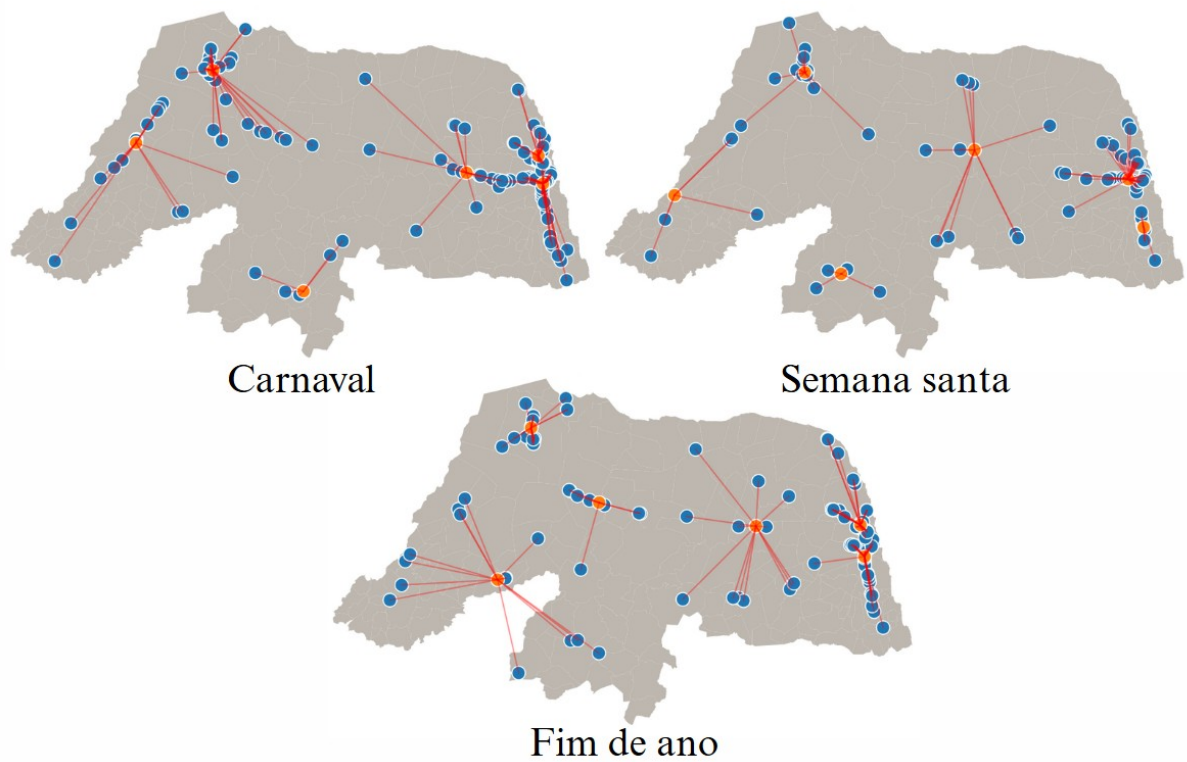
Fonte: Elaborado pelos autores (2025)

Figura 14 — Pontos de práticas preventivas no estado do Rio grande do Norte para $p = 5$



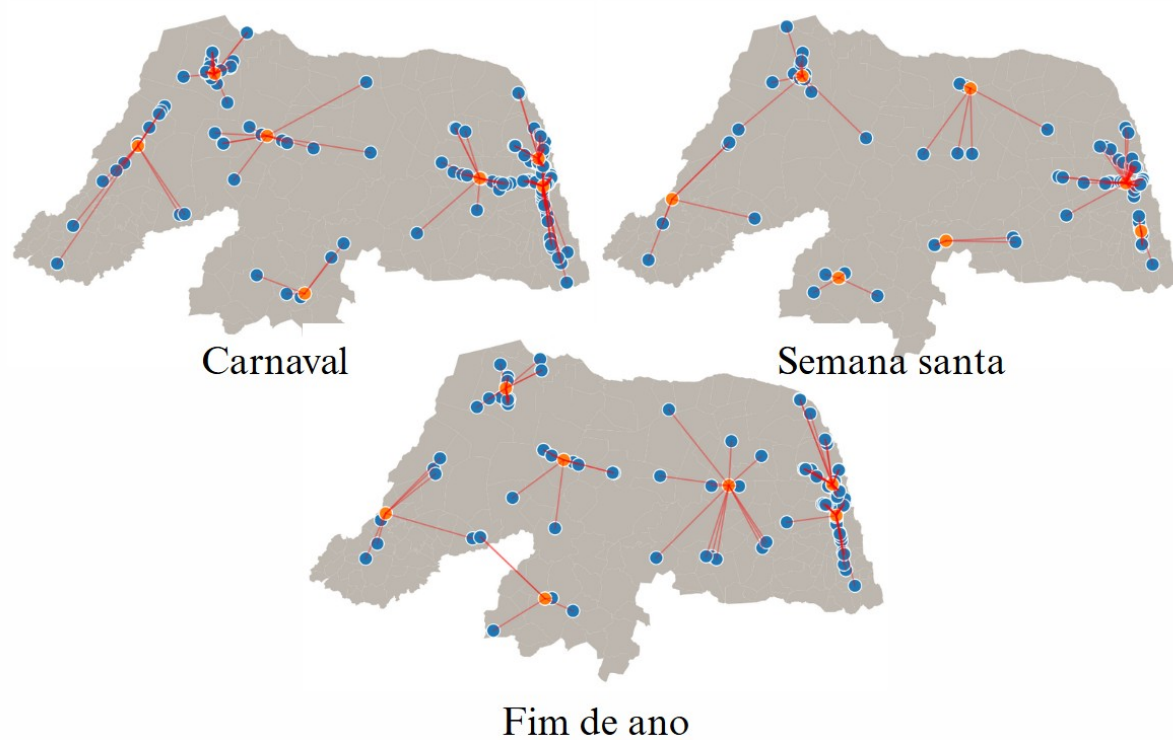
Fonte: Elaborado pelos autores (2025)

Figura 15 — Pontos de práticas preventivas no estado do Rio grande do Norte para $p = 6$



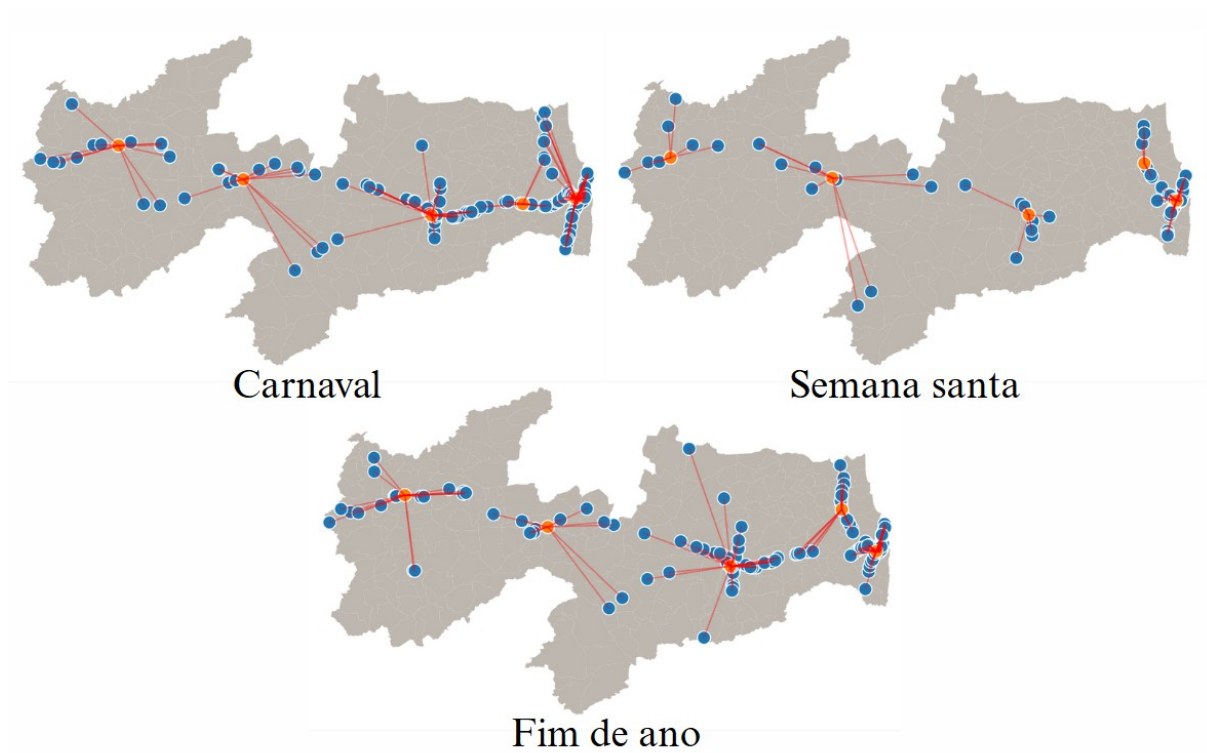
Fonte: Elaborado pelos autores (2025)

Figura 16 — Pontos de práticas preventivas no estado do Rio grande do Norte para $p = 7$



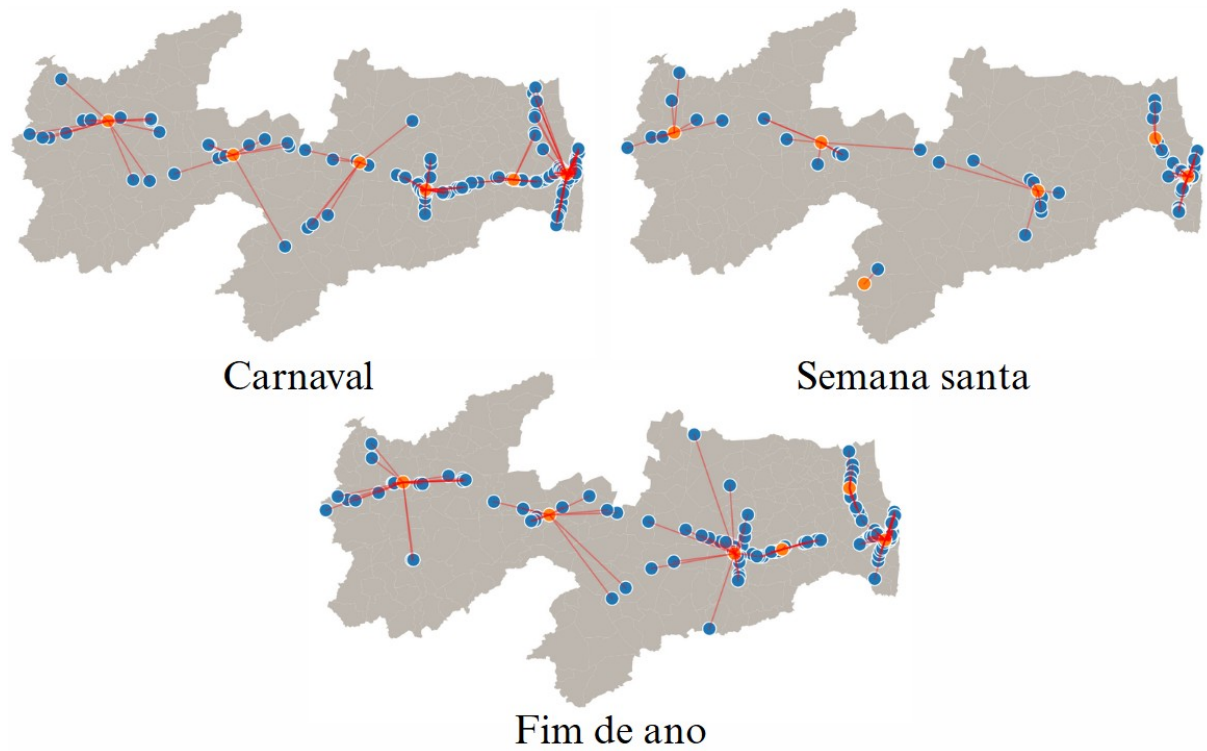
Fonte: Elaborado pelos autores (2025)

Figura 17 — Pontos de práticas preventivas no estado da Paraíba para $p = 5$



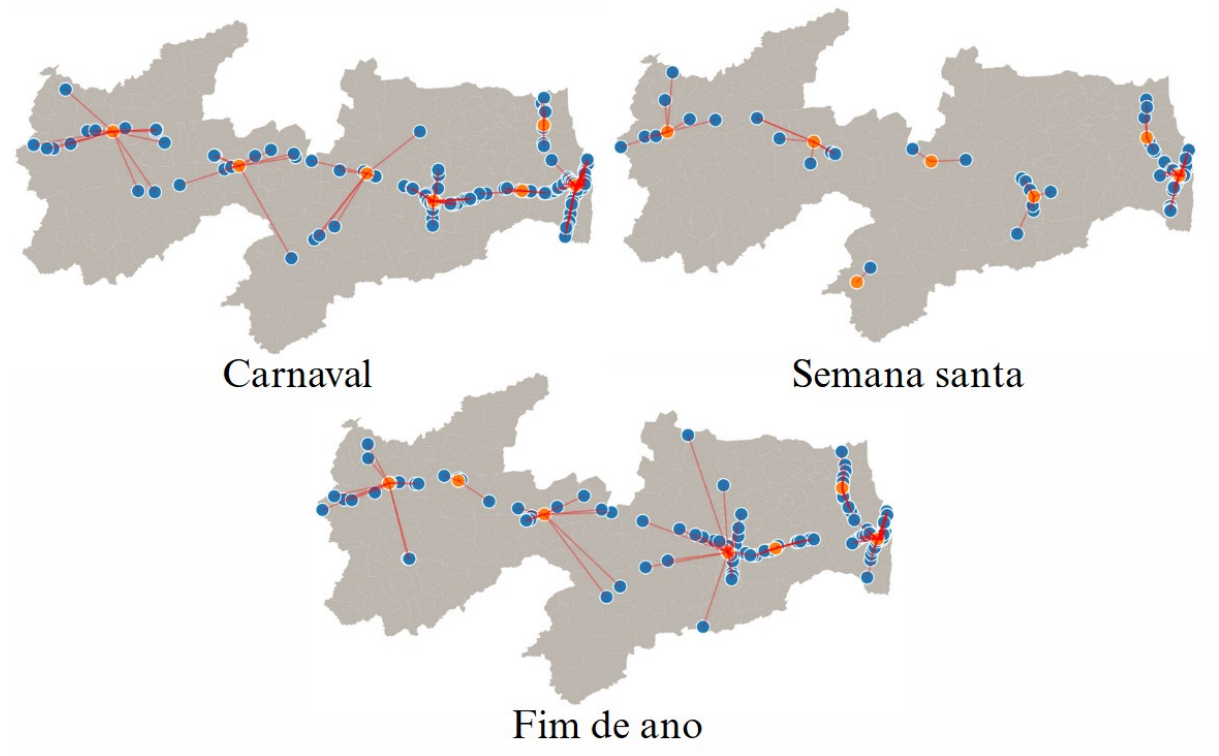
Fonte: Elaborado pelos autores (2025)

Figura 18 — Pontos de práticas preventivas no estado da Paraíba para $p = 6$



Fonte: Elaborado pelos autores (2025)

Figura 19 — Pontos de práticas preventivas no estado da Paraíba para $p = 7$



Fonte: Elaborado pelos autores (2025)