CHALMERS
UNIVERSITY OF TECHNOLOGY

Prediction of Energy Consumption for Battery Electric Vehicles

A study based on a data-driven machine learning approach

ADITYA SINGH, Complex Adaptive Systems
NICLAS VESTLUND, Systems, Control and Mechatronics

DEPARTMENT OF ELECTRICAL ENGINEERING
CHALMERS UNIVERSITY OF TECHNOLOGY
Gothenburg, Sweden 2021
www.chalmers.se

Master's thesis 2021

Prediction of Energy Consumption for Battery Electric Vehicles

A study based on a data-driven machine learning approach

ADITYA SINGH
NICLAS VESTLUND

Department of Electrical Engineering
Division of Systems and Control
Chalmers University of Technology
Gothenburg, Sweden 2021

Prediction of Energy Consumption for Battery Electric Vehicles
A study based on a data-driven machine learning approach
ADITYA SINGH & NICLAS VESTLUND

© ADITYA SINGH & NICLAS VESTLUND, 2021.

Supervisor: Viktor Larsson, Volvo Car Corporation
Examiner: Jonas Fredriksson, Chalmers University of Technology

Master's Thesis 2021
Department of Electrical Engineering
Division of Systems and Control
Chalmers University of Technology
SE-412 96 Gothenburg
Telephone +46 31 772 1000

Cover: Visualization of the navigation system used in some Volvo cars.

Typeset in LaTeX
Printed by Chalmers Reproservice
Gothenburg, Sweden 2021

Prediction of Energy Consumption for Battery Electric Vehicles
A study based on a data-driven machine learning approach
ADITYA SINGH & NICLAS VESTLUND
Department of Electrical Engineering
Chalmers University of Technology

Abstract

The market for battery electric vehicles (BEVs) is growing steadily because of the environmental benefits, compared to conventional vehicles. The limited range and long charging times are however large issues for BEVs. To diminish the issue of range-anxiety among drivers, the energy consumption prediction needs to be accurate when setting a destination in the satellite navigation.

This thesis aims to develop machine learning approaches that can predict the energy consumption for BEVs, split into propulsive and auxiliary consumption. The data used will consist of vehicle collected data from Volvo Cars and geographical data from a navigation supplier. Since a machine learning model's performance relies on its inputs, a major part of the thesis will be spent on choosing the correct parameters and modifying the data to create better predictions.

Linear regression, multi-layer perceptron, recurrent neural network, and gradient boosting were all machine learning models that were compared in the study.

After the study it can be concluded that, for the prediction of the propulsive energy consumption, the choice of model did not matter significantly. The performance of the auxiliary prediction was very similar for all models except the RNN, which showed worse results. It was also concluded that a more reasonable approach to show the predicted energy consumption for a trip, is to show a span of possible consumption instead of an absolute value. The main reason being that there is always a difference between the predicted and actual speed, acceleration, and total time for a trip.

Keywords: battery electric vehicles, energy prediction, machine learning, propulsive, auxiliary

Acknowledgements

We would first and foremost like to thank our supervisor at Volvo Cars, Viktor Larsson, who has helped us with both practical and theoretical challenges during our thesis. We would also like to show our gratitude to our examiner and supervisor at Chalmers, Jonas Fredriksson, who has always been very flexible and helpful. Finally, a special thanks to the employees we have been in contact with at Volvo Cars and the corporation in whole. Ole-Fredrik Dunderberg, Daniel Claesson, Niklas Åkerblom, and Sudhir Rao, all employees at Volvo Cars, have helped us with different things during our stay at the company. The organisation in whole has supplied us with facilities, test vehicles, raw data, and other equipment to make the thesis possible.

Aditya Singh & Niclas Vestlund, Gothenburg, June 2021

Contents

List of Figures

List of Tables

1
Introduction

1.1 Background

There is currently a strong trend towards battery electric vehicles (BEVs) within the automotive industry, where for example California has made the commitment to end the sales of new fossil fueled vehicles by 2035 [1]. The change is most certainly a response to rising awareness of global warming and the poor air quality in many major cities. BEVs typically produce fewer life cycle emissions than conventional vehicles and have the advantage of producing zero direct emissions, lowering the environmental impact both on a local and a global level [2]. However, the main disadvantages of BEVs, compared to conventional vehicles, are the limited driving range and the long re-charge times. The charging time being the main reason why customers would not consider a BEV when buying a new car, according to a survey done in California [1].

To diminish the associated range-anxiety effect, it is crucial to have a system that can estimate the vehicle energy consumption for a trip accurately. A good estimation is very important to gain trust from the customer, that the car will reach its promised destination. It is also important in order to facilitate intelligent trip planning, where charging stops can be inserted at appropriate places along the route, if needed.

Many car manufacturers are using a vehicle energy model (VEM) to predict the energy consumption of their BEVs. The VEM is generally optimized for computational efficiency and is based on physical models and heuristic rules defined by engineering experience. The approach can perform relatively well, however, might need manual adjustments for each specific vehicle configuration. With increased computational power in future cars, the opportunity to use a more adaptive and computationally heavy data-driven approach arises, which will be studied in this thesis.

1.2 Literature Review

Earlier studies have used different optimization methods to tackle the energy prediction problem. The methodologies can be roughly divided into either rule-based or data-driven methods according to [3]. Rule-based methods can be seen as "white-box" approaches, where the physics behind the system is studied in order make a good predictive model. The physics of BEVs include the powertrain, the batteries, the HVAC (Heating, Ventilation and Air Conditioning) control system, the wind and rolling resistance, etc. With a well-known system, the approach can be very accurate and no data collecting is needed to predict the outcome of the system. It can also benefit the correction for errors as the system is known and easier to debug. However, difficulty arises with either very complex physical relationships or new vehicle variants. Accurate calculations cannot be made without full knowledge of the system and it can sometimes be time-consuming, or even impossible, to ensure a good parameterization of all the physical relationships in a system, leading to assumptions being made that can vary more or less from reality. Data-driven methods, contrary to the rule-based methods, can instead be seen as “black-box” approaches, where the physics behind the system is neglected and large collection of data is used to model the relationship between the input and output variables of the system. The major strength of the approach is that even complex and fully, or partly, unknown systems can usually be modeled with enough data. Nevertheless, the methods often rely on having enough data and a decent distribution in the dataset, which could be a problem in the earlier stages of a BEV model’s life. They are also harder to debug, as the nature of making a “black-box” solution is to neglect what happens inside the system.

According to [3], who studied multiple research papers on BEV consumption estimation from 2011 to 2019, rule-driven methods were dominant up until the year of 2015, when data-driven methods became the more popular option. However, by looking at electric passenger cars only, instead of all electric vehicles, the data-driven methods have been the preferred choice for the entire studied period. The reason is most likely the complexity in a system with many stochastic parameters, for example driver behaviour and weather conditions, where a data-driven approach can find the complex relationships without the need to understand the inner system.

Data-driven approaches used for estimating consumption and remaining range in BEVs often use regression models to discover the relationships between parameters in a dataset. [4] uses a multivariate model to map the relationship of energy consumption against speed and acceleration. The study shows that, in a controlled environment, the consumption can be directly calculated with known speed and acceleration. [5] investigates the impact of road slope to the energy consumption of a BEV. A linear and a logarithmic regression model are used, where the logarithmic model shows the best result. Both [4] and [5] show that simple regression models are applicable to different problems regarding consumption estimation. However, both articles state that with more fluctuating and uncontrollable parameters, such as different temperatures and road conditions, the results will be harder to achieve with the selected models.

To gain better results, while including more parameters to the model, authors have proposed the usage of more advanced machine learning models. [6] displays a blended model, which fuses two machine learning algorithms, Extreme Gradient Boosting Regression Tree (XGBoost) and Light Gradient Boosting Regression Tree (LightGBM). These algorithms use regression trees instead of regular regression models and can therefore establish more complex relationships. Another benefit is the parallelism that can be achieved with the tree structure, which speeds up the process even for a large system. The authors' solution manages to make a better prediction of remaining range in the BEV than previous work with more common machine learning algorithms.

1.3 Aim

The objective of this thesis is to, together with Volvo Car Corporation, investigate the possibility to use a data-driven approach to predict the energy consumption of a BEV. The idea is to use data from Volvo to train a machine learning model, which is able to reliably predict the energy consumption for a trip, once a destination is set by the driver in the navigation system.

The majority of the data used in the thesis comes from a database containing data collected by over one hundred BEVs. The vehicles are leased by employees at Volvo Cars, where they have signed an agreement, allowing the company to collect data from the cars anonymously. Anonymously means no geographical information or other information that can point towards who the driver of the vehicle is. The data includes signals that are transmitted on the CAN and FlexRay busses in the vehicle, for example speed, acceleration, and road inclination. Most of the available signals are not relevant for the energy prediction, meaning that only a small subset will be used.

The available data also includes data from a navigation supplier, which contains the geographical information for a trip. The driver sets a destination in the infotainment system and the desired destination is then sent to the navigation supplier, which sends back the data needed to make predictions for the trip. The data received from the navigation supplier includes for example the average speed, road inclination, and distance for the trip. The data comes divided into different segments of the trip, where each segment contains average values for all the signals.

1.4 Limitations

- **Number of parameters:** The number of parameters, which affect the energy consumed during a trip, is large and it is a challenging task to be able to identify all the parameters affecting the energy consumption. Therefore, only the most significant parameters will be identified and used for predicting the consumption.
- **Data Collection Period:** All vehicle collected data was collected between December 2020 and April 2021 in Sweden, meaning that the ambient conditions are limited to Swedish Winter and Spring weather. The temperatures will mostly be around -5°C to 10 °C.
- **Vehicle Models:** Only one battery electric car model will be analyzed in the vehicle collected dataset. The reason being the limited time of the thesis.

1.5 Research Questions

- What machine learning model works best for the prediction of the propulsive energy consumption for BEVs?
- What machine learning model works best for the prediction of the auxiliary energy consumption for BEVs?

2

Theory

2.1 Vehicle Energy Model

To give a brief introduction to how the VEM is working, a general VEM from [7] can be shown. The VEM used by Volvo Cars is confidential and can therefore not be described in detail in this thesis.

Equation 2.1 shows the outside forces that are acting on a driving vehicle, where all in-going parameters are explained in table 2.1 and the resulting force, F_{prop}, is the force needed to propel the vehicle.

$$F_{prop} = \underbrace{f_r mgcos(\alpha)}_{\substack{\text{Rolling Resistance} \\ \text{Force}}} + \underbrace{\frac{1}{2}\rho C_d A_f (v - W)^2}_{\substack{\text{Aerodynamic Drag} \\ \text{Force}}} + \underbrace{mgsin(\alpha)}_{\substack{\text{Road Slope} \\ \text{Force}}} + \underbrace{m_{eff} a}_{\substack{\text{Acceleration} \\ \text{Force}}} \tag{2.1}$$

Table 2.1: Parameter explanation for VEM equation.

Parameter	Explanation
f_r	Rolling resistance coefficient
m	Vehicle mass
g	Gravitational acceleration
α	Road inclination angle
ρ	Air density
C_d	Air drag coefficient
A_f	Vehicle frontal area
v	Vehicle speed
W	Wind speed
m_{eff}	Effective vehicle mass (Mass + Equivalent mass of the motor and wheels inertia)
a	Vehicle acceleration

The resulting force, F_{prop}, is strongly related to the electrical propulsive power consumption, which is the wanted quantity when predicting the propulsive energy consumption. The equation giving the propulsive power consumption from the propelling force can be seen in equation 2.2, where P_{prop} is the propulsive power, μ is the drivetrain efficiency, v is the vehicle speed, and F_{prop} is the propelling force.

$$P_{prop} = \mu v F_{prop} \tag{2.2}$$

Equation 2.1 and 2.2 accounts for the propulsive power consumption of a vehicle, however, the auxiliary consumption is also a significant energy consumer and should be a part of the VEM.

Auxiliary power consumption can simply be stated as the equation seen in 2.3, where P_{aux} is the auxiliary power, P_{base} is the base power of the car, and $P_{climate}$ is the climate control power of the car.

$$P_{aux} = P_{base} + P_{climate} \tag{2.3}$$

The base power can be seen as the constant part of auxiliary consumption, including different electrical components of the car that are always on when the car is on. However, the more important part for modelling is the non-constant part of the consumption, the climate control power. It includes for example the climate control of the compartment and the battery heating.

With the full VEM in mind, containing both propulsive and auxiliary power, the parameters considered relevant for a data-driven prediction model can be chosen. The following two section go through all these parameters, divided into parameters relevant for the propulsive consumption and parameters relevant for the auxiliary consumption.

2.1.1 Relevant Parameters - Propulsive Energy Prediction

This subsection will go through all the parameters relevant for the propulsive energy prediction.

2.1.1.1 Speed

The speed of a vehicle is mostly affecting the aerodynamic drag force of the car, which can be seen in the square term of equation 2.1. The speed is also present as one of the multipliers when converting from force to power in equation 2.2. The speed is therefor considered important for the prediction. However, the subtracted wind speed in the same term can be neglected as it is not measurable from a car's sensors and is in most cases small compared to the speed of the vehicle.

2.1.1.2 Acceleration

Acceleration is considered relevant as it is needed for changing between speeds during a trip. The parameter is part of the acceleration force in equation 2.1, which in turn impacts the propulsive power consumption.

2.1.1.3 Road Inclination

Different road inclinations impacts both the rolling resistance force and the road slope force in equation 2.1, making it relevant for the parameter study.

2.1.1.4 Mass

Mass is chosen as it is present in many terms of equation 2.1. As with road inclination, the mass impacts both the rolling resistance force and the road slope force. It also increases the force needed to accelerate or decelerate the vehicle, being a part of the acceleration force. The effective mass is primarily made up by the vehicle mass, which is why equivalent mass of the motor and wheels inertia can be neglected.

2.1.1.5 Driver Behaviour

Acceleration is necessary during a regular trip to adjust the speed to the current speed limit. However, different driver behaviour affects how the acceleration or deceleration is performed. A driver with aggressive behaviour, i.e. a driver who breaks and accelerates harder than necessary, can increase the car's propulsive energy consumption. The opposite also applies to a calmer driver. Since driver behaviour is connected to the acceleration and deceleration of the car, the variation of the acceleration signal can show how the car is being driven. In mathematical terms, the variation of the signal can be shown with the standard deviation of the acceleration, which will be a parameter of interest in the thesis.

2.1.1.6 Weather Conditions

One factor that influences the propulsive energy consumption is the weather conditions. The different road surfaces created by snow or rain can change the rolling resistance coefficient and therefore increase the rolling resistance force in equation 2.1. The current, or future, weather is however very hard to establish with the available parameters in the car. The information could in theory be accessible from third party weather reports, for example provided by the navigation supplier, however, it is not. Instead the data from the car needs to be used, the most obvious one being the setting for the windshield wipers. If the wipers are switched on, there is likelihood of something covering the windscreen, most probably rain or snow. Therefore the wiper signal is considered a relevant parameter.

Also a part of the weather conditions surrounding the car is the ambient temperature. With changing temperatures, the air density, car tire pressure, and battery temperature change. The different air density causes a different aerodynamic drag force of the car, as seen in equation 2.1. The change in car tire pressure means that also the rolling resistance force will not be the same since the rolling resistance coefficient is dependent on the pressure of the wheels. The battery temperature has an indirect effect on the propulsive energy consumption as a cold battery cannot regenerate as much energy, compared to a warmer battery. All these effects make both ambient and battery temperature relevant for the energy prediction models.

2.1.2 Relevant Parameters - Auxiliary Energy Prediction

This subsection will go through all the parameters relevant for the auxiliary energy prediction.

2.1.2.1 Compartment, Ambient, and Set Temperature

The main part of the auxiliary consumption is due to the climate control system of the car. The task for the climate control system is to keep the compartment temperature at a comfortable level for the driver and passengers of the car. With a compartment temperature that deviates from the set temperature region, the car needs to use energy in order to reach the set temperature. The set temperature is controlled by the user and changes in theory how the climate system works. Another important temperature to keep track of is the ambient temperature as it influences how much the car cools down or heats up from the environment, which the car needs to counteract. All these three temperatures; compartment, ambient, and set temperature, can therefore be important inputs to predict the auxiliary consumption.

2.1.2.2 Battery Temperature

The temperature of the battery influences how well the battery can be charged and discharged, as well as the lifespan of the battery. There are specific temperatures that are optimal for both discharging and charging the battery, usually in the higher spectrum in respect to colder climate, meaning that the battery needs to be heated in some situations. The heating of the battery is a part of the climate power in equation 2.3 and thus the battery temperature is a relevant parameter.

2.1.2.3 Wiper Speed and Windshield Heating

Other than the temperature control in the car affected by the different temperatures inside and outside the car, there are also other auxiliary energy consuming factors in the moving part of the VEM. One example is the dehumidifier in the car, which is working at different rates depending on the ambient conditions. The humidity in the ambient air can change depending if it is raining or not. Rain can be detected by the windshield wiper setting, which means that the setting can be relevant as a predictor of the auxiliary energy consumption.

Additional examples of smaller energy consumers are the different heating systems for specific parts in the car, excluding the compartment heating, for example the windshield heater. Heating increases the auxiliary energy consumption in the car, meaning that the windshield heater setting can be a relevant parameter to track.

2.2 Basic Statistics

This section will give some general knowledge about two statistical methods used in the processing and selection of the data before the machine learning application. First, sample correlation coefficient will be explained, followed by the variance inflation factor.

2.2.1 Correlation

To establish relationships between parameters it is important to have measures of how much influence one parameter potentially has on another parameter. One easy and common way of measuring the linear correlation is to use the sample correlation coefficient [8]. The coefficient is a value between -1 and 1, showing the linear correlation between two features [8]. The formula can be seen in equation (2.4), where X and Y are two datasets of equal size, n is the dataset size, X_i and Y_i are the data points with index i in the datasets, and $\bar{X}$ and $\bar{Y}$ are the means of the datasets.

$$R = \frac{\sum_{i=1}^{n} Y_i(X_i - \bar{X})}{\sqrt{\sum_{i=1}^{n}(X_i - \bar{X})^2 \sum_{i=1}^{n}(Y_i - \bar{Y})^2}} \tag{2.4}$$

A correlation coefficient close to 1 means that there is a positive linear relationship and a correlation coefficient close to -1 means that there is a negative linear relationship between the datasets. The way to present these numbers is usually with a correlation matrix that shows the linear correlation between all individual parameters in a dataset. The correlation matrix can give a hint about which parameters impact each other.

An important note is that correlation should not be confused with causation, which is the deduction that a change in one parameter is a consequence of changing another one [8]. A correlation coefficient close to 1 or -1 of two variables A and B only shows that if variable A changes, variable B also changes. However, the change of B is not necessarily because of the change of A, some other variable can be the real driving factor.

2.2.2 Multicollinearity

One phenomenon that can occur in multiple regression problems is multicollinearity. Multicollinearity is said to be present when one input variable can be linearly predicted using another input variable with significant accuracy [8]. The phenomenon causes confusion when studying the correlation or relationship between the input variables and an output variable, as it is not clear which of the variables that impacts the output the most. The problem can be harmful to a machine learning model as it can learn untrue relationships without the user's knowledge. An example of an untrue relationship is if someone wants to predict the number of people drowning with the input variables temperature and ice-cream sales. There is most probably a correlation between temperature and number of drownings since people swim more when it is warmer outside. However, there is probably a correlation between ice cream sales and temperatures as well, meaning that the change in number of drownings can be picked up in the ice-cream sales parameter. If a machine learning algorithm sees the ice-cream sales as the driving parameter, then that is an untrue relationship learned since temperature is probably a more important input than ice-cream sales.

The variance inflation factor, often referred to as VIF, is a way of measuring the multicollinearity between a set of parameters [8]. VIF gives a value above or equal to 1, quantifying how much the variance of an estimated regression coefficient is increased because of the correlation between two variables [8]. The lower the number the better, however, there are different opinions about what are acceptable values. Some authors propose that a VIF number above 10 indicates a problem with multicollinearity [8]. However, other authors seem to think that the limit is too liberal and that the limit should be around 4 or 5 instead [8]. The latter will be the baseline for this thesis to avoid any kind of problems with multicollinearity.

2.3 Machine Learning Models

In this section, different machine learning models employed for the purpose of propulsive and auxiliary energy prediction are discussed in a general sense. In the thesis, energy consumption prediction is treated as a regression problem in machine learning, which involves learning the relationship between input and output variables.

2.3.1 Linear Regression

Linear regression (LR) is one of the simplest regression techniques that can be used. LR will serve as a benchmark for more advanced machine learning models. The technique assumes that the output variables have a linear relationship with the input variables. The equation showing the relationship is shown in equation 2.5, where, y is the dependent variable or the output, x_i are the input variables and β_i for $i = 0, 1, ..., n$ are the regression parameters to be determined from data.

$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + ... + \beta_n x_n \tag{2.5}$$

If the number of input variables are more than one, then it becomes a multiple linear regression problem. Since there will typically be more than one input signal affecting the output power consumption, multiple linear regression will be used for prediction. The performance of LR can be improved significantly if the data to be used is filtered and the extreme outliers are removed.

2.3.2 Multi-Layer Perceptron

One type of feed forward neural network is the multi-layer perceptron (MLP), which contains an input layer, an output layer, and hidden layers. The number of hidden layers can be more than one depending on the requirement. Commonly used activation functions in the neurons are tangent hyperbolic and sigmoid [12]. Figure 2.1 shows a multi-layer perceptron with an input layer with three inputs, a hidden layer with five hidden units, and an output layer with two output units. These networks are essentially feed-forward fully connected networks and are commonly trained using gradient descent for backpropagation [10].

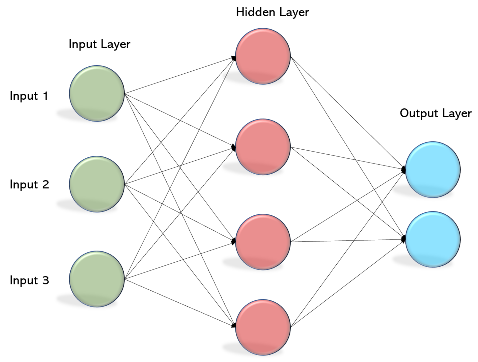


Figure 2.1: Schematic diagram for a simple multi-layer perceptron with one hidden layer [11].

2.3.3 Recurrent Neural Networks

Another important class of neural network is the recurrent neural network (RNN), which allows for temporal information processing. At any given time step, the output would depend not only on the inputs at that time but also indirectly on the inputs and outputs from previous time steps. Recurrent neural networks enable learning across time and building relationships between subsequent time steps. These networks perform best with time series data and in cases where there is some relation, whether long term or short term, between data points in series. The temporal relationship in the series can be learnt quite efficiently by RNN models.

One of the most popular neural network with a recurrent architecture is the Long short-term memory (LSTM). LSTM units are able to handle the vanishing gradient problem by the usage of a cell and gates [13]. A cell remembers information over multiple time steps and different gates regulate the information flow with time as shown in figure 2.2.

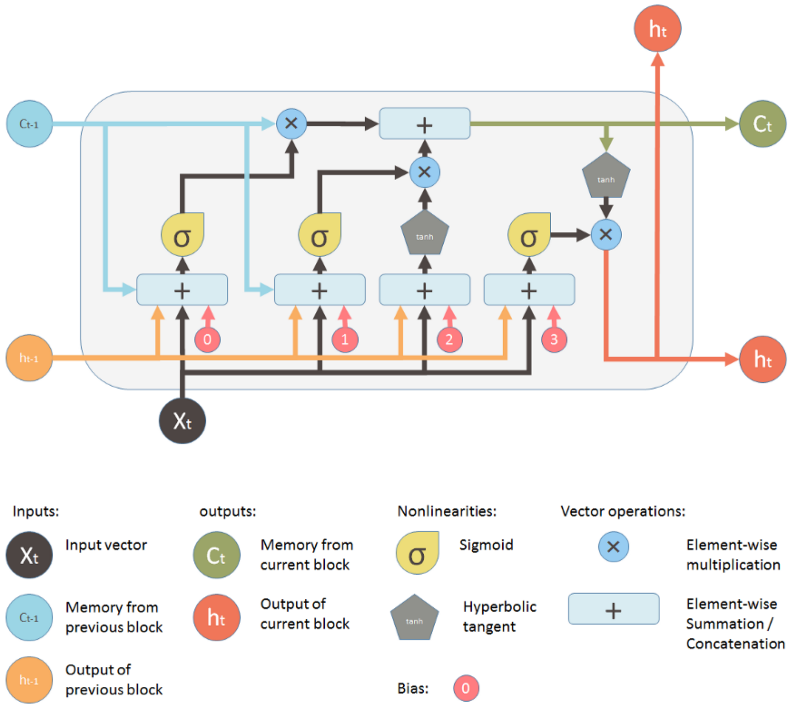


Figure 2.2: A typical LSTM unit [14]

There are different types of recurrent neural networks depending on number of inputs and outputs and their relative positioning. These are shown in the figure 2.3. Among the architectures, the ones of interest are many-to-one and many-to-many RNNs. Specific implementation details are discussed in the section 3.4.3. More information about RNN models and different architectures can be found in [13].

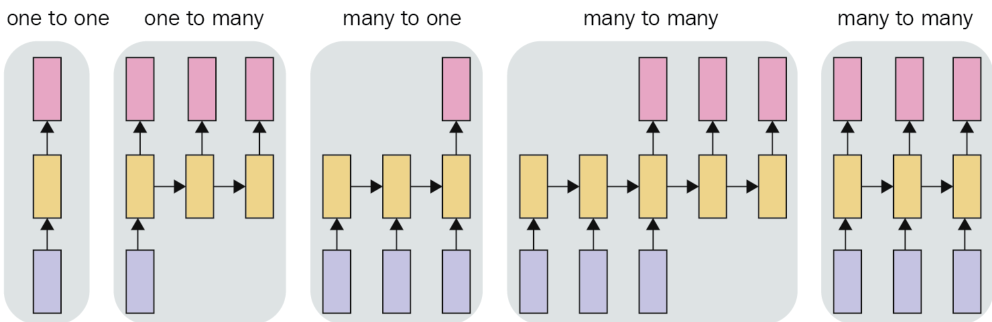


Figure 2.3: Different types of RNN architectures [15]

2.3.4 Gradient Boosting

Decision trees have long been a staple structure for taking decisions based on some features. It consists of root node, internal node and leaf nodes. Each tree starts with a root node, which branches out into internal nodes or leaf nodes. Each internal node in the tree takes a binary decision, i.e. a simple true or false based on a threshold decided by the data. A leaf node is the end node which contains a class or a numerical value. This structure is depicted in figure 2.4. It is difficult for a single decision tree to perform more complex tasks as it is prone to overfitting and can be inaccurate. A common strategy is to use ensemble learning [16], where more than one decision tree is used to make the overall decision. A collective decision always gives better results than the decision of an individual tree.

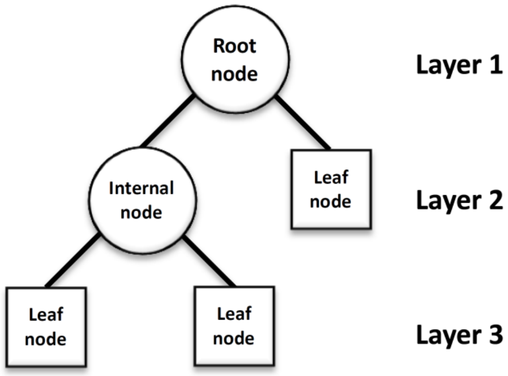


Figure 2.4: Schematic for a single decision tree [17].

Gradient boosting (GB) is an ensemble learning technique, which employs several weak learners to make an overall strong learner. A learner is called weak if it can make predictions better than random guessing, however, with no high enough accuracy. Each learner in the series tries to correct the mistakes of previous learners. Gradient boosted trees use decision trees as weak learners, as shown in figure 2.5. [18] contains a detailed description of decision trees and gradient boosting.

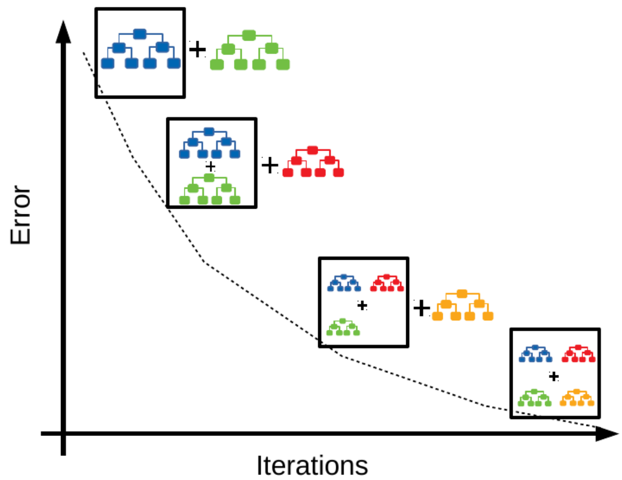


Figure 2.5: Gradient boosted decision trees [19]. More trees help reduce the error.

2.3.4.1 Feature Importance

Decision trees use features to make specific decisions and use them to split the tree at different nodes. Thus, it is possible to measure the relative importance of the features by counting the number of times a particular feature occurs in the trees of a gradient boosted tree model. The number gives an idea of how important a particular feature is in deciding the outcome, whether it is regression or classification. Feature importance plots display the relative importance of the features, thereby enabling picking out of redundant features with far less importance than others. Removal of redundant features can make the model simpler and still achieve similar performance, which helps reduce variance in the model and thus risks of overfitting the training data.

3

Methods

3.1 Data Processing

The flow of the data used in this thesis can be described by the flowchart seen in figure 3.1.

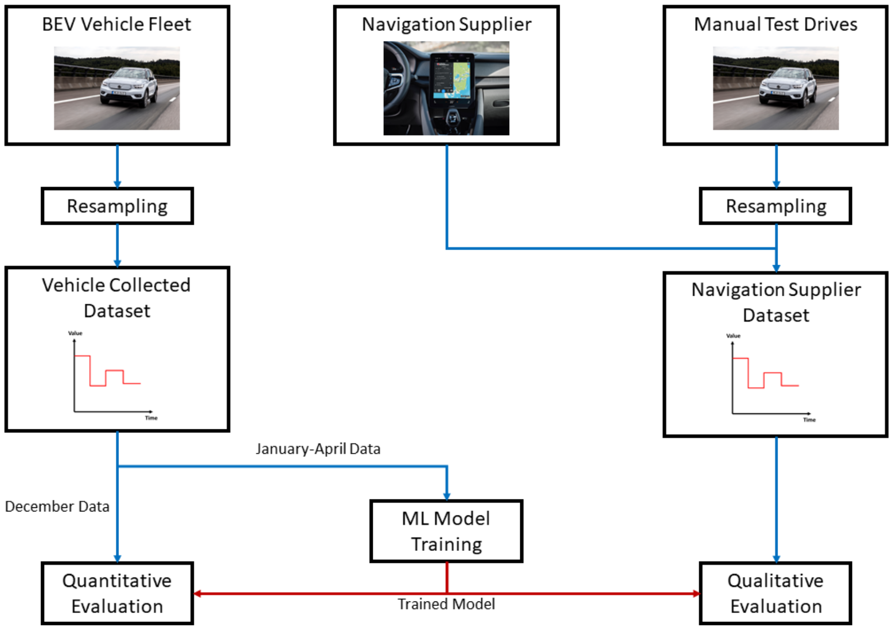


Figure 3.1: Flowchart of data in the thesis.

It can be seen that there are three sources for data; data collected by the Volvo car fleet, data sent by the navigation supplier, and data collected manually by the authors of this thesis. The following section will describe the flow of data, using the three sources. There will also be a more in-depth description of the nodes called resampling in the flowchart, as well as a more technical explanation of how the data extraction was done.

3.1.1 BEV Vehicle Fleet

As mentioned in the introduction, the majority of the data used in this thesis was collected from Volvo's vehicle fleet, called BEV Vehicle fleet in figure 3.1. The data consists of around 18 000 trips between December 2020 and April 2021, and was first resampled, as explained in section 3.1.3. It was then divided into two subsets; one with data from January to April data and one with December data. The January to April data was used to train and validate the machine learning models. The December data was instead used for quantitative evaluation of the trained machine learning models, where evaluation metrics were calculated to compare the different models to each other, see section 3.6.3 for more in-depth explanation of the evaluation.

The data was beneficial to use for training because of its large size and that all interesting parameters were available, nevertheless, there was one drawback with using the vehicle collected dataset for training. The dataset is more precise than the data used when predicting the energy consumption in a real scenario. To have a precise dataset is often a desirable property, except when the data for training is different from the data used in the actual application. In a real scenario, the data comes from the navigation supplier, which gives an estimate of how the trip will look like. The same kind of information can be simulated from the vehicle collected data by only using the same parameters that are available from the navigation supplier. However, the data from the vehicle was collected in advance, giving almost perfect data for the upcoming trip, where the navigation supplier data would be estimated. For example an unexpected traffic accident, causing unexpected traffic jam, would be impossible to predict for the navigation supplier. The performance loss of training the machine learning model on the vehicle collected data can be very small, yet should be considered.

3.1.2 Navigation Supplier and Manual Test Drives

The navigation supplier data is identical to the data that will be used in a real prediction scenario at the beginning of a planned trip, which in theory makes it ideal to use for training, validating, and evaluating the machine learning models. However, the dataset is very limited in size and could therefore only be used for qualitative evaluation of the trained models, see section 3.6.4 for more explanation about the evaluation. The lack in data is mostly due to how the data is collected. The navigation supplier does not supply any power consumption information for the trip since it is specific for the vehicle driving the trip. With no power consumption as the output for the machine learning model, it is impossible to predict it. Therefore, all geographical data given by the navigation supplier needs to be matched with trips run by an actual vehicle, called manual test drives in figure 3.1. The manual test drives data was resampled in the same way as the vehicle collected dataset, see section 3.1.3 for details, and then the power consumption data was added to the data provided by the navigation supplier data to make up the navigation supplier dataset. The dataset was used for qualitative evaluation with the machine learning models trained on the vehicle collected dataset.

3.1.3 Resampling

An issue with all the data collected by vehicles is how the segments are divided. The sampling of the signals is done through internal software from Volvo Cars and the sampling rate is decided to be 10 $samples/s$, or 0.1 $s/sample$, which means that each sample gives the information for a segment of 0.1 seconds. The segments from the navigation supplier are not constant and are for the most part somewhere between 0 and 15 seconds.

To address the issue of different segment lengths for the training and real data, resampling of the vehicle collected data was done. The general idea of the resampling is to take a couple of samples at a time and calculate the mean values for those samples, creating a new segment containing the data. The process will then be done for all the trips. The goal is to create new trips of vehicle collected data with a distribution of segment times that better matches the distribution of segment times of the navigation supplier data. However, finding the pattern of how the 0 to 15 second segments were chosen by the navigation supplier was not intuitive. Instead, the distribution of the segment times was studied to make sense of the data. The probability density distribution of the segment times of the current navigation supplier can be seen in figure 3.2.

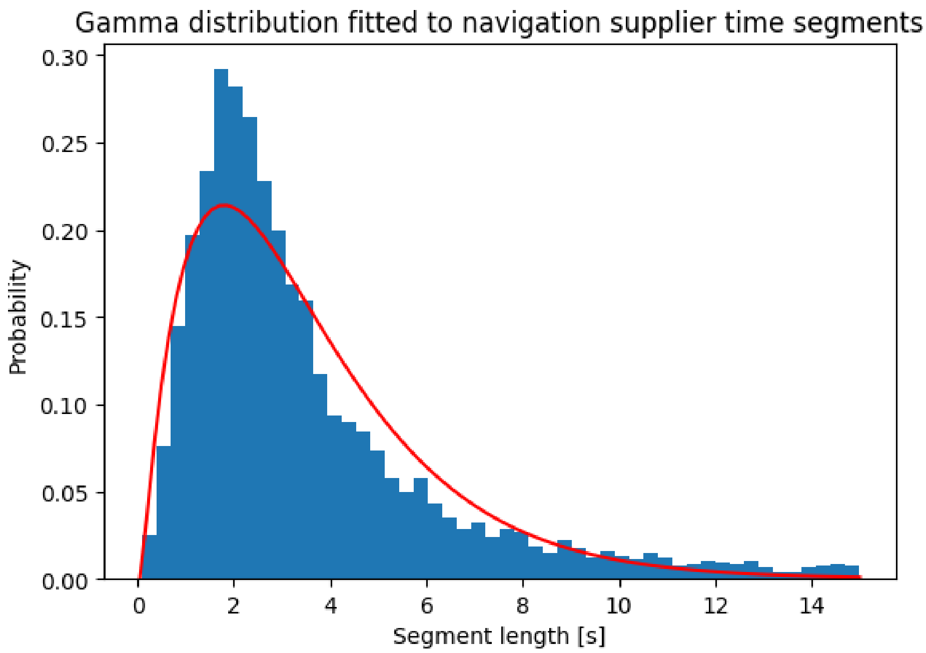


Figure 3.2: Distribution of segment times in navigation supplier data.

It can be seen that the distribution, shown with the blue bars, is very similar to a gamma distribution. The red line in the figure shows a gamma distribution fitted to the time segment distribution, which shows that it fits very well overall. It was therefore decided to draw the segment times for the resampling from a gamma distribution with parameters matching the fitted distribution. Two other resampling methods with constant time segments and time segments drawn from both a uniform distribution were made as reference. Nonetheless, the gamma distribution was proven to be best suited by training the different machine learning models and comparing the results, which is why it was chosen as the final choice for resampling the vehicle collected data.

Figure 3.3 shows an example of how the timeseries data for a specific parameter looks like before and after the resampling. The data was taken from a short snippet of 100 seconds of a manually collected trip and shows the speed signal in that snippet. The blue plot shows the speed with no resampling done and the red plot shows the resampled speed.

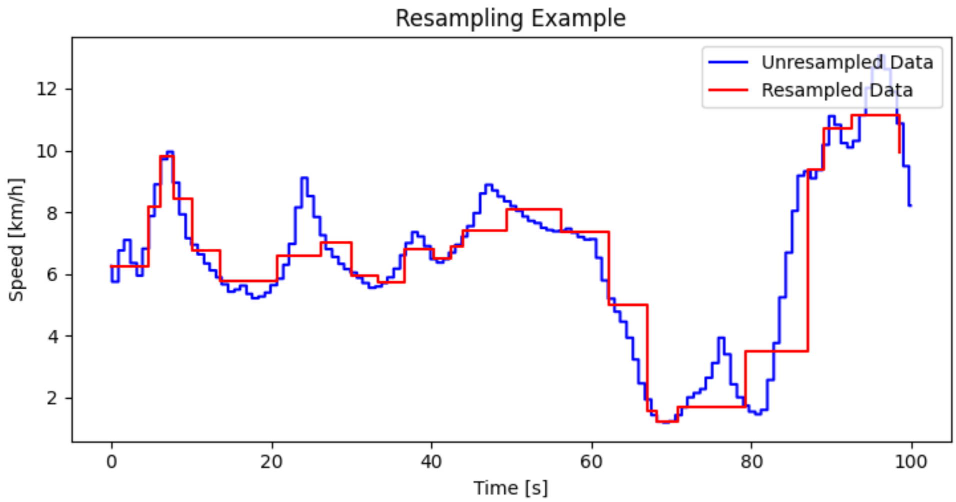


Figure 3.3: Example of speed timeseries data before and after resampling.

3.1.4 Data Extraction

All processed data in this project was timeseries data in the file format of comma-separated values, more commonly known as CSV-files. The file format is easy to work with through a range of different softwares and even the raw CSV-files are easy to understand. The column names were the variables used, for example speed and acceleration, and every row represented each segment of the trip.

The vehicle collected data can be accessed through a database available for Volvo Cars employees and extracted as CSV-files. The employee cars that are part of the data collection project sends automatically the data to the database, making it continuously updated. The original CSV-files were very large and contained a couple of weeks of data, meaning they needed to be separated into one CSV-file for each trip. During the separation a filter was added, which excluded all trips under 10 minutes. The reason for this filtration is that the very short trips can be very different from the trips, that are usually made when a destination is set in the car. The short trips can for example include trips when the driver is just moving around in a parking lot or opens the car door. The selection of 10 minutes was arbitrary, nevertheless, should filter out most of the strange shorter trips.

The CSV-files from the navigation supplier were extracted in a slightly different way. The extraction was done manually in a desktop test bench using Volvo's internal software. The files were extracted for the specific routes driven by the authors, using a test vehicle with the purpose of gathering evaluation data.

3.2 Selection of Parameters

The parameters in section 2.1 are all candidates to be used when predicting a car's energy consumption. However, the parameters needed to be investigated further in order to say if they would add value to the machine learning models. Since machine learning is a data-driven approach, the result of the algorithm is going to be dependent on the quality and the relevancy of the data used. The final parameter choice was therefore strengthened further.

The upcoming three subsections explain how the sample correlation coefficient, a performance comparison in a simple machine learning model, and a multicollinearity test, can be utilized to make a final parameter choice. Then these three concepts are applied to all parameters of interest, in a similar fashion as in section 2.1, to argue if the parameter data should be used or not. The subsection also includes some explanations of how the parameter data was used and if the parameter data was modified before usage. Finally, specific tables of the selected parameters are shown to make make it easier to see the end result of all the parameter analysis.

3.2.1 Correlation Matrices

A correlation matrix is a common way to see if two parameters are related to each other. The matrices used in this study shows the sample correlation coefficient between all parameters in the selected dataset, establishing if a linear relationship between the parameters is present. With a coefficient close to 1 or -1, there is a great chance that the machine learning algorithm can pick out a relation between the two parameters, which makes the correlation matrix a great tool to use before picking out inputs and outputs to a machine learning model. However, one should always be cautious when analyzing a correlation matrix, as correlation does not necessarily mean causality. There can also be problems with multicollinearity in a dataset, explained more in detail in section 3.2.3.

The correlation matrix for the relevant parameters for the propulsive energy prediction can be seen in figure 3.4 and the corresponding matrix for the auxiliary energy prediction can be seen in figure 3.5.

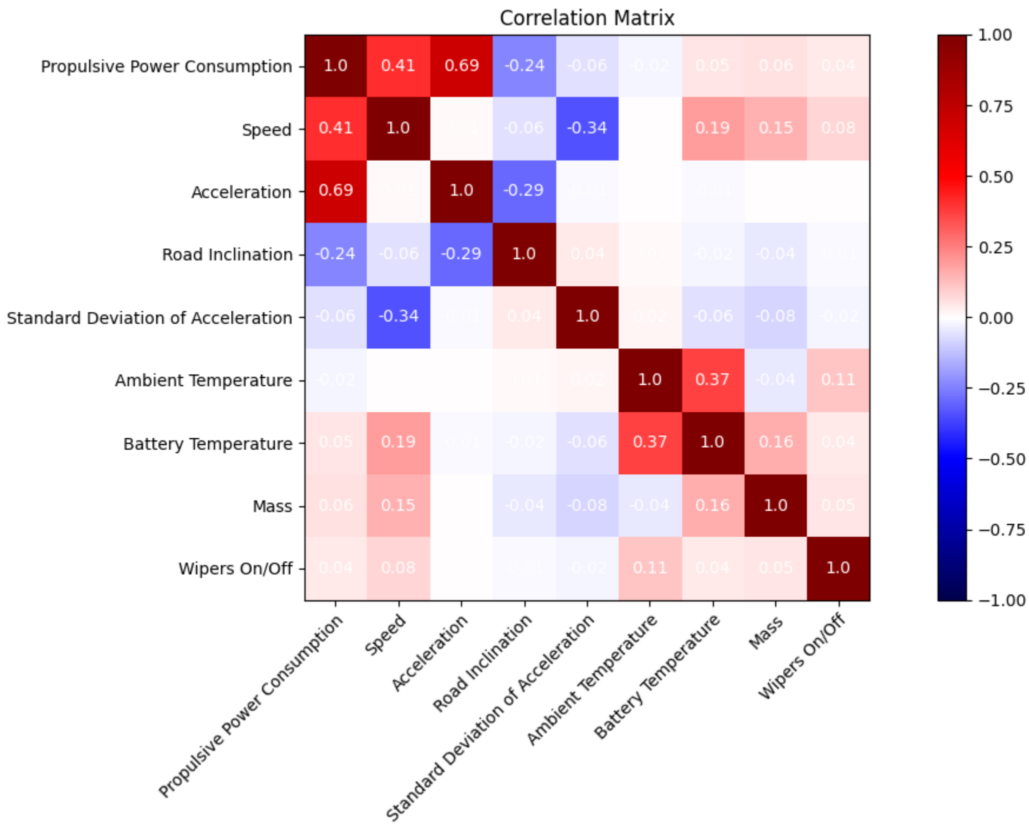


Figure 3.4: Correlation matrix for propulsive energy prediction parameters.

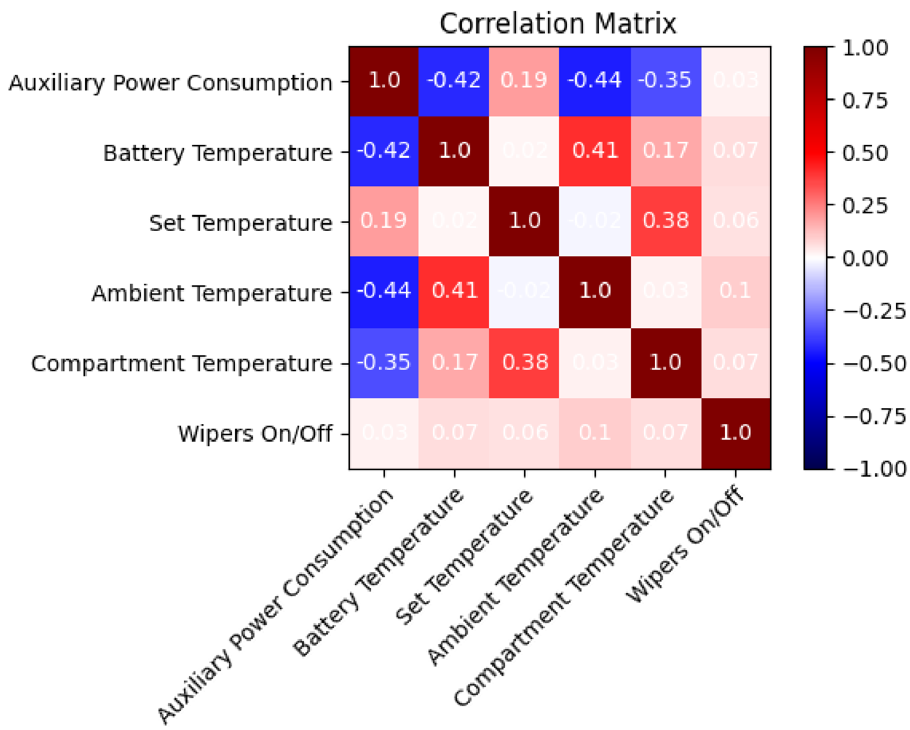


Figure 3.5: Correlation matrix for auxiliary energy prediction parameters.

3.2.2 Performance Comparison

To see how the parameters influence each other in a real machine learning algorithm, the parameters can be put in a simple machine learning model and the performance can be compared. A MLP model, described in section 3.4.2, was used in this study to compare different parameter combinations. However, with huge amounts of possible combinations, some conclusion were made before the tests to reduce the number of test cases. The conclusions were based on some of the presented analysis seen in section 3.2.4 and 3.2.5.

The performance comparison for the relevant parameters for the propulsive energy prediction can be seen in figure 3.4. The output in the testing is the propulsive power consumption, which was the most obvious candidate to use as an output, see 3.2.4.1 for more evidence. The base inputs were narrowed down to speed and acceleration, which are shown in section 3.2.4 to be the two most prominent predictors of the output in the model. The other reasonable inputs were then added one by one in the performance comparison to see if any of the new combinations were better than only the two base inputs. The values seen on the right side in the figure refers to the evaluation metric segment MAE, explained in section 3.6.5.2, where a lower number is better. The three columns, all showing different values, are three different runs of evaluations to make sure that the values are reliable when repeated since there is a stochastic change between runs of training.

Table 3.1: Performance comparison for the propulsive energy prediction parameters.

Inputs	**Segment MAE [Wh/km]**		
Speed, Acceleration	163.00	159.67	161.05
Speed, Acceleration, Mass	160.54	161.72	163.04
Speed, Acceleration, Road Inclination	137.48	137.88	139.75
Speed, Acceleration, Standard Deviation of Acceleration	156.71	157.72	159.58
Speed, Acceleration, Ambient Temperature	165.91	162.55	162.61

Similarly, the performance comparison for the parameters of interest for the auxiliary energy prediction can be seen in figure 3.5. The output in the testing was the auxiliary power consumption, which was the most obvious candidate to use as an output, see section 3.2.5.1 for more evidence. The base inputs were narrowed down to ambient temperature and compartment temperature, which are shown in section 3.2.5 to be the two most prominent predictors of the output in the model. The other reasonable inputs were then added one by one in the performance comparison to see if any of the new combinations were better than only the two base inputs. The values seen on the right side in the figure are the evaluation metric segment MAE values, presented in the same way as in figure 3.4

Table 3.2: Performance comparison for the auxiliary energy prediction parameters.

Inputs	Segment MAE [Wh/min]		
Ambient temp., Compartment temp.	12.72	12.78	12.86
Ambient temp., Compartment temp., Battery temp.	12.43	13.02	13.07
Ambient temp., Compartment temp., Set temp.	12.12	12.14	12.21
Ambient temp., Compartment temp., Battery temp., Set temp.	12.83	12.55	12.53

3.2.3 Multicollinearity Testing

Multicollinearity can be a property to look out for when using multiple inputs to make a predictive machine learning model. To make sure that multicollinearity is not a major concern for the chosen inputs, the VIF values for possible parameters can be checked. The procedure to check the VIF values and remove suspicious parameters can be done as follows [9]:

1. The VIF values for all the possible inputs are calculated.
2. Check if any of the values are high enough to raise concern. If true, move on to step 3. Otherwise, move on to step 4.
3. Remove the parameter with the highest VIF value or, in some special cases, another parameter that is causing high VIF values. Move back to step 2 again. Repeat this process of doing step 2 and 3 until the VIF values are satisfactory.
4. The multicollinearity between the remaining parameters is to be considered not harmful. If any parameters were removed, these should in general not be used with the other parameters since they correlate with them and can cause problems for the machine learning model.

The procedure was done for both the relevant parameters for the propulsive energy prediction and the corresponding parameter for the auxiliary energy prediction.

3.2.3.1 VIF Check - Propulsive Energy Prediction

The VIF check for the potential input parameters are seen in Table 3.3, where the different runs means every time the VIF values were being checked.

Table 3.3: Performance comparison for the propulsive energy prediction parameters.

Parameters	**VIF values Run 1**	**VIF values Run 2**	**VIF values Run 3**
Speed	5.53	2.91	1.51
Acceleration	1.09	1.09	1.09
Road Inclination	1.11	1.11	1.11
Standard Deviation of Acceleration	1.89	1.36	1.28
Ambient Temperature	1.65	1.64	1.30
Battery Temperature	4.78	3.89	-
Mass	9.17	-	-
Wipers On/Off	1.15	1.15	1.15

In the first run the VIF values of speed, battery temperature, and mass were to be considered high, where three of them were above 4. Mass, which had the highest value, was therefore removed and the remaining parameters were re-valued. In the second run, battery temperature still had a suspect number close to 4. Thus, battery temperature was removed. In run 3, all VIF values were very close to 1, meaning that the multicollinearity could be considered very low or non-existent. Combining the parameters remaining in run 3 can be done without significant harm. Both mass and battery temperature should be avoided if possible due to possible multicollinearity.

3.2.3.2 VIF Check - Auxiliary Energy Prediction

The VIF check for the potential input parameters are seen in Table 3.4. As in the propulsive case, the different runs means every time the VIF values were being checked.

Table 3.4: Performance comparison for the auxiliary energy prediction parameters.

Parameters	**VIF values Run 1**	**VIF values Run 2**	**VIF values Run 3**
Battery Temperature	4.58	4.55	-
Set Temperature	40.42	-	-
Ambient Temperature	1.64	1.64	1.34
Compartment Temperature	41.73	3.85	1.43
Wipers On/Off	1.15	1.15	1.15

In run 1, the VIF values of the set and compartment temperature were very high. The reason can be explained by the relationship that is present between the two parameters, where the climate system in the car tries to keep the compartment temperature close to the set temperature. However, only the initial data of each trip was used as input to the machine learning models predicting auxiliary energy consumption, see more explanation for this in section 3.5. The initial set and compartment temperatures should by intuition have no correlation if the car has been standing for some time without the climate system on. The compartment temperature will then be somewhere between the set temperature and the ambient temperature surrounding the car, i.e. there is no significant correlation between set and compartment temperature. Therefore the set temperature could be removed from the parameters, not because it was dangerous to use set temperature together with compartment temperature, rather because it disturbed the VIF testing. Then, in run 2, the compartment temperature VIF value was much lower and more reasonable. The battery temperature was however significant and was removed. Finally, in run 3, all the remaining parameters had very low VIF values and the multicollinearity could be considered very low. The result of the testing was that all parameters except battery temperature can be used together with ease.

3.2.4 Analysis - Propulsive Energy Prediction

This subsection shows the analysis of all the parameters for the propulsive energy prediction one by one based on the correlation matrix, performance comparison, and VIF check. It will also cover if any modifications were done to the parameters before usage.

3.2.4.1 Propulsive Power Consumption

The propulsive power consumption was the choice of output to the machine learning model since the goal is to predict the total energy consumption over a trip. It is available in the vehicle collected data and was used as output during the training, as well as both the qualitative and quantitative evaluation. However, in the qualitative evaluation, it needed to be manually collected, as explained in section 3.1.2.

One modification was done to the propulsive energy consumption data in order to make it better suited for the predictions. The modification was that the propulsive energy consumption in each time segment was converted to energy per distance unit instead of energy per time unit, or more specifically Wh/km instead of W. The conversion can be seen in equation 3.1, where $\frac{1}{3.6}$ is to correct the units and the rest of the terms are the values for one particular time segment.

$$consumption/km[Wh/km] = \frac{1}{3.6} \times \frac{consumption\ [W] \times time\ segment\ duration\ [s]}{time\ segment\ length\ [m]} \tag{3.1}$$

To convert to Wh/km matches the objective of the prediction better as it focuses on energy per distance unit instead of energy per time unit. The assumption is that propulsive energy consumption is more closely related to the distance travelled rather than the time passed. The testing of the MLP model with both Wh/km and W as output also showed that the Wh/km gave better results in general.

3.2.4.2 Speed

The speed was highly correlated with the propulsive power consumption, compared to most of the other parameters, with a correlation coefficient of 0.41 in figure 3.4.

The physical reasoning about the importance of speed, in combination with the high correlation, makes it very believable that speed should be used as an input to the machine learning models. It is available both in the vehicle collected data and the navigation supplier data. The data is also very reliable and works well with other parameters, as seen in the VIF check in section 3.2.3.1.

3.2.4.3 Acceleration

Acceleration is the parameter with the highest correlation to the propulsive power consumption of 0.69, seen in 3.4. Similar to the conclusion of why speed was being used as an input, the physical properties and the correlation coefficient is what favors the acceleration to be an input to the machine learning models. The VIF check in section 3.2.3.1 shows that it works in tandem with speed and was therefore our second chosen input to the models.

The acceleration signal is collected with an accelerometer in the vehicle, giving reliable and usable data in the vehicle collected dataset. However, the navigation supplier provides no acceleration data, which was an issue if acceleration was to be used to predict energy consumption. The acceleration data was instead created as seen in equation 3.2, where a_i is the acceleration of time segment i, $v_i - v_{i-1}$ is the velocity difference between time segment i and $i-1$, and Δt_i is the time duration of time segment i.

$$a_i = \frac{v_i - v_{i-1}}{\Delta t_i} \tag{3.2}$$

The construction, chosen to be called pseudo acceleration, creates a value with a similar trend to the real average acceleration for each time segment. The statement can be strengthened by looking at the correlation coefficient between acceleration and propulsive energy consumption compared to the correlation coefficient between the pseudo acceleration and propulsive energy consumption. The first one is 0.69 and the latter is 0.5. The latter one is slightly worse, however still shows very promising results and was used as an input to account for acceleration.

3.2.4.4 Road Inclination

Road inclination is not as correlated to the output as speed and acceleration are, however, still with a reasonable coefficient of -0.24, seen in 3.4. The testing in section 3.2.2 shows how it, in combination with the previously two mentioned inputs, increased the performance of the MLP model with a lower segment MAE overall than the other tested combinations. The performance test also supported that the road inclination should be the third chosen input. It is also available in both datasets and the VIF check proved that it does not interfere with the already chosen inputs, see section 3.2.3.1.

3.2.4.5 Mass

Even if the mass should have a significant impact on the propulsive energy consumption, it was not shown when used as input in the MLP model in section 3.2.2. The performance was very similar to the one where only speed and acceleration were the inputs. The correlation coefficient to the output is also very low at 0.06, see 3.4.

The non-existent increase of performance was probably due to two combined factors. Firstly, the mass data is estimated using other signals, which is a very hard task to do. Secondly, the mass difference between trips for the same vehicle is small compared to the total weight of the vehicle. The slightly unreliable signal in combination with the small mass fluctuations made it hard for the machine learning algorithm to find any useful connection between the mass signal and the propulsive power consumption. Mass was therefore not used as an input to the models.

3.2.4.6 Standard Deviation of Acceleration

The standard deviation of acceleration, combined with speed and acceleration, showed a slight increase of performance in section 3.2.2. It also showed close to none multicollinearity issues with the mentioned inputs in section 3.2.3.1. Nonetheless, it seem to have very small correlation to the output in 3.4 and it was very hard to utilize the signal in a real scenario. For training and quantitative evaluation, the signal was created in the resampling phase, where it was calculated for each segment of the trip. For qualitative evaluation, the signal is not obtainable since the future acceleration for a trip is unknown. This makes the signal very hard to use in a real scenario even if the performance can increase slightly. It was therefore decided not to use the parameter as input to the machine learning models.

3.2.4.7 Wiper Speed, Ambient Temperature, and Battery Temperature

Using the wipers to tell the outside weather is an interesting concept. Nevertheless, the information says nothing about the future weather conditions, making it unusable even if the machine learning model can distinguish between the energy consumption in different rainfalls. If weather reports were to be accessible to the car's prediction system in the future, the wiper speed could be an interesting parameter to analyze.

The relationship between the different temperatures and the propulsive energy consumption is very complex and when the ambient temperature was tested in the MLP model as input in section 3.2.2, no performance increase was seen, rather a decrease. The correlation in 3.4 is also very low for both battery and ambient temperature. Thus, the two temperatures were left out as inputs to the machine learning models. Better knowledge about the power losses and efficiency of a battery and further investigation is needed before using temperatures as a possible input to the models. An increased temperature span in the dataset could also make relationships observable since the current dataset mostly contains ambient temperatures between -5°C and 10°C.

Wiper Speed, Ambient Temperature, and Battery Temperature were not used in the predictive models because of the above arguments.

3.2.5 Analysis - Auxiliary Energy Prediction

This subsection shows the analysis of all the parameters for the auxiliary energy prediction one by one based on the correlation matrix, performance comparison, and VIF check. It will also cover if any modifications were done to the parameters before usage.

3.2.5.1 Auxiliary Power Consumption

The auxiliary power consumption was the choice of output to the machine learning model since the goal is to predict the total energy consumption over a trip. It is available in the vehicle collected data and was used as output during the training, as well as both the qualitative and quantitative evaluation. However, in the qualitative evaluation, it needed to be manually collected, as explained in section 3.1.2.

The auxiliary energy consumption is closely related to the time passed. Therefore no conversion to Wh/km was made, as done with the propulsive energy consumption.

An important note is that when auxiliary power consumption was used as an output to the machine learning model, time was used as one of the inputs by default. The reason is that the power consumption changes with time. The initial power consumption is usually higher for the climate control system, in order to change the compartment temperature to the set temperature. When the compartment temperature is close to the set temperature, the system can go into a more steady state where it keeps the temperature as it is. See more about how this assumption can be utilized in section 3.2.5.5.

3.2.5.2 Compartment, Ambient, and Set Temperature

Compartment, ambient, and set temperature are all correlated to the auxiliary power consumption with correlation coefficients of -0.35, -0.44, and 0.19, in figure 3.5. Ambient and compartment temperature stands out with the slightly higher coefficients and from a physical standpoint they seem very important to be able to predict the auxiliary energy consumption. With the climate control system as the major consumer of energy, the two parameters are the driving factor of this system. Set temperature is also important for the climate control system. However, it usually moves in the area around 20°C, making the significance slightly less important in theory. The smaller theoretical significance is why it was not used among the base inputs in the performance comparison in section 3.2.2. Nonetheless, the addition of set temperature in the comparison made the best result and the parameter was therefore, together with compartment and ambient temperature, chosen as inputs for the machine learning models. The multicollinearity testing in section 3.2.3.1 also confirms that the parameters can work together as inputs.

3.2.5.3 Battery Temperature

The largest concern with battery temperature is the higher correlation with the already chosen input temperatures, which caused multicollinearity in the dataset, see run 2 in table 3.4. The correlation with other inputs can be traced to the correlation matrix in figure 3.5, where it especially has a high correlation with the ambient temperature of 0.41. The relationship is easy to understand since the battery, if not heated, is usually around the same temperature as the ambient temperature. Adding battery temperature to the inputs in the comparison, section 3.2.2, made the segment MAE more unstable and the model probably had a hard time because of the multicollinearity. With all reasons mentioned, battery temperature was not used as an input to the model.

3.2.5.4 Wiper Speed and Windshield Heating

The dehumidifier, windshield heater, and other auxiliary energy consumers are all important for the total consumption. However, the smaller contributors are relatively small individually compared to the largest consumers, which are the temperature controls of the compartment air and battery. If the windshield heater setting is used to predict the consumption for instance, it will most probably be overshadowed by fluctuations in the consumption, caused by all the parts consuming auxiliary power. There is no way to see the power consumption purely caused by the windshield heater when the output signal is the total auxiliary consumption. The concept can be shown in the correlation matrix in figure 3.5, where the wiper signal has a correlation with the power consumption close to none. The difficulty of establishing relationships gave no other choice but to avoid using the signals wiper speed and windshield heating. If there was a way of measuring the power consumption solely for these functions, maybe they could have been used.

3.2.5.5 Transient Time, Transient Power, and Steady-state Power

While analyzing the auxiliary power consumption it could be seen that most of them follow the same pattern over time. The pattern can be explained as a kind of step function and four examples can be seen in figure 3.6, where all sub-figures represent the auxiliary power consumption in the initial minutes of four different trips. They all start with high energy consumption, named the transient phase in this thesis. Then it decreases quite rapidly and moves into a more steady lower energy consumption, named the steady-state phase. The time when the change occurs is named the transient time.

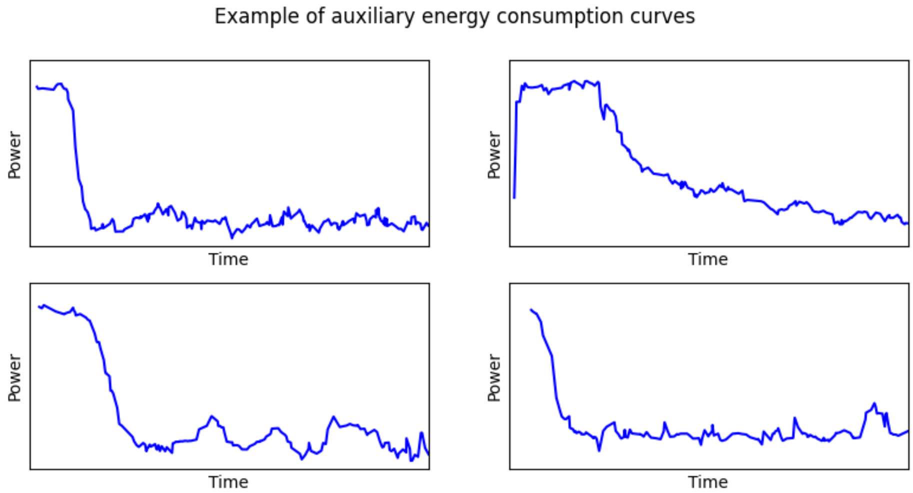


Figure 3.6: Example of consumption curve, polynomial function, and transient time.

With the three parts, the transient phase, the transient time, and the steady-state phase, it is possible to define a step function that can be predicted for a trip. The idea can then be utilized for simpler training since the machine learning model can learn the three properties of the step function instead of a more complex auxiliary power consumption behaviour over time. The issue is annotating the datasets with reasonable numbers for the three parameters. To find estimated values for the transient phase power and steady-state phase power, the median for these time periods can be used for each trip. However, with no transient time, the time periods cannot be decided. Therefore the transient time needed to be decided first and it needed to be done with reasonable precision since the values for the two phases would depend on its accuracy. The way to find the transient time can be described in a simplified version with the 4 steps seen in figure 3.7 from left to right. The first image on the left shows the beginning of a trip with the general step function behaviour previously described. Then a polynomial in red was fitted to the data points describing the auxiliary energy consumption, which is shown in the second image. It is a 10th order polynomial with the goal to smooth out the signal before finding the transient. The third image shows how the polynomial was used to find the beginning of the steady-state phase, indicated by the blue arrow. The point was found by selecting the first point in the polynomial, where the requirements $f'(x) = 0$ and $f''(x) > 0$ were fulfilled. It was then possible to find the point with the largest derivative before the estimated beginning of the steady-state phase, which will give a reasonable estimate of what the transient time is. The found transient time in the example can be seen in the last image represented by the dotted line.

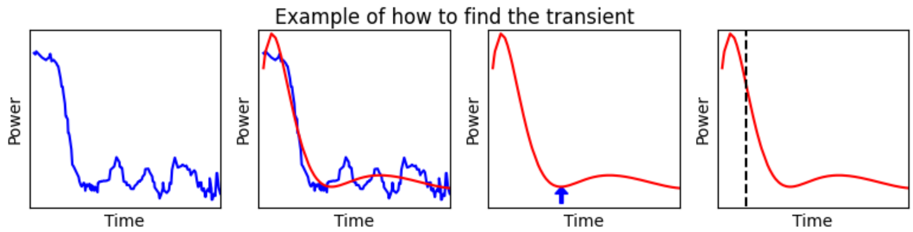


Figure 3.7: Example of how to find the transient time.

In addition to the simplified steps described in figure 3.7, certain other clauses needed to be passed in order to avoid poor estimations of the transient time. For example one clause decides if there is a transient phase at all, or if the whole trip is just steady-state phase. The latter can happen when the car is already heated up and needs no extra heating in the beginning of the trip. It can be due to that the car stayed in a garage, has a pre-heating setting, or has been driven very recently.

The same four auxiliary energy consumption examples shown in figure 3.6 can be seen with the additional polynomial fitting and transient time in figure 3.8. The examples show that the transient times were estimated accurately, even if it is not always perfect as seen in the upper right image, where the transient time estimation is slightly too early. The transient time estimation would probably be better if it was done manually, however, it was not possible to do manually with the amount of trips available. The annotation process gives a very good result, regarding that it was done autonomously. With the transient time set, it is also very easy to determine the transient phase power and the steady-state phase power, as mentioned earlier.

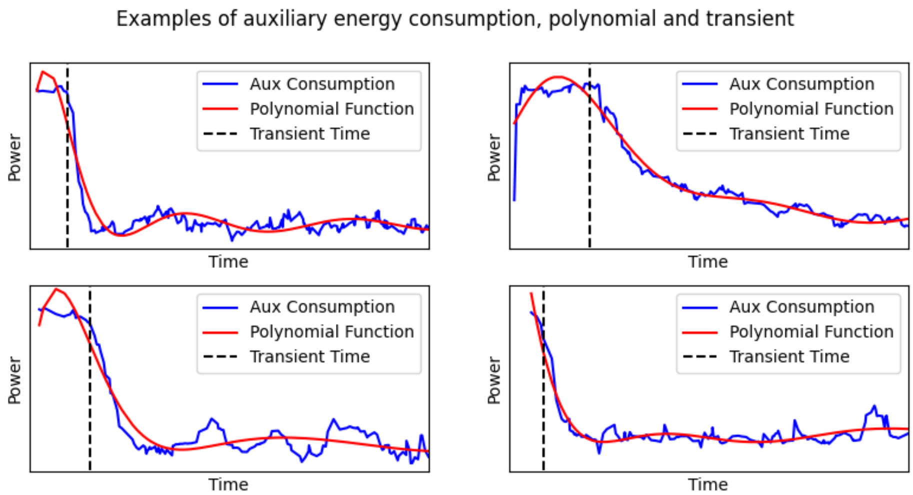


Figure 3.8: Example of consumption curve, polynomial function, and transient time.

It is important to note that the observations of a step-function behaviour not necessarily applies to all ambient temperatures. The data included in the study was collected between December 2020 and April 2021, most probably in Sweden, meaning that the ambient temperature is usually in the lower regions around $-10°C$ to $10°C$. The auxiliary energy consumption peak in the transient phase is then dominated by heating and the relationship does not need to be the same for higher ambient temperatures, where there can instead be a peak for cooling.

The three added signals, transient time, transient power, and steady-state power, were all used as output to the machine learning model if the assumption was made that it follows the step-function structure. The auxiliary power consumption was then removed as an output since the power is already represented in three new outputs.

3.2.6 Table of Final Selection

After all the analysis, all the input and output signals can be listed for simplicity. Table 3.5 shows the final parameter choice for the propulsive energy prediction. Table 3.6 shows the final parameter choice for the auxiliary energy prediction with the assumption of a step-function representing the power levels over time. Table 3.7 shows the final parameter choice for the auxiliary energy prediction without the previously mentioned assumption.

Table 3.5: Final parameter choice for the propulsive energy prediction.

Chosen Parameter	**Input**	**Output**
Propulsive Power Consumption		X
Speed	X	
Acceleration	X	
Road Inclination	X	

Table 3.6: Final parameter choice for the auxiliary energy prediction with the assumption of a step-function representing the power levels over time.

Chosen Parameter	**Input**	**Output**
Ambient Temperature	X	
Compartment Temperature	X	
Set Temperature	X	
Transient Time		X
Transient Phase Estimate		X
Steady-state Phase Estimate		X

Table 3.7: Final parameter choice for the auxiliary energy prediction without the assumption of a step-function representing the power levels over time.

Chosen Parameter	Input	Output
Auxiliary Power Consumption		X
Ambient Temperature	X	
Compartment Temperature	X	
Set Temperature	X	
Time	X	

3.3 Data Distribution Check

When all the parameters that are going to be used in the machine learning models are chosen, the distributions of the parameter data is important to check. The distribution of a parameter in the used dataset should correspond to the distribution of the parameter in the real world, i.e. the distribution should be reasonable considering what it shows.

The distribution of the four parameters used in the propulsive prediction model are seen in figure 3.9. The distributions are shown as probability density histograms and the propulsive power consumption is normalized to avoid giving away sensitive information about the used vehicles. For the other three sub-figures, the X-labels show the units of the parameters. It can be seen that both road inclination and acceleration, seen in the two lower sub-figures, correspond pretty well to the real world. It makes sense that both are normal distributions around zero since every acceleration has a corresponding deceleration and every positive slope has a corresponding negative slope. Nevertheless, both the power consumption and the speed are probably unreasonable. It is not possible to know exactly how the distributions should look like, however, the large peak around zero is suspicious. A general trip will most likely consist of small amounts of these very low speeds and the peak is probably present because of how the collection is done in the vehicles. If the vehicle stands still in a parking lot for ten minutes before going away, the first ten minutes will be considered a part of the trip, which they are not in the eyes of a navigation system. The navigation system assumes that the driver drives away immediately.

The mentioned skew in data can cause problems for the machine learning models as they learn, untruly, that the vehicle is very likely to stand still. Therefore the amount of very slow moving samples need to be reduced. The reduction is done by filtering out 90 percent of all samples under 5 km/h. The rest are kept to give some data for the model to learn from if the car actually will drive in the lower speed regions.

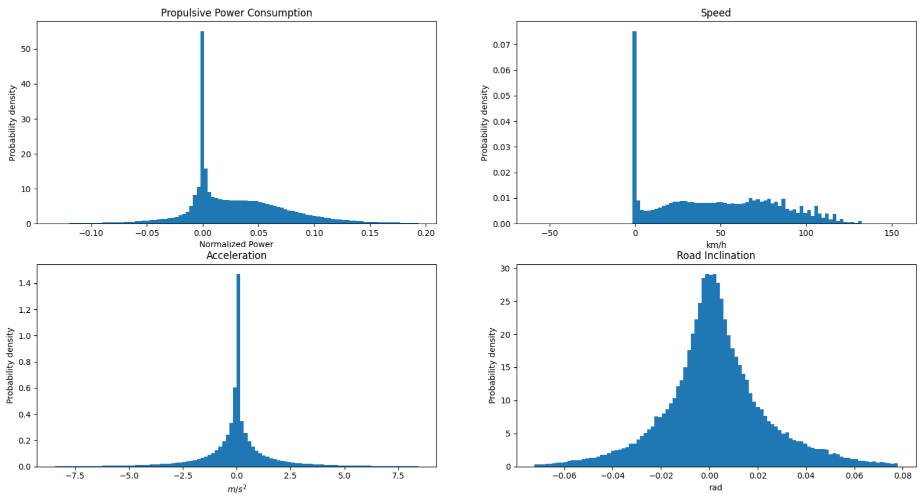


Figure 3.9: Example of consumption curve, polynomial function, and transient time.

The result of the filtering of low speed segments can be seen in figure 3.10. The propulsive power consumption and speed distributions are not perfect. Nonetheless, much better than in figure 3.9.

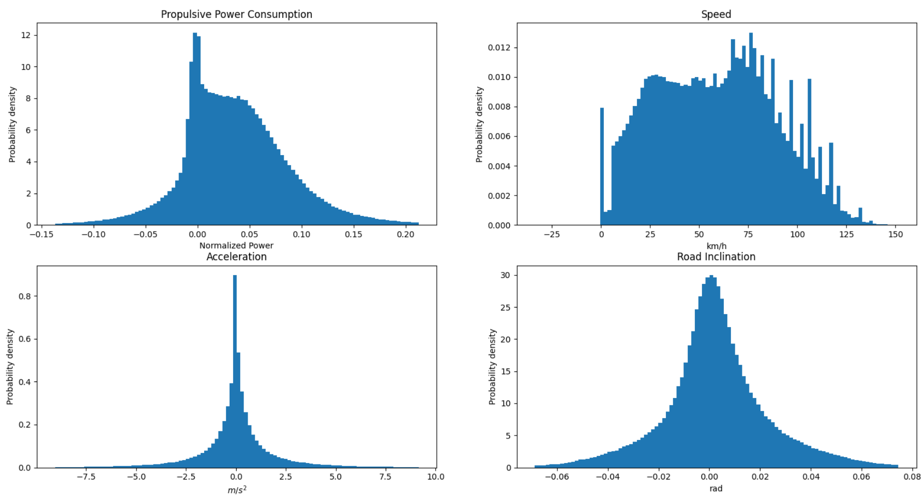


Figure 3.10: Example of consumption curve, polynomial function, and transient time.

The distribution of the parameters used in the auxiliary prediction model are seen in figure 3.11. Similar to the propulsive parameters, the auxiliary power consumption is normalized and the rest have the correct units. The distribution of auxiliary power consumption looks as expected with two separate peaks, where the higher peak is the usual steady-state power consumption and the lower peak is the usual transient power consumption, matching the assumption of a step-like function explained in section 3.2.5.5. Ambient, compartment, and set temperature all look reasonable, considering that the data is collected between December and April in Sweden.

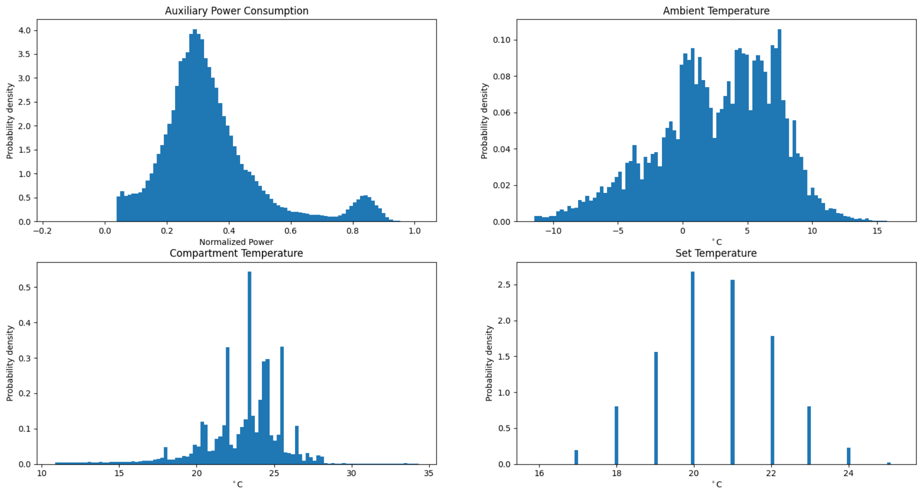


Figure 3.11: Example of consumption curve, polynomial function, and transient time.

3.4 Propulsive Consumption Prediction

The major part of the total energy consumption in battery electric vehicles is usually the propulsive part. One thing to be mentioned is that, in this study, the propulsive part also includes all the electrical losses in the powertrain and battery. Section 3.2.4 mentions the selected signals, which affect the propulsive consumption.

For a machine learning model, the problem of estimating power required for traversing a road segment, using the segment information, can be converted into a regression problem. The required output power is to be estimated as a function of all the input signals. One important thing to be remembered is that even though one can use a large number of signals to make the predictions extremely accurate, in reality, only a few signals like the vehicle speed and road inclination in each segment will be known beforehand. This makes choosing the input signals required to estimate the consumption crucial, as they will be key to deciding the bias and variance in the machine learning models.

In most of the neural network models, the loss function used is mean squared error and the weights are updated via Adaptive Moment Estimation, or Adam in short. This is primarily due to the fact that Adam was found to be much faster than stochastic gradient descent method with momentum, while giving very similar results. All the models use some form of early stopping mechanism to prevent overfitting, thus resulting in a better performance on test trips. In the subsections that follow, the machine learning models used for energy consumption prediction and the key ideas behind using them will be discussed.

3.4.1 Linear Regression

The linear regression model served as a baseline for the more complex machine learning models discussed in the sections ahead. In the LR model, output consumption per kilometer is linearly dependent on the selected input signals. The loss function used for training the model was mean squared error. For each road segment, the propulsive consumption was computed using the input signals for the individual segments.

3.4.2 Multi-Layer Perceptron

A feed-forward network goes a step beyond and tries to find non linear relationships between the propulsive power and chosen input signals. The activation function used in the layers was the rectified linear unit (ReLU). Two hidden layers were used and the output layer was simply a weighted sum of previous layer's output to allow full range of numbers in the output.

3.4.3 Recurrent Neural Network

The trip data is sequential in nature, since it is time series data. It therefore becomes natural to think that propulsive power consumption in each segment will depend not only on the input signals of that segment but also on the information from segments that have come before it. Recurrent neural networks are efficient in finding sequential relationships between data points. Thus, an architecture with an LSTM layer, followed by two fully connected layers, was used. The last layer did not have an activation function to allow for full range of values in the output.

Some of the trips in this study can be long, potentially having hundreds of segments. Using very large number of segments can make it harder for the network to learn something meaningful as information from segments far back in time is forgotten. The power consumption in a segment will depend strongly on a fewer number of more recent segments and less on segments far back in time. Thus the training data was used to create small sub-trips, where each sub-trip was 10-30 segments long. The stride size can also be decided to choose the amount of data. For example, a stride size of 1 would mean that segments 1-20 make one sub-trip, followed by segments 2-21 and so on. A small stride size will create a larger amount of data and thereby increase the time to train the model. Thus, a stride size of 2 or more is typically chosen. Towards the end of a trip, when the number of remaining segments is less than the sub-trip size, the remaining segments were not included in training data. There can be two ways of training the model, one being many-to-many structure and other being many-to-one. The training methods are discussed in detail in the next sections 3.4.3.1 and 3.4.3.2.

3.4.3.1 Many-to-many

In the many-to-many architecture, at each time step all the input signals for that particular time step were used as input and the output consumption per kilometer was generated for all the time steps. Thus, if there are, for example, 20 segments in one sub trip, then this model will output 20 values corresponding to each segment. Training was performed by using the loss function on all the outputs. In this way, the model learns the relationship using the outputs from all the segments.

3.4.3.2 Many-to-one

Contrary to the many-to-many architecture, in many-to-one, only the last output in a sub-trip was used to train the model. Thus, if there are 20 segments in a sub-trip, the model will generate one output once all the input signals for each segment have been fed. The loss function only takes in this output and uses it to train the model for each such sub-trips. This essentially means that the final output is more strongly dependent on all the previous segments since now the loss function is only considering the final output instead of outputs from all the segments.

3.4.4 Gradient Boosting

The gradient boosting implementation used in this thesis was the extreme gradient boosting (XGBoost), which is highly efficient, optimized, and extremely flexible [20]. XGBoost uses several optimization techniques and parallelization to achieve the best out of the computational devices and can work with huge amounts of data. This is beneficial, as the model has to learn over thousands of trips each with hundreds of segments. It also uses higher order regularisation terms and second order terms in Taylor expansion of the loss function [21], which helps the models to generalize better. In order to prevent overfitting, which can happen with deeper trees and more number of iterations, moderate tree depths and early stopping was used. DMatrix data structure was used in the training since these are the internal structures used by XGBoost, which are optimized for both memory and training. After completion of training, the trained model can be used to see the relative feature importance of the input features. Feature importance plot obtained for propulsive consumption prediction model is shown in figure 3.12 with the features on y-axis and their importance on x-axis.

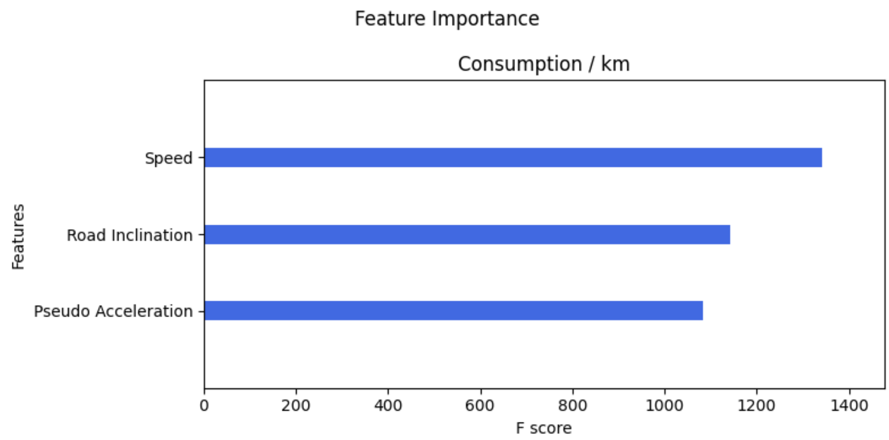


Figure 3.12: Feature importance plot for gradient boosting model based on F scores.

3.5 Auxiliary Consumption Prediction

Propulsive consumption is one part of the total consumption. The other part is the auxiliary consumption, which includes all the auxiliary systems in the car including the battery heating, cooling systems, stereo system, displays, wipers, defrosters, seat heating, etc. In this analysis, the auxiliary part only contains the actual auxiliary power demand without the associated electrical losses as these losses were lumped together in the propulsive consumption part. Section 3.2.5 mentions the selected signals, which affect the auxiliary consumption.

One key aspect in auxiliary consumption prediction is that only the initial values of the input signals can be used. This is due to the fact that for most of the input signals like compartment temperature, wiper speed or status, weather conditions, etc., only the initial conditions before the start of the trip are known. It is impossible to know what the set temperature is going to be in future. Even though it is possible to know the weather conditions like ambient temperature, wind speed, rain condition, etc. along a trip ahead in time, in this report these signals are not provided from the navigation supplier and thus only the initial values of the mentioned signals can be used. The only information that can be used from navigation supplier's data in auxiliary prediction is the time information, or how long it would take to traverse each segment, and thus the entire trip time in total.

3.5.1 Mixed Model

The auxiliary consumption problem can also be converted into a regression problem with a couple of modifications. Some assumptions were also made in order to successfully predict auxiliary consumption. The assumption was made on the nature of auxiliary power demand signal, based on observations from a large number trips. Most of the trip data used in this report is from colder climate regions and the auxiliary power in these trips has a transient phase with relatively higher consumption, followed by a steady state phase, where the consumption is more or less constant throughout the rest of the trip, mentioned in section 3.2.5.5. The problem can be converted into an intermediate problem of predicting the transient state time (T_{ts}), the transient state power level (P_{ts}), and the steady-state power level (P_{ss}) using the initial conditions, i.e. the values of the chosen input signals at the start of a trip. The intermediate output values of transient state time, transient state power, and steady state power were obtained from the model. Using the assumption made on the nature of auxiliary power demand, the intermediate output values were put together using a Heaviside function to make the final prediction, as shown in equation 3.3, where $H(x)$ is the Heaviside function.

$$P_{aux} = P_{ts}H(T_{ts} - t) + P_{ss}H(t - T_{ts}) \tag{3.3}$$

The machine learning models used for predicting the three intermediate outputs, i.e. T_{ts}, P_{ts}, and P_{ss}, will be called mixed models and are discussed in following three subsections. Once the intermediate outputs are obtained, the procedure to receive the final output with equation 3.3 is same for all of them.

3.5.1.1 Linear Regression

As in the case of propulsive consumption, linear regression was the simplest model to start from as it can serve as the baseline for all other machine learning models. The matrix form of the regression problem is shown in equation 3.4, where, x_j are the input signals e.g. ambient temperature, set temperature, compartment temperature etc., and β_{ij} are the parameters to be found out.

$$\begin{bmatrix} T_{ts} \\ P_{ts} \\ P_{ss} \end{bmatrix} = \begin{bmatrix} \beta_{11} & \beta_{12} & \dots & \beta_{1n} \\ \beta_{21} & \beta_{22} & \dots & \beta_{2n} \\ \beta_{31} & \beta_{32} & \dots & \beta_{3n} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} \tag{3.4}$$

3.5.1.2 Multi-Layer Perceptron

It is more likely that the relationship between the inputs and outputs is non linear, and thus it would require better machine learning models to come into play. Multi-layer perceptron was the first step towards enabling non-linear features to be learnt. The model has multiple inputs, each corresponding to the initial value of an input signal, some hidden layers, and an output layer with three units corresponding to T_{ts}, P_{ts} and P_{ss}, seen in figure 3.13.

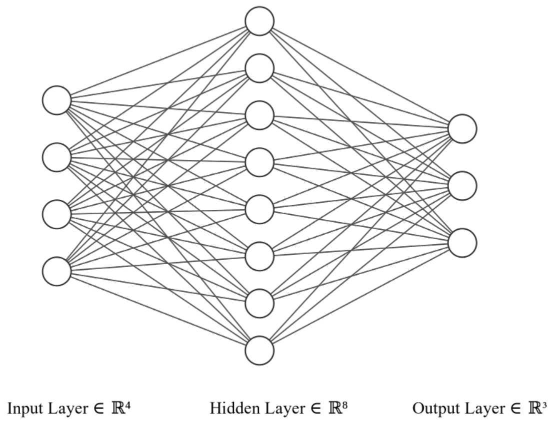


Figure 3.13: MLP model used in the mixed model

3.5.1.3 Gradient Boosting

Decision trees perform well for regression related tasks and thus gradient boosted decision trees can perform even better, given that it takes into account mistakes from all the individual weak learners.

As there are three intermediate outputs for the mixed model, there needs to be one gradient boosting model for each of the three outputs. The outputs of the three trees were concatenated to give an output structure similar to the ones obtained by previous models. These were then combined using step function to get the final auxiliary power consumption prediction.

Feature importance plot obtained for auxiliary consumption prediction model is shown in figure 3.14 with the features on y-axis and their importance on x-axis. The three subplots contain the feature importance for models corresponding to prediction of the three intermediate outputs.

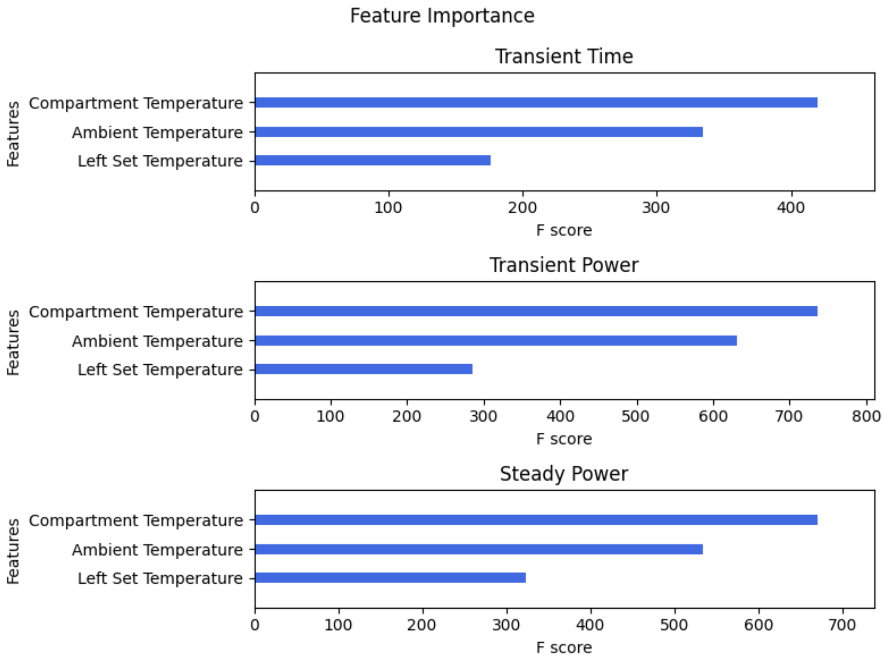


Figure 3.14: Feature importance plot for gradient boosting model based on F scores.

3.5.2 Recurrent Neural Network

As in the case of propulsive power prediction, it is natural to use an RNN model for auxiliary power prediction as well. The model is different from the mixed model as no assumption was made on the nature of the auxiliary power consumption, i.e. the assumption that there is a transient phase followed by a steady state phase. Not having the mentioned assumption makes the RNN model learn the behavior of the auxiliary power consumption. However, it comes at the cost of increased complexity, because the model has many more parameters and settings, thus making it harder to tune the model and receive robust results. The LSTM layer in the RNN model can be stacked, meaning that more than one LSTM layer can be used. It was followed by two fully connected layers to convert the LSTM layer's output to the final output. The output layer was simply weighted sum of previous layer's output, i.e. it did not use an activation function to allow the full range of numbers. Using only the initial values of the input signal values, obtained at the beginning of a trip, makes the predicting auxiliary power consumption more challenging.

It became important to use the trip time, defined as the time elapsed from the beginning of the trip, as an input to the model since the model had to know the amount of time elapsed at any given point in a trip and thus switch from transient state to auxiliary state smoothly depending on the time and the initial conditions. Using the running time in seconds as an input is always tricky because it is monotonically increasing quantity and can reach large values for a longer trip. In this work, trip time was split into hours, minutes, and seconds elapsed. All three values were used as three different inputs at every time stamp to the model. This is a common way of using continuous time as an input to a machine learning model. Apart from using time as an input, there are other settings which can affect the model's training and performance. The next two subsections discuss the two ways of implementing the recurrent network model used in this work.

3.5.2.1 Initializing Hidden States

The hidden states of the LSTM layer are generally initialized to zero by default. In this thesis, the initial values of the input signals were used instead to initialize the hidden states of the LSTM layers. More specifically, some of the elements of the initial hidden state vector were set to the scaled initial values of the input signals. The training was carried out using output from all the time steps and for each time step the only input signals used were the hours, minutes, and seconds calculated from the running trip time. An example can be seen in figure 3.15.

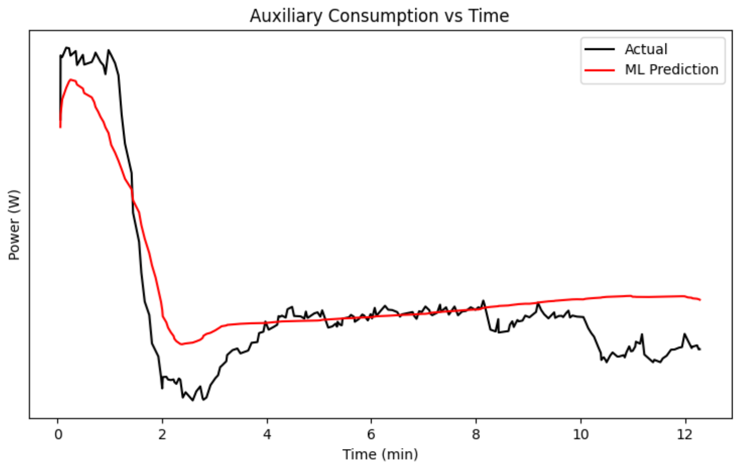


Figure 3.15: RNN model using time as input and hidden states initialized with temperature signals.

Sometimes discontinuities were seen in generated power at every second and even more prominently after every minute and hour since the input time signal used is also discontinuous, for example the 'seconds' signal wraps back to zero after reaching value of 60 or one minute and similarly 'minutes' signal wraps back to zero after one hour. Thus, to prevent the discontinuity another trick was used, which was to use sine or cosine transform on each of the hour, minute, and second time signals. It would mean that when the seconds go from 60 to 0, the transformed second signal would not have a similar discontinuity as sine and cosine always vary smoothly. Even this transformation was insufficient to prevent the small fluctuations in the model's output. The model was incorporating patterns in input time as a feature in the predicted output signal. From the analysis, it seemed like using only time as an input was not sufficient and additional inputs were necessary to stabilize the output. The idea is discussed in the next subsection.

3.5.2.2 Autoregressive Approach

In this approach, the output signals consist of the auxiliary power along with the compartment temperature. Thus, the model learned to generate both auxiliary power and also learned to predict the compartment temperature. The initial values were not used to initialize the hidden state of the LSTM layers. All the input signals along with the time signal, i.e. hour, minute, second, were used as input to the model at every time step. The input temperature signals were shifted forward by one time step or one segment, which enabled the model to see the signals from previous time step as an input for current time step. Values of temperature signals from all segments were used when training the model, whereas while evaluating on test data, the initial values of all temperature signals except the compartment temperature, were used for all the segments. For compartment temperature, the output of previous time step was used as input for the current time step since it was generated along with the auxiliary power. The model learned to generate both auxiliary power and compartment temperature simultaneously. One example is shown in figure 3.16.

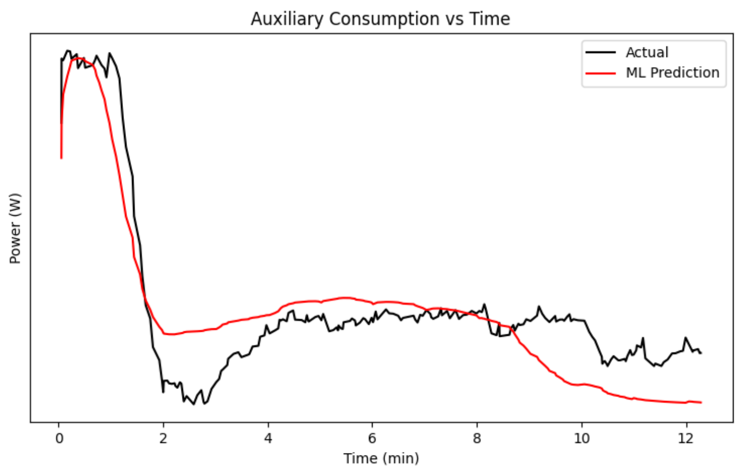


Figure 3.16: RNN model using time and temperature signals as input.

The autoregressive approach also suffers from the problem of discontinuous time as mentioned in the previous subsection. One solution to avoid using the time as an input would be to use constant resampling, i.e. a resampling of segments, where each segment has same length in time, as mentioned in section 3.1.3. Constant resampling can be done by resampling the originally resampled segments into constant ones and then the model output can be resampled back into the original segment lengths. The resulting output is dependent only on input temperature signals and overcomes the problem of discontinuities in time. It generates a constant steady state as seen in figure 3.17.

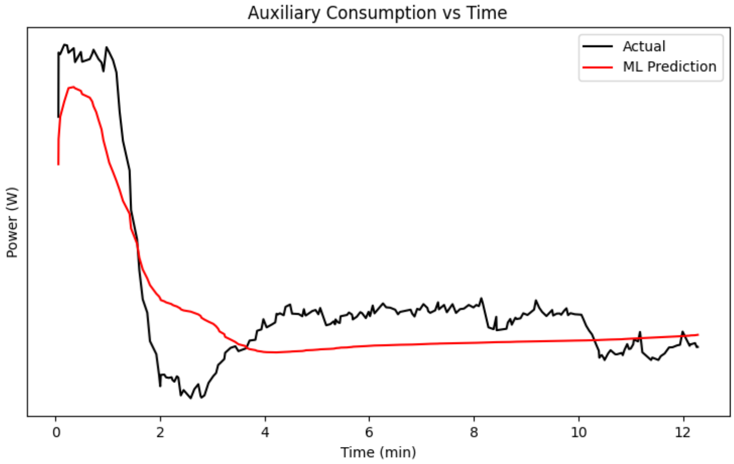


Figure 3.17: RNN model using only temperature signals as input.

3.6 Evaluation

This section will describe the evaluation of the ML models. First the difference between the evaluation of the propulsive and auxiliary models will be explained. Then the two types of evaluation, qualitative and quantitative, will be discussed. Finally, the specific evaluation metrics are shown.

3.6.1 Propulsive Consumption

The propulsive consumption machine learning models can be trained to predict the power consumption per segment or the consumption per kilometer for each segment. If the output is power (Watt) then it is multiplied with the time of the segment converted to hours (h), shown in equation 3.5. Thus, the energy for every segment in Watt-hour (Wh) is obtained. Similarly, when the output of models is consumption per kilometer (Wh/km), then it is multiplied with the length of each segment converted to kilometers (km), shown in equation 3.6, thereby giving the energy for each segment (Wh).

$$\text{Case 1: } E_{prop}^{segment}[i] = P_{prop}^{segment}[i] \times \Delta t^{segment}[i] \tag{3.5}$$

$$\text{Case 2: } E_{prop}^{segment}[i] = C_{prop}^{segment}[i] \times d^{segment}[i] \tag{3.6}$$

In the equations, P is power, C is consumption per kilometer, d is segment distance, t is segment time and $[i]$ denotes i-th segment. The energy per segment is then summed up to give the total energy consumption, shown in 3.7. The total energy consumed subtracted from initial battery energy gives the expected final battery energy.

$$E_{prop}^{total} = \sum_i E_{prop}^{segment}[i] \tag{3.7}$$

It was found that using consumption per kilometer gave better results compared to power since in the segments where the car is standing still, for example in a traffic, if the model's output is slightly non zero then equation 3.5 will be non zero since segment time would be non zero as well, whereas equation 3.6 would be zero as the segment distance would be zero. Thus, in all the evaluations on test trips, consumption per kilometer was used as an output for the models.

In order to asses a particular model's performance, one needs to look at the outputs both quantitatively and qualitatively, which will be described in more detail in sections 3.6.3 and 3.6.4.

3.6.2 Auxiliary Consumption

The models developed to predict auxiliary consumption were trained on the average power required for each segment since auxiliary consumption is dependent on the time elapsed, not the distance travelled. The predicted power is multiplied with the respective segment time, see equation 3.8, and then summed up to get the total predicted energy consumption due to auxiliary loads, see equation 3.9. In the equations, P is power, t is segment time and $[i]$ denotes i-th segment.

$$E_{aux}^{segment}[i] = P_{aux}^{segment}[i] \times \Delta t^{segment}[i] \tag{3.8}$$

$$E_{aux}^{total} = \sum_{i} E_{aux}^{segment}[i] \tag{3.9}$$

Similar to propulsive consumption prediction, to receive a complete picture of the models developed, both qualitative and quantitative evaluations are required, discussed in sections 3.6.3 and 3.6.4.

3.6.3 Quantitative Evaluation

The models were trained on the vehicle collected dataset, which consists of thousands of individual trips. A fraction of these trips, around two weeks of vehicle data, was set aside to be used as test data on which the evaluation was performed. The number of trips is sufficiently high to be able to reliably receive the various metrics seen in section 3.6.5 over all the test trips. Averages like mean, median, or mode can be calculated over all the trips to look at overall performance of the model. Maximum and minimum values can be checked to see if there is any anomaly, i.e. if the model performs particularly worse on some trips, and then the identified trips can be analyzed individually to check where the problem lies. All these assessments allow to receive robust numbers indicating the model's performance on various aspects.

The quantitative evaluations were done on the logged vehicle data, which contains all the signals used as input. However, while performing evaluation only the signals which are also available from the navigation supplier's data, like vehicle speed, segment time, segment distance, and road inclination, are used to allow for a fair evaluation. A default value was used for all other signals used as input to the model, however, cannot be known ahead of time. The default values for signals were either based on average values calculated from the training data or based on engineering intuition. Quantitative evaluations cannot be done with navigation supplier's data since, as mentioned earlier, there was not enough number of trips to do so.

3.6.4 Qualitative Evaluation

There were very few test trips with the navigation suppliers data present along with the logged vehicle data. These trips can be used to do a more qualitative evaluation of the trained models. Each trip was looked at individually using both the numbers showing the model's performance metrics and various plots, showing the predicted energy consumption behavior for different types of trips. In this assessment, it was possible to see the model's prediction on the navigation supplier's data, which represents the real data that the model will use when commencing an actual trip. This can be compared to the evaluation on the corresponding logged vehicle data, which represents data where the perfect signal values for the trip were known ahead of time. Thus, a comparison of the predictions on navigation supplier's data and vehicle collected data, and the original consumption can be used to do further analysis.

When evaluating auxiliary models on the navigation supplier's data, only the time information was used from the data since it does not contain any of the temperature signals. To handle this, the initial values of the signals were collected from the corresponding logged vehicle data and used in the models. This is a reasonable workaround, because in a real scenario, the model would have both the navigation supplier's data and the initial sensor readings at the beginning of a trip.

3.6.5 Metrics

To quantify any of machine learning models' performance, a set of metrics were needed. The models were trained on a loss function and the value of the loss function itself can serve as a metric in some cases. However, in the case of energy consumption prediction, it will not be sufficient to just use the loss function to evaluate a model's performance. There are a few reliable metrics that can be used and are discussed below.

3.6.5.1 Absolute Error (AE)

Absolute error metric is defined as the absolute value of the difference between the predicted energy consumption and the actual energy consumption at the end of a trip. The energy can be from propulsive, auxiliary, or combined consumption, i.e. including both propulsive and auxiliary consumption. The difference is calculated for the total energy or the summation of energy consumption for all the segments, i.e. energies in equations 3.7 and 3.9. The calculation is shown in equation 3.10.

$$\mathrm{AE} = |E_{\mathrm{actual}} - E_{\mathrm{predicted}}| \tag{3.10}$$

The absolute error at the end of a trip is dependent on the length of a trip. For very short trips, the absolute error would probably be very small, i.e. a fraction of battery's state of charge, whereas if the trip is longer, then the AE will be larger. Thus, it is reasonable to normalize the absolute error with the length of the trip. The normalized quantity is measured in units of Watt-hour per kilometer (Wh/km) for propulsive consumption and Watt-hour per minute (Wh/min) for auxiliary consumption. Wh/km shows the error in energy prediction for every kilometer driven and Wh/min shows the error in energy prediction for every minute of driving. The normalized metric is defined mathematically in equation 3.11, where total length is the total distance in kilometers for propulsive consumption and total trip time in minutes for auxiliary consumption.

$$\text{NAE} = \frac{|E_{\text{actual}} - E_{\text{predicted}}|}{\text{Total Length}} \tag{3.11}$$

3.6.5.2 Mean Absolute Error (MAE)

Mean absolute error takes into account the energy predictions in all the segments individually instead of just the final integrated value. Mathematically it is defined as in equation 3.12, where n is the total number of segments in a trip and i is the i-th segment.

$$\text{MAE} = \frac{1}{n} \sum_{i=1}^{n} \left| E_{\text{actual}}^{\text{segment}}[i] - E_{\text{predicted}}^{\text{segment}}[i] \right| \tag{3.12}$$

The MAE is calculated for energy (Wh) per segment and also for energy per unit (Wh/unit), where unit is kilometer (km) for propulsive consumption and minutes (min) for auxiliary consumption. Segment MAE for energy is affected by the length of the segments as the error would be proportional to the segment length. Segment MAE for energy per unit is a better metric than segment MAE for energy as the errors are normalized by the segment lengths. The segment MAE for energy per unit will be referred to as 'Segment MAE' from hereon.

All the machine learning models are trained to predict the power or consumption for each segment. Segment MAE for energy per unit directly measures the difference between the model's raw output and corresponding actual value. Therefore, segment MAE remains fairly constant when same model is trained multiple times with different random seeds. Due to randomness in weight initialization and shuffling of training trips, trained models are slightly different from one training to other. Total energy at the end of trip takes into account the integrated error from all the segments, which could be noticeable for different training runs. AE and normalized AE metrics get affected by this to a minor degree but the segment MAE is almost unaffected. Thus, segment MAE is a more robust measure of gauging how well a particular model was trained.

3.6.5.3 Mean Absolute Percentage Error (MAPE)

Another metric, which is very popular in the domain of energy forecasting, is the mean absolute percentage error. MAPE measures the average values of the relative error in a prediction. In this, the absolute difference in each segment is divided by the corresponding actual energy consumption. This is then converted to a percentage by multiplying with 100, depicted in equation 3.13, where n is the total number of segments in a trip and i is the i-th segment.

$$\text{MAPE} = \frac{1}{n} \sum_{i=1}^{n} \left| \frac{E_{\text{actual}}^{\text{segment}}[i] - E_{\text{predicted}}^{\text{segment}}[i]}{E_{\text{actual}}^{\text{segment}}[i]} \right| \tag{3.13}$$

The metric can be unstable when the actual value used in the denominator of equation 3.13 is very small or close to zero. If the predicted value is also numerically very small or zero, the result can be a division by a small number, which can blow up the overall MAPE value. In the case of vehicle energy consumption, especially propulsive energy consumption, the actual values of energies in some segments can be very close to zero or even exactly zero, as in the case of a vehicle standing still. This can lead to very large MAPE values, which do not represent the model's predictive performance. Therefore, metrics discussed in earlier subsections will be favored in final evaluation.

4

Results

4.1 Propulsive ML Models

This section presents the results obtained for the models trained for predicting propulsive consumption. The first subsection contains results from evaluation on a large number of test trips from the vehicle collected dataset. The second subsection contains results from evaluation on a few trips with both navigation supplier data and the corresponding manually logged data from the vehicle.

4.1.1 Quantitative Evaluation

The results of the quantitative evaluation for all propulsive machine learning models can be seen in table 4.1. It can be seen that the difference between the models is quite significant when it comes to the segment MAE. The LR model is clearly the worst with the highest value. MLP and XGBoost are slightly better and the two RNN networks have the best values. Segment MAE is a metric showing how well the predictions points in the same direction as the real energy consumption trajectory. It is therefore believed that both RNNs are better than the other models at following the trend of the real energy consumption. Nonetheless, when looking at the normalized AE, it can be seen that the RNN models are actually performing slightly worse than the other models. The end error per kilometer is very similar for LR, MLP and XGB models. The slight differences in the values are probably just statistical noise.

Table 4.1: Performance comparison for the propulsive energy prediction models.

Propulsive ML Model	**Segment MAE [Wh/km]**	**Norm AE [Wh/km]**	**AE [Wh]**
LR	157.28	21.45	276.20
MLP	129.83	21.65	270.61
XGBoost	122.27	21.33	271.45
RNN Many-to-One	102.40	23.82	297.72
RNN Many-to-Many	101.70	23.28	298.40

While the average values of different metrics give important information, it is also important to look at their spread. Distributions can give a more complete idea about the models' performance. Figure 4.1 shows the segment MAE error distributions for energy and energy per kilometer for the RNN many-to-many model. The plots on the left are the segment MAEs plotted for all the test trips and the plots on the right are their corresponding distributions. The vertical dotted line in the histograms represents the median value for the distribution. It can be seen that both the histograms show roughly normal distribution along with some outliers. Therefore, the median value gives a better representation of the average value. A lower segment MAE value suggests that the model was trained better as lowering of segment MAE was the objective of training the models.

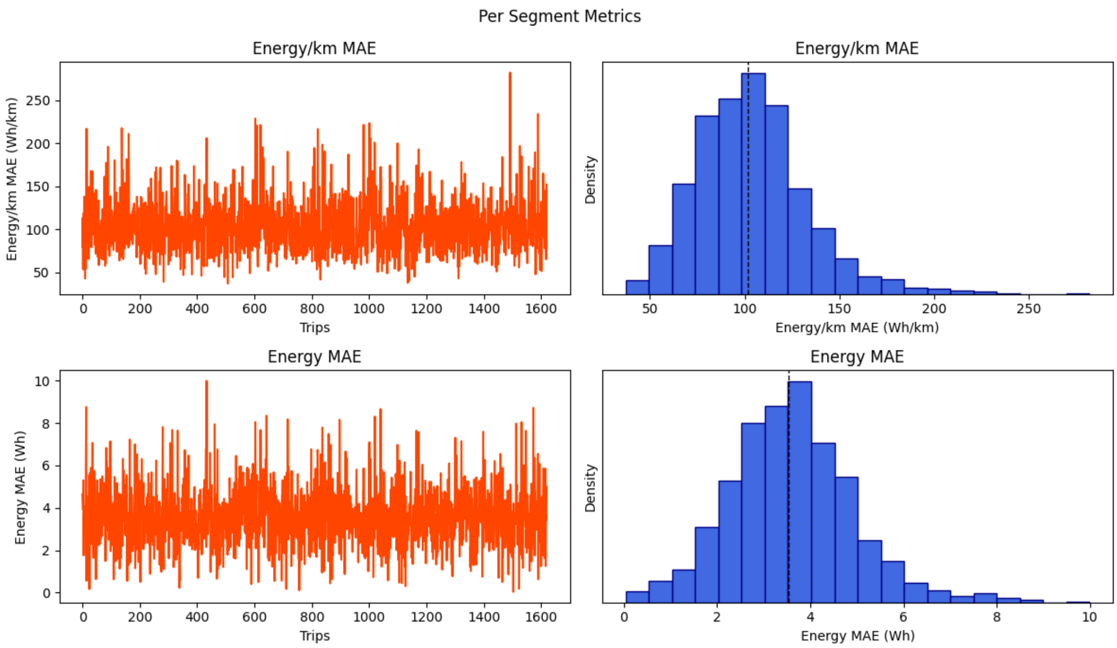


Figure 4.1: Distribution of segment MAE of energy and energy per kilometer, calculated on test trips from vehicle logged data for the RNN many-to-many model.

Figure 4.2 shows the absolute error distribution for the RNN many-to-many model. The left plots show the absolute error for each individual trip and the right plots show the distribution of the errors in a histogram. The top plots show normalized AE and the bottom plots show the AE without normalization. The vertical dotted lines in the histograms represent the median value of the distribution. It can be seen that AE distribution has outliers with large values far from the median value. The outliers seen in AE distribution are taken care of by the normalization procedure. This effect can also be seen in the left plots where the maximum AE value, at around 850th trip, gets normalized and the new value gets much closer to the median value.

Another observation to be made is that the distributions are skewed towards zero, meaning the majority of the AE and normalized AE values are closer to zero. The density quickly falls off for larger error values. Similar behavior is seen for all the other models. This is a good indication as it shows that most of the AE and normalized AE values are small, which says that all the models have a good performance.

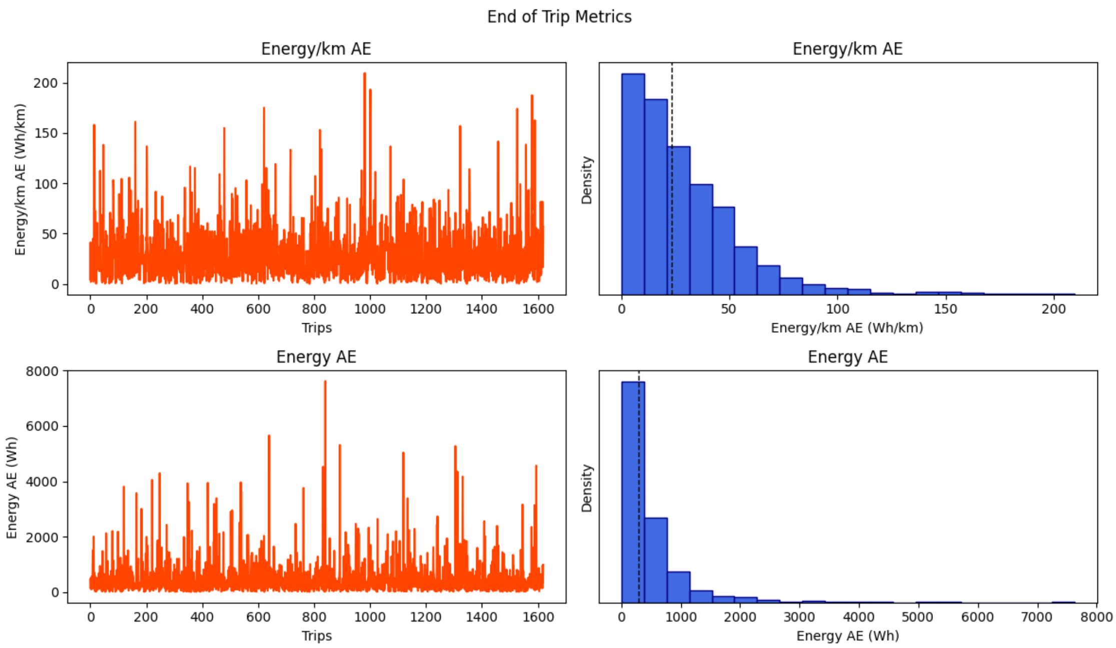
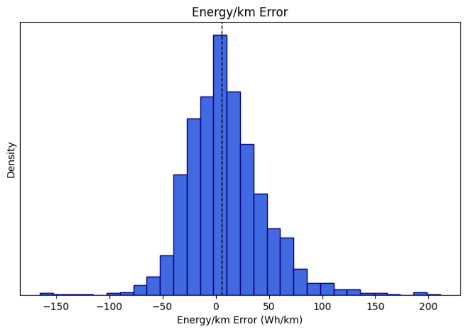


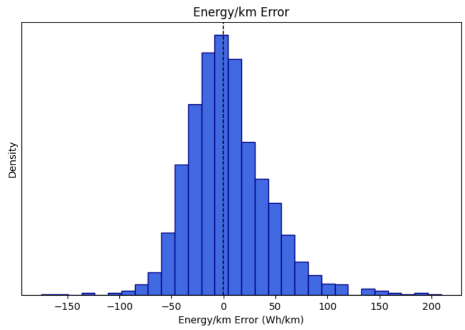
Figure 4.2: Distribution of AE and normalized AE of energy at end of trip, calculated on test trips from vehicle logged data for the RNN many-to-many model.

As seen in lower left plot in figure 4.2, there can be some trips with exceptionally high end absolute error values, seen by the high peaks in the left plots. One reason for high value can be due to the trip being very long as the small errors in each segment accumulate over time and give larger error values. Another reason can be that sometimes there can be problems with signal logging. Problems include missing signal values or wrong values of signals getting logged. This was the case in one of the trips with very high error value when actual vehicle acceleration instead of pseudo acceleration was used as one of the input signals. When the trip was inspected individually, it was found that while all other signals were normal, the acceleration signal was clamped to a constant value of 1 m/s^2 somewhere after half of the trip. A value of 1 m/s^2 sustained for longer duration is abnormal, thereby explaining a much higher predicted propulsive energy consumption. The final trained models assume that the vehicle mass is constant. This assumption holds when the number of people sitting in car is one or two, however, tends to fail when there for example four people with heavy luggage. More total vehicle weight could lead to underprediction by the models as the actual energy consumption would be higher than usual.

Usage of absolute values of different error metrics does not make distinction between positive errors and negative errors. While it is important to look at absolute errors, it is equally important to also look at the distribution of errors without taking absolute values. Distribution of trip end errors are shown in figure 4.3a for MLP model and in figure 4.3b for RNN many-to-many model. The distributions are centered roughly around zero with a majority of the points lying near zero. An interesting observation is when the median values are compared. The median value of error distribution for MLP model is 5.93 Wh/km whereas median value is -0.33 Wh/km for RNN many-to-many model. The median value for LR and XGB models is also above 5 Wh/km and very close to zero for RNN many-to-one model. The median values suggest that the RNN models have a higher accuracy than other models. The sign of median error also gives an idea if the model is underpredicting or overpredicting. The end error is the difference between actual and predicted values of energy consumption. A positive median error value signifies that the model would slightly underpredict the consumption and a negative median error means overprediction. Therefore on an average, LR, MLP and XGB models slightly underpredict whereas both the RNN models neither underpredict nor overpredict. A complete list of median error and median absolute deviation error for different models is given in table 4.2.



(a) Histogram for MLP model errors with median value of 5.93



(b) Histogram for RNN-mtm model errors with median value of -0.33

Figure 4.3: End error distributions for two different models. Dotted vertical line represent the median value.

Table 4.2: Median and MAD values for trip end error for different models for auxiliary consumption prediction models.

Propulsive ML Model	**Norm Error [Wh/km]**	**Norm Error MAD [Wh/km]**
LR	5.32	21.11
MLP	5.93	21.69
XGBoost	5.49	21.41
RNN Many-to-One	1.14	23.77
RNN Many-to-Many	-0.33	23.19

4.1.2 Qualitative Evaluation

There were a handful of test trips, which had both the navigation supplier data and logged vehicle data to make the predictions on. For propulsive energy consumption, the evaluations were performed on trip number 2 and trip number 5. Figure 4.4 shows the speed profile for trip 2 for both data sets and gives some idea about the nature of the trip. Trip 2 is almost 90 kilometers long and duration of the trip is around 75 minutes.

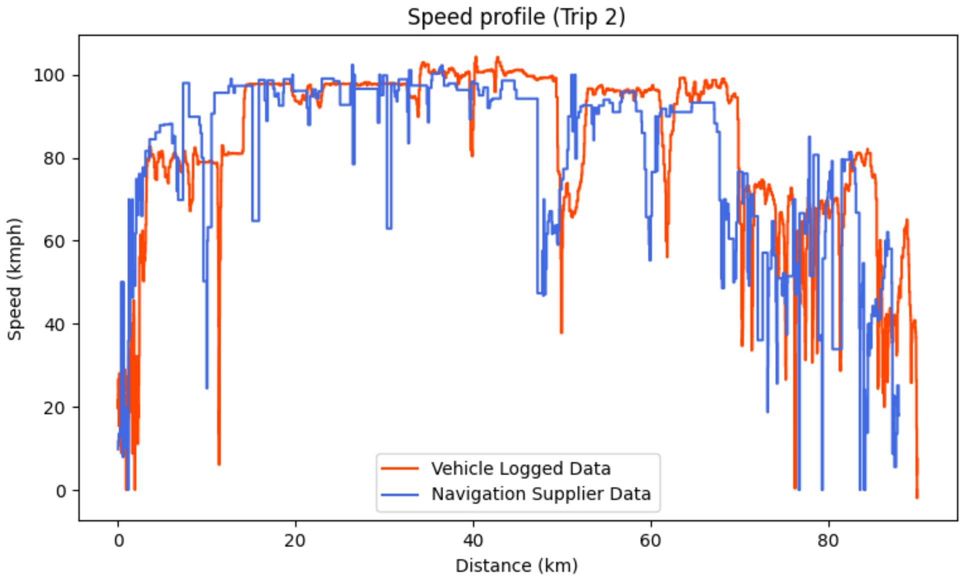


Figure 4.4: Speed profile for trip 2 from both navigation supplier data and vehicle logged data

It has already been stated that the navigation supplier data and vehicle logged data are different in some ways. Therefore, it is essential to look at the model predictions on both types of data in order to compare the performance. Figure 4.5 shows the prediction made on propulsive energy consumption by the RNN many-to-many model on both the datasets. Trip time for navigation data has been scaled to match the actual trip time. It can be seen that the RNN model makes a very good prediction on logged data, which is very close to the actual consumption. Prediction in between the trip is a bit away from actual curve partly due to scaling of time but mainly due to difference in the two datasets. The end value of prediction on the navigation supplier data is also very close to the actual consumption value. It is interesting to see that the prediction on the vehicle logged data follows every pattern of the actual consumption, i.e. all the small scale features very well. This is not the case with prediction on navigation data as it contains more averaged out values of input signals compared to more precise values of logged data.

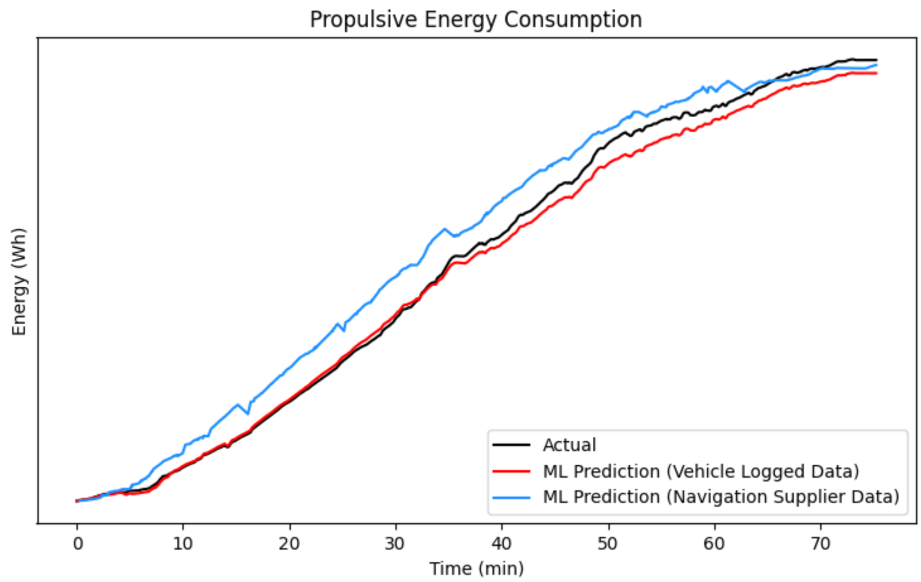


Figure 4.5: Propulsive energy consumption for trip 2. The graph shows actual consumption value and predictions on both navigation supplier data and vehicle logged data obtained by RNN many-to-many model.

For trip 2 predictions from all models on the navigation supplier data were obtained and are shown in figure 4.6. It is important to compare model performance on the navigation data as it represents the real scenario when the destination is set by the user. It is clear that RNN many-to-many model performs the best on this particular trip. LR performs the worst as it is less adaptive to a individual trip's consumption pattern. The RNNs are more adaptive than other models, seen from the predictions. Gradient boosting and MLP model seem to have very similar performance. The end error value is less than 0.1% of total battery energy, which is an extremely low value for a trip of given length. Metrics for trip 2 including the absolute end error in terms of battery percentage are listed in table 4.3.

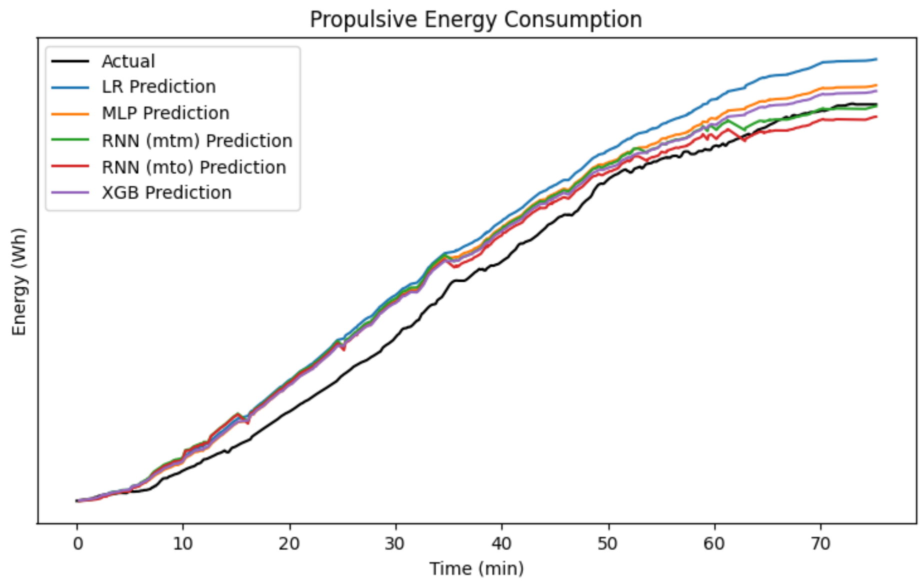


Figure 4.6: Propulsive energy consumption predictions for trip 2 obtained from all the propulsive ML models on navigation supplier data.

Table 4.3: Performance comparison for the propulsive energy prediction models on Trip 2.

Propulsive ML Model	**Norm AE [Wh/km]**	**AE [Wh]**	**AE [% battery]**
LR	21.30	1872.52	2.39
MLP	9.02	793.08	1.01
XGBoost	6.31	554.40	0.71
RNN Many-to-One	5.81	510.92	0.65
RNN Many-to-Many	0.83	73.29	0.09

The second qualitative evaluation is done on trip number 5. This trip was made during relatively warmer days in the month of April. The speed profile for the trip is shown in figure 4.7 for both the datasets.

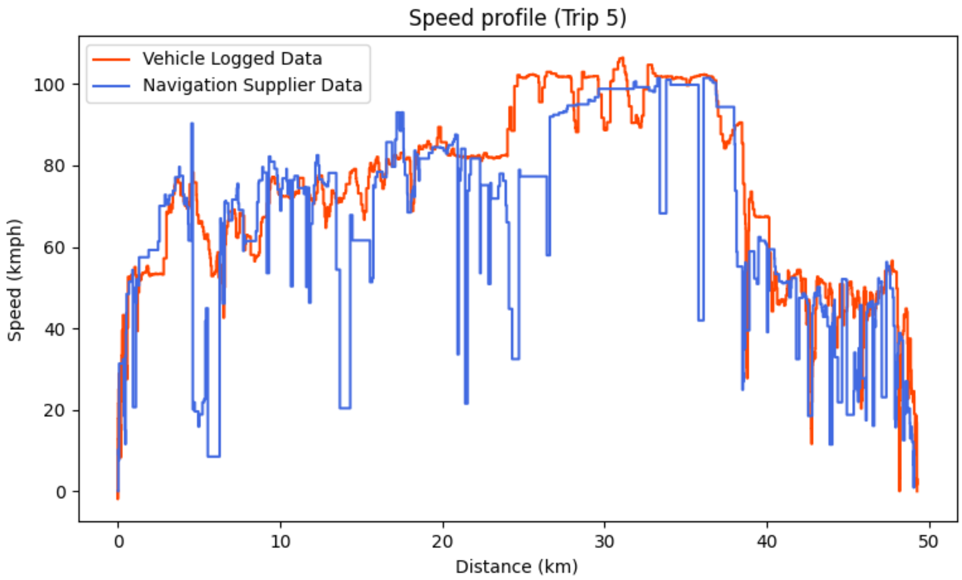


Figure 4.7: Speed profile for trip 5 from both navigation supplier data and vehicle logged data

Predictions from all the models on trip 5 are shown in figure 4.8 made on the navigation supplier data. The trip time from navigation data has been re-scaled to match the actual trip time. Trip 5 represents a trip, on which the models perform relatively bad. On the same trip, the RNN many-to-one model performs the best out of all the models followed closely by the other many-to-many RNN model. The LR model performs the worse of all as it fails to adapt fast enough with the trip. The gradient boosting and MLP models have similar performances for this trip. The difference in the predictions diverges rapidly after 30 minute mark, corresponding to slower vehicle speeds seen from figure 4.7. (WHY? check altitude also). The RNN models once again prove to be more adaptive in this situation, similar to trip 2. The absolute end error value on trip 5 for RNN many-to-many model is less than 2.1% of total battery energy, which is a very small change to notice over a trip of length 49 kilometers. Metrics for other models for trip 5 along with trip AE in terms of total battery energy are listed in the table 4.4.

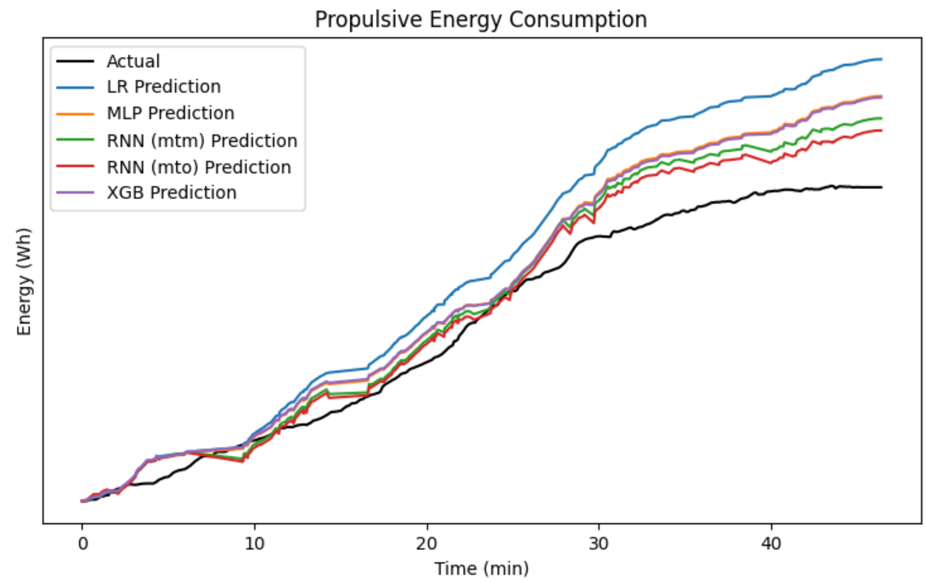


Figure 4.8: Propulsive energy consumption predictions for trip 5 obtained from all the propulsive ML models on navigation supplier data.

Table 4.4: Performance comparison for the propulsive energy prediction models on Trip 5.

Propulsive ML Model	Norm AE [Wh/km]	AE [Wh]	AE [% battery]
LR	61.64	3024.28	3.86
MLP	43.90	2153.55	2.75
XGBoost	43.26	2122.58	2.71
RNN Many-to-One	27.33	1340.84	1.71
RNN Many-to-Many	33.18	1627.79	2.08

4.2 Auxiliary ML Models

In this section, results obtained on test trips for the models trained for auxiliary consumption prediction are presented. The first subsection contains results from evaluation on a large number of trip data collected from vehicles. In the second subsection, results from evaluation on trips with both navigation supplier data and the manually logged data from a test vehicle is shown.

4.2.1 Quantitative Evaluation

The quantitative evaluation for auxiliary energy prediction is done on over one thousand individual trips. These are the same trips used for predicting propulsive energy. The evaluation metrics for auxiliary consumption prediction models are listed in the table 4.5. The error values represent median values obtained from evaluation over 1500 individual test trips from the month of December. The first three models, with suffix 'Mix', make use of the assumption on existence of transient and steady state in the auxiliary power whereas the RNN model does not. Segment MAE values from table show that the mixed models perform better than the RNN model. Among the mixed models, gradient boosting and MLP model have slightly better segment MAE values. The normalized absolute error values are also lower for mixed models than the RNN model. Gradient boosting model has the lowest absolute error and normalized absolute error of all the auxiliary energy prediction models. The normalized AE is very similar for linear regression and MLP models. Corresponding to a higher segment MAE, RNN also has a higher AE and normalized AE. The table suggests that among the auxiliary models, mixed model with gradient boosting has a marginally better performance than other models as it has both lower segment MAE and lower normalized AE.

Table 4.5: Performance comparison for the auxiliary energy prediction models.

Auxiliary ML Model	**Segment MAE [Wh/min]**	**Norm AE [Wh/min]**	**AE [Wh]**
Mix LR	11.83	7.50	148.70
Mix MLP	11.36	7.52	147.02
Mix XGBoost	11.42	7.25	139.23
RNN	12.57	9.48	178.54

The distributions of the metrics can provide a different way of looking at the models' performance. Figure 4.9 shows the segment MAE values of energy and energy per minute on test trips for mixed model with gradient boosting. Segment MAE values are plotted on the left and the corresponding histograms showing their distributions on the right. The dotted vertical line in the histograms represents the median value. Median statistic is more relevant here since there are a few outliers present in the results, which can distort the mean value. It can be seen that most of the values lie near the median value and their density falls off going away from the median value. A lower segment MAE value shows that the model was able to achieve a lower training loss since the segment MAE for energy per minute is directly related to the training objective. The segment MAE value also tells how close the predicted output power is to the actual power consumption.

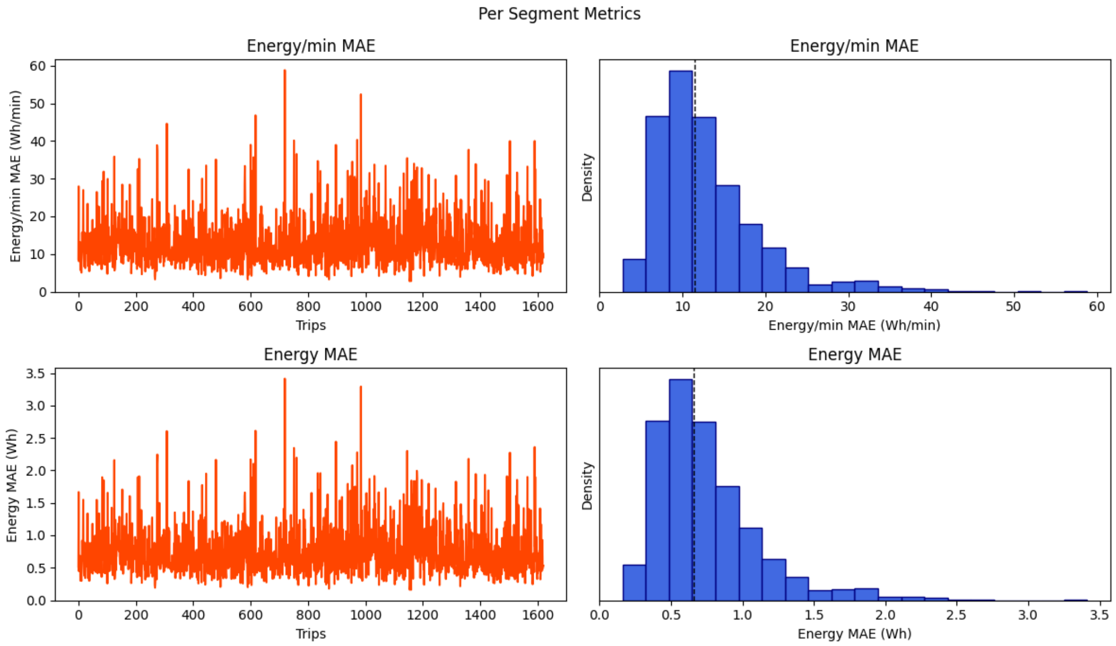


Figure 4.9: Distribution of segment MAE of energy and energy per minute, calculated from the test trips for the mixed model with gradient boosting for auxiliary consumption prediction.

Figure 4.10 shows the normalized absolute error and absolute error for the mixed model with gradient boosting. Plots on the left are the AE values for each trip and right plots are the histograms for the corresponding left plots. There a few outliers with very large values seen in the AE distribution for example near trip 400 and trip 1250. These outliers are for longer trips as longer trips can have larger errors. After normalization with respective total trip times, these outliers get close to the median normalized AE value. Another noticeable pattern is that the distributions are skewed heavily towards zero meaning most of the errors are small and there are fewer trips with larger errors.

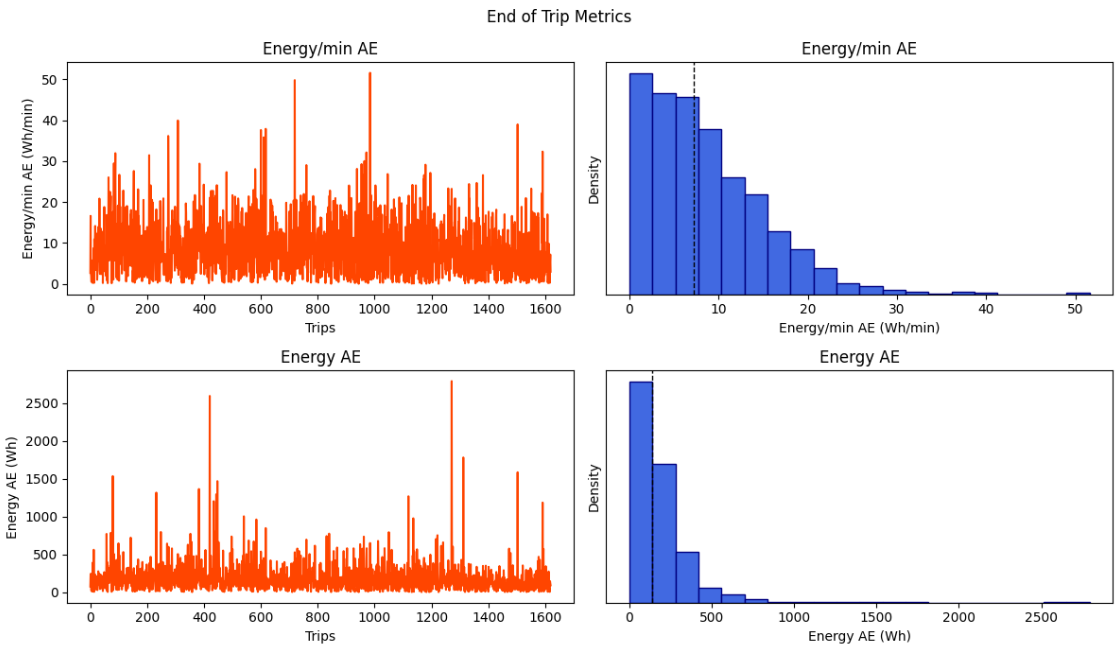
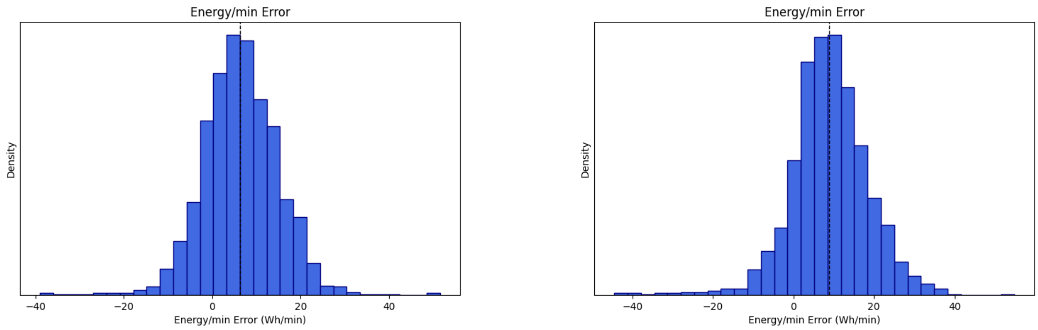


Figure 4.10: Distribution of AE and normalized AE of energy at end of trip, calculated from the test trips for the mixed model with gradient boosting.

To create a complete picture of the models' performance, the normalized end of trip errors, i.e. without taking absolute values need to be analyzed. Distribution of normalized trip end errors are shown in figure 4.11a for mixed model with gradient boosting and in figure 4.11b for the RNN model. The most important observation is that the median value for the normalized error is positive for all the models. The median value is around the order of median absolute deviation for the distributions. A positive median value implies that the auxiliary consumption prediction models always underpredict on an average. A explanation could be that the true auxiliary consumption is dependent on a large number of signals some of which are impossible to predict ahead of time. Due to a simplified view of the consumption and the fact that only the initial values can be used to make the prediction in future, the models use fewer, however, more important signals as input. These include the compartment temperature, ambient temperature, and set temperature. The simplifications and assuming dependence on a fewer signals means that some part of auxiliary consumption cannot be completely explained by the chosen input signals. The limitation comes from the nature of the problem, not from the complexity of the machine learning models. Therefore, all the models always ends up underpredciting the auxiliary consumption. Table 4.6 lists the median and MAD values for all the auxiliary energy prediction models.



(a) Histogram for Mix-XGB model errors with median value of 6.28

(b) Histogram for RNN model errors with median value of 8.80

Figure 4.11: End error distributions for two different models. Dotted vertical line represent the median value.

Table 4.6: Median and MAD values for trip end error for different auxiliary consumption prediction models.

Auxiliary ML Model	Norm Error [Wh/min]	Norm Error MAD [Wh/min]
Mix LR	6.41	5.16
Mix MLP	6.72	5.20
Mix XGBoost	6.28	5.53
RNN	8.80	5.48

4.2.2 Qualitative Evaluation

For auxiliary consumption, the qualitative evaluation was performed on trip number 1 and trip number 6. Trip 1 was taken on a colder day whereas trip 6 was taken on a slightly warmer day. Thus, evaluations on these two trips will give a picture of models' performance in different ambient conditions. Auxiliary consumption is heavily affected by the length of the trip in time. The navigation data for all the test trips used in this section were obtained at a different date and time than the date and time of the actual trip. It can give different expected trip time from the navigation supplier as the traffic conditions can be different on different days and at different times of the day. Therefore, evaluations for different models were made on the vehicle logged data since the time information in the logged data is the true travel time.

Temperature profile for trip 1 including the ambient, compartment, and set temperatures, is shown in figure 4.12. The figure gives some idea about the nature of the trip with respect to the signals affecting auxiliary consumption. The trip was 18 kilometer long and took around 26 minutes for completion.

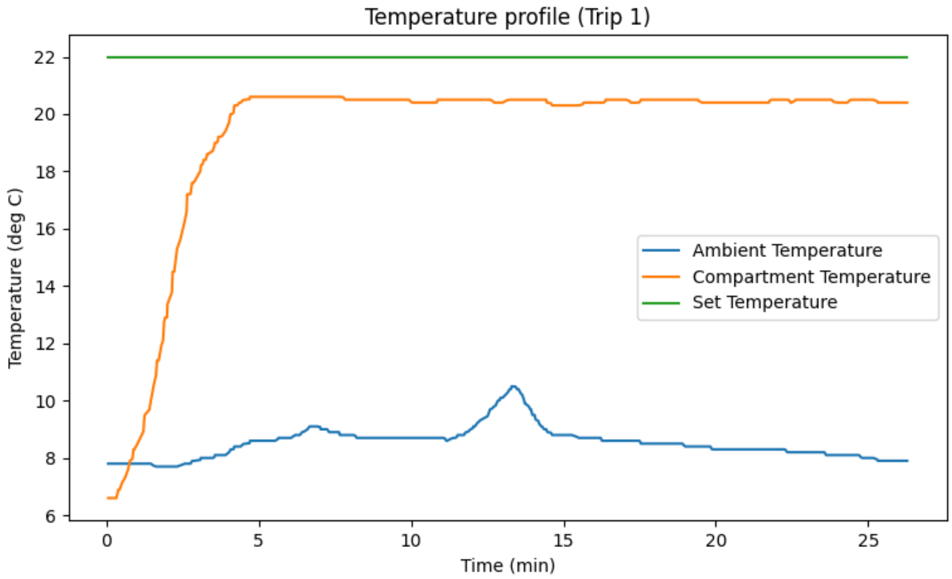


Figure 4.12: Temperature profile for trip 1 from vehicle logged data.

Figure 4.13 shows the auxiliary power predictions obtained from the mixed model with gradient boosting on trip 1. A transient phase can be clearly seen at the beginning of the trip followed by steady state phase with power levels around an average value. The second peak seen in the graph is not considered a transient and is probably due to some other factors. It can be seen that the model is able to predict the length of transient and power levels for both transient and steady state phases very close to the average values for the trip. The prediction uses segment time information from vehicle logged data and navigation supplier separately, however, uses the initial value of the input signals from the manually logged vehicle data. The slight difference in prediction seen at the end of the transient is simply due to the different arrangement of segments in the datasets. Time axis is not scaled for the navigation supplier data to show that for auxiliary consumption the trip time estimated by the navigation supplier is very important in determining the energy that will be consumed.

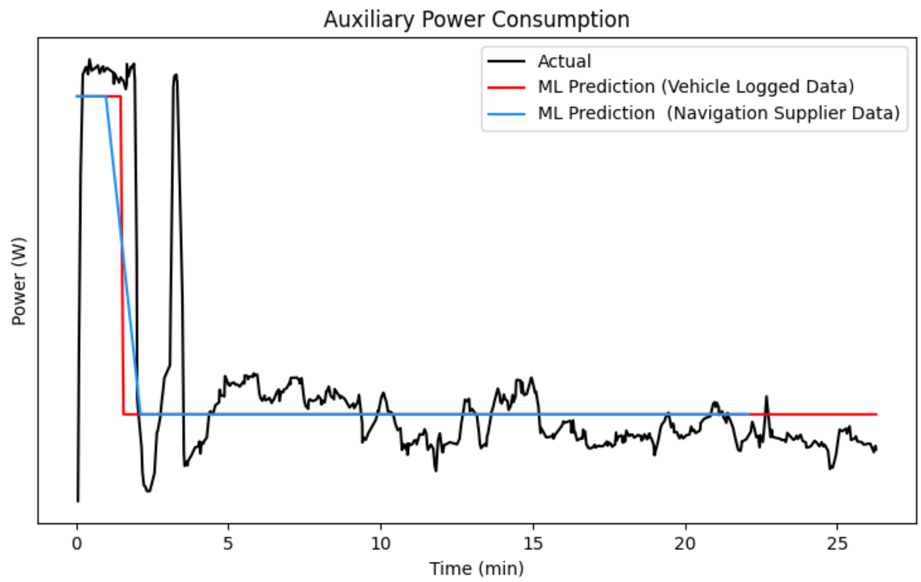


Figure 4.13: Auxiliary power predictions for trip 1 obtained from mixed model with gradient boosting. The model is able to predict the duration and power level of transient phase reasonably well.

The output power shown in figure 4.13 when integrated with time, gives the energy consumption for the trip, seen in figure 4.14. The initial region with higher slope represents the transient phase followed by a smaller slope, which represents the steady state phase. As stated earlier, the major difference between the predictions on the two datasets is mainly due to the difference in estimated and actual total trip time. For this trip, the prediction is following the actual consumption closely and happens to almost coincide towards the end of the trip.

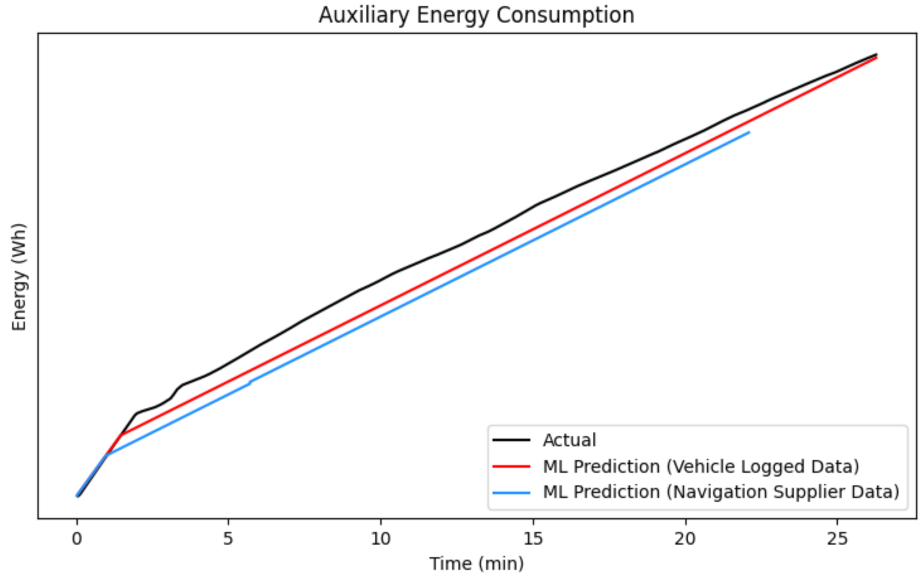


Figure 4.14: Auxiliary energy predictions for trip 1 obtained from mixed model with gradient boosting on both navigation supplier and vehicle logged data.

To compare predictions from all the models, vehicle logged data was used. The predictions are shown in figure 4.15. It can be seen that the predictions from the mixed models are very close to each other. The RNN model is able to model the transient phase well, nonetheless, is underpredicting the steady state power level. The transition between transient and steady state is precisely defined for the mixed model, whereas RNN makes a gradual transition to the steady state. The energy consumption for the models is shown in figure 4.16. Mixed model with linear regression and gradient boosting perform the best followed by the MLP model. RNN performs the best during the transient phase as seen in the first three minutes of the trip, however, due to underprediction of steady state power, diverges from the actual consumption over time.

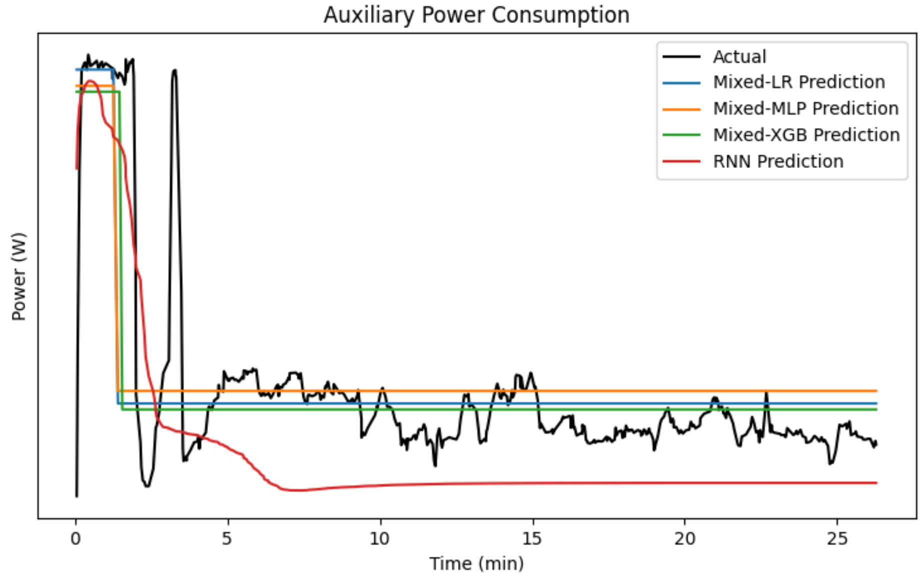


Figure 4.15: Auxiliary power predictions for trip 1 obtained from all the models on vehicle logged data. The trip was taken on a colder day and hence the transient phase is pronounced.

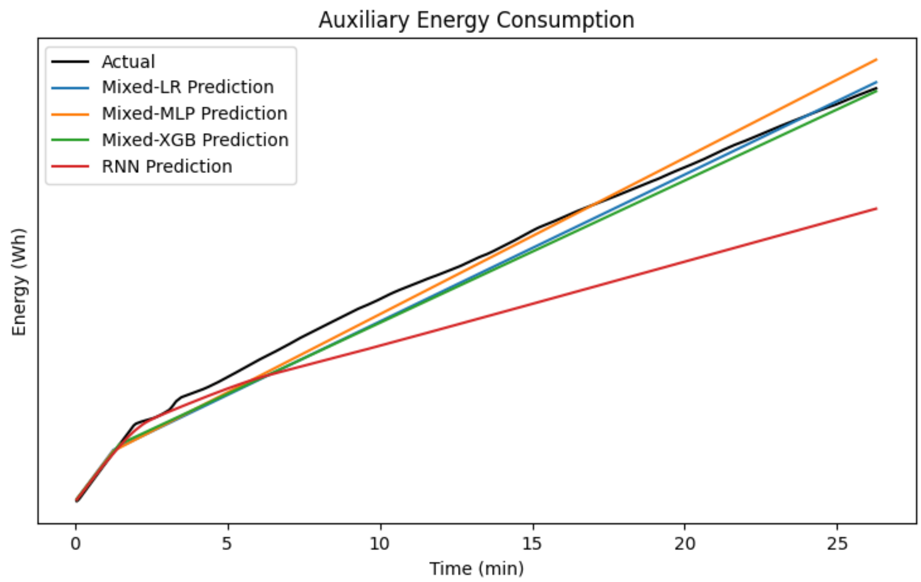


Figure 4.16: Auxiliary energy consumption predictions for trip 1 obtained from all the models on vehicle logged data. Predictions from different models except RNN are very close to each other and the actual consumption.

The numerical values for the metrics obtained for trip 1 along with the absolute error at end of trip expressed in percentage of total battery energy are listed in the table 4.7. The table shows that the lowest error obtained is a mere 0.01% of total battery energy. The largest error, obtained from RNN model, is less than 0.5% of total battery energy. The higher number is due to the RNN model underpredicting the steady state power by a large amount.

Table 4.7: Performance comparison for the auxiliary energy prediction models on Trip 1.

Auxiliary ML Model	**Norm AE [Wh/min]**	**AE [Wh]**	**AE [% battery]**
Mix LR	0.73	19.23	0.02
Mix MLP	3.48	91.57	0.12
Mix XGBoost	0.36	9.38	0.01
RNN	14.66	385.19	0.49

All the input temperature signals for trip 6, including the ambient, compartment, and set temperatures, are shown in figure 4.17. The figure gives some idea about the nature of the trip with respect to the signals affecting the auxiliary consumption. Trip 6 was around 46 kilometers long and took about 53 minutes to finish. The trip was made shortly after completion of another trip, thus the compartment was preheated, seen in the initial value of compartment temperature in the same figure.

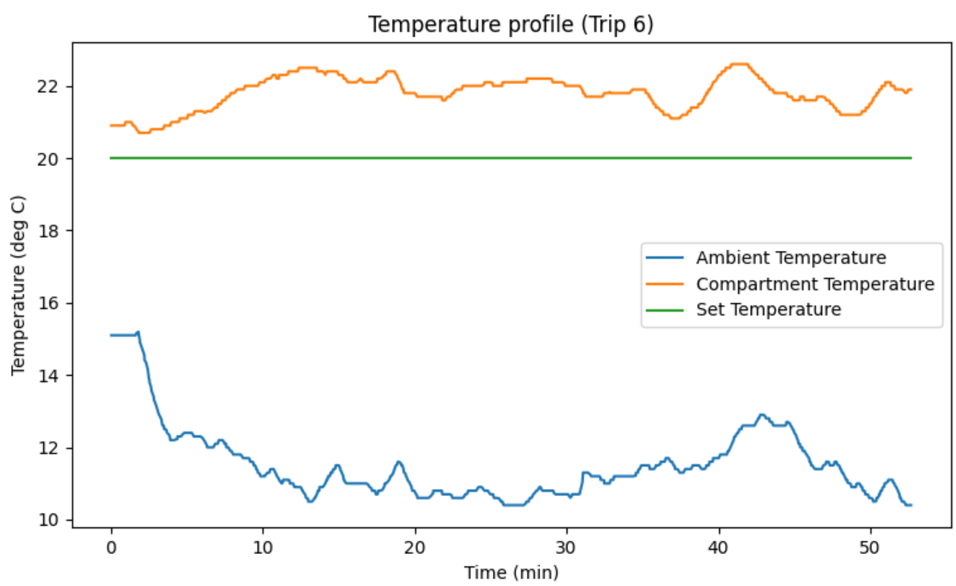


Figure 4.17: Temperature profile for trip 6 from vehicle logged data.

The auxiliary power predictions for trip 6 are shown in figure 4.18. The transient phase is almost non existent and thus the length of transient phase is also negligible in the predictions. Mixed models with MLP and gradient boosting have the best predictions as the steady state value represents the average consumed power. Upon integrating the auxiliary power predictions, the energy predictions are obtained, as shown in figure 4.19. In the plot, it is clear that the mixed models with MLP and gradient boosting follow the actual consumption very closely. The linear regression and RNN models underpredict the energy consumption and diverge from the true consumption with time.

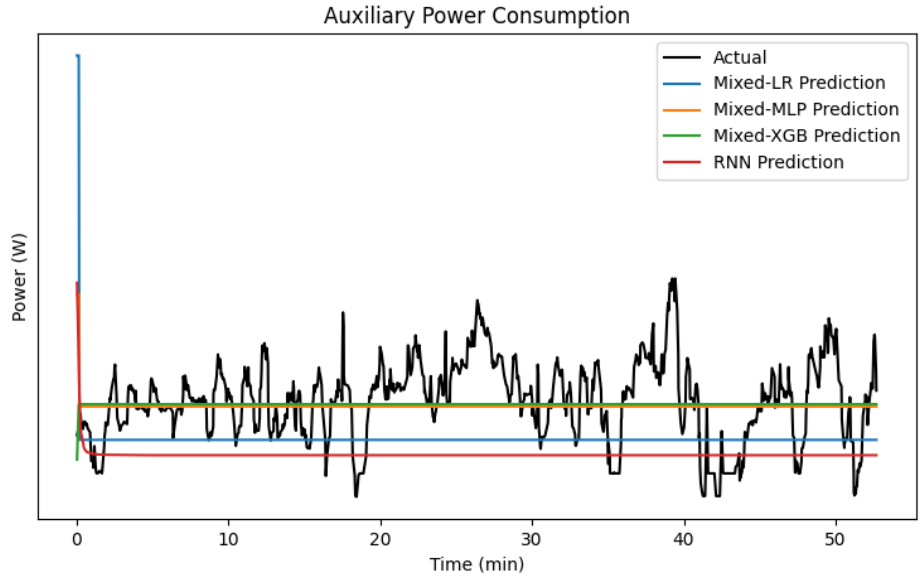


Figure 4.18: Auxiliary power predictions for trip 6 from all the auxiliary ML models. Due to slightly warmer day and preheated compartment, the transient phase is negligible.

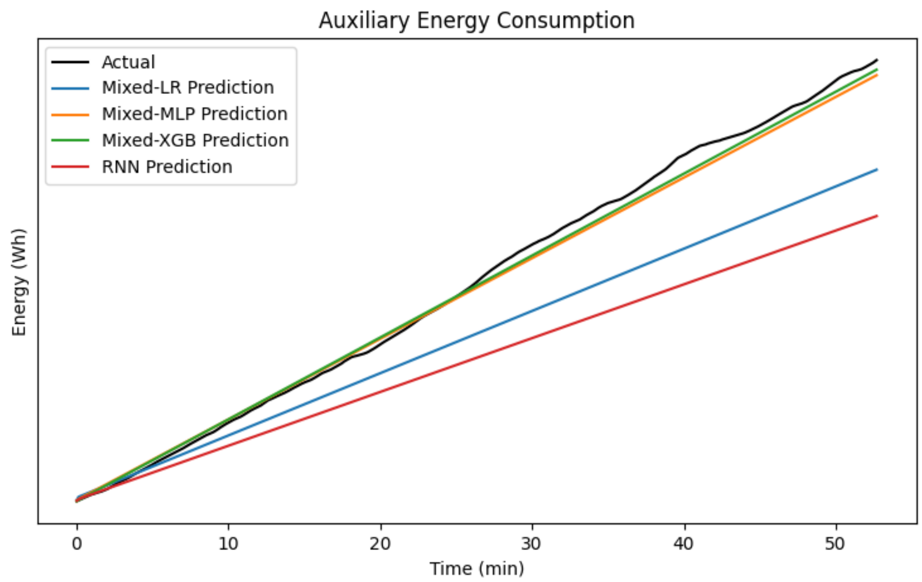


Figure 4.19: Auxiliary energy predictions for trip 6 from all the auxiliary ML models. The predictions have almost a constant slope throughout the trip as transient phase is negligible.

The numerical values for the metrics for trip 6 along with the absolute error in terms of battery percentage are summarised in table 4.8. The lowest absolute error at trip end is about 0.03% of total battery energy. The largest absolute error is 0.53% of total batter energy, obtained from the RNN model. As the transient phase is not present in trip 6, the RNN performs the worst as it underpredicts the steady state power. LR follows a similar trend of underprediction. MLP and XGB models were able to predict the steady state power very close the the true average power consumed in that trip.

Table 4.8: Performance comparison for the auxiliary energy prediction models on Trip 6.

Auxiliary ML Model	**Norm AE [Wh/min]**	**AE [Wh]**	**AE [% battery]**
Mix LR	5.54	291.85	0.37
Mix MLP	0.76	40.17	0.05
Mix XGBoost	0.48	25.44	0.03
RNN	7.89	415.40	0.53

As mentioned in the section 4.2.1, it was apparent from the error distributions that all the auxiliary prediction models were underpredicting on an average. The behavior is verified for both the trips presented in the qualitative analysis for auxiliary consumption prediction. In trip 1, the RNN model underpredicted and in trip 6, mixed model with LR and RNN model underpredicted.

4.3 Prediction Interval

All the machine learning models, for both propulsive and auxiliary consumption prediction, give an estimate of predicted consumption based on the trip data. The estimate for total energy consumption for the whole trip is a single numerical value. The models are deterministic and for a given set of input signals always produce the same output. Sometimes it is important to get a range for the estimated quantities, in this case estimated total energy consumption. A range is important because driving styles can differ from person to person driving the same vehicle under different circumstances. The predictions in real scenario are made on the navigation supplier data which contains expected speed values for different segments in a trip. The vehicle might actually be driven at slightly higher or lower speeds than the expected speed from navigation data, depending on the driver behavior, actual traffic conditions and other unforeseen factors. A range gives a lower and upper bound for the energy consumption allowing for charging stops to be placed earlier or later along a trip. It also allows to determine the amount to which a BEV needs to be charged for both higher and lower than usual consumption. There are two ways in which a range can be obtained from the models, discussed in the sections ahead.

4.3.1 Using Input Range

The input signals used for predicting propulsive consumption are vehicle speed, pseudo acceleration and road inclination. Out of the three, road inclination does not change along a fixed route and is considered a constant for a given segment in a trip. Vehicle speed is variant and can be higher or lower than the expected speed received from the navigation supplier data. Thus, a range for vehicle speed can be used as an input to the propulsive models. Pseudo acceleration is calculated as shown in equation 3.2 and increasing or decreasing speed would change the pseudo acceleration. The equation can be rewritten as:

$$a_i = \frac{v_i(v_i - v_{i-1})}{d_i} \tag{4.1}$$

where d_i is the length of segment i. If the speed is increased by a factor of 1.1 or 10% in every segment, then the pseudo acceleration increases by square of the factor i.e. 1.21 or 21%. Thus feeding lower and higher speeds and accelerations would give a corresponding lower and upper bound to energy consumption. Without any scaling of speed would give the mean expected consumption.

Due to scaling of speed, the time spent in each segment would also decrease and hence the total trip time. This will affect the auxiliary consumption as it depend on the trip duration. For a 10% increase in speed, the total trip time decreases by roughly 9.1%, reducing the auxiliary consumption as well. Using ranges for both propulsive and auxiliary consumption, the combined range can be obtained. Figure 4.20 shows the propulsive energy prediction interval with upper bound corresponding to 10% higher speed and lower bound corresponding to 10% lower speed. Similarly, figure 4.21 shows the auxiliary energy prediction interval with upper bound corresponding to 10% lower speed and lower bound corresponding to 10% higher speed. For the figure, the time scales for upper and lower bound are scaled to the original time scale to show the difference in energy consumption.

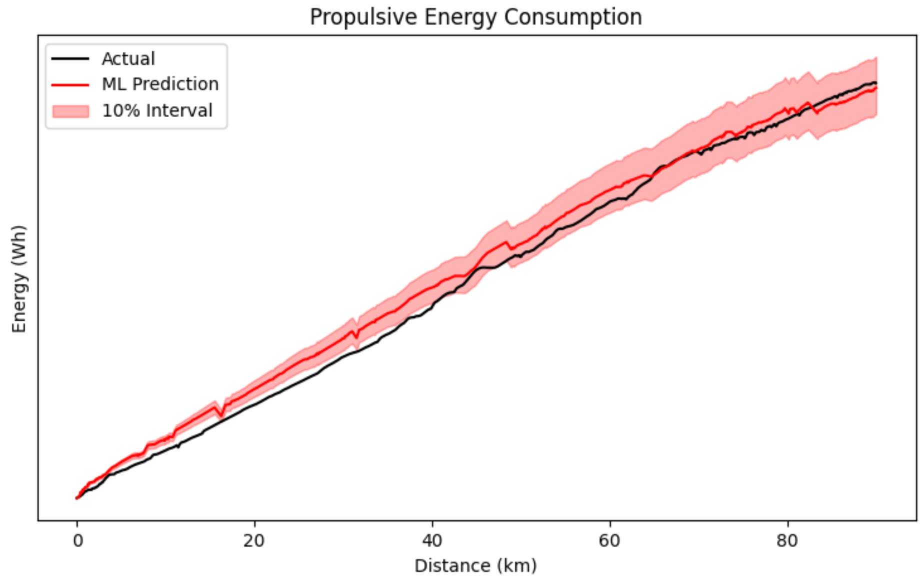


Figure 4.20: Propulsive energy consumption prediction interval for trip 2 obtained from RNN many-to-many model on navigation supplier data for 10% speed variation.

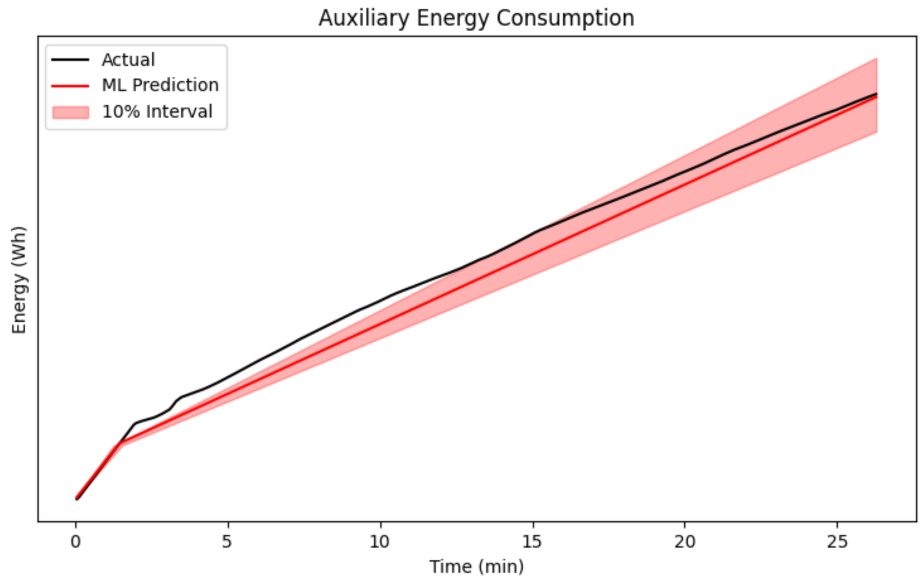


Figure 4.21: Auxiliary energy consumption prediction interval for trip 1 obtained from mixed-XGB model on vehicle logged data for 10% speed variation.

4.3.2 Using Statistical Range

Different error metrics were presented in the quantitative analysis sections for predictions on both propulsive and auxiliary consumption. The median value of normalized AE when multiplied with the expected total trip distance and time gives the value for expected absolute error for propulsive and auxiliary consumption respectively. The interval obtained in this case would be with respect to the absolute error. To obtain an interval with respect to the true error, shown in figure 4.3 and figure 4.11, the mean and median absolute deviations can be used. The interval for expected error at the end of trip is given by $\mu \pm \mathrm{MAD}$, where μ is the median error and $textMAD$ is the median absolute deviation in error. Then the interval for the energy prediction can be obtained by:

$$[\, E_{pred} + (\mu - \mathrm{MAD})L, E_{pred} + (\mu + \mathrm{MAD})L \,] \tag{4.2}$$

where E_{pred} is either propulsive or auxiliary energy prediction, μ and MAD are corresponding median and MAD values, and L is the total distance for propulsive and total time for auxiliary consumption. The median and MAD values for different propulsive consumption models are listed in table 4.2 and for auxiliary consumption models in table 4.6. Interval for a combined energy consumption including propulsive and auxiliary parts is obtained by adding the upper and lower interval limits for each part.

5

Conclusions and Future Work

5.1 Conclusions

- The difference between the propulsive consumption prediction models when evaluating the results is quite small, leading to a conclusion that a more advanced model does not necessarily gives a better result for the problem at hand. The reason probably being that there is to much uncertainty in the input parameters, which are estimated by the navigation supplier before the trip. Road inclination is hopefully accurate and constant, however, speed and acceleration can both change with different traffic conditions. It is also very uncommon that the driver drives exactly as the navigation supplier believes, changing the final consumption significantly.
- For the auxiliary models, it can be concluded that the models with the mix-prefix, based on the assumption with a transient phase and a steady-state phase, perform better than the RNN. The mixed models do a very good job at predicting the energy consumption and seem to benefit by the simplified assumption on how the power usually changes over the trip. Nevertheless, it is important to remember that all the data was in colder weather and it is difficult to say how the models would work in warmer weather.
- One factor that both the propulsive and auxiliary prediction have in common is how there are some inputs that cannot be predicted perfectly before the trip has been run. The speed and acceleration for the propulsive prediction and, connected to the speed, the time of the trip for the auxiliary prediction. All drivers will differ more or less from the navigation supplier estimation. It can be concluded that to give a better estimation of the energy consumption for a trip, a span of potential energy consumption should be set, rather than an absolute value. The lower and upper bounds of the span can be set in a way that gives the driver for example 95% certainty that the actual consumption falls within the span. A span will never be 100% perfect, however, it can be an easy way to understand how the choice of driving style impacts the resulting energy consumption.

5.2 Future Work

One way to continue the work done in this thesis is to adapt the predictions of a trip while the trip is being done.

To adapt the propulsive energy consumption prediction during a trip can be difficult. However, there are two factors that would be interesting to use in an adaptive scenario. The first one being driver behaviour, which is a topic that has been touched upon earlier in the thesis. With data from the current trip a classification of the driver's behaviour can in theory be made, for example by looking at the standard deviation of acceleration or jerk. The classification can then be used to change the energy consumption prediction for the trip. The other factor that would be interesting to investigate is how the energy consumption earlier in the trip compares to the initial prediction. The difference might tell if the model is under-predicting or over-predicting and then adapt the remaining prediction accordingly.

The auxiliary prediction is easier to adapt if the power consumption follows the assumption made in the report with a transient state and a steady-state. After a part of the trip is done and the consumption is in steady-state, there is available data of the steady-state power consumption earlier in the trip. The average of the steady-state power data is, in theory, a perfect estimate for the remaining steady-state power, which can be utilized. One idea could be to do a running average adaptation, where both a new steady-state prediction, based on the average of the current trip, and the initial prediction can be combined. The running average reduces the risk of including noise into the prediction.

Another way of adapting over a longer duration of time and more specific to an individual vehicle is with online learning. The ML models used in this thesis were trained over trips made with different vehicles of the same type. The more generic ML models trained over the entire dataset can be adapted for an individual vehicle by training on newer trips the vehicle makes. With time, more trips will accumulate and the models can be updated using the newer trips. Update can be done for one trip at a time or can be done once a certain number of trips are accumulated in a batch. Adapting in this way allows the ML models to adapt to specific long term driving behaviors of a driver and consumption patterns of a given vehicle. Another benefit is that long term adaptation can adapt for seasonal changes, for example the consumption would be different for the same vehicle during summer and winter. Thus, both trip level adaptation and long term adaptation would be an essential part to improve the models' predictive power specific to individual vehicles.

Bibliography

[1] Bruce, G. What's stopping Americans from buying electric cars? YouGovAmerica [Internet]. Oct 23, 2020 [June 1, 2021]; Available from: https://today.yougov.com/topics/consumer/articles-reports/2020/10/23/whats-stopping-americans-buying-electric-cars

[2] Office of Energy Efficiency & Renewable Energy. Reducing Pollution with Electric Vehicles [Internet]. Washington DC: Office of Energy Efficiency & Renewable Energy; - [-;1 June, 2021]. Available from: https://www.energy.gov/eere/electricvehicles/reducing-pollution-electric-vehicles

[3] Chen, Y., Wu, G., Sun, R., Dubey, A., Laszka, A., Pugliese, P. A Review and Outlook of Energy Consumption Estimation Models for Electric Vehicles. Cornell University. 2020;-:-

[4] Galvin, R. Energy consumption effects of speed and acceleration in electric vehicles: Laboratory case studies and implications for drivers and policymakers. Transportation Research Part D: Transport and Environment. 2017;53(-):234-248.

[5] Liu, K., Yamamoto, T., Morikawa, T. Impact of Road Gradient on Energy Consumption of Electric Vehicles. Transportation Research Part D: Transport and Environment. 2017;54(-):74-81.

[6] Zhao, L., Yao, W., Wand, Y., Hu, J. Machine Learning-Based Method for Remaining Range Prediction of Electric Vehicles. IEEE Access. 2020;8(-):212423-212441.

[7] Wang, J. Battery electric vehicle energy consumption modelling, testing and prediction: a practical case study [PhD Thesis]. Eindhoven, the Netherlands: Eindhoven University of Technology; 2016 [23 March, 2021]. Available from: https://pure.tue.nl/ws/files/41303833/20161031_Wang.pdf

[8] Montgomery, D., Runger, G. Applied Statistics and Probability for Engineers. 3rd Edition. United States of America: John Wiley & Sons; 2003

[9] Bhandari, A. What is Multicollinearity? Here's Everything You Need to Know. 2020, March 20 [April 10, 2021]. In: Analytics Vidhya [Internet]. Unknown Place of Publication: Analytics Vidhya; March 20, 2020. Available from: https://www.analyticsvidhya.com/blog/2020/03/what-is-multicollinearity/

[10] Dabbura, I. Gradient Descent Algorithm and Its Variants. 2017, Dec 21 [March 2, 2021]. In: towards data science [Internet]. Unknown Place of Publication: towards data science; Dec 21, 2017. Available from: https://towardsdatascience.com/gradient-descent-algorithm-and-its-variants-10f652806a3

[11] Mohanty A. Multi layer Perceptron (MLP) Models on Real World Banking Data. 2019, May 15 [2021, April 13]. In: Medium [Internet]. Unknown place of publication. Available from: https://becominghuman.ai/multi-layer-perceptron-mlp-models-on-real-world-banking-data-f6dd3d7e998f

[12] Fortuner B. Activation Functions. 2018 Aug 18 [2021, April 18]. In Github [Internet]. Unknown place of publication. Available from: https://github.com/bfortuner/ml-glossary/blob/master/docs/activation_functions.rst

[13] Ian Goodfellow, Yoshua Bengio and Aaron Courville. Deep Learning, MIT Press, 2016

[14] Yan S. Understanding LSTM and its diagrams. 2016, March 13 [2021, April 17]. In Medium [Internet]. Unknown place of publication. Available from: https://blog.mlreview.com/understanding-lstm-and-its-diagrams-37e2f46f1714

[15] Ayyadevara K. Neural Networks with Keras Cookbook. In O'reilly [Internet]. Unknown place of publication. Available from: https://www.oreilly.com/library/view/neural-networks-with/9781789346640/178fa4d1-3349-4033-b8d3-d72ee249ad43.xhtml

[16] Brownlee J. Ensemble Learning Methods for Deep Learning Neural Networks. 2018 Dec 19 [2021, April 21]. Unknown place of publication. In Machine Learning Mastery [Internet]. Available from: https://machinelearningmastery.com/ensemble-methods-for-deep-learning-neural-networks/

[17] Mei-Hung C. The use of facial micro-expression state and tree-forest model for predicting conceptual-conflict based conceptual change. 2016, January [2021, April 19]. In ResearchGate [Internet]. Unknown place of publication. Available from: https://www.researchgate.net/figure/Basic-structure-of-a-decision-tree-All-decision-trees-are-built-through-recursion_fig3_295860754

[18] Boehmke B, Greenwell B. Hands-On Machine Learning with R. 2020 Feb 1 [2021, April 21]. In BradleyBoehmke [Internet]. Unknown place of publication. Available from: https://bradleyboehmke.github.io/HOML/gbm.html

[19] Vasiloudis T. Block-distributed Gradient Boosted Trees. 2019, Aug 26 [2021, May 10]. In Tvas [Internet]. Unknown place of publication. Available from: http://tvas.me/articles/2019/08/26/Block-Distributed-Gradient-Boosted-Trees.html

[20] Readthedocs. XGBoost Documentation. 2020 [2021, May 7]. In readthedocs [Internet]. Unknown place of publication. Available from:

https://xgboost.readthedocs.io/en/latest/

[21] Readthedocs. Introduction to Boosted Trees. 2020 [2021, May 7]. In readthedocs [Internet]. Unknown place of publication. Available from: https://xgboost.readthedocs.io/en/latest/tutorials/model.html

DEPARTMENT OF ELECTRICAL ENGINEERING
CHALMERS UNIVERSITY OF TECHNOLOGY
Gothenburg, Sweden
www.chalmers.se

AVANCEZ
1829

CHALMERS
UNIVERSITY OF TECHNOLOGY