RIO

PUC

Dissertação de Mestrado

# Machine Learning and Heuristic Map Matching for Estimating NOx Emissions of Heavy-Duty Vehicles

Renan Morais Florias

Pontifícia Universidade Católica do Rio de Janeiro
Centro Técnico Científico
Departamento de Informática

Rio de Janeiro, 4 de setembro de 2025

Pontifícia
Universidade
Católica do
Rio de Janeiro

Dissertação de Mestrado

# Machine Learning and Heuristic Map Matching for Estimating NOx Emissions of Heavy-Duty Vehicles

Renan Morais Florias

Orientação: Professor Paulo Ivson Netto Santos

Dissertação apresentada como requisito parcial para a obtenção do grau de Mestre em Informática pelo programa de Pós-Graduação em Informática, no Departamento de Informática da Pontifícia Universidade Católica do Rio de Janeiro.

Rio de Janeiro, 4 de setembro de 2025

Pontifícia
Universidade
Católica do
Rio de Janeiro

# Machine Learning and Heuristic Map Matching for Estimating NOx Emissions of Heavy-Duty Vehicles

Renan Morais Florias

**Dissertação apresentada como requisito parcial para a obtenção do grau de Mestre em Informática pelo programa de Pós-Graduação em Informática, no Departamento de Informática da Pontifícia Universidade Católica do Rio de Janeiro Aprovada pela Comissão examinadora abaixo:**

**Prof. Paulo Ivson Netto Santos**
Orientador
Departamento de Informática – PUC-Rio

**Professor Alberto Barbosa Raposo**
Departamento de Informática – PUC-Rio

**Professor Rafael Martinelli Pinto**
Departamento de Informática – PUC-Rio

Rio de Janeiro, 4 de setembro de 2025

Pontifícia
Universidade
Católica do
Rio de Janeiro



Renan Morais Florias

Graduou-se em Ciências Econômicas pela Pontifícia Universidade Católica do Rio de Janeiro.

Ficha Catalográfica

Florias, Renan Morais

Machine Learning and Heuristic Map Matching for Estimating NOx Emissions of Heavy-Duty Vehicles / Renan Morais Florias; advisor: Paulo Ivson Netto Santos. – 2025.

57 f: il. color. ; 30 cm
Dissertação de Mestrado - Pontifícia Universidade Católica do Rio de Janeiro, Departamento de Informática, 2025.

Inclui bibliografia

1. Informática – Teses. 2. Correção de trajeto. 3. Emissões de NOx. 4. Aprendizado de Máquina. 5. Modelagem de transporte. I. Ivson, Paulo Netto Santos. II. Pontifícia Universidade Católica do Rio de Janeiro. Departamento de Informática. III. Título

CDD: 004

To my family, especially my parents,
for their support, care, and the opportunities they have given me.

## Acknowledgments

First and foremost, I thank God for the wisdom and the opportunity to continue my studies.

To my parents, for their education, care, and constant support.

To Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq) and PUC-Rio for the financial support, without which this work would not have been possible.

To my advisor, Paulo Ivson, for his encouragement throughout the development of this work.

To the professors who participated in the examination committee.

To all the university and computer science department staff for their assistance and services.

This study was financed in part by the Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brasil (CAPES) - Finance Code 001.

# Abstract

Florias, Renan Morais; Ivson, Paulo Netto Santos (Advisor). **Machine Learning and Heuristic Map Matching For Estimating NOx Emissions of Heavy-Duty Vehicles**. Rio de Janeiro, 2025. 57p. Masters Dissertation – Departamento de Informática, Pontifícia Universidade Católica do Rio de Janeiro.

This study investigates the main factors influencing nitrogen oxide (NOx) emissions from heavy-duty vehicles by integrating GPS trajectory data with road network and environmental characteristics. The research focuses on evaluating the contribution of map-matched variables such as elevation, curved distance, and segment geometry in comparison to features derived from raw GPS data, such as Haversine-based distance and simple speed estimates. The analysis explores the relative importance of each variable in predicting NOx emissions and assesses whether map matching significantly enhances model performance. Results indicate that while map matching enables the extraction of richer spatial features, its overall impact on prediction accuracy is limited in expressway-dominated routes. These findings offer practical insights into the relevance of geographic and operational features for emissions modeling and provide guidance for the efficient design of data-driven tools for sustainable logistics and environmental regulation.

## Keywords

Map matching; NOx emissions; Machine Learning; Transportation Modeling.

## Resumo

Florias, Renan Morais; Ivson, Paulo Netto Santos. **Aprendizado de Máquina e Heurística de Map Matching para Estimativa de Emissões de NOx de Veículos Pesados**. Rio de Janeiro, 2025. 57p. Dissertação de Mestrado – Departamento de Informática, Pontifícia Universidade Católica do Rio de Janeiro.

Este estudo investiga os principais fatores que influenciam as emissões de óxidos de nitrogênio (NOx) provenientes de veículos pesados, integrando dados de trajetórias de GPS com características da malha viária e do ambiente. A pesquisa foca na avaliação da contribuição de variáveis obtidas por map matching, como elevação, distância ao longo da curva e geometria dos segmentos, em comparação com variáveis derivadas diretamente dos dados brutos de GPS, como distância baseada na fórmula de Haversine e estimativas simples de velocidade. A análise explora a importância relativa de cada variável na predição das emissões de NOx e avalia se o uso do map matching melhora significativamente o desempenho dos modelos. Os resultados indicam que, embora o map matching permita a extração de características espaciais mais ricas, seu impacto na acurácia preditiva é limitado em rotas dominadas por vias expressas. Essas conclusões oferecem insights práticos sobre a relevância de atributos geográficos e operacionais para a modelagem de emissões e fornecem orientações para o desenvolvimento eficiente de ferramentas baseadas em dados voltadas à logística sustentável e à regulação ambiental.

## Palavras-chave

Correção de trajeto; Emissões de NOx; Aprendizado de Máquina; Modelagem de transporte.

# Table of contents

# List of figures

## List of tables

## List of algorithms

## List of Abreviations

ANN – Artificial Neural Networks

EGR – Exhaust Gas Recirculation

GPS – Global Positioning System

IBGE – Instituto Brasileiro de Geografia e Estatística

IoT – Internet of Things

kWh – Kilowatt-Hour

MAPE – Mean Absolute Percentual Error

NOx – Nitrogen Oxides

OBD – On-Board Diagnostics

OSM – OpenStreetMap

PEMS – Predictive Emission Monitoring Systems

PPM – Parts Per Million

*The first thing I learned was that there were no limits to what I could learn. The second was that not everyone would accept that.*

**Isaac Asimov**, *Bicentennial Man.*

# 1
# Introduction

Nations are modernizing their road freight fleets by importing or manufacturing vehicles equipped with hybrid or fully electric (battery-powered) engines. However, the transition process remains gradual, especially for heavy-duty vehicles, which require higher engine power to transport large loads. In Brazil in particular, freight logistics and transportation are carried out predominantly by trucks operating across interstate routes. The road transport sector accounts for 64.9% of all freight moved in the country (Confederação Nacional do Transporte, 2023), and the vast majority of vehicles still rely on fossil fuels. In 2020 alone, Brazil consumed 40 billion liters of mineral diesel, representing 78.2% of the total diesel available in the country (Confederação Nacional do Transporte, 2023). Given this context, it is crucial to continue developing methodologies for estimating NOx emissions, thus supporting more effective policy discussions and strategies aimed at reducing such emissions on a national scale.

Despite its relevance to national emissions, accurately estimating NOx remains challenging, particularly when relying on real-world GPS data collected from freight vehicles. These datasets often present inconsistencies, noise, and irregular sampling intervals, which hinder direct application in emission models.

Estimating vehicular emissions poses methodological challenges that vary according to the pollutant type and the data source employed. In the study by Zhou, Xie et al. (2020), $CO_2$ emissions are indirectly estimated using On-Board Diagnostics (OBD) data, which provide engine operation features such as speed, acceleration, and load. This approach enables scalable predictive modeling, such as the use of Support Vector Machines (SVM), and is well suited for eco-routing applications. In contrast, Fang, Li et al. (2021) adopt a different strategy by utilizing Portable Emissions Measurement Systems (PEMS) to directly measure real-world $NO_x$ and $CO_2$ emissions from a heavy-duty vehicle. Although this method offers higher precision, it involves greater complexity and cost, making it less practical for large-scale applications. Comparing both studies reveals that while $CO_2$ can be reasonably estimated using driving behavior features, $NO_x$ requires more sensitive data related to engine thermal and chemical conditions, making it more difficult to model through OBD data alone. These differences highlight the need for pollutant-specific modeling strategies, particularly when representing $NO_x$ emissions in

real and heterogeneous operational contexts.

Map matching is defined as a process of approximating or associating GPS points (usually obtained from vehicles or pedestrians in urban areas) in order to match the sequences of these points to a road or accessible path in a road network or digital map (CHAO et al., 2019). GPS data are inherently noisy due to positioning errors caused by signal obstructions (e.g., urban canyons, tunnels), low sampling frequency leading to gaps between consecutive points, abrupt jumps caused by signal interference, and temporal synchronization issues (CHAO et al., 2019). Several studies have leveraged map-matching algorithms to enhance trajectory reconstruction and speed estimation from noisy GPS data. For example, Newson e Krumm (2009) developed a Hidden Markov Model-based approach to align GPS points to road networks, significantly improving the accuracy of speed calculations. Other works, such as Lou et al. (2009), have proposed probabilistic and urban-focused methods respectively to reconstruct realistic vehicle paths.

In this study, we use GPS data collected via PEMS-based sensors integrated into the IoT system. The objective of this study is to analyze the main factors influencing $NO_x$ emissions in heavy-duty vehicles, with a focus on identifying the most relevant features for accurate prediction. We also evaluate the benefits of incorporating a simplified map-matching technique to enhance input features such as elevation and road geometry, particularly in geographic contexts characterized by heterogeneous terrain and constraints of lower-frequency sensor data acquisition. By comparing model performance with and without map matching, we aim to assess the real impact of map-matched features in emission estimation.

The remainder of this study is organized as follows: **Section 2** outlines the theoretical background, covering GPS-based emission monitoring, map-matching techniques, road curvature, and the machine learning models employed in this study. **Section 3** reviews related work on vehicular emission modeling and map-matching techniques, highlighting key findings and gaps in integrating both approaches for emission reduction. **Section 4** describes the methodology, including data preprocessing, feature selection, the implementation of a heuristic-based map-matching technique for feature extraction, and the machine learning models applied. **Section 5** discusses the results, focusing on the identification of key predictive features and comparing model performance with and without map matching to assess its practical benefits. Finally, **Section 6** concludes the study with a summary of findings, emphasizing feature importance and discussing the implications of map matching on emission modeling, along with suggestions for future research.

## 1.1 Objectives

The primary objective of this study is to develop a predictive framework capable of estimating $NO_x$ emissions in specific road segments using real-world GPS data from heavy-duty vehicles. As secondary objectives, the research aims to implement a heuristic-based map-matching procedure to enhance trajectory alignment and enrich input features such as elevation and road geometry, to evaluate the contribution of map-matched features to the predictive performance of machine learning models, to identify and analyze the most relevant factors influencing $NO_x$ emissions in heterogeneous operational contexts, and to compare model performance with and without map matching in order to assess its practical benefits for emission estimation.

# 2
# Theoretical Background

In this chapter, we present a brief discussion of the technical terms used throughout this work. In the section 2.1, we describe how GPS data is collected through different types of sensors. Next, in 2.2 section, we introduce the functioning and theoretical background of the most commonly used map matching algorithms in the literature, as well as the different use of formula to calculate the distance between two points. The 2.3 section discusses how we apply the concept of road curvature based on the geometry of the network. Finally, in section 2.4 we explore machine learning models from a mathematical perspective, aiming to understand how different models interact with the data and how performance metrics reveal these behaviors.

## 2.1
## GPS Sensors

There are several types of sensors used for measuring pollutants in land vehicles. Within the scope of this work, we highlight three main types: sensors embedded in on-board systems (OBD - On-Board Diagnostics), sensors used exclusively in laboratory settings (chassis dynamometer), and portable systems or sensors for real-world, in-field measurements (PEMS - Portable Emissions Measurement Systems). Below, we provide a brief overview of how each of these systems works.

### 2.1.1
### OBD - On-Board Diagnostics

On-board system sensors are permanently installed in vehicles and, in most cases, are connected to the engine control unit (ECU). They are mainly used to monitor engine operation and collect data such as engine speed (RPM), engine temperature, and the air-fuel mixture. In addition, they allow for the reading of indirect emission indicators such as oxygen levels (O), nitrogen oxides (NO  only in diesel vehicles), and exhaust gas temperature. These sensors continuously collect data. Although there is no direct conversion to emission mass (g/km), the data provided can be used to estimate emissions. Fig. 2.1 and

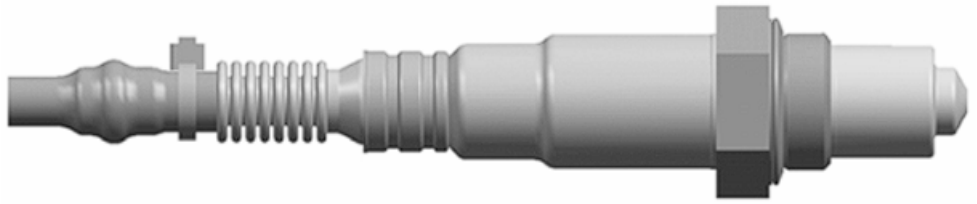

Figure 2.1: Casing of a sensor used in OBD systems, as typically found in vehicles for emission monitoring. Source: (KAMP, 2010).

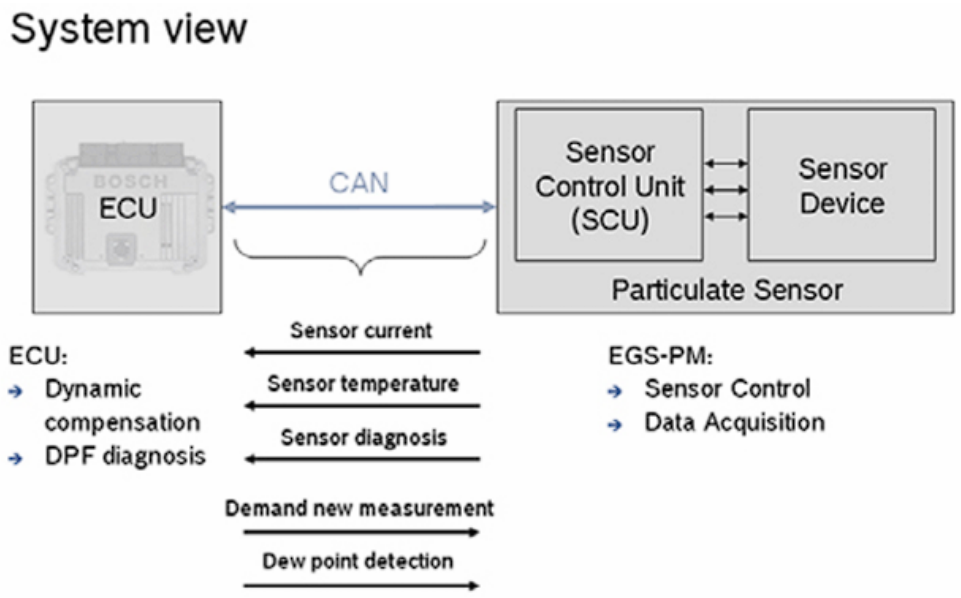


Figure 2.2: Framework of the particulate matter sensor integrated with the engine control unit (ECU), as used in OBD systems. Source: (KAMP, 2010).

### 2.1.2 Chassis Dynamometer Laboratory Sensors

Basically, chassis dynamometer sensors are connected directly to the vehicle's exhaust system, meaning they are an integral part of the exhaust. This setup is complex and requires a gas suction system feeding into a computational unit that analyzes the gases to achieve precise measurement of emitted pollutants. Additionally, filters are used to prevent unwanted particulates.

These measurements are conducted in controlled laboratory environments. In fact, the term "chassis dynamometer" refers to the motorized roller system on which the vehicle is placed to simulate driving conditions over rollers. The sensors used in this setup share similarities with those found in Portable Emissions Measurement Systems (PEMS), particularly in terms of their measurement objectives and integration with the vehicle's exhaust.

As described by Wang et al. (2024), the computational unit may incorporate optical, electronic, or mechanical sensors to ensure high precision and low uncertainty in emission measurements.

Laboratory test sensors have improved in their simulations but are still considered inferior to real-world tests (PEMS sensors). The graphical summary in the article (Fig. 2.3) shows how the sensors diverge in CO and NOx emissions even at the same vehicle speeds (TRIANTAFYLLOPOULOS et al., 2019). On the other hand, ongoing research supports the development of chassis dynamometer sensor technologies. Wang et al. (2024) present a new optical precision measurement technique that reduces uncertainty in the detection of gases emitted by diesel vehicles. Since sensors used in simulation-based tests are designed to capture high-frequency data, even without real variations from routine truck driving, the applied technique allows these simulations to achieve greater variability and accuracy in emission measurements.

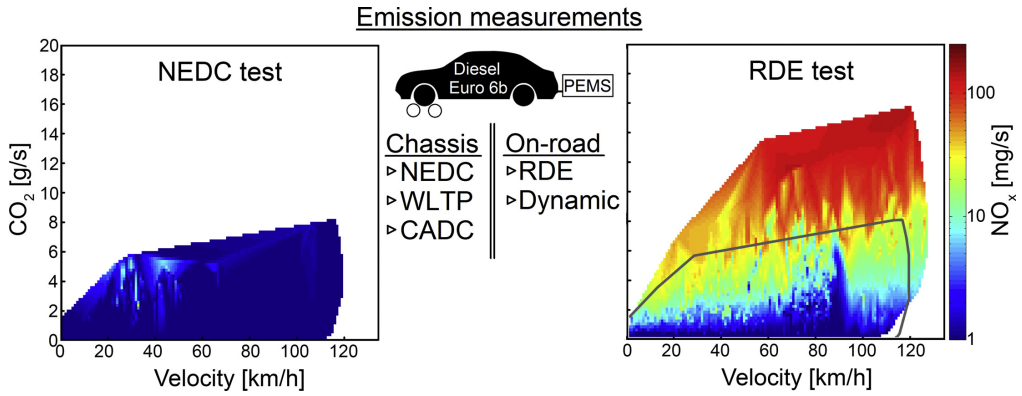


Figure 2.3: Graphical summary of the differences in emissions between simulation-based sensors and real-world measurement sensors. Source: (TRIANTAFYLLOPOULOS et al., 2019)

### 2.1.3 PEMS - Portable Emissions Measurement System

As discussed in the previous subsection, both PEMS sensors and those used in chassis dynamometer tests are applied at the same detection level in vehicles, namely the exhaust system. However, PEMS models are designed to be portable and operate under real driving conditions, whether in urban or highway environments. A key distinguishing feature of PEMS sensors is their insulation from external factors at the exhaust installation point, which protects them from moisture, temperature fluctuations, and dust. It is important to emphasize that PEMS-type sensors were employed in the vehicles analyzed in this study. This distinction is crucial for understanding the

methodological differences between our approach and other studies that rely on different test setups, measurement techniques, or NO prediction models.

Zhao et al. (2023) use a PEMS system to evaluate the impact of driving behavior on $NO_x$ emission levels. The tests are conducted on heavy-duty vehicles, primarily varying the vehicle payload. Traffic conditions are also observed; however, vehicle speed is measured using an OBD sensor. The authors employ an embedded system, shown in (Fig. 2.4), integrating the PEMS sensor with the vehicles onboard OBD engine sensor to synchronize both systems, allowing for a precise variation analysis between $NO_x$ emissions and speed. This synchronization is a key differentiator of the present study, in which we later propose a methodology to infer vehicle speed in the final dataset for evaluation purposes.

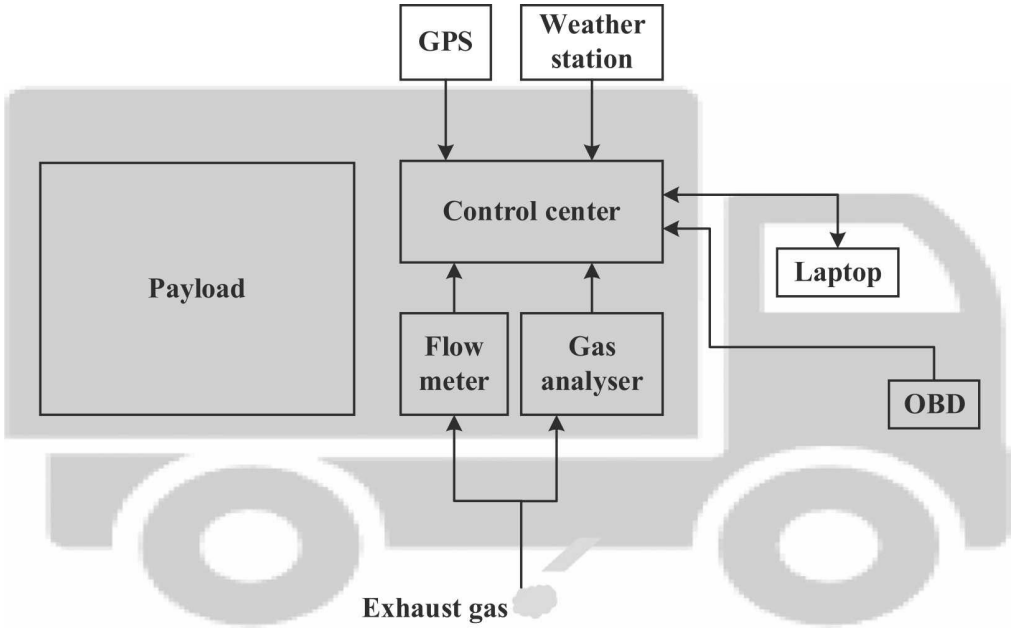


Figure 2.4: Layout of PEMS devices. Source: (ZHAO et al., 2023)

## 2.2 Map Matching Algorithms

In this section, we present the frameworks and, where applicable, the calculations used in the map matching algorithms found in the literature. There are several models derived from the more traditional algorithms. As a disclaimer, we will focus on the foundational algorithms, those that serve as the basis in the literature.

### 2.2.1 Heuristic-Based Algorithms

Chawathes study combines road geometry and GPS data with a heuristic approach aimed at refining road segments. In other words, it employs algorithms that weigh vehicle movement through the road network. These distances, known as the *Hausdorff distance* and the *Fréchet distance*, are used respectively to filter potential candidates on the road network between points A and B, and to perform a more accurate evaluation among the best candidates identified by the first distance (EITER; MANNILA, 1994; HUTTENLOCHER; KLANDERMAN; RUCKLIDGE, 1993; CHAWATHE, 2007).

Mathematically, the Hausdorff distance is defined as:

$$d_H(A, B) = \max \left\{ \max_{a \in A} \min_{b \in B} \|a - b\|, \max_{b \in B} \min_{a \in A} \|b - a\| \right\} \tag{2-1}$$

Where:

- $a \in A$ and $b \in B$ are individual points in the sets $A$ and $B$;
- $\|a - b\|$ represents the Euclidean distance between points $a$ and $b$;
- $\min_{b \in B} \|a - b\|$ computes the distance from point $a$ to its nearest neighbor in $B$;
- $\max_{a \in A} \min_{b \in B} \|a - b\|$ selects the largest of these minimum distances across all points in $A$;
- The symmetric term $\max_{b \in B} \min_{a \in A} \|b - a\|$ ensures the distance is measured in both directions.

The Fréchet distance $\delta_F(f, g)$ measures the similarity between two curves $f$ and $g$, taking into account the order in which points are traversed. It can be visualized as the minimal leash length if a person walks along one curve and a dog along the other, without backtracking.

The Fréchet distance is defined as:

$$\delta_F(f, g) = \inf_{\alpha, \beta} \max_{t \in [0,1]} \|f(t) - g(t)\| \tag{2-2}$$

Where:

- $f(t)$ and $g(t)$ are points along curves $f$ and $g$, parameterized by $t \in [0, 1]$;
- $\alpha$ and $\beta$ are continuous, non-decreasing reparameterizations that allow walking along each curve at feature speeds;

- $\max_{t\in[0,1]} \|f(t) - g(t)\|$ computes the largest distance at any point along the reparameterized curves;
- $\inf_{\alpha,\beta}$ then finds the reparameterization that minimizes this maximal distance, giving the Fréchet distance.

Both distances are widely used in map-matching algorithms: Hausdorff focuses on pointwise closeness, while Fréchet captures both closeness and the ordering of points along a path.

### 2.2.2 Topology-Based Algorithms

One of the pioneering works studying the relationship between topology and map matching algorithms White, Bernstein e Kornhauser (2000), the study compares Point-to-Point Matching (P2P) and Point-to-Curve Matching (P2C) algorithms, which form the basis of our heuristic calculated with the Haversine distance equation Robusto (1957) to associate the correct vehicle trajectory. In Fig. 2.5, created by the authors, there is a discussion of how the algorithms associate each point and how topology influences the choice of the next segment.

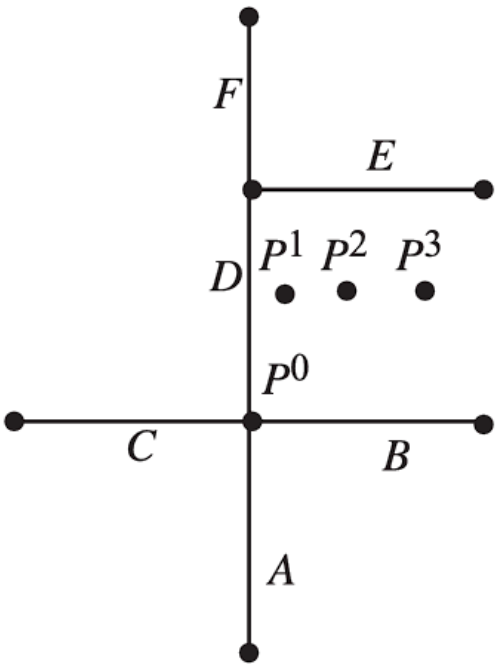


Figure 2.5: "The road network topology (connectivity between roads) prevents errors in GPS trajectory mapping by restricting subsequent points (1 and 3) only to physically connected roads (A, B, C or D) from the origin point 0, while discarding topologically inaccessible options such as Road E, even when geographically close" (WHITE; BERNSTEIN; KORNHAUSER, 2000)

The Haversine formula estimates the great-circle distance between two points on the surface of a sphere, such as the Earth. Unlike a simple straight-line distance, it accounts for the Earth's curvature, providing a more accurate measure of the actual distance traveled along the surface.

The Haversine distance is calculated by:

$$d = 2R \cdot \arcsin\left(\sqrt{\sin^2\left(\frac{\Delta\phi}{2}\right) + \cos(\phi_1) \cdot \cos(\phi_2) \cdot \sin^2\left(\frac{\Delta\lambda}{2}\right)}\right) \tag{2-3}$$

Where:

- $d$ is the distance between the two points along the Earth's surface;
- $R$ is the Earth's mean radius (approximately 6,371 km);
- $\phi_1$ and $\phi_2$ are the latitudes of the first and second points, measured in radians;
- $\lambda_1$ and $\lambda_2$ are the longitudes of the first and second points, in radians;
- $\Delta\phi = \phi_2 - \phi_1$ is the difference in latitude;
- $\Delta\lambda = \lambda_2 - \lambda_1$ is the difference in longitude;
- $\sin^2(\Delta\phi/2)$ computes the squared sine of half the latitude difference, which captures the north-south component of the distance;
- $\cos(\phi_1) \cdot \cos(\phi_2) \cdot \sin^2(\Delta\lambda/2)$ captures the east-west component, scaled by the cosine of the latitudes to account for convergence of meridians toward the poles;
- $\sqrt{...}$ and $\arcsin(...)$ transform these components into an angular distance along the sphere;
- Multiplying by $2R$ converts the angular distance into a linear distance along the Earth's surface.

In summary, the Haversine formula combines both the north-south and east-west differences between two points, adjusts for the Earth's curvature, and gives a single distance value that closely approximates the shortest path along the Earth's surface.

## 2.3 Road Curvature

The approximation between GPS points and the road network is neglected if we don't specify exactly where a projected point is located on the associated edge. This neglect is justified if we consider the low frequency of GPS data obtained through sensors, as is the case in this study.

The association of GPS points with edges will be discussed later in the methodology section of this work. However, it is important to understand how we consider the curves of each edge in the road network.

To determine the distance traveled along a road geometry between an arbitrary point and a reference node, we propose a function which operates through three fundamental mathematical stages:

1. **Orthogonal Projection**: Following computational geometry principles established by Berg et al. (2008), given a geometry $\Gamma$ (represented as a *LineString*) and a point $P$, we calculate the linear projection $d_{proj}$ through Euclidean distance minimization:

$$d_{proj} = \underset{t \in [0,L]}{\operatorname{argmin}} \|\Gamma(t) - P\| \tag{2-4}$$

   Here:

   - $\Gamma$ = the road geometry, modeled as a curve (a *LineString*);
   - $P$ = the GPS point to be projected;
   - $t$ = the position parameter along the road, which varies from 0 (start) to $L$ (end);
   - $L$ = the total length of the road;
   - $d_{proj}$ = the location on $\Gamma$ where the distance to $P$ is smallest;
   - $\|\Gamma(t) - P\|$ = the Euclidean distance between the point on the road and $P$.

   In simple terms, this step finds the point on the road $\Gamma$ that is closest to the GPS point $P$. It is equivalent to dropping a perpendicular line from the GPS point onto the road.

2. **Subsegment Extraction**: According to the `target_node` parameter:

$$\Gamma_{sub} = \begin{cases} \Gamma(0 \to d_{proj}) & \text{if } \texttt{target\_node} = \text{“início”} \\ \Gamma(d_{proj} \to L) & \text{if } \texttt{target\_node} = \text{“fim”} \end{cases} \tag{2-5}$$

   Here:

   - $\Gamma_{sub}$ = the selected subsegment of the road;

- `target_node` = user-defined option indicating which part of the road to keep;
- $\Gamma(0 \rightarrow d_{proj})$ = the portion from the start (0) to the projection point;
- $\Gamma(d_{proj} \rightarrow L)$ = the portion from the projection point to the end ($L$).

This step cuts the road into two pieces at $d_{proj}$ and selects either the first or the second, depending on whether the target node is the start or the end.

3. **Distance Calculation**: The length of subsegment $\Gamma_{sub}$ is calculated numerically using the discrete segment summation method, analogous to the approach proposed by Peucker e Douglas (1975):

$$D = \int_{\Gamma_{sub}} ds \approx \sum_{i=1}^{n-1} \|\Gamma_{sub}(t_{i+1}) - \Gamma_{sub}(t_i)\| \tag{2-6}$$

Here:

- $D$ = the total length of the subsegment;
- $\int_{\Gamma_{sub}} ds$ = the exact curve length expressed as a continuous integral;
- $t_i$ = discrete sample points along the subsegment;
- $\Gamma_{sub}(t_i)$ = the coordinates of the curve at each sampled point;
- $\|\Gamma_{sub}(t_{i+1}) - \Gamma_{sub}(t_i)\|$ = the straight-line distance between consecutive sampled points.

In practice, this means we approximate the length of the curved road by dividing it into many small straight pieces, measuring each one, and then adding them together. The finer the division, the closer the approximation is to the true continuous length.

## 2.4 Machine Learning Models

Advanced statistical models, including machine learning approaches, are often employed in combination due to their shared inferential objectives. However, the literature also seeks to understand why certain models outperform others, even when their hyperparameters have been similarly tuned to the dataset (ALHARTHI et al., 2020). This perspective directs us to the foundational theories of these models, where we can examine the specific statistical scenarios in which each performs optimally.

In the following subsections, we will detail the unique characteristics of each model used in this study, highlighting their methodological construction and differences.

### 2.4.1 Random Forest

The ensemble Random Forest model was first proposed by Breiman (2001). Breiman says that the creation of Random Forest originates from the idea of ensembles, where each tree votes for the most frequent class that appeared in the training set. The created trees come from random vectors and these vectors influence how each branch of the tree expands, it's important to understand that the vectors are independent and after generating many trees, the voting decides the final class, informing the classification or regression of the dataset, this is the process and foundation of Random Forest. This operation is briefly explained in the margin concept that Breiman calls Random Forest Convergence, shown in the equation below.

$$\text{mg}(X,Y) = \frac{1}{K}\sum_{k=1}^{K} I(h_k(X) = Y) \; - \; \max_{j \neq Y} \frac{1}{K}\sum_{k=1}^{K} I(h_k(X) = j)$$

Here:

- $\text{mg}(X,Y)$ = the **margin** of classification for instance $X$ with true class $Y$. It measures the difference between the proportion of trees that correctly classify $X$ and the highest proportion of trees that vote for any incorrect class.
- $X$ = the **input instance**, represented as a feature vector describing the data point.
- $Y$ = the **true class** of the instance $X$.
- $K$ = the **number of trees** in the Random Forest ensemble.
- $h_k(X)$ = the prediction of the $k$-th tree for the instance $X$.
- $I(\cdot)$ = the **indicator function**, which returns 1 if the condition inside is true and 0 otherwise.

In practical terms, the first term, $\frac{1}{K}\sum_{k=1}^{K} I(h_k(X) = Y)$, counts the fraction of trees that correctly predict the true class $Y$. The second term, $\max_{j \neq Y} \frac{1}{K}\sum_{k=1}^{K} I(h_k(X) = j)$, finds the highest fraction of trees voting for any incorrect class. The margin is simply the difference between these two quantities: a positive margin means the majority of trees voted correctly, while a negative margin indicates misclassification.

Followed by the probability of error, which indicates the chance that the classifier incorrectly predicts the class, that is, votes for the wrong class, resulting in a negative margin. The equation is as follows:

$$P_E^* = P_{X,Y}\Big(\text{mg}(X,Y) < 0\Big)$$

Here:

- $P_E^*$ = the **theoretical probability of error** of the classifier; it measures the likelihood that a random instance is misclassified.
- $P_{X,Y}$ = the **joint probability distribution** over the input instances $X$ and their true classes $Y$.
- $\text{mg}(X,Y)$ = the **margin**, as defined in the previous equation.

In other words, this equation calculates the probability that the margin is negative, meaning that the ensemble Random Forest voted incorrectly for a given instance. It provides a formal measure of the classifiers expected error over the entire data distribution.

A Fig. 2.6 demonstrates a basic example of the tree growth process in this method through to its final output. The key distinction between Random Forest's regression and classification models is relatively minor, essentially resulting in very similar algorithmic frameworks for tree construction.

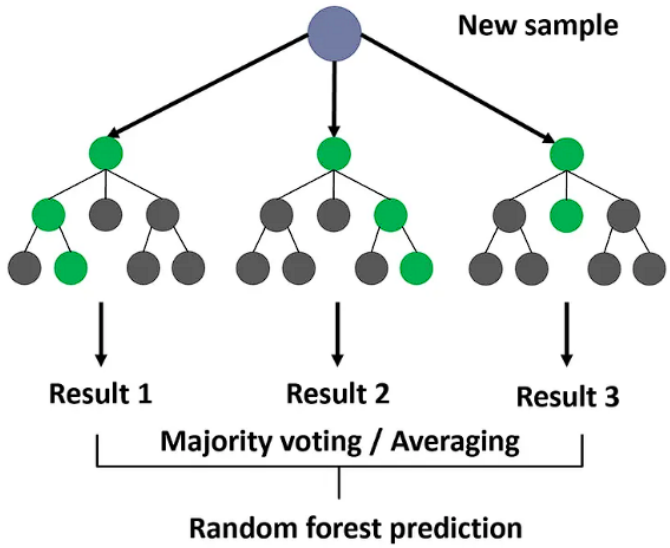


Figure 2.6: Random Forest Model source: (YEHOSHUA, 2023)

### 2.4.2 XGBoost - LightGBM - HistGradBoost

Two other deep decision-tree models used in this work are XGBoost and LightGBM. XGBoost was first introduced in the literature in 2016; it is also a decision treebased model, optimized for inference processing speed, enhancing prediction performance compared to other algorithms, and designed

for scalabilitycapable of processing terabytes of data (CHEN; GUESTRIN, 2016). The LightGBM model also emerged recently, first introduced in 2017. It follows the same principles as XGBoost, aiming to increase processing speed, offering improvements of up to 20 times faster than XGBoost on datasets with more than 10 million rows. Its accuracy is comparable to other state-of-the-art algorithms while significantly reducing computational cost (KE et al., 2017). The Histogram-based Gradient Boosting method was formalized by Ke et al. (2017) as part of LightGBM, although earlier versions of histogram-based approaches had been explored before. This technique leverages histogram binning to accelerate training and reduce memory usage while maintaining high predictive performance (KE et al., 2017).

### 2.4.3 k-Nearest Neighbors (k-NN)

The $k$-Nearest Neighbors (k-NN) algorithm is the only non-parametric machine learning method used in this work. It predicts the output for a new data point based on the average of its $k$ nearest neighbors within the feature space defined by the selected features, and it can be applied to both regression and classification tasks. Unlike decision tree models, k-NN does not perform well with a large number of features; that is, there is a loss of interpretability among the $k$ nearest neighbors when the feature space becomes too complex for the averaging process in regression (COVER; HART, 1967). The Fig. 2.7 demonstrates the k-NN Regression algorithm, comparing predictions using uniform weighting (equal influence for all neighbors) and distance-based weighting (closer neighbors have greater impact), showing how the weights parameter affects the model's smoothness and adaptability to the data.

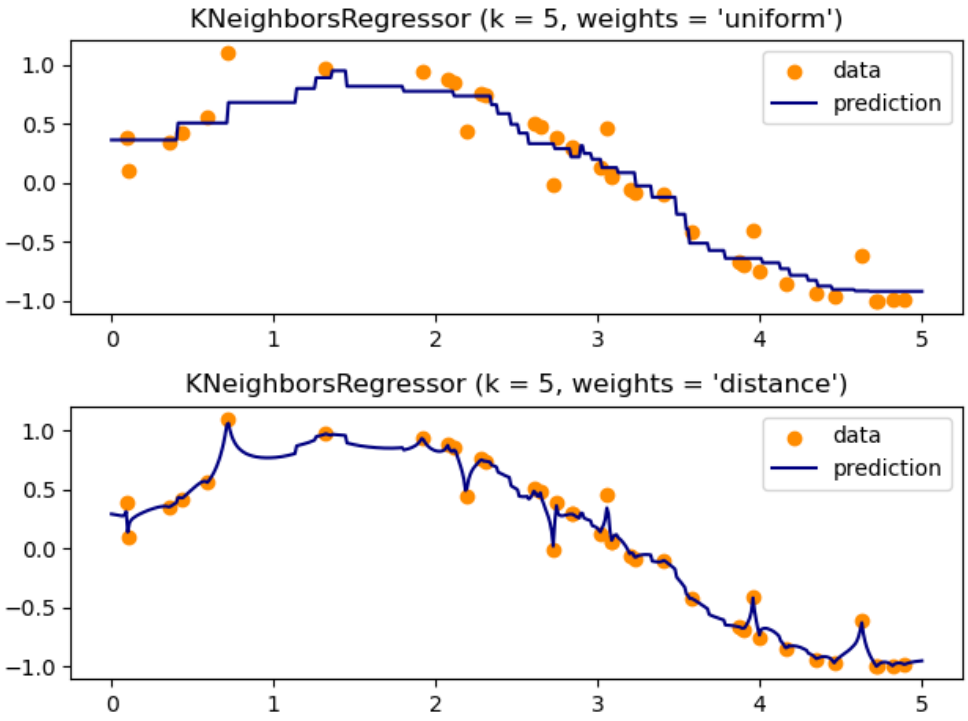


Figure 2.7: K-Nearest Neighbors source: (LEARN, 2023)

# 3
# Related Work

This study covers two main topics in the literature: emissions prediction using GPS data and the application of map-matching techniques. First, we discuss the widely adopted approaches for emissions estimation and the methods employed for prediction in each context, including studies involving the two types of devices mentioned in the introduction: OBD and PEMS. Subsequently, we examine map-matching approaches and the primary categories of applications reported in the literature.

## 3.1
## Vehicular Emission Modeling Approaches

Several approaches in the literature focus on the study of vehicular emissions. Although many studies rely on empirical data, not all incorporate map matching techniques or machine learning models to identify key features influencing emissions. For instance, Hao et al. (2023) propose a methodology based solely on On-Board Diagnostics (OBD) data combined with GPS to estimate $NO_x$ and $CO_2$ emissions from heavy-duty diesel vehicles operating under real-world conditions. However, their method is entirely deterministic, relying on predefined physical equations rather than advanced statistical modeling to predict emissions along road segments. In contrast, Lee et al. (2021) adopt deep learning approaches, particularly artificial neural networks (ANNs), which significantly enhance prediction accuracy. Nevertheless, their model is limited in scope, as it is based on data from only two vehicles and does not account for important vehicle-specific characteristics such as load or vehicle age, which are known to influence emission behavior. Complementing the discussion on different approaches to vehicle emissions, Ntziachristos e Samaras (2000) propose an analysis that is closer to the actual functioning of the vehicle. The authors indicate that the relationship between the catalytic converter (a device installed in the exhaust system of vehicles with the purpose of reducing the amount of pollutants) and the operation of the engine is dynamic. In other words, emissions are influenced in real time, bringing a more realistic perspective to measurements and variations for example, the main emission factor they identify is vehicle speed.

Another approach that considers the dynamic nature of vehicle engines is the study by Ahn et al. (2002). Their research performs a controlled experiment and provides data that adds robustness to the literature on the effects of speed

and acceleration on fuel consumption and pollutant emissions.

The functioning of each vehicle tends to deteriorate and require more maintenance over the years. Even if maintenance is performed, the engine performance is no longer the same as it was in the beginning. Complementing this point, newer vehicle technologies and regulations that encourage engines with lower emissions have proven to be effective, resulting in vehicles emitting less per kilometer traveled (ZACHARIADIS; NTZIACHRISTOS; SAMARAS, 2001). Frey, Rouphail e Zhai (2008) do not apply map-matching techniques in their analysis, but their methodology emphasizes the role of roadway segments in emission modeling. Specifically, they estimate average emission factors based on the type of road segment, meaning that emission calculations are weighted according to the specific link the truck is traveling on, rather than using raw trajectory data.

## 3.2 Fundamentals and Applications of Map Matching in GPS-Based Emission Studies

Kempinska, Davies e Shawe-Taylor (2016) propose a probabilistic map-matching algorithm based on particle filters aimed at improving the accuracy of GPS trajectory alignment to road networks. Although the study does not focus on emission estimation, precise trajectory reconstruction is a fundamental preprocessing step for reliable emission modeling from GPS data (KEMPINSKA; DAVIES; SHAWE-TAYLOR, 2016). Incorporating a robust map-matching step into the preprocessing of GPS data is essential for ensuring the reliability of derived features such as speed, acceleration, and road gradient. Several studies have shown that omitting this step can significantly compromise the accuracy of these features, leading to errors that propagate into subsequent modeling stages. As a result, emission estimates based on poorly aligned trajectories may be considerably biased, particularly in urban environments with complex road geometries and topographical variation.

Quddus, Noland e Ochieng (2005) demonstrated that the application of map matching reduced GPS positional error to within 6 meters, substantially improving the extraction of kinematic features. Similarly, Kealy, Retscher e Brzezinska (2006) showed that combining map matching with odometer calibration lowered mean GPS error from approximately 53.7 meters to 8.8 meters, enhancing the accuracy of elevation and speed estimation. Therefore, precise trajectory alignment not only improves the spatial representation of vehicle movement but also strengthens the consistency and reliability of emission models that depend on such attributes.

Geometric map-matching methods associate each GPS point with the nearest road segment based solely on spatial proximity, which may lead to errors in areas with closely spaced or overlapping roads. In contrast, topological methods consider the connectivity of the road network ensuring a more consistent trajectory by preserving the logical sequence of travel. More advanced approaches include probabilistic methods which model GPS uncertainty and incorporate temporal and motion constraints to enhance matching accuracy especially under noisy or sparse conditions. As highlighted by Chao et al. (2022), probabilistic and advanced map-matching algorithms such as those based on Hidden Markov Models (HMMs) or particle filters offer improved robustness and are particularly well suited for real-world applications with imperfect GPS data.

Although particle filters, such as those used by Kempinska, Davies e Shawe-Taylor (2016), represent a robust approach, the literature reveals an even greater diversity of map-matching algorithms. Additionally, Lou et al. (2009) develop a robust model focused on low-sampling-rate GPS data (around 2 to 5 minutes), concluding that their algorithm significantly improves processing time and accuracy for such data compared to algorithms designed for high-frequency sampling. Finally, Hashemi (2014), in an empirical comparative study, evaluates different map-matching algorithms, highlighting the trade-off between accuracy and computational cost. The author points out that simpler methods are generally faster but less precise, while probabilistic or particle filter-based methods offer higher accuracy at the expense of greater complexity and processing time. This analysis reinforces the importance of selecting the most suitable algorithm according to the specific characteristics of the dataset and the application requirements.

To the best of the authors knowledge, no studies have been found in the literature that simultaneously integrate map matching techniques with machine learning models and focus specifically on the reduction of vehicular emissions. This study aims to fill this gap by proposing an integrated framework and making available a real-world dataset that contributes significantly to understanding and modeling this relationship.

# 4
# Methodology

We illustrate in Fig. 4.1 the modeling pipeline adopted in this study, encompassing the application of a map-matching heuristic, data preprocessing, model training, and evaluation.

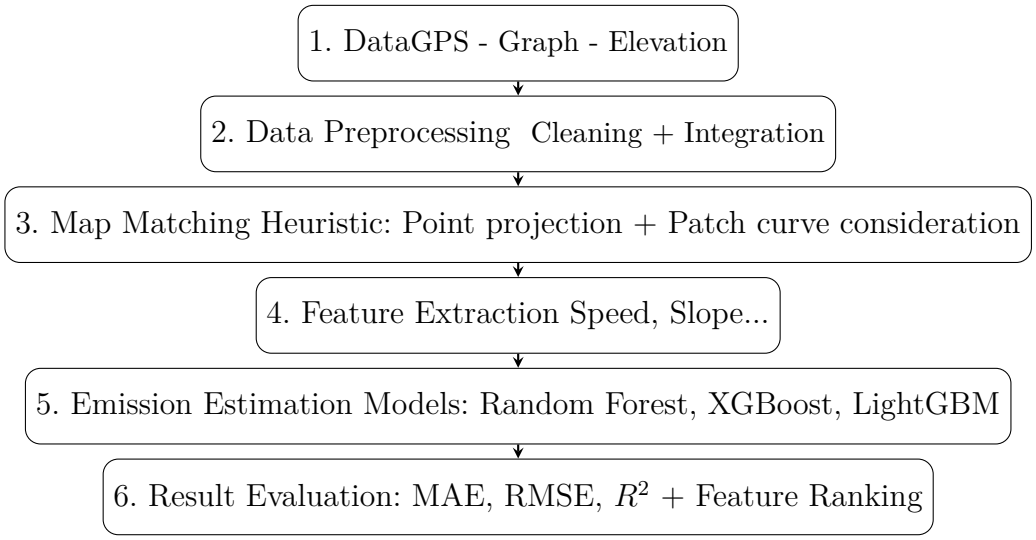


Figure 4.1: Overview of the emission estimation workflow.

The methodology is organized into distinct yet interconnected stages designed to estimate vehicular emissions based on spatiotemporal and contextual data. The process begins with the collection and preparation of GPS and elevation data, followed by the integration of additional vehicle and environmental attributes. After preprocessing and feature construction, supervised learning models are trained and evaluated according to standard performance metrics. The subsequent subsections present the data used and provide a concise description of each methodological component.

## 4.1
## Data Sources: GPS, Graph, and Elevation

The first stage of the workflow combined three key sources of information: GPS-based $NO_x$ measurements, the street network, and elevation data. Vehicle trajectories were collected from PMES onboard sensors, providing minute-by-minute records of position (latitude and longitude), vehicle ID, trip information, and average $NO_x$ concentration. The pollutant was measured in the exhaust system with a zirconium dioxide electrochemical sensor (range 0–1500 ppm, accuracy ±10 ppm), and values outside 100–1500 ppm were discarded.

Concentrations were converted into emission factors (g/kWh) through an empirical calibration:

$$\mathrm{NO}_x\,(\mathrm{g/kWh}) = 6.636 \times 10^{-3} \times \mathrm{NO}_x\,(\mathrm{ppm}) \tag{4-1}$$

GPS data were then temporally aligned with emissions to enable spatiotemporal analysis. The road network of Rio de Janeiro was obtained from OpenStreetMap and represented as a graph of nodes (intersections) and edges (road segments). To focus the analysis, only trips within Rio de Janeiro and Duque de Caxias were retained, identified via spatial join with municipal boundaries (Instituto Brasileiro de Geografia e Estatística (IBGE), 2023). Elevation information, sourced from AMBDATA at 1 km resolution (GUIMARãES, 2021), was linked to road segments to estimate slope, a key determinant of vehicle energy demand and emissions. The integration of these three sources resulted in a unified database, where each vehicle observation was enriched with spatial and topographic attributes, forming the foundation for preprocessing, feature extraction, and modeling.

## 4.2 Data Preprocessing

In the preprocessing stage, it was necessary to integrate three fact tables: the NOx measurements, the trip information, and the vehicle information. The NOx table contains the records of vehicular emissions, while the trip table includes details about each delivery, such as the transported weight and the associated timestamp. Additionally, the vehicle table provides characteristics such as model, brand, and year of manufacture. To enable a proper merge of these datasets, it was essential to adjust the timestamps from the NOx and trip tables, since they were not fully synchronized. Therefore, a temporal approximation strategy was adopted to ensure that each emission record could be correctly matched with its corresponding trip and vehicle information.

Following this integration, it was also necessary to implement filtering procedures to guarantee that the dataset remained consistent and representative of the study context. Geographic coordinates were validated by applying spatial constraints on latitude and longitude, thereby excluding invalid records and restricting the analysis to trips that effectively occurred within the city of Rio de Janeiro. This step was crucial for mitigating the impact of spatial outliers that could otherwise bias the results. Furthermore, an upper bound was imposed on NOx values, since some observations displayed unrealistically high concentrations, reaching approximately 1900 g/kWh. According to the technical specifications of the sensor manufacturer, the maximum reliable

measurement is 1500 g/kWh; thus, this threshold was adopted to improve data reliability and reduce the influence of extreme outliers on subsequent modeling.

During the preprocessing stage, it is important to consider that vehicle attributes, such as the weight being carried, can change throughout the routeespecially in logistics operations involving multiple deliveries. In some datasets, this information may already be embedded or recorded by onboard sensors. However, in many cases, as in this study, it is necessary to reconstruct this feature using complementary sources.

When weight data is not directly available for each point along the trajectory (such as GPS records), it may be necessary to integrate different datasets, such as trip logs or delivery records, to estimate the vehicles weight over time. This involves aligning delivery timestamps with trajectory data, allowing the estimation of the cargo weight at each point along the route. Including this type of feature is essential for improving the accuracy of emission modeling, since vehicle weight directly influences fuel consumption and pollutant emissions.

To capture potentially relevant temporal patterns in vehicle behavior and emission levels, several features were extracted from the `timestamp` column. These features were included to account for variations in traffic and operational patterns, considering that the company's demand may be both fixed and sporadic, with deliveries occurring during early morning hours and on weekends, which can affect vehicle speed and emissions.

- **Day of the week**: extracted in two formats, a categorical feature with the weekday name (e.g., *Monday*, *Tuesday*), suitable for one-hot encoding, and a numeric format (0 to 6), where 0 corresponds to Monday.
- **Time of day**: classified into four periods based on the hour of the observation: *Night* (00:00–05:59), *Morning* (06:00–11:59), *Afternoon* (12:00–17:59), and *Evening* (18:00–23:59).
- **Vehicle age**: computed as the difference between the observation year and the vehicles manufacturing year (`vehicle_year`), this feature is expected to correlate with emission levels due to aging effects.

The GPS dataset contains coordinates spanning a large portion of the state of Rio de Janeiro, including trips beyond the capital. However, the highest trip density is concentrated in the city of Rio de Janeiro. To reduce computational complexity and focus on the most representative area of operation, we restricted the analysis to trips that occurred within the municipalities of Rio de Janeiro and Duque de Caxias. This was achieved by using official municipal boundary shapefiles provided by the (Instituto

Brasileiro de Geografia e Estatística (IBGE), 2023). A spatial join was then performed between the GPS coordinates and the municipal polygons to retain only the observations falling within the defined area of interest.

These preprocessing and feature engineering steps were guided by the hypothesis that vehicle emissions are influenced by both technical and operational factors. Vehicle attributes, such as age and cargo weight, are expected to directly affect NOx levels due to mechanical aging and load-dependent fuel consumption. Temporal features, including time of day and day of the week, were included to capture variations in traffic conditions and operational demand, considering that deliveries may occur during early morning hours, late at night, or on weekends. Spatial factors were also considered, restricting the analysis to the municipalities of Rio de Janeiro and Duque de Caxias to ensure that the dataset reflected the most representative operational area. Together, these assumptions informed the construction and filtering of features, aiming to provide a dataset that accurately represents the conditions under which emissions occur.

## 4.3 Map-matching Heuristic and Feature Extraction

We illustrate below the two algorithms that compose the proposed map-matching heuristic. The main objective of this heuristic is to accurately reconstruct the trajectory followed by the trucks, observation by observation. This allows for the estimation of the vehicles speed along each segment based on the geometries of the road network taking into account road curvature and the exact location of the reference edge where the truck was positioned at the time of the sensor recording. Both algorithms produce a DataFrame suitable for training the predictive models.

The algorithm 1 projects GPS points onto their nearest road edges to accurately associate each observation with a segment of the street network. The input road network (`gdf_roads`) is a GeoDataFrame representing the street mesh extracted from OpenStreetMap (OSM), ensuring a detailed and up-to-date representation of the Rio de Janeiro road infrastructure (OpenStreetMap contributors, 2025). The input GPS points (`gdf_points`) are provided as a GeoDataFrame containing vehicle locations.

A search radius (`search_radius`) parameter is used to define the maximum distance (set here to 60 meters) within which nearby road edges are considered for projection. This radius accounts for practical driving scenarios, where trucks often leave the main roads temporarily while making deliveries, parking, or maneuvering off-road. By allowing the algorithm to search for

candidate edges within this radius, it can better handle GPS points that fall slightly outside the mapped network, improving the robustness and accuracy of the map-matching process.

The output of the algorithm is the original `gdf_points` GeoDataFrame augmented with three additional columns: the identifier of the matched road edge (`matched_road_id`), the projected point on that edge (`projected_point`), and the length of the matched road segment (`matched_road_length`).

**Algorithm 1:** Project GPS points onto nearest road edges

```
Build spatial index for gdf_roads
for each point in gdf_points.geometry do
    Create a buffer of radius search_radius around point
    Find candidate edges intersecting the buffer using the spatial index
    if no candidates found then
        Find candidate edges intersecting the bounds of point
        if still none found then
            Assign NULL values for match ID, projection, and length
            continue
    Compute distances from point to candidate edges
    Select the nearest edge
    Project point onto the nearest edge
    Retrieve matched edge ID and projection length
return Augmented gdf_points
```

The algorithm 2 processes consecutive GPS observations of a vehicle to derive structured movement metrics based on the underlying road network and elevation data. For each pair of consecutive points, it determines whether the vehicle remained on the same road segment or transitioned to another. In both cases, it computes the traveled distance using the road geometry, the elapsed time between timestamps, and the resulting speed in kilometers per hour. Implausible speeds are filtered to ensure data reliability. Additionally, the algorithm calculates topographic features, including initial and final elevation, elevation change, road grade percentage, and whether the segment represents an ascent. Importantly, the algorithm also incorporates a function that accounts for the curvature of each road edge and, consequently, of the corresponding traveled segment, as explained in the Theoretical Background. Moreover, it attempts to correct elevation values by following the adjusted trajectory ob-

tained through the map matching procedure, ensuring that the vertical profile of the route remains consistent with the actual road network. This guarantees that, for each pair of consecutive observations, the derived metrics reflect not only the linear distance but also the geometric and topographic complexity of the trajectory. The output is an enriched dataset containing semantically meaningful features for downstream analysis, such as emissions estimation or vehicle trajectory profiling.

**Algorithm 2:** Compute elevation, distance, and speed metrics between consecutive GPS observations

DataFrame *df* with GPS points, matched road edges, and projections; road network graph $G$ *df* with distance, speed, elevation change, and road grade metrics

Initialize new columns in *df* for distance, elevation, road grade, time, and speed;

**for** $i = 0$ **to** $|df| - 2$ **do**

  $row_{curr} \leftarrow df[i]$; $row_{next} \leftarrow df[i+1]$;

  **if** $matched\_road\_id_{curr} == matched\_road\_id_{next}$ **then**

    Compute distance between projected points on the same edge;

  **else**

    • Distance from $row_{curr}$ to the end of its edge; • Shortest path along $G$ between end of current edge and start of next edge; • Distance from $row_{next}$ to the start of its edge; Sum the three distances to get total traveled distance;

  Compute time difference $\Delta t$ between timestamps; Compute speed $v = \frac{d}{\Delta t}$ in m/s and convert to km/h;

  **if** *speed within valid range* **then**

    Compute elevation change and classify as ascent/descent; Compute road grade percentage; Update *df* at index $i$ with all computed metrics;

**return** *df*

The aforementioned algorithms aim to correct the trajectory estimated using the Haversine method by aligning it with the shortest path over the road network. This adjustment leads to greater accuracy in estimating vehicle speed based on the actual distance traveled, while also considering the curvature of the road edges for each pair of consecutive observations. In addition, the elevation profile is refined by following the map-matched trajectory, which allows for a more reliable estimation of vertical movement and road grade in

line with the actual road network.

Fig. 4.2 illustrates this difference: the left side shows a segment of the trajectory calculated using the Haversine method, while the right side presents the proposed map-matching heuristic, which aligns the trajectory more closely with the actual road layout.

Figure 4.2: Comparison between the trajectory obtained using the Haversine method (left) and the trajectory adjusted by the proposed map matching algorithm (right).

The final dataset used in this study was built through a series of preprocessing steps, which included the spatial filtering of GPS data, the integration of vehicle and trip metadata, the creation of temporal features, and the computation of speed and acceleration based on map-matched trajectories.

The resulting `DataFrame` includes continuous and categorical features, each contributing relevant dimensions that will be further discussed and analyzed in the results section.

Continuous features capture physical and mechanical aspects, such as speed, acceleration, elevation change, and vehicle weight. Categorical features enable the segmentation of NOx emission behavior across different vehicles. Table 4.1 summarizes all features used in the analysis.

Table 4.1: Description of the features used in the model.

| **Feature** | **Description** |
|---|---|
| *Continuous features* | |
| speed_kmh | Estimated average speed (km/h) |
| acceleration_m_s2 | Estimated average acceleration (m/sš) |
| elevation_diff | Elevation difference (m) |
| elevation_abs | Absolute elevation (m) |
| vehicle_age | Vehicle age (years) |
| current_weight | Estimated vehicle weight (kg) |
| *Categorical features* | |
| day_of_week | Day of the week |
| time_of_day | Time of day |
| vehicle_id | Vehicle identifier |

The features presented in Table 4.1 constitute the final dataset used in this study. The combination of continuous, binary, and categorical features captures both the physical characteristics of vehicle operation and the contextual conditions of each trip. This diversity is essential for a comprehensive understanding of the factors influencing NOx emissions and will serve as the foundation for evaluating the models discussed in the following section.

## 4.4 Exploratory Analysis, Training and Evaluation of Machine Learning Models

Before training the models, an exploratory analysis was conducted on the pre-processed data. This included examining the number of observations per vehicle, the distribution of the target feature ($NOx$), scatter plots comparing new features, speed, and acceleration with $NOx$, and the distributions of the new features. Speed clusters were also analyzed, separating observations into low, medium, and high-speed groups. Additionally, statistical analyses of the features were performed to compare the original model based on the Haversine distance with the map-matched model. This exploratory analysis guided the selection of the machine learning models and informed decisions regarding feature normalization when necessary.

Following this analysis, multiple regression models were trained to predict $NOx$ concentrations in vehicles using features related to vehicle dynamics, vehicle characteristics, and temporal context. Three types of predictors were considered: continuous, binary, and categorical features. Continuous features included speed (*speed_kmh*), acceleration (*acceleration_m_s2*), abso-

lute elevation (*elevation_abs*), vehicle age (*vehicle_age*), current weight (*current_weight*), and elevation difference (*elevation_diff*). The binary feature was the uphill flag (*uphill_flag*), and categorical features were day of the week (*day_of_week*), time of day (*time_of_day*), and vehicle identifier (*vehicle_id*).

Continuous features were normalized using *MinMax Scaling* to ensure all values were between 0 and 1, while categorical features were encoded using *One-Hot Encoding*. The binary feature *uphill_flag* was explicitly converted to integer for consistency.

The dataset was split into training (70%) and testing (30%) sets with a fixed random state to ensure reproducibility. Several regression models were trained, including Random Forest, XGBoost, Gradient Boosting, K-Nearest Neighbors (KNN), HistGradientBoosting, and LightGBM. Each model was selected to capture different types of relationships in the data: tree-based models (Random Forest, XGBoost, Gradient Boosting, HistGradientBoosting, and LightGBM) are capable of modeling non-linear interactions without assuming linearity among features; KNN captures local patterns based on similarity in the feature space; and linear or additive models (if included) assume linear relationships between predictors and the target. The specific functionality and assumptions of each model were described in detail in Chapter Theoretical Background. The training of these models was implemented within an integrated pipeline, which ensured that the pre-processed features were consistently applied across all stages, from normalization and encoding to model fitting and evaluation.

Model performance was evaluated using multiple metrics: MAE (Mean Absolute Error), RMSE (Root Mean Squared Error), $R^2$ (coefficient of determination), MAE (%) and RMSE (%) relative to the mean of the target feature, and MAPE (Mean Absolute Percentage Error). These metrics allowed for consistent comparison between models and identification of the best-performing approach.

For tree-based models (Random Forest, XGBoost, and LightGBM), feature importance was computed to determine the contribution of each feature to the predictions. This analysis highlighted the most influential factors affecting *NOx* emissions, such as speed, elevation difference, and vehicle weight.

Results were summarized in a table sorted by RMSE to facilitate comparison among models, and feature importance plots were generated to visually present the most relevant features. This step enabled the selection of the most efficient model while providing insights into the main determinants of *NOx* emissions.

## 4.5 Experimental Setup

All experiments were conducted in Python 3.13.5 using the Jupyter Notebook IDE. The code was executed on a personal computer with an AMD Ryzen 5 5500U processor, a 6-core, 12-thread CPU from the Ryzen 5000 series (Lucienne generation) running at a base clock of 2.1 GHz and a maximum boost of 4.0 GHz, with an integrated Radeon Graphics GPU, 20 GB of RAM, and Windows 11 (64-bit).

Table 4.2: Average processing time for each step of the data workflow

| **Step** | **Description** | **Average Time** |
|---|---|---|
| Algorithm 1 | Project GPS points onto nearest road edges | 13 minutes |
| Algorithm 2 | Compute elevation, distance, and speed metrics between consecutive GPS observations | 1 hour |
| Final Pipeline | Processing the final model input after all pre-processing steps | 4 minutes |

Table 4.2 shows that the most time-consuming step in the workflow was computing elevation, distance, and speed metrics between consecutive GPS observations, which took on average 1 hour per trip. Overall, the processing times indicate that the pipeline is efficient, with the final step for preparing the data for model input requiring only a few minutes.

# 5
# Results and Discussion

In this section, we will thoroughly discuss the results obtained at each stage of data preprocessing, from cleaning and integrating various data sources to feature engineering for the machine learning models. Additionally, we will present and analyze the performance metrics of the developed models, highlighting factors that influenced prediction quality and the importance of selected features. This analysis will provide insight into both the effectiveness of the data processing steps and the models ability to capture patterns related to vehicle emissions.

## 5.1
## Results of Pre-processing Steps

Table 5.1 provides a summary of the vehicles analyzed in this study, including the model, year of manufacture, and the number of valid observations after preprocessing with map-matching heuristics. This overview highlights the diversity of the fleet and the amount of trajectory data available per vehicle, which is essential for the subsequent emission modeling. The study area is limited to the municipalities of Rio de Janeiro and Duque de Caxias. Due to sensor malfunctions, the statistics presented in Table 5.1 are also used as the basis for the experiments conducted without the map-matching procedure.

Table 5.1: Summary of the Vehicles Used in the Analysis

| Vehicle ID | Vehicle Model | Vehicle Type | Year | Obs. Count |
|---|---|---|---|---|
| 001 | Model A | smallTruck | 2013 | 46170 |
| 002 | Model B | mediumTruck | 2014 | 31692 |
| 003 | Model A | smallTruck | 2013 | 26900 |
| 004 | Model B | mediumTruck | 2014 | 25816 |
| 005 | Model B | mediumTruck | 2013 | 21159 |
| 006 | Model C | mediumTruck | 2013 | 19327 |
| 007 | Model A | smallTruck | 2013 | 19252 |
| 008 | Model D | smallTruck | 2019 | 16997 |
| 009 | Model B | mediumTruck | 2020 | 12020 |
| 010 | Model B | mediumTruck | 2013 | 8227 |
| 011 | Model B | mediumTruck | 2013 | 5529 |

Figure 5.1 displays the distribution of the target feature (NOx). Special attention should be given to the distributions tail behavior particularly the kurtosis as it may influence model sensitivity to outliers and impact performance metrics discussed in the results section.

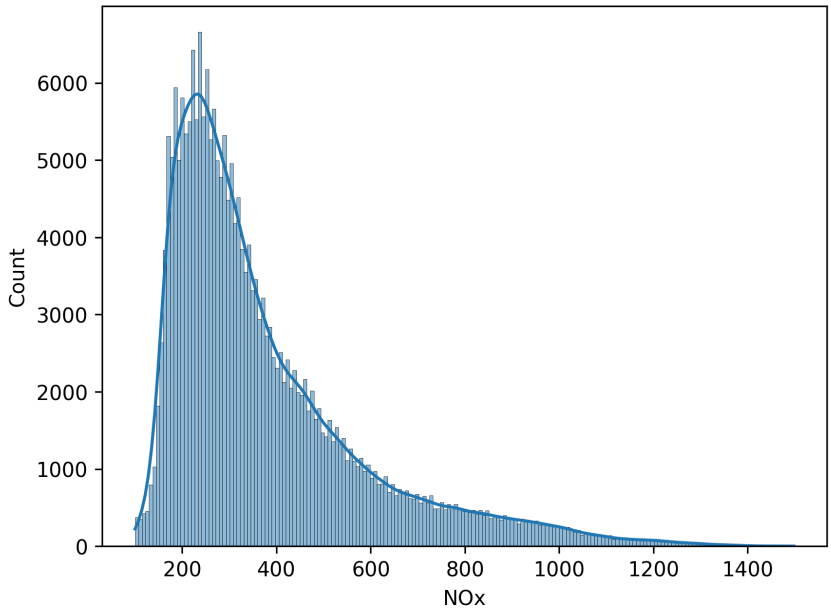


Figure 5.1: Distribution of NOx (nitrogen oxides) emissions recorded for the vehicles in the sample. The distribution is right-skewed (skewness = 1.59), with slightly heavier tails than the normal distribution (kurtosis = 2.53), indicating the presence of extreme emission values.

Following the preprocessing steps, we applied map-matching heuristics with the aim of refining vehicle trajectories and, more importantly, obtaining the velocity and acceleration features. The elevation feature is now applied to the projection of the corrected observation obtained through map-matching, which brings a subtle adjustment compared to using only the original latitude and longitude provided by the GPS. For comparison purposes, as illustrated in Fig. 4.2, the Haversine model is also considered in the statistics and visualizations presented after preprocessing.

Fig. 5.2 highlights the small differences between the Haversine model and the map-matching heuristic. These differences, observed in the velocity and acceleration features, indicate that a large portion of the trips takes place on expressways, where trucks are not frequently subjected to sharp turns. Additionally, the scatterplot of NOx versus acceleration shows that the acceleration values obtained through map-matching are more concentrated and consistent, whereas the values derived from the Haversine model display a wide and unrealistic dispersion.

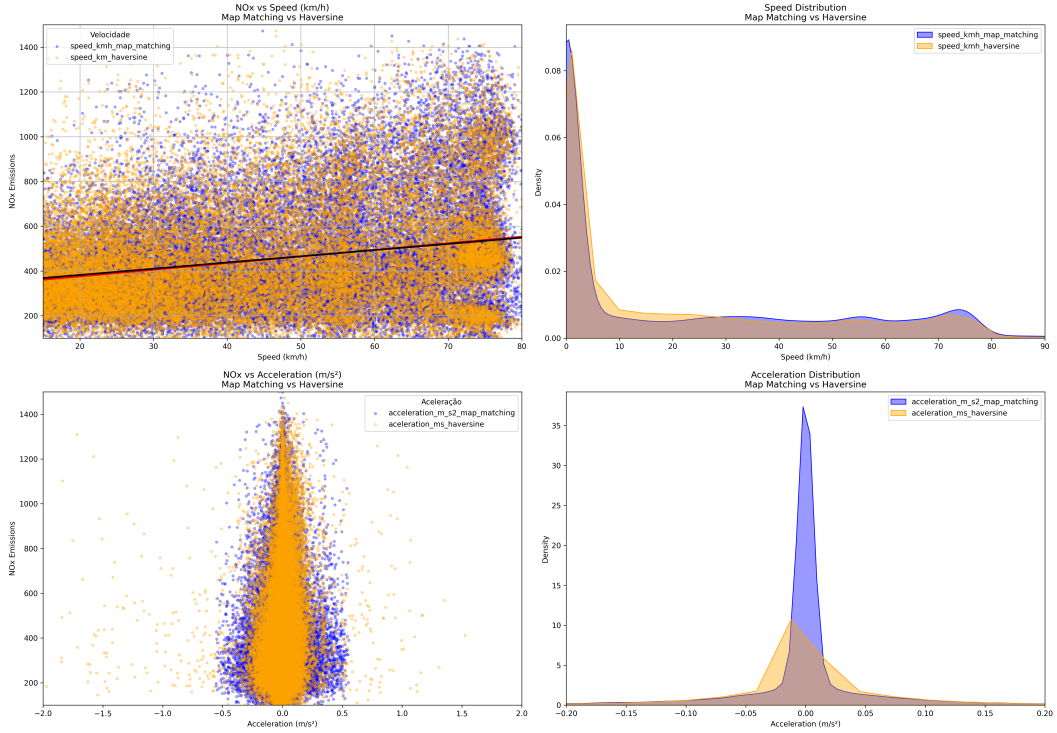


Figure 5.2: Comparison between speed and acceleration estimated by Map Matching and Haversine. Top: relationship with $NO_x$ (left) and distributions (right). Bottom: same for acceleration.

According to the optimized model with map matching, the regression line indicates that vehicles in this study emit, on average, **363.49 g/h** of NOx when the speed is **0 km/h**. In other words, even when stationary, the engine continues to run and release pollutants.

The next figure Fig. 5.3 illustrates how vehicle speed and NOx emissions behave across defined speed ranges. This representation brings the analysis closer to real-world driving conditions by associating emission patterns with the typical speed limits of different road types. Understanding these patterns helps identify which road segments trucks are likely to access and, when available, what their average speeds are. The statistical distribution of NOx emissions by speed range can inform routing strategies that prioritize specific types of roads to reduce overall emissions in urban areas.

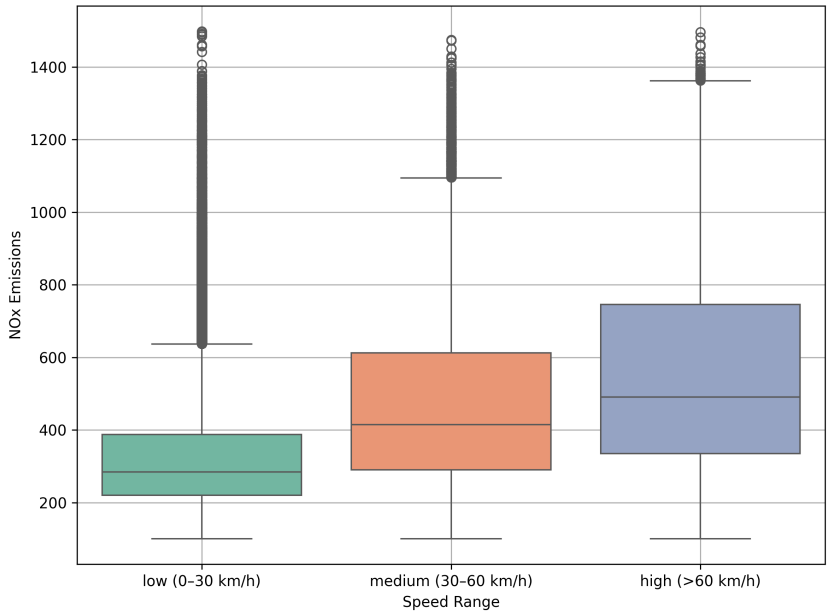


Figure 5.3: Boxplot of NOx emissions grouped by speed range (low, medium, high). This figure highlights how emissions vary according to typical road speed categories.

Table 5.2 shows that there are no significant statistical gains between the models. features related to elevation (`elevation_m`) remain identical, suggesting that the interpolation of projected elevation with or without map matching results in equivalent values. Regarding speed (`speed_kmh`), we observe an increase in both the mean and median after applying the map matching method, accompanied by a substantial decrease in the maximum recorded value. This indicates that the heuristic may be correcting anomalous peaks present in the haversine-based model.

Table 5.2: Descriptive statistics of the features with and without map matching

| **Feature** | **Metric** | **Haversine - standard projection** | **With Map Matching** | **Difference** |
|---|---|---|---|---|
| `speed_kmh` | Mean | 18.72 | 21.15 | +2.43 |
| | Std Dev | 27.83 | 27.69 | -0.14 |
| | Min | 0.00 | 0.00 | 0.00 |
| | 25% | 0.11 | 0.01 | -0.10 |
| | Median | 1.77 | 1.53 | -0.24 |
| | Max | 843.75 | 119.98 | -723.77 |
| `acceleration_m_s2` | Mean | -0.0006 | -0.0002 | +0.0004 |
| | Std Dev | 0.0824 | 0.0757 | -0.0067 |
| | Min | -3.73 | -0.55 | +3.18 |
| | 25% | -0.0031 | -0.0012 | +0.0019 |
| | Median | 0.00 | 0.00 | 0.00 |
| | Max | 1.95 | 0.56 | -1.40 |
| `elevation_abs` | Mean | 26.08 | 26.08 | 0.00 |
| | Std Dev | 50.48 | 50.48 | 0.00 |
| | Min | -75.00 | -75.00 | 0.00 |
| | 25% | 5.00 | 5.00 | 0.00 |
| | Median | 10.00 | 10.00 | 0.00 |
| | Max | 541.00 | 541.00 | 0.00 |
| `elevation_diff` | Mean | 0.0109 | 0.0648 | +0.0539 |
| | Std Dev | 16.28 | 13.76 | -2.52 |
| | Min | -532.00 | -410.00 | +122.00 |
| | 25% | 0.00 | 0.00 | 0.00 |
| | Median | 0.00 | 0.00 | 0.00 |
| | Max | 531.00 | 423.00 | -108.00 |

Acceleration (`acceleration_m_s2`) also shows a slight reduction in both standard deviation and maximum value, suggesting smoother trajectories with the heuristic approach. Moreover, the higher mean value of delta elevation in the map-matched model may reflect greater sensitivity of the heuristic to actual terrain variations, likely due to better alignment with the road network edges.

Overall, the heuristic method yields minor improvements in data coherence without compromising the statistical stability of the features used in modeling.

Fig. 5.4 shows the altimetric distribution of the GPS points collected from the fleet of vehicles along the routes traveled in the cities of Rio de Janeiro and Duque de Caxias. The map results from the application of map matching for elevation inference, providing trajectories more accurately aligned with the actual road network. In the following sections, we apply machine learning models to assess the importance of each proposed feature in explaining NOx emissions.

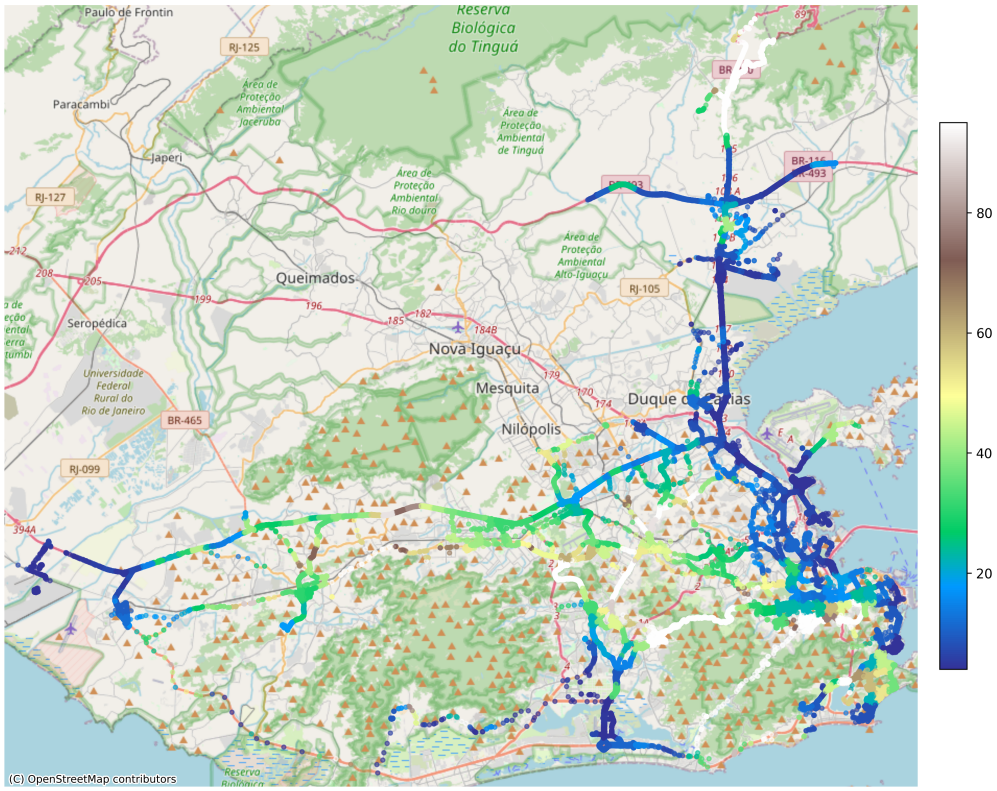


Figure 5.4: Spatial distribution of the fleet's GPS elevation points projected onto the OpenStreetMap base map. Point colors represent altitude in meters, with the color scale constrained between the 5$^{th}$ and 95$^{th}$ percentiles to enhance the visualization of local elevation variations.

## 5.2 Results of Models

The results presented in Table 5.3 compare the predictive performance of machine learning models trained using two different methodologies for processing GPS data: a simplified approach based on the haversine formula for speed calculation and elevation projection, and a more precise map-matching technique that aligns GPS points with the road network. While the map-matching method aims to improve feature accuracy, the differences in model performance metrics between the two approaches are generally modest. This suggests that, for this dataset and context, simpler processing techniques may be sufficient for reasonably accurate NOx emission predictions. We split the dataset into 70% for training and 30% for testing. The training set was used to fit the machine learning models, and the testing set was kept entirely separate during training and used as a hold-out validation set to evaluate the generalization performance. This approach ensures that models are assessed on previously unseen data.

All models were configured with fixed random seeds to preserve reproducibility. Hyperparameters were kept consistent across experiments to facilitate comparative analysis: tree-based models (Random Forest, Gradient Boosting, LightGBM, and XGBoost) were set with 300 estimators, with learning rates of 0.1 for LightGBM and 0.01 for XGBoost, respectively. The KNN regressor, implemented with $k = 5$ neighbors, adopts a non-parametric, instance-based strategy, relying on local data similarity rather than model training. HistGradientBoosting, while tree-based, uses histogram binning for feature discretization, enhancing computational efficiency on large datasets.

Table 5.3: Comparison of MAPE (%) and $R^2$ for models with and without map matching

| **Model** | **With Map Matching** | | **With Haversine** | |
|---|---|---|---|---|
| | **MAPE (%)** | $R^2$ | **MAPE (%)** | $R^2$ |
| Random Forest | **24.55** | **0.645** | 24.57 | 0.645 |
| LightGBM | 28.66 | 0.591 | 28.64 | 0.590 |
| HistGradBoost | 30.31 | 0.552 | 30.28 | 0.554 |
| KNN | 27.35 | 0.551 | 27.37 | 0.551 |
| XGBoost | 32.30 | 0.510 | 32.30 | 0.510 |
| GradBoost | 32.82 | 0.498 | 32.90 | 0.495 |

Among the tree-based models evaluated, the Random Forest model stands out with the best results, achieving an $R^2$ of 0.645 and a MAPE of 24.55% This indicates that tree-based models provide a reasonable predictive capacity when using GPS sensor data collected at relatively large time intervals.

Fig. 5.5 illustrates the feature importance from the best performing model, Random Forest. Consistent with previous studies, speed and acceleration emerge as the most influential features, as they directly affect engine operation and fuel consumption, thereby impacting NOx emissions (ZHANG; BATTERMAN, 2013).

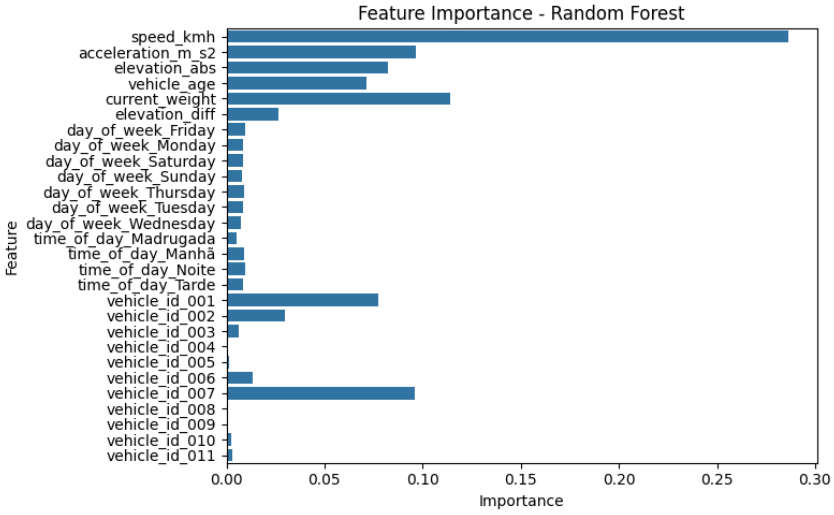


Figure 5.5: Feature importance from Random Forest model.

Fig. 5.5 presents the features that most influence $NO_x$ emissions in the Random Forest model using map-matched data. As expected, vehicle speed appears as the most important feature, which is consistent with Ntziachristos e Samaras (2000) and Ahn et al. (2002), who identify speed as a key factor in emission behavior. Other relevant features include current_weight, elevation, acceleration, and vehicle age. These results support the notion that both the vehicles operational state and the road environment directly affect $NO_x$ emissions.

The feature current_weight reflects the load being carried by the vehicle at each moment and ranks among the top predictors in the model. Heavier loads demand greater engine effort, particularly on uphill segments or in situations involving frequent acceleration, thereby increasing emissions. Despite its relevance, such a feature is often not incorporated in emission models. Lee et al. (2021), for example, do not account for vehicle load or age, which are known to impact emissions significantly.

Vehicle-specific identifiers (e.g., vehicle_number_001, vehicle_number_007) also appear as influential, suggesting that emissions vary across vehicles likely due to differences in engine wear, maintenance levels, or compliance with emission standards. This observation aligns with Zachariadis, Ntziachristos e Samaras (2001), who discuss how aging and technological differences among engines affect emission patterns over time.

The presence of road-related features such as elevation is also noteworthy and consistent with the approach proposed by Frey, Rouphail e Zhai (2008), who emphasize the role of roadway characteristics in emission modeling. Their method estimates average emission factors based on segment type, even though it does not incorporate map matching. In contrast, the model presented here

combines map-matched data with machine learning techniques, allowing for a more detailed and data-driven analysis of how emissions vary along different road segments and vehicle conditions.

Figure 5.6 presents the partial dependence plots (PDP) for the selected features in the NOx prediction model. In the following discussion, each feature is individually analyzed, highlighting its observed relationship with NOx emissions and the potential influence of the map matching process on these patterns.

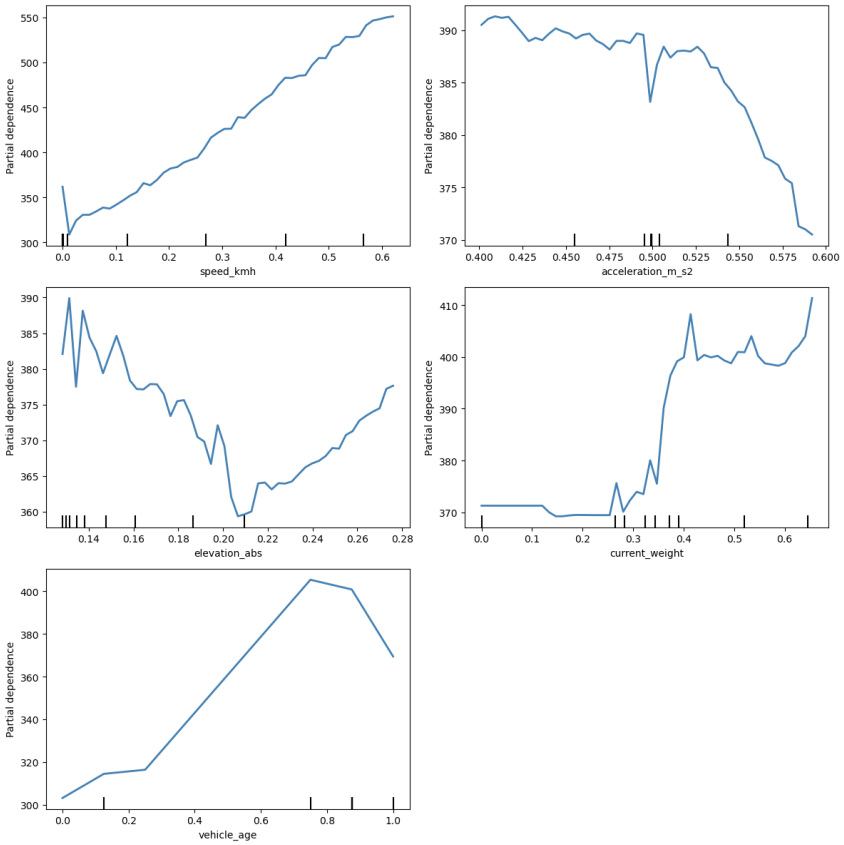


Figure 5.6: Partial dependence plots (PDP) for the five most relevant features in the NOx prediction model.

The analysis of the partial dependence plots (PDP) revealed relevant patterns in the relationship between vehicular and environmental features and the predicted NOx emissions, with their interpretation strongly linked to the accuracy of the map matching process used to align GPS trajectories to the road network. The feature speed_kmh showed a strongly increasing effect, indicating that higher average speeds are associated with significant increases in predicted emissions. Inaccuracies in map matching could lead to over or underestimation of speed, directly affecting this observed relationship. The acceleration_m_s2 feature presented a relatively stable pattern up to approximately $0.53 m/s^2$, followed by a sharp drop in predictions. Since acceleration

is computed from sequentially matched positions, any spatial misalignment could smooth or exaggerate acceleration profiles, thereby altering the modeled effect on NOx. The elevation_abs feature exhibited a non linear behavior, with emissions decreasing until around 0.22 and then increasing again, a pattern that may reflect variations in engine load associated with topographic changes; here, precise map matching to road segments with correct elevation attributes is critical for capturing such effects. For current_weight, emissions remained stable for lower values but increased significantly from approximately 0.35, reinforcing the influence of transported load on energy demand and emissions; in this case, map matching affects the weights interaction with slope and speed profiles derived from the matched route. Finally, vehicle_age displayed an increasing relationship up to about 0.8 (older vehicles), followed by a slight decline, possibly reflecting the lower efficiency of aging vehicles and specific patterns of maintenance or usage within the analyzed fleet. Although this feature is not directly impacted by spatial positioning, its interaction with other map matched features (such as slope and speed) can influence the modeled effect on NOx emissions.

# 6
# Conclusion

This study investigated the impact of applying map-matching techniques on the predictive performance of machine learning models for estimating NOx emissions in heavy-duty vehicles. Our results indicate that map matching does not lead to a significant improvement in overall model accuracy when compared to simplified methods for computing speed and elevation, particularly in expressway-dominated contexts where routes tend to be smooth and less complex. As illustrated in Figure 5.2 and Figure 5.4, the distribution between the two models does not show substantial differences, but rather a subtle adjustment that reinforces the notion that the trajectories are more linear.

Nonetheless, feature importance analysis revealed that speed, which is directly influenced by the choice of spatial computation method (Haversine model vs. Heuristic model), is consistently the most critical predictor across all models evaluated. Although map matching did not improve accuracy at the aggregate level, its application played a crucial role in refining this key feature. By enabling more realistic representations of travel distance and speed profiles, map matching indirectly contributes to the robustness of NOx emission estimates.

In addition, the analysis highlighted the relevance of other spatial and operational features, such as elevation difference and vehicle weight, which, despite having lower overall importance scores, may contribute to fine-tuning models in specific conditions or regions. These insights underscore the importance of carefully selecting and engineering features in transportation emission modeling.

Future research should aim to increase the granularity of the road network by proposing a new feature to represent curve angularity, such as distinguishing between sharp and moderate turns, since curvature itself is already considered during the map-matching process; this would allow the models to better capture the effect of driving dynamics on emissions, as sharper turns typically require changes in acceleration and gear shifting that may influence pollutant levels. In addition, future studies could benefit from more refined datasets that incorporate higher-frequency sensor data, allowing for more precise capture of instantaneous speed fluctuations and transient operating conditions, which would enhance the accuracy of emission modeling, since rapid variations in speed are strongly linked to fuel consumption and emission spikes. The inclusion of additional exogenous features, such as weather and microclimate data,

may also provide valuable insights into how environmental conditions affect emission patterns, because precipitation and temperature influence road friction and combustion efficiency, which in turn alter emission levels. Another direction is to enhance the robustness of the map-matching algorithm by connecting previous and subsequent observations to ensure greater path consistency and coherence, thereby reducing trajectory noise and misclassification of road segments and leading to more reliable estimation of vehicle dynamics and emissions. Expanding the analysis to different vehicle categories beyond heavy-duty trucks, including light-duty vehicles and buses, would help assess the generalizability of the findings, making it possible to evaluate whether the proposed methods are adaptable to different driving behaviors and operational contexts. Finally, integrating emerging data sources, such as connected vehicle telemetry and real-time traffic information, could further improve the robustness of predictive models and support their application in dynamic traffic management systems, since accounting for live traffic conditions and vehicle-to-infrastructure communication would make these models more applicable for policy design and real-world operational scenarios.

# 7
# Bibliography


AHN, K. et al. Estimating vehicle fuel consumption and emissions based on instantaneous speed and acceleration levels. **Journal of Transportation Engineering**, American Society of Civil Engineers, v. 128, n. 2, p. 182–190, 2002.

ALHARTHI, H. et al. Systematic ensemble model selection approach for educational data mining. **Knowledge-Based Systems**, v. 196, p. 105810, 2020. ISSN 0950-7051. Disponível em: <https://www.sciencedirect.com/science/article/pii/S0950705120302999>.

BERG, M. d. et al. **Computational Geometry: Algorithms and Applications**. 3rd. ed. [S.l.]: Springer, 2008.

BREIMAN, L. Random forests. **Machine Learning**, v. 45, n. 1, p. 5–32, 2001.

CHAO, P. et al. A survey on map-matching algorithms. **arXiv preprint arXiv:1901.02723**, 2019.

CHAO, P. et al. A survey on map-matching algorithms. **IEEE Transactions on Intelligent Transportation Systems**, 2022.

CHAWATHE, S. S. Segmentbased map matching. In: **2007 IEEE Intelligent Vehicles Symposium**. [S.l.: s.n.], 2007. p. 1190–1197.

CHEN, T.; GUESTRIN, C. Xgboost: A scalable tree boosting system. In: ACM. **Proceedings of the 22nd ACM SIGKDD**. [S.l.], 2016. p. 785794.

Confederação Nacional do Transporte. **Pesquisa CNT de Rodovias 2023**. Brasília, Brasil, 2023. Acesso em: 19 jun. 2025. Disponível em: <https://cdn.cnt.org.br/diretorioVirtualPrd/907973a7-6dc6-4006-b683-9e6ef6bc1505.pdf>.

COVER, T. M.; HART, P. E. Nearest neighbor pattern classification. **IEEE Transactions on Information Theory**, IEEE, v. 13, n. 1, p. 21–27, 1967.

EITER, T.; MANNILA, H. Computing discrete fréchet distance. **Technical Report CD-TR 94/64, Christian Doppler Laboratory**, 1994.

FANG, K.; LI, J. et al. Real-world measurement and mechanical-analysis-based verification of nox and co emissions from an in-use heavy-duty vehicle. **Atmospheric Measurement Techniques**, v. 14, p. 21152131, 2021.

FREY, H. C.; ROUPHAIL, N. M.; ZHAI, H. Linkbased emission factors for heavyduty diesel trucks based on realworld data. **Transportation Research Record: Journal of the Transportation Research Board**, SAGE Publications, n. 2058, p. 23–32, 2008.

GUIMARãES, R. **WorldClim v2.1 Brazil subset with PCA components at 1km resolution**. 2021. <https://www.worldclim.org/data/index.html>. Organized for Honorato et al. (2021), Instituto Evandro Chagas/SVSA/MS.

HAO, L. et al. Assessment of heavy-duty diesel vehicle $no_x$ and $co_2$ emissions based on obd data. **Atmosphere**, MDPI, v. 14, n. 9, p. 1417, 2023.

HASHEMI, S. M. An empirical study of map matching algorithms. **ISPRS International Journal of Geo-Information**, v. 3, n. 4, p. 1409–1437, 2014.

HUTTENLOCHER, D. P.; KLANDERMAN, G. A.; RUCKLIDGE, W. J. Comparing images using the hausdorff distance. **IEEE Transactions on Pattern Analysis and Machine Intelligence**, IEEE, v. 15, n. 9, p. 850–863, 1993.

Instituto Brasileiro de Geografia e Estatística (IBGE). **Malhas Territoriais**. 2023. Available at: <https://www.ibge.gov.br/geociencias/organizacao-do-territorio/malhas-territoriais/15774-malhas.html>. Accessed: July 9, 2025.

KAMP, B. **Particulate Matter Sensor for On Board Diagnostics (OBD) of Diesel Particulate Filters (DPF)**. [S.l.], 2010.

KE, G. et al. Lightgbm: A highly efficient gradient boosting decision tree. **Advances in Neural Information Processing Systems**, v. 30, 2017.

KEALY, A.; RETSCHER, G.; BRZEZINSKA, D. A. Gps accuracy estimation using map matching for odometer calibration. In: INTERNATIONAL ASSOCIATION OF INSTITUTES OF NAVIGATION. **Proceedings of the International Symposium on GPS/GNSS**. 2006. p. 1–6. Disponível em: <https://www.researchgate.net/publication/268748204>.

KEMPINSKA, K.; DAVIES, T.; SHAWE-TAYLOR, J. A probabilistic map-matching algorithm based on particle filters. **Transportation Research Part C: Emerging Technologies**, v. 68, p. 45–57, 2016.

LEARN scikit. **Nearest Neighbors Regression**. 2023. Disponível em: <https://scikit-learn.org/stable/auto_examples/neighbors/plot_regression.html>.

LEE, J. et al. Machine learning applied to the $no_x$ prediction of diesel vehicle under real driving cycle. **Applied Sciences**, MDPI, v. 11, n. 8, p. 3758, 2021.

LOU, Y. et al. Map-matching for low-sampling-rate gps trajectories. In: ACM. **Proceedings of the 17th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems**. [S.l.], 2009. p. 352–361.

NEWSON, P.; KRUMM, J. Hidden markov map matching through noise and sparseness. In: ACM. **Proceedings of the 17th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems**. [S.l.], 2009. p. 336–343.

NTZIACHRISTOS, L.; SAMARAS, Z. Speed-dependent representative emission factors for catalyst passenger cars and influencing parameters. **Atmospheric Environment**, Elsevier, v. 34, n. 27, p. 4611–4619, 2000.

OpenStreetMap contributors. **OpenStreetMap**. 2025. <https://www.openstreetmap.org>. Accessed: June 24, 2025.

PEUCKER, T. K.; DOUGLAS, D. H. Detection of surface-specific points by local parallel processing of discrete terrain elevation data. **Computer Graphics and Image Processing**, v. 4, p. 375–387, 1975.

QUDDUS, M. A.; NOLAND, R. B.; OCHIENG, W. Y. Validation of map-matching algorithms using high-precision positioning with an integrated gps and inertial navigation system. **Journal of Navigation**, Cambridge University Press, v. 58, n. 2, p. 257–271, 2005.

ROBUSTO, C. The cosine-haversine formula. **The American Mathematical Monthly**, Mathematical Association of America, v. 64, n. 1, p. 38–40, 1957.

TRIANTAFYLLOPOULOS, G. et al. A study on the co and no emissions performance of euro 6 diesel vehicles under various chassis dynamometer and on-road conditions including latest regulatory provisions. **Science of The Total Environment**, v. 665, p. 1118–1128, 2019. Disponível em: <https://doi.org/10.1016/j.scitotenv.2019.02.144>.

WANG, M. et al. Rapid and highprecision cavityenhanced spectroscopic measurement of hono and no: Application to emissions from heavyduty diesel vehicles in chassis dynamometer tests and in mobile monitoring. **Talanta**, v. 285, p. 127386, 2024.

WHITE, C.; BERNSTEIN, D.; KORNHAUSER, A. Some map-matching algorithms for personal navigation assistants. **Transportation Research Part C: Emerging Technologies**, Elsevier, v. 8, n. 1-6, p. 91–108, 2000.

YEHOSHUA, R. **Random Forests Understanding Ensemble Methods**. 2023. Medium article. Disponível em: <https://medium.com/@roiyeho/random-forests-98892261dc49>.

ZACHARIADIS, T.; NTZIACHRISTOS, L.; SAMARAS, Z. The effect of age and technological change on motor vehicle emissions. **Transportation Research Part D: Transport and Environment**, Elsevier, v. 6, n. 3, p. 221–227, 2001.

ZHANG, K. M.; BATTERMAN, S. Air pollution and health risks due to vehicle traffic. **Science of The Total Environment**, v. 450-451, p. 307–316, 2013.

ZHAO, X. et al. The impact of the variation in driving conditions on the nox emissions characteristics in pems test for heavy-duty vehicle. **Discover Environment**, Springer, 2023.

ZHOU, Y.; XIE, Z. et al. Application of the support vector machine and heuristic k-shortest path algorithm to determine the most eco-friendly path with a travel time constraint. **Transportation Research Part D: Transport and Environment**, v. 86, p. 102445, 2020.