Journal of Internet Services and Applications, 2024, 15:1, doi: 10.5753/jisa.2024.3805

Using Electric Vehicle Driver's Driving Mode for Trip Planning and Routing

Marcelo dos-Reis [**Pontifical Catholic University of Minas Gerais** | ***marceloreis@pucpcaldas.br***]
Celso Iwata Frison [**Pontifical Catholic University of Minas Gerais** | ***celso@pucpcaldas.br***]
Fabiano Costa Teixeira [**Pontifical Catholic University of Minas Gerais** | ***teixeira@pucpcaldas.br***]
Humberto Torres Marques-Neto [**Pontifical Catholic University of Minas Gerais** | ***humberto@pucminas.br***]

Institute of Hard Sciences and Informatics (ICEI), Pontifical Catholic University of Minas Gerais – PUC Minas, Caixa Postal 30535-901, Belo Horizonte, MG, Brazil

Received: 23 November 2023 • **Accepted:** 12 July 2024 • **Published:** 22 September 2024

Abstract With the increasing adoption of electric vehicles worldwide, some limitations have emerged in their usage. The main limitations include low autonomy and a scarcity of charging points. In this work, we describe a software architecture for planning a stop at charging stations along a trip, by prediction of battery charge to be spent along the path. We describe the main components of this architecture and evaluate regression methods for the car consumption prediction module. We also use a real dataset built from an electric vehicle usage to validate the architecture concept and its viability analyzing multiple linear regression machine learning models. To further validate the architecture, we make comparisons between simulated and real trips.

Keywords: Driving range, Energy consumption estimation, Electric vehicles, Range prediction, Drivabilty, Recommender architecture

1 Introduction

The electric cars usage in the world has rapidly grown in the last decade and could represent 13% of the world car fleet in 2030, with an average growth of 36% per year between 2019 and 2030 [Gorner, 2020]. However, even with tax over automotive vehicles exemption or reduction in some countries (like Brazil) and discount on the import tax of electric vehicles, largely used strategies to accelerate the adoption [Ali and Naushad, 2022], the high cost of acquisition has been preventing its widespread adoption [Velandia Vargas *et al.*, 2020]. Companies such as Ambev, Coca-Cola and JBS have been using the distribution fleet electrification to achieve carbon footprint reduction goals. It is expected that by 2040, about 85% of the fleet of cars per application will be electric [McKinsey, 2023]. Reducing the negative environmental impact per capita of cities is a key objective of SDG 11 (Sustainable Development Goal 11 - Sustainable cities and communities) and the recharge planning of electric vehicles in long-distance trips can be an important factor to encourage more people to adopt electric vehicles.

Another common problem with the large-scale adoption of electric vehicles is the recharges and low autonomy problems [Lebrouhi *et al.*, 2021], such as: (i) high frequency of recharges, due to the limited capacity of the batteries; (ii) long wait to recharge, typically 30 minutes to 2 hours; (iii) recharge peaks, as there are more popular times to recharge, such as lunch breaks and others. In Brazil, it is also important to highlight the limited supply of public electric stations on roads and cities, as shown in **Figure 1**. In part, it happens due

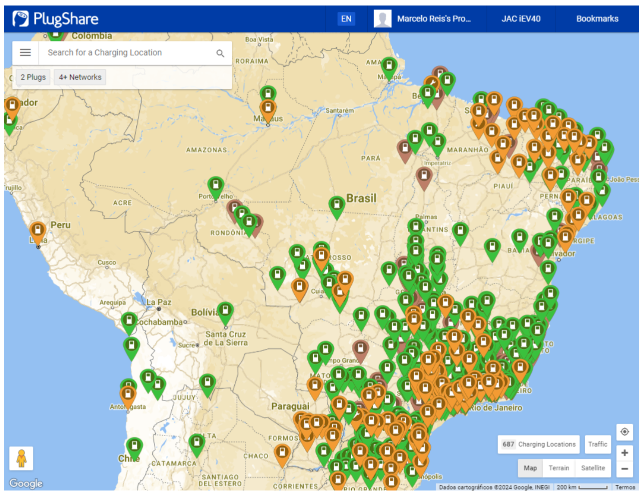

Figure 1. EV charging infrastructure in Brazil in April, 2024. **Source:** Available at `https://www.plugshare.com`. Accessed on 04/02/2024

to the fact that only in June 2018 ANEEL[1] published a resolution giving permission to commercial exploitation of electric vehicle supply [Velandia Vargas *et al.*, 2020]. In other countries, similar issues with the charging infrastructure exists, as we can observe in Italy, which lacks investments in Southern regions, shown in **Figure 2**. Turkey's deficiencies in the existing charging infrastructure is illustrated in the density map in **Figure 4**, and in **Figure 3** we can see the same problem of poor distribution of stations in Australia.

For these reasons, the correct autonomy prediction and the consequent recommendation of charging stations along the way is one of the approaches to partially solve some of these problems [Alanazi, 2023]. Indeed, one of the most important

[1]Brazilian National Electric Energy Regulator.

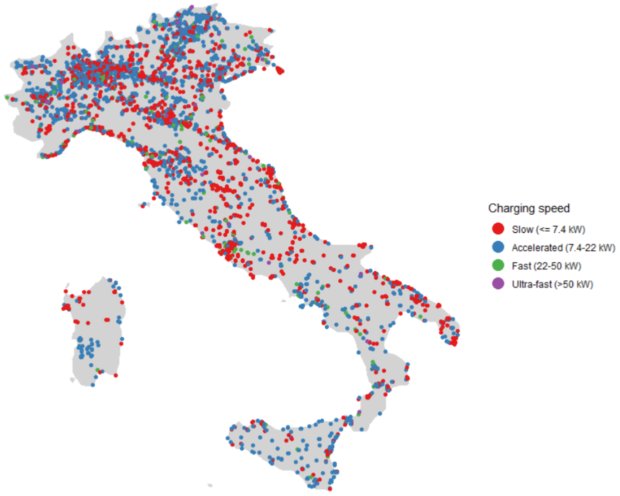

Figure 2. EV charging stations' locations in Italy, as shown in Noussan [2020], Fig. 2

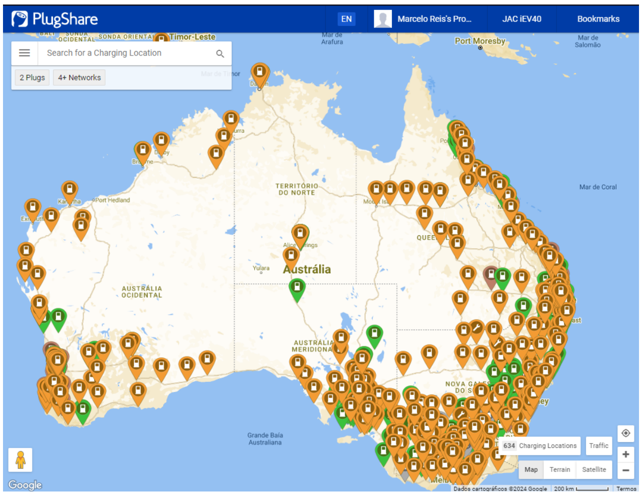

Figure 3. EV charging infrastructure in Australia in April, 2024. **Source:** Available at `https://www.plugshare.com`. Accessed on 04/02/2024

factor that can influence the autonomy prediction is the driving manner [Delnevo *et al.*, 2019]. How a person drives a vehicle is a complex concept that can be defined as the way the driver operates the vehicle's control in the context of driving and external conditions. The characterization of the way of driving can be decisive for a series of situations, such as predicting dangerous behaviors, avoiding traffic accidents, and others [Fugiglando *et al.*, 2018]. From now on, we call way of driving as *drivability*.

There are works predicting the consumption or autonomy of electric vehicles using telemetry data and physics formulation [Bailey *et al.*, 2022], regression algorithms [Chandran *et al.*, 2021] or deep neural network [Adedeji and Kabir, 2023]. Other articles focuses on the driving mode versus fuel economy relation [Stanton and Allison, 2020], while several papers studies diverse ways of driving [Siami *et al.*, 2020]. In this paper, we propose a system architecture that makes use of a motorist type classifier and a model for autonomy prediction. We use telemetry data to determine and classify drivers by their driving style and to forecast the energy consumption of each leg of a trip. Our goal is to make the prediction considering the type of driver and the context data of the

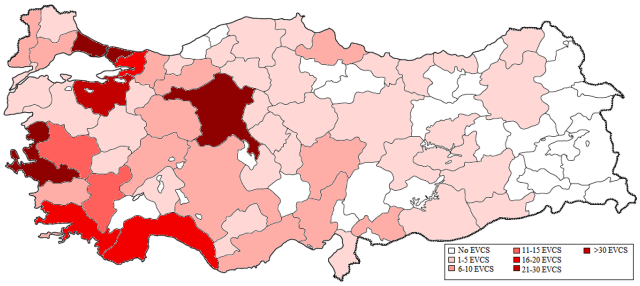

Figure 4. EV charging stations density map of Turkey, as shown in Gönül *et al.* [2021], Fig. 8

segment. In this way, it is possible to recommend charging stations along the route, considering the calculated ranges, to offer a better experience to the driver. Although there are proprietary applications, such as Tesla's and Volvo's, this work describes an open architecture, with API that can be used by more popular vehicles and integrated with existing applications, such as *Waze* and *Google Maps*.

Thus, we propose a system capable of predicting the autonomy of electric vehicles and allow recommendation of charging points. Moreover, (a) we assess whether the proposed architecture meets the purpose of proposing charging stations along a trip; (b) we identify a very suitable Machine Learning algorithms to predict the range of electric vehicles; and (c) we incorporate the *drivability* and travel context evidence into selected recommendation strategies. This approach differentiates our method from other studies that typically address only individual aspects of SOC prediction, consumption forecasting, or driving pattern analysis.

After analyzing real data collected directly from an electric vehicle, we used k-means to classify the drivers and also verified some regression methods that obtained satisfactory results to predict the amount of battery energy consumed in travel segments. The Random Forest Regressor method obtained an absolute average error of 0.249 pp. Therefore, through the routing application, it is possible to predict the autonomy of a projected trip, using the consumption forecast section by section. In addition, based on the calculated range and the route traced, we propose to recommend charging stations.

The remainder of this paper is organized as follow. Section II describes several works in order to verify how the problems of classification of the driving style and prediction of autonomy of electric vehicles are being solved. Section III presents a summary of the main technologies used in the project. The proposed architecture is presented in the Section IV, also describing how the main modules work. In Section V, we present the methodology used in the evaluation of the modules. Section VI presents the preliminary results, with the data currently collected. Finally, Section VII discusses future works and the conclusion.

2 Related Work

Many papers have studied predicting the energy consumption of electric vehicles and estimating battery SOC (State of Charge). Chandran *et al.* [2021] used features extracted from the battery with various machine learning algorithms to estimate the SOC. Another paper included those features along with the instantaneous speed and vehicle mileage [Adedeji and Kabir, 2023] as input to a deep neural network to estimate the SOC.

Several articles have linked driving mode to fuel economy, either to indicate to the driver the most economical mode [Stanton and Allison, 2020], or to analyze the impact on fuel consumption at a macro level [Ping *et al.*, 2019]. Regarding battery power savings, Huang *et al.* [2020] proposed a model using LSTM, and Chen *et al.* [2019b] used a neural network algorithm to relate driving mode and battery consumption. Additionally, Tiwary *et al.* [2022] study the impact of driving patterns on energy consumption, while Goujon *et al.* [2022] analyze eco-driving, defined as driving intentionally to economize energy expenditure.

The work presented on Bolovinou *et al.* [2014] analyzes how altitude loss or altitude gain data on a route influence the prediction of the range of electric vehicles. Another article uses Federated Learning to average various neural network and linear regression models [Thorgeirsson *et al.*, 2021]. And Bailey *et al.* [2022] use telemetry data to predict electric vehicle's autonomy. Finally, Tran *et al.* [2023] used road characteristics, like elevation, plus the air conditioning expenditure, to improve the accuracy of the SOC estimation. However, they did not take into account diverse ways of driving.

There are several papers on the identification of driving mode. One paper identified 29 different driving styles in a database of 2,500 drivers, classifying them based on telemetry data [Siami *et al.*, 2020]. In Fugiglando *et al.* [2018], a modified K-means algorithm is used to classify drivers with bad habits. Models using CAN or OBDII network data, including a re-sampling method to reduce the amount of data in order to classify drivers, have also been described in several papers [Fugiglando *et al.*, 2018; Chen *et al.*, 2019c,a; Ping *et al.*, 2019; Krishnamurthy *et al.*, 2019]. Complex networks of past behaviors have also been used in the identification of dangerous drivers [Aiyitibieke *et al.*, 2019] and other techniques detecting abnormal behavior [Jia *et al.*, 2020].

Another work compares machine learning algorithms for driving pattern discovery, notably Random Forest [Chen *et al.*, 2019c]. This paper and others detail and evaluate the most used metrics [Chen *et al.*, 2019a] or among 51 features [Chen *et al.*, 2019c], or specific ones such as brake pedal pressure [Delnevo *et al.*, 2019].

In these several studies of characterization of drivability, several machine learning techniques were used, from Deep Learning [Sama *et al.*, 2020], convolutional neural networks [Jia *et al.*, 2020], recurrent neural networks [Li and Kang, 2020], K-Means [Fugiglando *et al.*, 2018], Spectral Clustering [Chen *et al.*, 2019c], or other new approaches [Siami *et al.*, 2020; Krishnamurthy *et al.*, 2019; Ping *et al.*, 2019; Liu *et al.*, 2019].

Some papers have also been published that use pattern recognition methods to classify driving modes, such as artificial neural networks and random trees [Krishnamurthy *et al.*, 2019], K-means [Warren *et al.*, 2019; Liu *et al.*, 2019], and deep neural networks [Sama *et al.*, 2020] and combined techniques such as Kohonen maps, convolutional neural networks and Long short-term memory [Jia *et al.*, 2020] and autoencoder [Siami *et al.*, 2020].

In **Table 1** we briefly point out the differences between several recent studies and ours. None of these previous works proposes a complete architecture. In our previous work [dos Reis *et al.*, 2023] (in Portuguese), we presented an architecture that integrates: (a) a driver classification algorithm dividing them by driving mode, using features extracted from telemetry data, with (b) a linear regression algorithm to determine consumption section by section, taking into account the type of driver and the trip context data, to determine (c) the range and recommend charging stations along the journey. In this paper, we extend the results by testing with a dataset with more drivers.

3 Supporting Technologies

For the proposed architecture, we use several technologies integrated into modules and we use telemetry data in classification and regression algorithms to predict battery consumption and autonomy. The main technologies are described below.

3.1 Telemetry

With the increase in the fleet, the usage of telemetry data from electric vehicles can be an important source of data to determine the way of driving, to predict how energy is expended and to predict the vehicle's autonomy. The CAN network (Controller Area Network) is a high-speed serial bus system common in embedded systems. It is a technology widely used in modern vehicles from the adoption of various micro-controllers, such as electronic injection, ABS and others. Through this bus, the data of acceleration, speed, odometer, pressure on the accelerator pedal and other data of electric vehicles are obtained [Fugiglando *et al.*, 2018].

This data can be transmitted over the Internet, through a communication protocol. With the increase in the usage of cell phones, the need for standardization of communication protocols arose, culminating in the specification of GSM (Global System for Mobile Telecommunications). Furthermore, with the need for data transmission and access to the Internet, the GPRS (General Packet Radio System) protocol arises, specifically for connecting to the Internet [Halonen *et al.*, 2004]. By using GPRS, available in 3G and 4G coverage areas, it is possible to cover almost all Brazilian cities and 44.2%, which corresponds to 55,711 km, of their highways, according to ANATEL[2]'s infrastructure data panel [ANATEL, 2022].

[2] Brazilian National Telecommunication Regulator.

Table 1. Comparative analysis of recent works on SOC (State of Charge) prediction

Ref	Telemetry data	Trip context data	*Drivability*	Consumption prediction	Method	Route planning
[Adedeji and Kabir, 2023]	yes	no	no	yes	ANN	no
[Tran *et al.*, 2023]	yes	yes	no	yes	Mathematical	yes
[R *et al.*, 2023]	yes	no	no	yes	LTSM	no
[Adedeji, 2023]	yes	no	no	yes	ANN	no
[Liu *et al.*, 2022]	yes	yes	yes	yes	ANN	no
[Mashkov and Karova, 2023]	yes	no	no	yes	Mathematical	no
[Kocaarslan *et al.*, 2022]	yes	yes	yes	yes	Mathematical	no
[Tiwary *et al.*, 2022]	yes*	no	yes	no	-	no
[Goujon *et al.*, 2022]	yes	yes	yes	yes	Mathematical	no
[Wei *et al.*, 2022]	yes	yes	no	yes	Mathematical	no
[Hong *et al.*, 2020]	yes	yes	yes	yes	LTSM and Regression Models	no
[Paul *et al.*, 2023]	yes	yes	no	yes	LTSM, DNN, CNN and Regression Models	no
[Bailey *et al.*, 2022]	yes	yes	no	yes	Mathematical	yes
this paper	yes	yes	yes	yes	Regression Models	yes

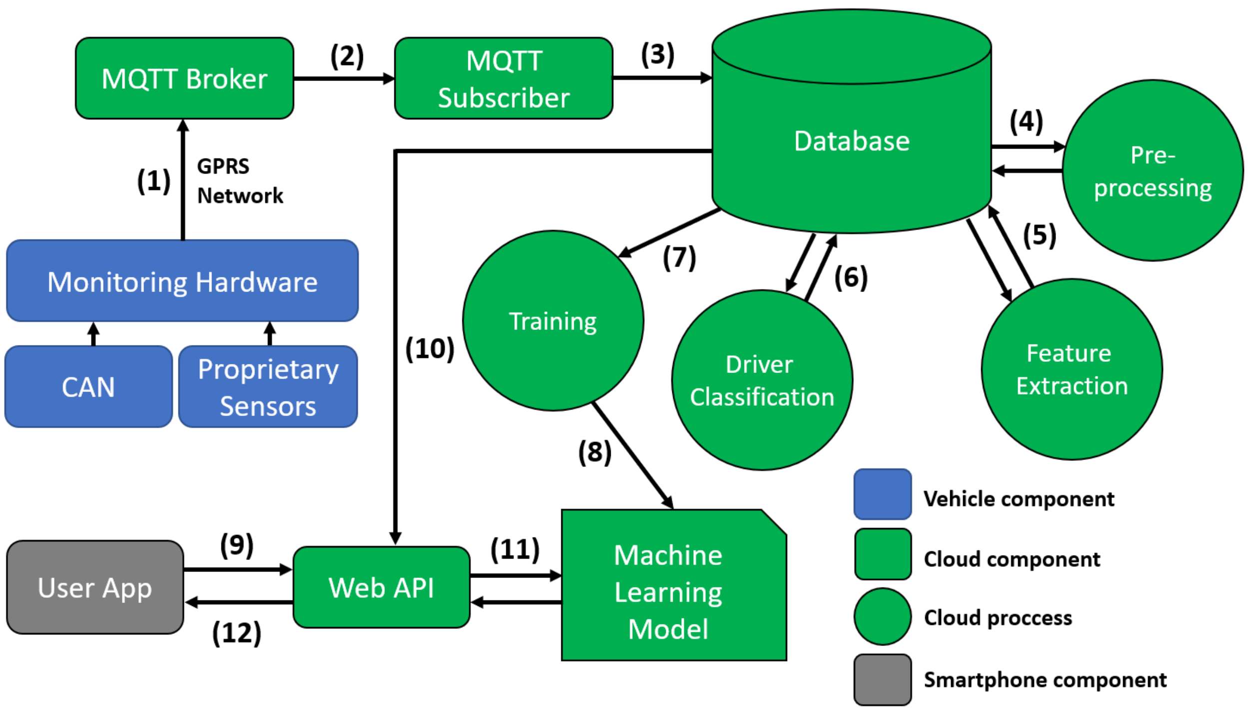

Figure 5. Proposed system architecture - ArchDriva

3.2 Classification

To classify drivers by driving mode, we propose an unsupervised method of clustering data analysis. Using unsupervised clustering methods, we can determine which parameters are more suitable to identify groups from the analyzed data [Arabie *et al*., 1996].

The accuracy of clustering methods deeply depends on the number of groups and features. In addition, the number of analyzed features influences in the groups characterization, increasing or decreasing the number of needed classes for differentiation. Also, having few groups may result in very generic groups with few similarities, but many groups would make the model very complex and computationally expensive. Finding the right number of groups and features, without losing information, has already been the subject of study [Fugiglando *et al*., 2018; Rodriguez *et al*., 2019] and is one of future goals.

3.3 Regression

We use linear regression algorithms to determine the behavior of a dataset and extrapolate output values based on the input data. We seek to find a correlation between the independent variables and the results. Multiple linear regression algorithms are an extension of the simple linear regression model, where multiple prediction variables are used for only one outcome of a continuous set [Eberly, 2007].

To forecast of consumption, we propose to use a multiple regression algorithm, using travel context data, section by section, with its consumption and type of driver. And, based on this information, predict the consumption and autonomy, for that type of driver, for the next calculated stretches.

4 The Architecture

Our proposed architecture comprises multiple integrated components and processes collaborating to provide charging station recommendations. These components include both in-vehicle elements and cloud-based functionalities, as illustrated in **Figure 5**, with arrows denoting the flow of information between them. Importantly, all these components are designed to be modular, ensuring ease of replacement as required.

Figure 5 also shows the monitoring hardware consisting of an equipment developed by a research team of PUC Minas (our University) to capture, through the CAN network and its own sensors, the necessary telemetry data. This equipment collects the data and sends it to a MQTT Broker, through a GPRS connection (1), over the Internet. The MQTT broker receives this telemetry data (1) and transmits it to its subscribers (2). The MQTT subscriber receives messages warning of the data change and reads it (2), writing it to a Database (3).

In the pre-processing module, the collected data, read from the database (4), identified with the respective drivers, are grouped, normalized, re-sampled and completed. The features are extracted in the feature extraction process, consisting of two sets: (i) features to classify drivers, like (a) throttle and break pedal pressure; (b) speed; (c) inclination; (d) acceleration; and (ii) features to linear regression, which consists of analyzing the data (5) of each trip, extracting, in ten-kilometer windows: (a) vehicle; (b) driver; (c) average speed; (d) altitude variation; (e) use of air conditioning; (f) economy mode; (g) consumption.

Also shown in **Figure 5**, the driver classification process reads data from table of features (6), limited to a maximum period of one year and, using K-Means algorithm, groups drivers into driver's classes. In the training stage, the regression model is trained with data from the features table, through data-flow (7), to forecast consumption, generating and saving a model (8), which calculates the energy consumption in that stretch, according to the type of driver and the data of the trip context, such as altitude variation, speed, air conditioning usage and economy mode. The Machine Learning Model is called directly by the API and receives: (a) driver identification and (b) trip data, section by section, and returns all sections to the API, with their respective consumption.

This Web API is used by a mobile application that traces the route according to the user's starting point and the chosen destination. This application can use one route API to determine the data from the stretches as accurately and with the greatest possible sampling. This API receives route information and driver identification (9). The driver's class is determined by consulting the table in the database (10), according to the classification of drivers. Then, the model is called with the driver classification and trip data, returning the consumption estimate section by section (11). The User Application is responsible for connecting the driver to the vehicle and its main function is to trace the travel route and recommend stops for recharging along the way.

4.1 Pre-processing

Data pre-processing is performed once a day, analyzing the data stored from the previous day and saving the processed data to the database. For this processing, some considerations were made:

1. each trip consists of a collection of chronologically sequential data, from the same vehicle and the same driver on the same day.
2. the beginning and end of each trip are marked when the speed is zero and the brake and accelerator pedal are not pressed, indicating car at rest.
3. when a trip comprises more than one day, it is separated into more than one trip.
4. every one-minute period with no data, a new trip is also considered, as it is not feasible to check the missing values.

4.2 Features

With the analysis of this data, we verify the best features to identify driving patterns, with respect to instantaneous battery consumption. Some metrics used in the classification of drivers are acceleration, inclination, speed, and pedal position, among others.

In order to identify these patterns, as it is not possible to define the number of classes or which drivers will be of each class a priori, we used an unsupervised classification learning techniques, K-means. In this paper, we used 4 classes and the following features:

- **AX**, **AY** and **AZ**: the acceleration in each axis, given in g-force;
- **GX**, **GY** and **GZ**: the inclination in each axis, in °/s;
- **BRK**: position of the break pedal;
- **ACC**: position of the throttle pedal;
- **SPD**: instantaneous velocity, in meters per second.

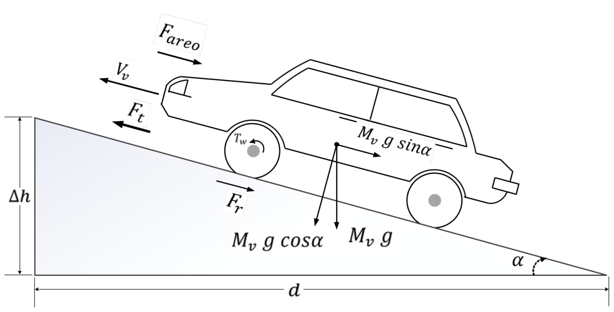

Figure 6. Forces acting on a vehicle moving up a grade, as in Akl *et al.* [2019], Figure 1

We created another set of features for use in estimating consumption, based on the type of driver and the trip context data. In Akl *et al.* [2019], the forces acting on a moving car are described, as seen in **Figure 6**. Thus, the force necessary to propel the car at a given moment is calculated by

$$F_t = F_{aero} + F_r + M_v g \sin\alpha \tag{1}$$

where; M_v, g and α are the vehicle mass, gravitational acceleration and the slope of the road, respectively. The air drag resistance force, F_{aero}, is correlated to speed [Hucho and Ahmed, 1998] and the gravity force applied depends on the inclination of the slope.

In addition, electric vehicles have an economy mode, which works opposite to sport mode on regular cars, spending less energy when switched on. Besides that, the main auxiliary energy burner is the air conditioning system, which drains energy directly from the battery. The energy is drained a in

$$E_s = E_t + E_a \tag{2}$$

where; E_t and E_a is the energy necessary to generate F_t through a given time (E_t) and the energy spent on auxiliary systems on the same time (E_a). Based on that, we selected some features that might have influence on energy consumption.

These features are described below, along with how they are calculated, for each ten-kilometer stretch:

- **DIST**: travelled distance in kilometers;
- **VEIC**: vehicle identification - each vehicle has its characteristics and battery charge capacity;
- **COND**: identification of the driver's class - for each class, there is a way to drive, more or less economical, depending on the route data;
- **VARALT**: altitude variation in the section in meter per meter - calculated as the difference in altitude at the start and end of the section divided by its size in meters;
- **SPD**: average speed in the segment, in meters per second - calculated as the distance covered in the segment by the time spent;
- **ARC**: use of air conditioning in the section - 0 for off and 1 for on - intermediate values indicate the percentage of use in the section;
- **ECO**: use of economy mode in the section - 0 for off and 1 for on - intermediate values indicate the percentage of use in the section;
- **CONS**: percentage of battery consumed in the section, with an accuracy of 0.5 pp, as informed by the vehicle.

4.3 Driver Classifying

It consists of reading the data from table (6) of features, for a maximum period of one year, generating histograms of 10 equally divided bins, for each driver and vehicle. We used the conversion of values into histograms to reduce the number of values of each feature, aiming to simplify the processing of the classification model, without losing important information, as in Fugiglando *et al.* [2018].

Using the histograms of each driver as input, an unsupervised classifier groups the drivers into classes, saving the driver and class tuples in a database table (6).

According to Alpaydin [2020], one of the most used classification algorithms is K-Means, which is an unsupervised method of classification and grouping of data that analyzes the distribution of data and identifies centroids to generate groups with similar characteristics.

Typically, K-Means input parameters are: an integer k, representing the number of classes, and a sample x = x_1, x_2, ..., x_n. Thus, the clusters are decided by randomly generating centroids and choosing the closest centroid for each x_i. Next, the center of each cluster is calculated and it is verified whether each object remains closer to these new centroids, repeating these steps until there is convergence. Each element in the sample can have one or more features that characterize the element.

4.4 Training

Using the data from the features table (7), replacing the driver's identification for his class, a model is generated, through multiple linear regression, to forecast consumption, section by section. This model analyzes, according to the type of driver and the data of the trip context, such as altitude variation, speed, air conditioning use and economy mode, the percentage of battery used in that segment. Adding up the cost section by section, we have the percentage value of battery used in the total journey.

This model is saved as a file in the cloud (8) for use in forecasting consumption section by section and this training can be carried out daily or at longer intervals according to the available computational resources.

4.5 Machine Learning Model

The model is called directly by the API (11) and receives:

- The driver's identification

- Trip data, leg by leg

This module loads the linear regression model saved from the "Training" module and, for each segment of the input, the model predicts the consumption in that segment, according to the class of the driver. All snippets, with their respective consumption, are returned to the API (11).

4.6 Web API

The API is used by a mobile application that traces the route according to the user's starting point and the chosen destination. This application can use a routes API to determine the data from the stretches as accurately and with the greatest possible sampling. This API accepts route information and driver identification (9) and, as illustrated in **Algorithm 1**, returns the consumption of each leg.

$alt \leftarrow Input.start.altitude$;
$class \leftarrow$ get_class($Input.driver$)
for *leg in Input.path* **do**
  if *leg.varalt is None* **then**
    $leg.varalt \leftarrow (leg.end.altitute - alt)/legdist.distance/1000$;
  end
  $cons \leftarrow$ model.predict($\{class, leg.speed, leg.varalt, Input.airconditioning, Input.economode\}$);
  $alt \leftarrow leg.final.altitude$;
  $leg.consumption \leftarrow cons * distance$;
end

Algorithm 1: Predicting battery consumption

To call this routine, you can access the following endpoint:

```
POST /predict
```

Here's an example request:


```
{
  "cond": 1,
  "arc": 1,
  "eco": 0,
  "inicio": {
    "lat": -21.799384,
    "lon": -46.597957,
    "alt": 1314
  },
  "caminho": [
    {
        "final": {
            "lat":  -21.7920362,
            "lon": -46.5937391,
            "alt": 1292,
        },
        "veloc": 7.222,
        "dist": 1.3,
        "varalt": 0,
        "gasto": 0
    }, ...
  ]
}
```


This function returns the original object while populating `varalt` and `gasto` with the altitude variation and fuel consumption of each leg, respectively. The legs of the path are provided within the `caminho` array, where each leg includes the final coordinate (`final`), average speed (`veloc`), and length (`dist`).

The driver's class is determined by consulting a database table (10) that contains driver classifications. Subsequently, the model is invoked with both the driver's classification and trip data, generating a consumption estimate for each section of the trip (11). This estimate is then relayed to the User App (12) for journey planning and recommendations regarding recharging stops.

4.7 User Application

This application is responsible for connecting the driver to the vehicle. It also has the function of tracing the travel route, starting from the selected origin and destination, and recommending stops for recharging along the way.

After the route is traced by a trip routing API, as in **Algorithm 2**, the segments data and the driver are passed to the Web API (9), which returns the expected expenditure of the route (12), to support the recommendation of recharging stations.

5 Methodology

To evaluate this architecture, we used the telemetry data from a real electric vehicle. These data were collected in October 2023, through an ANEEL R&D, which acquired the vehicle for DME Distribuição S.A. (DMED).

We developed a device to capture, through the CAN network, the following telemetry data: (a) Instantaneous speed; (b) State of charge - % of electric vehicle battery; (c) Odometer; (d) Engine voltage; (e) Engine current; (f) Tilt of the accelerator pedal; (g) Brake pedal tilt; (h) Air conditioning (on/off); (i) Economy mode (on/off); (j) Expected autonomy.

In addition to these data, the equipment also provides GPS data, such as: (a) Positioning (latitude and longitude); (b) Electric vehicle tilt (gyroscope); (c) Acceleration (accelerometer); (d) Altitude; (e) Date; (f) Time.

This equipment collects the data and sends it to an MQTT Broker, through a GPRS connection, that reads and saves to a database. The collection routine continuously generates data at a frequency of two or three seconds between each collection.

5.1 Dataset

In our data collection study, we enlisted the participation of nine drivers to traverse the same round-trip route connecting Poços de Caldas, MG, and São João da Boa Vista, SP. This route spans approximately 76 kilometers in length and exhibits a notable altitude variation of approximately 550 meters, visually represented in Figure 7, using OpenStreetMap[3].

Thus, the database is composed of 24,277 telemetry records, collected in October 2023. After pre-processing, the

[3] Available at https://www.openstreetmap.org/

```
steps ← gmap.directions(Input.start_coord,
  Input.end_coord);
lat ← −1;
long ← −1;
elevation ← −1;
for step in steps do
    points ← step.points;
    distance ← step.distance;
    duration ← step.duration;
    elevations ←gmaps.elevation_along_path(points);

    speed ← distance/duration;
    for coordinate in elevations do
        if lat ≠ −1 or long ≠ −1 then
            step_distance ←
              geopy.distance((lat, long),
              (coordinate.latitude, coordinate.longitude)
              ).km;
            valalt = (coordinate.elevation −
              elevation)/step_distance/1000;
            data ← data.append({step_distance,
              Input.driver, speed, varalt,
              Input.airconditioning,
              Input.economode});
        end
        lat ← coordinate.latitude;
        long ← coordinate.longitude;
        elevation ← coordinate.elevation;
    end
    lat ← −1;
    long ← −1;
    elevation ← −1;
end
```

Algorithm 2: Sample algorithm for application, using Google Maps API

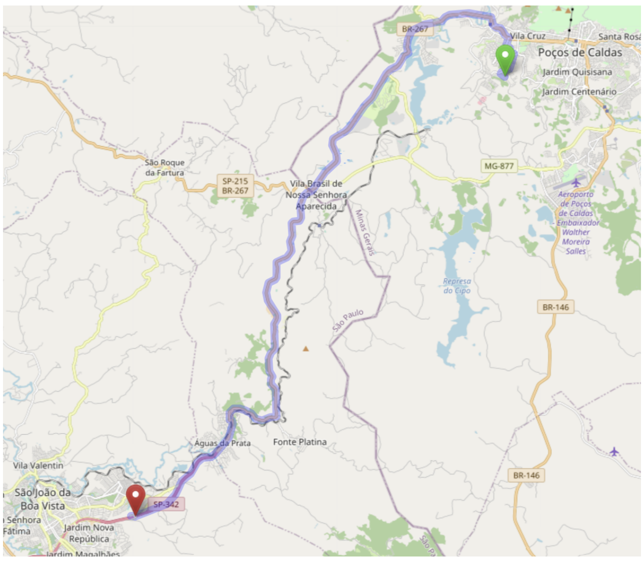

Figure 7. Travel round-trip route - 76 km - OpenStreetMap

resulting dataset consisted of 24,113 records, with 696.4 km of travel recorded, divided in 9 identified trips, as detailed in **Table 2**. **Table 3** show the dataset summarized by the economy mode and the air conditioning status.

Table 2. Trips after pre-processing

Driver	Distance	Bat. Discharge	Avg. Speed
#1	76.4 km	36.0%	57.96 km/h
#2	75.8 km	37.5%	63.46 km/h
#3	76.3 km	46.0%	64.07 km/h
#4	76.0 km	35.5%	57.07 km/h
#5	68.9 km	24.0%	55.03 km/h
#6	76.0 km	33.0%	53.08 km/h
#7	76.2 km	38.0%	53.92 km/h
#8	76.2 km	35.5%	53.62 km/h
#9	19.3 km	12.5%	55.23 km/h

Table 3. Dataset after pre-processing

Economy Mode	Air Conditioning	Number of Records
on	on	364
	off	721
off	on	10,518
	off	12,510

With data from ten-kilometer stretches, the feature extraction phase extracted 5,285 records. To extract the stretches and maximize the number of records to improve the regression model, we use windows of 10 km for each 0.1 km variation of the same trip, overlapping 99% of each window, as shown in **Figure 8**. We have chosen 10 km for the window size to have relevant battery discharge and 0.1 km because that is the minimum step in the car odometer available from sensor.

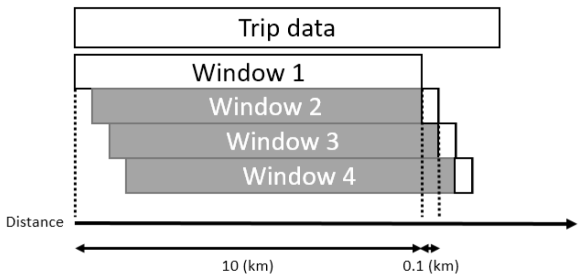

Figure 8. Moving window with 10 km length and 99% overlap

6 Evaluation

After conducting the driver classification routine, we ended up with four distinct classes into which the drivers were clustered, as depicted in **Figure 10**, where drivers #2 and #3 are highlighted. The number of clusters was determined empirically based on the patterns of drivers who accelerate or brake too quickly or too slowly. The histograms of ACC, BRK, and SPD for drivers #2 and #3 are presented in **Figure 9**.

In **Figure 11**, we present scatter plot of each feature against consumption per kilometer. We can observe that

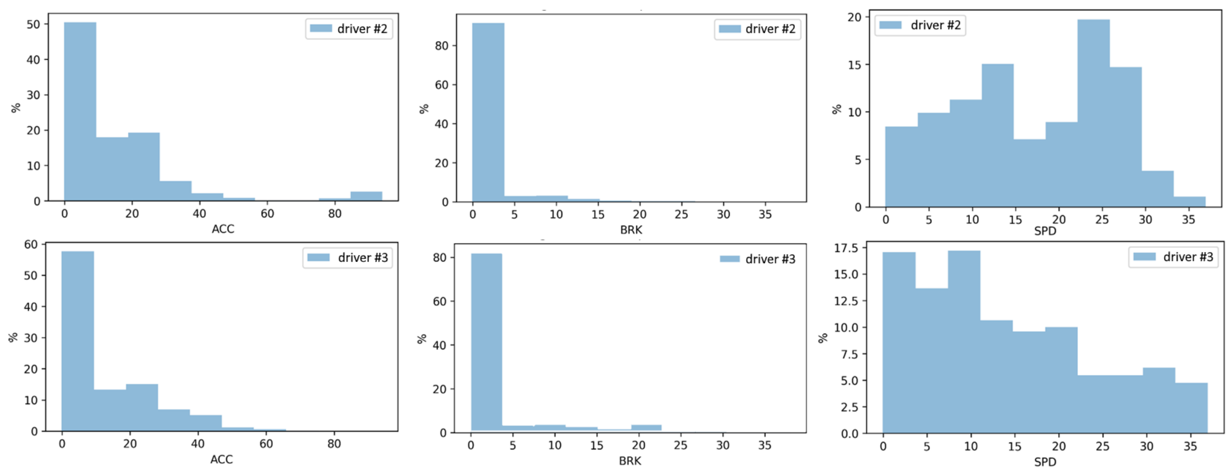

Figure 9. Histograms of ACC, BRK and SPD for drivers #2 and #3

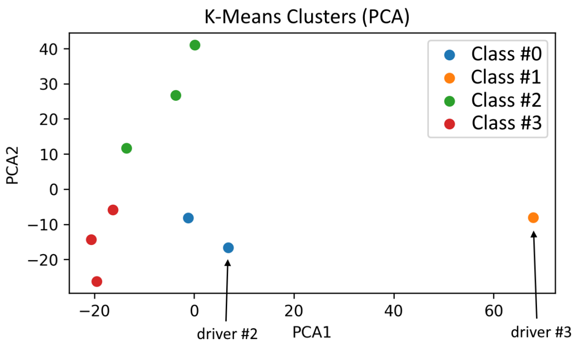

Figure 10. K-means Clustering

driver in class #1 spends more per kilometer and has the higher average speed of all. We can also see that the altitude variation has the highest correlation, with mild average speed correlation. Air conditioning and economode usage correlations are inconclusive, as class #1 used air conditioning almost all the time and class #3 no time.

We evaluate the proposed architecture verifying ten machine learning techniques for continuous function multiple regression: Random Forest Regressor (RFR), Support Vector Regression (SVR), Linear Regression (LR), Ridge Regression (Ridge), Lasso Regression (Lasso), Elastic Net Regression (ENR), K-Neighbors Regressor (KNR), Decision Tree Regressor (DTR), Ada Boost Regressor (Ada) and Gradient Boosting Regressor (GBR). For all these regressors we use the default hyperparameters of the scikit-learn package, except for n_estimators in the Random Forest and ADA Boost, which we used 500, instead of 100. Also in SVR, we used the rbf kernel. The initial evaluation of the techniques took place through 10-fold cross-validation. The indicator used in this first evaluation was the mean absolute error (MAE), with its standard deviation. For each technique, data from four different situations were evaluated: (a) without driver information or trip context, only average speed, (b) only driver information and average speed, (c) only trip context information, and (d) with driver information and travel context. The results of the ten techniques are presented in **Table 4**.

Considering that the battery charge is obtained in steps of 0.5 pp., we evaluated that the methods that had an average error below this value are good candidates. In this table, we observe that the RFR and DTR methods obtained better results in relation to the mean absolute error, indicating an error of 0.196 with a standard deviation of 0.012 and 0.198, with a standard deviation of 0.018 respectively, for the case where all features were used in the regression.

When analyzing the results, we also observed that the inclusion of travel context data significantly reduces the mean absolute error, indicating that the use of these data improves the prediction made by the model. Including the identification of the conductor, a decrease in error is also noted, but in a less significant way. If only average velocity data are used, the model has similar errors across the different models, but more than doubles the 0.5 pp. accuracy of the battery charge meter.

From this analysis, we applied the Bootstrap re-sampling method 1,000 times, in the dataset (d), to measure the mean error of the population. For this stage, the methods best placed in the previous evaluation (RFR and DTR) were tested. The results are shown in **Figure 12** and **Figure 13**. Considering the 95% confidence interval, the mean absolute errors of the methods were 0.238 to 0.260 and 0.239 to 0.282, respectively. The calculated errors are still smaller than the order of magnitude of the battery charge measurement and indicate the RFR method as the best method to predict the consumption.

In the final phase of our study, we conducted a comprehensive comparison between the actual energy consumption of each trip and the predicted consumption generated by the regressor. The results of this comparison are presented in detail in **Table 5**.

In this extended study, we aimed to not only validate the performance of our machine learning model but also to mitigate overfitting by incorporating external data for predictions. Specifically, we utilized route calculations and altitude data from Google Maps API to predict battery consumption. This approach helps ensure that the model's predictions are generalizable and not overly tailored to the training data. This phase aimed to assess the accuracy of the predicted consumption versus the real consumption recorded for seven drivers. Two drivers were not considered as their telemetry data was not complete.

Each driver's route was mapped and the consumption was

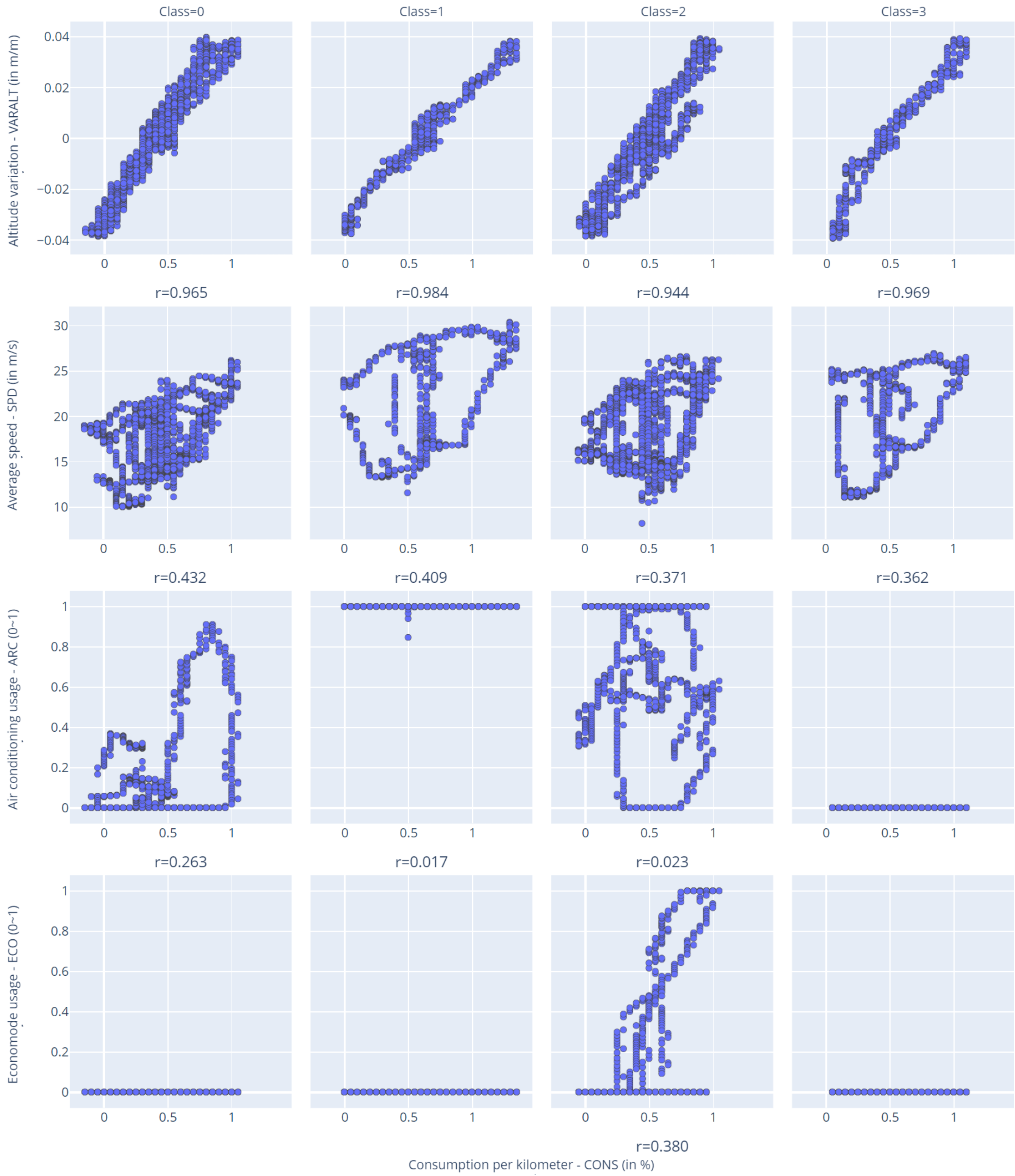

Figure 11. Consumption per kilometer x (VARALT, SPD, ARC and ECO)

Table 4. Medium absolute error and standard deviation.

	(a)	(b)	(c)	(d)
Random Forest Regressor	1.990 ± 0.066	1.682 ± 0.061	0.287 ± 0.016	0.196 ± 0.012
Support Vector Regression	1.999 ± 0.069	2.002 ± 0.060	2.005 ± 0.067	1.990 ± 0.065
Linear Regression	2.110 ± 0.069	2.106 ± 0.067	0.691 ± 0.026	0.691 ± 0.026
Ridge Regression	2.110 ± 0.069	2.106 ± 0.067	0.998 ± 0.048	0.994 ± 0.046
Lasso Regression	2.102 ± 0.074	2.102 ± 0.074	2.102 ± 0.074	2.102 ± 0.074
Elastic Net Regression	2.103 ± 0.072	2.103 ± 0.072	2.103 ± 0.072	2.103 ± 0.072
K-Neighbors Regressor	2.092 ± 0.101	1.594 ± 0.052	1.103 ± 0.062	1.144 ± 0.049
Decision Tree Regressor	1.999 ± 0.068	1.808 ± 0.067	0.299 ± 0.025	0.198 ± 0.018
Ada Boost Regressor Regressor	2.020 ± 0.053	1.999 ± 0.046	0.722 ± 0.029	0.641 ± 0.013
Gradient Boosting Regressor	1.940 ± 0.062	1.777 ± 0.042	0.474 ± 0.019	0.360 ± 0.018

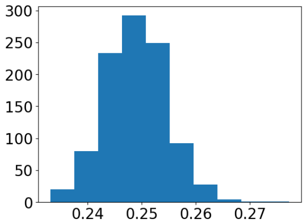

Figure 12. MAE Random Forest

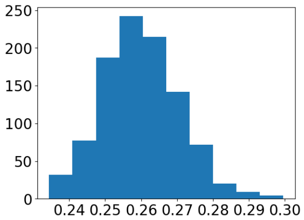

Figure 13. MAE Decision Tree

Table 5. Comparision between real and predicted consumption

Driver	Real discharge	Predicted discharge	Difference
#1	36.0	38.1	+2.1
#2	37.5	36.5	-1.0
#3	46.0	44.4	-1.6
#4	35.5	36.3	+0.8
#5	24.0	25.4	+1.4
#6	33.0	33.8	+0.8
#7	38.0	39.8	+1.8
#8	35.5	35.8	+0.3
#9	12.5	10.3	-2.2

calculated based on the distance, altitude and estimated travel time provided by the Google Maps API. The differences between the actual consumption and each prediction method were analyzed to determine the accuracy and reliability of the predictions, as shwon in **Table 6**.

Table 6. Comparision between real and predicted consumption with Google API route information

Driver	Real discharge	Predicted discharge	Difference
#1	36.0	33.7	-2.3
#2	37.5	33.5	-4.0
#3	46.0	37.9	-8,1
#4	35.5	33.4	-2,1
#6	33.0	30.2	-2.8
#7	38.0	35.6	-2.4
#8	35.5	31.8	-3.7

To further analyze the discrepancies and understand the underlying causes, we divided the route into 18 stretches and analyzed the time each stretch took to accomplish. We compared these times with the actual time taken by each driver. The analysis is summarized in Table 7, showing the relation between the predicted and actual times for each leg.

We found that the variations in time taken for each stretch by different drivers significantly influenced the prediction accuracy. These time differences affect the average speed, which is a critical factor in predicting battery consumption accurately.

To address this, we adjusted our prediction model to account for the actual times taken by drivers on each stretch, using a factor $k_{i,j}$ for each leg i and driver j. By incorporating these adjustments, we observed a significant improvement in the prediction accuracy. This suggests that considering the individual driving patterns and the specific time taken on each segment can lead to more precise predictions of battery consumption, as we can see in Table 8.

Furthermore, to show the versatility of our API for integration with third-party applications, our research involved the

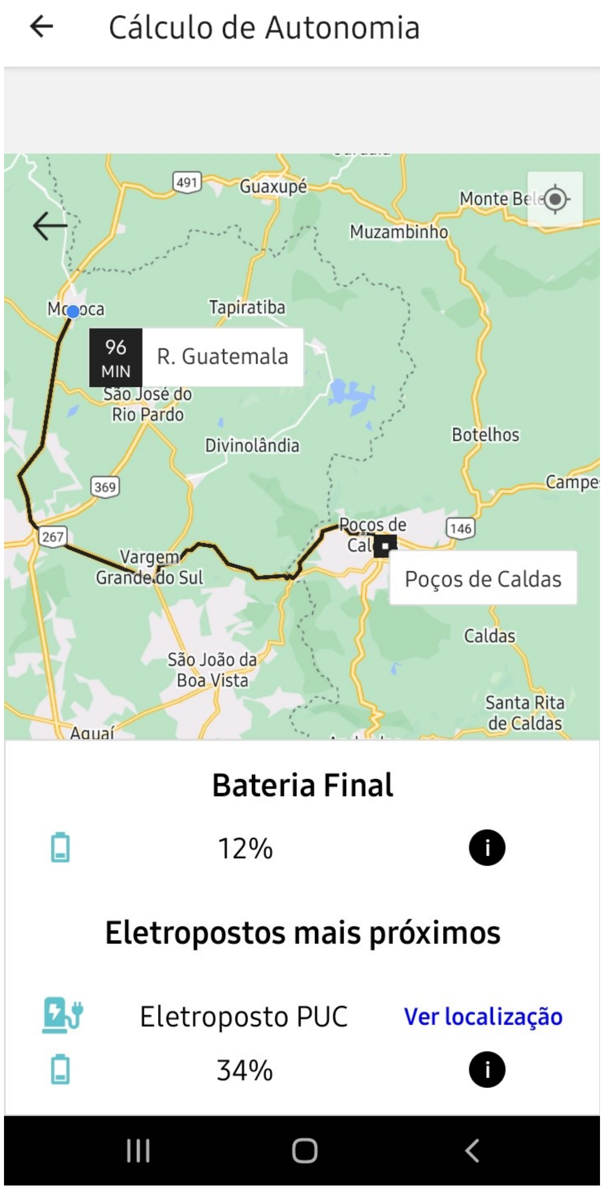

Figure 14. Simulated travel route - 104 km - User App MOV+, displaying autonomy prediction (Cálculo de Autonomia), final battery charge (Bateria Final) until target, 12%, and the nearest charging station (i.e., Eletroposto PUC), at 34%

utilization of a User App prototype called MOV+. This innovative tool empowered us to offer valuable recharging station recommendations proactively, well before the battery charge level reaches a critical 34%, as depicted in **Figure 14** (Note: The figure is in Portuguese).

7 Conclusion

With the increasing adoption of electric vehicles (EVs) worldwide, challenges related to limited driving range and charging infrastructure are becoming more pronounced. Accurate range prediction and strategic charging point recommendations are essential for enhancing the EV user experience.

Our proposed architecture addresses these challenges by leveraging telemetry data to classify drivers based on their driving behavior and analyzing trip segment data to predict energy consumption and range, to help applications to sug-

Table 7. Comparison of leg completion times between Google predicted and driver's actual times

Leg	Google	Driver #1	Driver #2	Driver #3	Driver #4	Driver #6	Driver #7	Driver #8
1	3.00	4.07	2.67	4.42	5.33	3.92	8.40	4.60
2	3.00	5.05	3.17	4.33	3.28	3.35	4.30	3.38
3	4.00	2.68	2.78	2.48	3.37	3.12	3.48	2.73
4	5.00	3.92	3.73	3.25	3.82	4.35	4.07	4.27
5	6.00	4.68	4.37	3.82	4.80	5.27	4.55	5.35
6	10.00	9.05	7.60	7.70	11.08	10.33	11.25	9.48
7	4.00	3.23	4.73	4.20	3.45	7.02	3.30	4.03
8	5.00	5.32	6.83	5.33	4.40	5.60	4.67	5.33
9	4.00	4.27	4.18	3.97	3.80	4.52	3.70	4.02
10	5.00	3.93	3.85	3.22	3.67	4.63	3.25	3.67
11	4.00	4.12	3.27	3.47	4.02	3.53	4.60	3.93
12	3.00	2.78	2.38	2.70	2.68	3.03	2.75	2.57
13	10.00	8.87	7.63	6.68	7.77	9.15	8.27	7.95
14	6.00	4.40	4.25	3.65	4.73	4.67	4.70	4.82
15	5.00	3.77	3.65	3.15	3.90	4.23	3.98	4.68
16	4.00	3.75	2.40	3.85	3.95	3.85	2.98	3.62
17	2.00	1.97	1.15	1.37	1.75	1.55	2.25	1.68
18	3.00	2.68	2.52	3.43	3.60	3.28	3.83	3.13
Total	86.00	78.53	71.17	71.02	79.40	85.40	84.33	79.25

Table 8. Comparision between real and predicted consumption, using factor $k_{i,j}$ along the route

Driver	Real discharge	Predicted discharge	Difference
#1	36.0	36.9	+0.9
#2	37.5	37.1	-0.4
#3	46.0	44.4	-1.6
#4	35.5	35.8	+0.3
#6	33.0	31.2	-1.8
#7	38.0	38.9	+0.9
#8	35.5	33.8	-1.7

gest optimal charging stops along the route. Machine learning techniques are employed for driver classification and consumption prediction models.

Our analysis reveals that the Random Forest regression method can effectively predict battery consumption for a given trip segment when driver and contextual data are available. Routing applications provide crucial information such as altitude variations and average speed, while additional data like driver behavior, air conditioning usage, and economy mode settings are inputted via an app. The primary objective of this study —to validate the proposed architecture's feasibility — was achieved.

Our findings indicate that prediction accuracy is heavily dependent on data precision. For instance, with battery charge variations as small as 0.5 percentage points (pp), a mean absolute error of 0.249 pp is deemed acceptable. However, there is potential for further accuracy improvements. A significant discrepancy was noted between estimated and actual travel times, impacting average speed and subsequent consumption predictions.

In conclusion, our research underscores the importance of considering individual driving styles and contextual factors in developing reliable battery consumption predictive models. We identified the necessity of applying a correction factor to the average speed of each significant trip segment. This factor, termed $k_{i,j}$, represents the ratio of actual average speed to that predicted by routing applications.

Additionally, expanding the dataset to include more instances of economy mode usage — currently less than 10% of the total dataset — could enhance prediction precision.

Future work should explore whether K-Means is the optimal algorithm for classifying driving modes and determine the ideal number of driver classes, given the limited driver data in this study. Further investigation is needed to assess the impact of dataset time intervals on regression model training, as the collected data spans a limited period. Examining additional contextual factors such as temperature, travel time, and geographic coordinates may also improve consumption forecasts. Finally, further studies should explore the relationship between the $k_{i,j}$ factor and driver behavior.

Declarations

Acknowledgements

This work was developed with the support of FAPEMIG (PPM-XII), DME Distribuição (DMED) and the Agência Nacional de Energia Elétrica (ANEEL).

Authors' Contributions

Competing interests

The authors declare that they have no competing interests.

Availability of data and materials

The datasets generated and/or analysed during the current study are available in https://github.com/mreis76/electric-vehicle-telemetry-dataset

References

Adedeji, B. P. (2023). Electric vehicles survey and a multifunctional artificial neural network for predicting energy consumption in all-electric vehicles. *Results in Engineering*, 19:101283. DOI: 10.1016/j.rineng.2023.101283.

Adedeji, B. P. and Kabir, G. (2023). A feedforward deep neural network for predicting the state-of-charge of lithium-ion battery in electric vehicles. *Decision Analytics Journal*, 8:100255. DOI: 10.1016/j.dajour.2023.100255.

Aiyitibieke, N., Wang, W., Wu, N., Sun, Y., Li, X., and Sun, Y. (2019). An empirical study on high-risk driving behavior to urban-scale pattern in china. *IEEE Access*, 7:43654–43665. DOI: 10.1109/ACCESS.2019.2906930.

Akl, M. M., Ahmed, A. A., and Rashad, E. E. M. (2019). A wide component sizing and performance assessment of electric drivetrains for electric vehicles. In *2019 21st International Middle East Power Systems Conference (MEPCON)*, pages 834–839. IEEE. DOI: 10.1109/MEPCON47431.2019.9008195.

Alanazi, F. (2023). Electric vehicles: Benefits, challenges, and potential solutions for widespread adaptation. *Applied Sciences*, 13(10). DOI: 10.3390/app13106016.

Ali, I. and Naushad, M. (2022). A study to investigate what tempts consumers to adopt electric vehicles. *World Electric Vehicle Journal*, 13(2):26. DOI: 10.3390/wevj13020026.

Alpaydin, E. (2020). *Introduction to machine learning*. MIT press. Book.

ANATEL (2022). Anatel - infraestrutura. Available at:`https://informacoes.anatel.gov.br/paineis/infraestrutura/`, Accessed 7 January 2022.

Arabie, P., Hubert, L., and De Soete, G. (1996). *Clustering and classification*. World Scientific. Book.

Bailey, C., Jones, B., Clark, M., Buck, R., and Harper, M. (2022). Electric vehicle autonomy: Realtime dynamic route planning and range estimation software. In *2022 IEEE 25th International Conference on Intelligent Transportation Systems (ITSC)*, pages 2696–2701. DOI: 10.1109/ITSC55140.2022.9922244.

Bolovinou, A., Bakas, I., Amditis, A., Mastrandrea, F., and Vinciotti, W. (2014). Online prediction of an electric vehicle remaining range based on regression analysis. In *2014 IEEE International Electric Vehicle Conference (IEVC)*, pages 1–8. IEEE. DOI: 10.1109/IEVC.2014.7056167.

Chandran, V., Patil, C. K., Karthick, A., Ganeshaperumal, D., Rahim, R., and Ghosh, A. (2021). State of charge estimation of lithium-ion battery for electric vehicles using machine learning algorithms. *World Electric Vehicle Journal*, 12(1). DOI: 10.3390/wevj12010038.

Chen, M.-S., Hwang, C.-P., Ho, T.-Y., Wang, H.-F., Shih, C.-M., Chen, H.-Y., and Liu, W. K. (2019a). Driving behaviors analysis based on feature selection and statistical approach: a preliminary study. *The Journal of Supercomputing*, 75(4):2007–2026. DOI: 10.1007/s11227-018-2618-9.

Chen, S., Sun, W., Li, Y., and Shi, L. (2019b). On the relationship between energy consumption and driving behavior of electric vehicles based on statistical features. In *2019 Chinese Control Conference (CCC)*, pages 3782–3787. DOI: 10.23919/ChiCC.2019.8866414.

Chen, W., Lin, Y., and Chen, W. (2019c). Comparisons of machine learning algorithms for driving behavior recognition using in-vehicle can bus data. In *2019 Joint 8th International Conference on Informatics, Electronics Vision (ICIEV) and 2019 3rd International Conference on Imaging, Vision Pattern Recognition (icIVPR)*, pages 268–273. DOI: 10.1109/ICIEV.2019.8858531.

Delnevo, G., Di Lena, P., Mirri, S., Prandi, C., and Salomoni, P. (2019). On combining big data and machine learning to support eco-driving behaviours. *Journal of Big Data*, 6(1):64. DOI: 10.1186/s40537-019-0226-z.

dos Reis, M., Teixeira, F., and Marques-Neto, H. (2023). Utilizando o modo de dirigir do motorista de veículo elétrico para o planejamento e roteirização de viagem. In *Anais do VII Workshop de Computação Urbana*, pages 15–28, Porto Alegre, RS, Brasil. SBC. DOI: 10.5753/courb.2023.740.

Eberly, L. E. (2007). Multiple linear regression. *Topics in Biostatistics*, pages 165–187. DOI: 10.1007/978-1-59745-530-5$_9$.

Fugiglando, U., Massaro, E., Santi, P., Milardo, S., Abida, K., Stahlmann, R., Netter, F., and Ratti, C. (2018). Driving behavior analysis through can bus data in an uncontrolled environment. *IEEE Transactions on Intelligent Transportation Systems*, 20(2):737–748. DOI: 10.1109/TITS.2018.2836308.

Gönül, Ö., Duman, A. C., and Güler, Ö. (2021). Electric vehicles and charging infrastructure in turkey: An overview. *Renewable and Sustainable Energy Reviews*, 143:110913. DOI: 10.1016/j.rser.2021.110913.

Gorner, M. (2020). Electric vehicles: Analysis - tracking report: June 2020. Accessed 19 July 2020.

Goujon, A., Haghighi, H., and Amine Hamri, M. E. (2022). Impact of eco-driving on energy consumption of an electric vehicle. In *2022 7th International Conference on Environment Friendly Energies and Applications (EFEA)*, pages 1–6. DOI: 10.1109/EFEA56675.2022.10063838.

Halonen, T., Romero, J., and Melero, J. (2004). *GSM, GPRS and EDGE performance: evolution towards 3G/UMTS*. John Wiley & Sons. Book.

Hong, J., Wang, Z., Chen, W., Wang, L.-Y., and Qu, C. (2020). Online joint-prediction of multi-forward-step battery soc using lstm neural networks and multiple linear regression for real-world electric vehicles. *Journal of Energy Storage*, 30:101459. DOI: 10.1016/j.est.2020.101459.

Huang, Y., Zhu, L., Sun, R., Yi, J., Liu, L., and Luan, T. H. (2020). Save or waste: Real data based energy-efficient driving. *IEEE Access*, 8:133936–133950. DOI: 10.1109/ACCESS.2020.3007508.

Hucho, W. and Ahmed, S. (1998). *Aerodynamics of Road Vehicles: From Fluid Mechanics to Vehicle Engineering*. Legendary Women. Society of Automotive Engineers. Book.

Jia, S., Hui, F., Li, S., Zhao, X., and Khattak, A. J. (2020). Long short-term memory and convolutional neural network for abnormal driving behaviour recognition. *IET Intelligent Transport Systems*, 14(5):306–312. DOI:

10.1049/iet-its.2019.0200.

Kocaarslan, I., Zehir, M. A., Uzun, E., Uzun, E. C., Korkmaz, M. E., and Cakiroglu, Y. (2022). High-fidelity electric vehicle energy consumption modelling and investigation of factors in driving on energy consumption. In *2022 4th Global Power, Energy and Communication Conference (GPECOM)*, pages 227–231. DOI: 10.1109/GPECOM55404.2022.9815789.

Krishnamurthy, V., Nezafati, K., Bae, J., Gursoy, E., Zhong, M., and Singh, V. (2019). Classification of driving behavior events utilizing kinematic classification and machine learning for down sampled time series data. In *2019 IEEE International Conference on Big Data (Big Data)*, pages 3789–3796. DOI: 10.1109/BigData47090.2019.9005982.

Lebrouhi, B., Khattari, Y., Lamrani, B., Maaroufi, M., Zeraouli, Y., and Kousksou, T. (2021). Key challenges for a large-scale development of battery electric vehicles: A comprehensive review. *Journal of Energy Storage*, 44:103273. DOI: 10.1016/j.est.2021.103273.

Li, H. and Kang, H. (2020). Research on user behavior prediction and profiling method based on trajectory information. *Automatic Control and Computer Sciences*, 54(5):456–465. DOI: 10.3103/S0146411620050065.

Liu, J., Shi, P., and Fan, J. (2022). Electric vehicle energy consumption model based on working condition recognition and k-means cluster analysis. In *2022 International Conference on Manufacturing, Industrial Automation and Electronics (ICMIAE)*, pages 130–134. DOI: 10.1109/ICMIAE57032.2022.00032.

Liu, S., Wang, N., and Wang, X. (2019). Multi-factor influenced integrated driving behavior clustering analysis evaluation model. In *2019 International Conference on Computer, Information and Telecommunication Systems (CITS)*, pages 1–5. DOI: 10.1109/CITS.2019.8862028.

Mashkov, V. and Karova, M. (2023). Estimation of power consumption for an electric vehicle through variable parameter modelling. In *2023 International Conference Automatics and Informatics (ICAI)*, pages 119–124. DOI: 10.1109/ICAI58806.2023.10339079.

McKinsey (2023). O futuro da mobilidade no brasil: uma rota para eletrificação. Available at: `https://www.mckinsey.com.br/our-insights/all-insights/o-futuro-da-mobilidade-no-brasil`, Accessed 27 January 2023.

Noussan, M. (2020). *Public EV charging in Italy: policies, business models and future scenarios*. MDPI. Book.

Paul, D., Mo, H., Elsayed, S., and Chakrabortty, R. K. (2023). Predicting energy consumption of battery-operated electric vehicles: A comparative performance assessment. In *2023 IEEE International Conference on Industrial Engineering and Engineering Management (IEEM)*, pages 1032–1036. DOI: 10.1109/IEEM58616.2023.10406558.

Ping, P., Qin, W., Xu, Y., Miyajima, C., and Takeda, K. (2019). Impact of driver behavior on fuel consumption: Classification, evaluation and prediction using machine learning. *IEEE Access*, 7:78515–78532. DOI: 10.1109/ACCESS.2019.2920489.

R, D., Venkateshkumar, M., and Kuppusamy, C. (2023). Driving range estimation using amrit for prognostics model of electric vehicles. In *2023 Innovations in Power and Advanced Computing Technologies (i-PACT)*, pages 1–6. DOI: 10.1109/i-PACT58649.2023.10434455.

Rodriguez, M. Z., Comin, C. H., Casanova, D., Bruno, O. M., Amancio, D. R., Costa, L. d. F., and Rodrigues, F. A. (2019). Clustering algorithms: A comparative approach. *PloS one*, 14(1):e0210236. DOI: 10.1371/journal.pone.0210236.

Sama, K., Morales, Y., Liu, H., Akai, N., Carballo, A., Takeuchi, E., and Takeda, K. (2020). Extracting human-like driving behaviors from expert driver data using deep learning. *IEEE Transactions on Vehicular Technology*, 69(9):9315–9329. DOI: 10.1109/TVT.2020.2980197.

Siami, M., Naderpour, M., and Lu, J. (2020). A mobile telematics pattern recognition framework for driving behavior extraction. *IEEE Transactions on Intelligent Transportation Systems*, pages 1–14. DOI: 10.1109/TITS.2020.2971214.

Stanton, N. A. and Allison, C. K. (2020). Driving towards a greener future: an application of cognitive work analysis to promote fuel-efficient driving. *Cognition, Technology & Work*, 22(1):125–142. DOI: 10.1007/s10111-019-00554-7.

Thorgeirsson, A. T., Scheubner, S., Fünfgeld, S., and Gauterin, F. (2021). Probabilistic prediction of energy demand and driving range for electric vehicles with federated learning. *IEEE Open Journal of Vehicular Technology*, 2:151–161. DOI: 10.1109/OJVT.2021.3065529.

Tiwary, A., Garg, S., and Mishra, S. (2022). Impact of driving behaviour on energy consumption of electric vehicle. In *2022 22nd National Power Systems Conference (NPSC)*, pages 872–877. DOI: 10.1109/NPSC57038.2022.10069293.

Tran, Q. T., Roose, L., Vichitpunt, C., Thongmai, K., and Noisopa, K. (2023). An improved method to estimate energy consumption for light duty electric vehicle based on road characteristics and ambient temperature. In *2023 Asia Meeting on Environment and Electrical Engineering (EEE-AM)*, pages 01–04. DOI: 10.1109/EEE-AM58328.2023.10395316.

Velandia Vargas, J. E., Seabra, J. E. A., Cavaliero, C. K. N., Walter, A. C. S., Souza, S. P., and Falco, D. G. (2020). The new neighbor across the street: An outlook for battery electric vehicles adoption in brazil. *World Electric Vehicle Journal*, 11(3). DOI: 10.3390/wevj11030060.

Warren, J., Lipkowitz, J., and Sokolov, V. (2019). Clusters of driving behavior from observational smartphone data. *IEEE Intelligent Transportation Systems Magazine*, 11(3):171–180. DOI: 10.1109/MITS.2019.2919516.

Wei, H., He, C., Li, J., and Zhao, L. (2022). Online estimation of driving range for battery electric vehicles based on soc-segmented actual driving cycle. *Journal of Energy Storage*, 49:104091. DOI: 10.1016/j.est.2022.104091.