

Article

On-Road Experimental Campaign for Machine Learning Based State of Health Estimation of High-Voltage Batteries in Electric Vehicles

Edoardo Lelli ¹, Alessia Musa ^{2,*} , Emilio Batista ¹, Daniela Anna Misul ^{2,*}  and Giovanni Belingardi ³ ¹ Hyundai Motor Europe Technical Center GmbH, Hyundai-Platz, 65428 Ruesselsheim, Germany² Department of Energy (DENEG), Politecnico di Torino, 10129 Torino, Italy³ Department of Mechanical and Aerospace Engineering (DIMEAS), Politecnico di Torino, 10129 Torino, Italy

* Correspondence: alessia.musa@polito.it (A.M.); daniela.misul@polito.it (D.A.M.)

Abstract: The present study investigates the use of machine learning algorithms to estimate the state of health (SOH) of high-voltage batteries in electric vehicles. The analysis is based on open-circuit voltage (OCV) measurements from 12 vehicles with different mileage conditions and focuses on establishing a correlation between the OCV values, the energy stored in the battery, and the battery SOH. The experimental campaign was conducted at the Hyundai Motor Europe Technical Center GmbH (Germany), and the data collection process took advantage of the ETAS Integrated Calibration and Application Tool (INCA) and the ETAS Measure Data Analyzer (MDA) software. Six machine learning algorithms are evaluated and compared, namely linear regression, k-nearest neighbors, support vector machine, random forest, classification and regression tree, and neural network. Among the evaluated algorithms, random forest (RF) exhibits the best performance in predicting the state of health of high-voltage batteries, both for the OCV and the capacity (C) estimation. Specifically, if compared to the worst algorithm (i.e., linear regression), RF achieves a remarkable improvement with a reduction of 96% and 97% in the mean absolute error for the OCV and the C estimation, respectively. Furthermore, the comparison highlighted the main differences in the performance, complexity, interpretability, and specific features of the six algorithms. The findings of the present study will contribute to the development of efficient maintenance strategies, thus reducing the risk of unexpected battery failures.

Keywords: high-voltage batteries; state-of-health (SOH); machine learning (ML) algorithms



Citation: Lelli, E.; Musa, A.; Batista, E.; Misul, D.A.; Belingardi, G.

On-Road Experimental Campaign for Machine Learning Based State of Health Estimation of High-Voltage Batteries in Electric Vehicles. *Energies* **2023**, *16*, 4639. <https://doi.org/10.3390/en16124639>

Academic Editor: Satoru Okamoto

Received: 14 April 2023

Revised: 24 May 2023

Accepted: 9 June 2023

Published: 11 June 2023



Copyright: © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

The automotive industry is heavily investing in electrified vehicles as a crucial solution to achieve carbon-neutral mobility and reduce pollution in the transportation sector. However, there are still challenges and limitations that need to be addressed [1,2]. Several countries and regions have announced plans to phase out the sale of petrol- and diesel-powered vehicles in the upcoming years. As a result, the hybrid vehicle market is expected to grow significantly over the next five years, with an estimated market size of USD 882.88 billion by 2026 and a Compound Annual Growth Rate (CAGR) of 20.6%. In particular, the Asia-Pacific region is expected to dominate, with major players such as Toyota, Honda, Nissan, Kia, BYD, and Hyundai leading the way [3]. The plug-in hybrid vehicle segment is poised to experience the highest growth rate during the forecast period, driven by the increasing popularity of electrified solutions and the growing availability of charging infrastructure. Similarly, the electric vehicle market is expected to witness significant growth, with an expected market size of USD 1393.33 billion by 2027 and a CAGR of 19.19% [4]. Even major European automakers such as Volkswagen, Mercedes Group, BMW, Volvo, and Renault are expected to increase their market share, reducing the gap with their Asian counterparts. This growth is driven by the increasing demand for fuel-efficient

and low-emission vehicles, government initiatives, and technological advancements that have improved the performance and range of electrified solutions. This scenario is driving research and development efforts towards improvements in battery technology and the performance of battery management systems. Effective battery management systems are critical for maximizing the performance and lifespan of batteries, which is essential to meet the demands of the market for electric and hybrid vehicles.

Related Works

Battery management systems in electrified vehicles are responsible for several functions, including monitoring the state of charge (SOC) and state of health (SOH), managing thermal conditions to ensure optimal performance, protecting against overcharge and overdischarge, and balancing the cells within the battery pack [5–13]. Such functions are designed to improve the battery performance, to extend its useful life, and to reduce maintenance and replacement costs, ultimately making electrified vehicles more accessible and cost-effective for consumers.

In high-voltage (HV) batteries, there are two distinct methods for assessing for the battery SOH: estimation and prediction. SOH estimation exploits measurements of the battery current and voltage, as well as a variety of other diagnostic tests, to determine the battery current SOH. Such an evaluation is useful to determine whether the battery is functioning as expected and whether it requires maintenance or replacement. In contrast, SOH prediction utilizes historical data and predictive models to estimate the battery's future performance and expected lifespan. This is typically accomplished by analyzing the battery degradation patterns over time and extrapolating these trends into the future. The purpose of SOH prediction is to provide a prediction of the remaining battery life and to aid in maintenance and replacement planning [14–17]. Both estimation and prediction are crucial for managing high-voltage (HV) batteries and ensuring their optimal performance and durability. SOH evaluation can be accomplished through either model-based or data-driven methods. Model-based methods rely on mathematical models that describe the electrochemical processes in the battery, whereas data-driven methods use statistical techniques and artificial intelligence (AI) algorithms to estimate the SOH. Although data-driven methods offer adaptability, the accuracy of the estimation is dependent on the feature selection and algorithms. Both methods have their own strengths and limitations, and the choice depends on the specific requirements of the selected application.

Examples of model-based approaches include equivalent circuit models [18–20], electrochemical [14,21–23] and grey box models [24]. Equivalent circuit models represent the battery as a circuit with a series of resistances and capacitors, and, by monitoring the voltage and current of the battery over time, they can evaluate the SOH of the battery [18,19]. Electrochemical models simulate the chemical and physical processes that occur inside the battery, including the transport of ions and electrons, chemical reactions, and heat generation. These models can provide very accurate predictions of the battery behavior, but they require many computational resources and are often too complex to be used in on-board real-time applications [14,21–23]. Grey-box models such as the extended kalman filter combine empirical data with mathematical models to predict the battery behavior. These models can be simpler than electrochemical ones, but still provide accurate predictions of the battery behavior. They use a combination of battery performance data and mathematical models to estimate the internal parameters of the battery, such as the SOC and the internal resistance [24]. One of the main drawbacks of model-based algorithms is their complexity, given that the development of accurate models to describe the behavior of a battery system can be a complex and time-consuming process. Additionally, model-based algorithms are sensitive to model parameters, and inaccurate or ill-defined parameters can lead to inaccurate results. Moreover, model-based algorithms may not be easily adaptable to new or different battery systems, as they are typically designed to work with a specific battery chemistry and defined operating conditions.

Examples of data-driven methods include a support vector machine (SVM) [25–29], random forest (RF) [30,31], artificial neural networks (ANNs) [32–37], recurrent-neural networks (RNNs) [38], and variants such as long short-term memory (LSTM) [39–44] and a nonlinear autoregressive network with exogenous inputs (NARX) [45–47]. ANNs learn from historical data to predict future behavior, i.e., the learning procedure exploits a dataset representative of the battery behaviour to make predictions. Both NARX and LSTM are examples of recurrent neural network (RNN) architectures that can be used for time-series prediction and control. However, there are some differences between the two architectures. The main difference is in the way they model the temporal dependencies in the input data. NARX uses a feedback loop to propagate information from previous time steps to the current time step [45–47], whereas LSTM uses a more complex memory cell that can selectively retain or discard information from previous time steps [39–44]. SVM models work by finding the best boundary that separates data into different classes [25–29]. RF models use multiple decision trees to make predictions [30,31]. The main drawbacks of these data-driven methods are the large amount of data required for the training and the computational resources. The selection of a data-driven method is dependent on the specific application and its requirements, taking into consideration factors such as the amount of data available, the required computational resources, and the model complexity. Although each of the presented methods is well known in the literature, comparative analyses and performance evaluations for the available algorithms are lacking, especially when real data are employed.

Therefore, this study focuses on the use of machine learning algorithms to estimate the SOH of HV batteries in electric vehicles. This analysis is based on the trends in open-circuit voltage evolution over the battery lifespan, which is collected from 12 prototype vehicles. The goal is to establish a correlation between these values and the energy stored in the battery, allowing for the determination of the SOH. Six machine learning algorithms, including linear regression, k-nearest neighbors, support vector machine, random forest, classification and regression tree, and neural network, are evaluated and compared to determine the most effective approach. The objective of this research is to provide a reliable estimation of the battery-replacement SOH, which is crucial for customers to minimize the risk of unexpected battery failure and accurately determine the battery's remaining useful life. Furthermore, this study offers insights into the machine learning algorithms that are employed, underlining the main differences in complexity, performance, interpretability, data requirements and preprocessing. The remainder of this paper is organized as follows: one section is dedicated to battery fundamentals, which encompasses a comprehensive glossary, a description of performance characteristics, and an overview of the main circuitual models. Subsequently, the methodology section will be presented, detailing the experimental data-collection procedures and the machine learning algorithms that are employed. Finally, the results of the study will be presented and conclusions will be drawn.

2. Fundamentals of HV Batteries

This section provides an overview of HV batteries, covering their technical specifications, fundamental characteristics, performance metrics, key technologies, and circuitual models. An introduction to the basics, including their physical and electrical properties, is presented. A discussion of the major technologies used, together with an explanation of their performance characteristics, immediately follows. Finally, circuitual models, which are used to simulate and predict battery behavior, are briefly described.

2.1. Technical Specifications

In the field of battery technology, technical specifications play a crucial role in describing the various characteristics and performance indicators of battery cells, modules, and packs. The nominal voltage is a key specification that addresses the reference voltage of the battery, whereas the cut-off voltage is the voltage at which a battery is considered fully

discharged. The nominal energy capacity of the battery, referred to as ‘capacity’ or ‘energy capacity’ in the present research work, is the total available Amp-hours when the battery is discharged at a specified current rate. In addition to these specifications, the nominal energy represents the total Watt-hours that are available when the battery is discharged at a specified current rate. The cycle life, which refers to the number of discharge–charge cycles a battery can undergo, is also listed. The specific energy and specific power describe the energy and power per unit mass of the battery, and the energy and power density describe the energy and power per unit volume of the battery. The technical specifications also cover the maximum continuous discharge current and the maximum 30 s discharge pulse current, which determines the most sustainable speed and acceleration of the vehicle. The recommended charge voltage, float voltage (i.e., the voltage at which a battery is maintained after being fully charged to maintain the capacity, compensating for the self-discharge of the battery), and charge current, as well as the maximum internal resistance (which is different for charging and discharging modes), are also specified. A basic glossary related to the main characteristics and performance indicators discussed in this article will be provided in the following section.

2.2. Battery Fundamentals

Cell, modules and packs—The HV battery in hybrid and electric vehicles is composed of individual modules and cells arranged in series and parallel. A cell is the smallest unit of a battery, typically ranging from 2.5 to 4.2 volts. Multiple cells are combined to form a module, which can be connected in series or parallel. The final battery pack is constructed by connecting multiple modules, either in series or parallel.

Battery Classifications—Battery performance and capabilities vary even among batteries of the same chemistry. The primary trade-off in battery design is between power and energy density, as batteries can either be high-power or high-energy-density, but not both. Manufacturers often categorize batteries based on these characteristics. A high-power battery is designed to deliver large amounts of power in a short period of time (e.g., engine start-up). High-power batteries typically have a low energy density, meaning that they cannot store as much energy as a high-energy-density battery of the same size. On the other hand, a high-energy-density battery is designed to store a large amount of energy in a small volume. This is crucial to providing the driving range in an electric vehicle. High-energy-density batteries typically have a lower power output, i.e., they cannot deliver as much power as a high-power battery of the same size. Another common classification is ‘high durability’, which indicates that the chemistry has been altered to increase the battery life, at the cost of power and energy.

C- and E- rates—The discharge current in batteries is often measured in terms of a C-rate to standardize it against the varying battery capacities. A C-rate represents the rate at which a battery is discharged in comparison to its maximum capacity. For example, a 1C rate for a battery with 100 Ah capacity means a discharge current of 100 Amps will discharge the entire battery in 1 h. Similarly, a 5C rate for the same battery would equate to 500 Amps, and a C/2 rate would be 50 Amps. Furthermore, the E-rate is used to describe the discharge power, with a 1E rate, indicating the power required to discharge the entire battery in 1 h.

2.3. Performance Characteristics

State of Charge (SOC)—This is a measure of the current battery capacity expressed as a percentage of the maximum capacity. It is calculated by tracking changes in the battery capacity over time using current integration.

Depth of Discharge (DOD)—This is the amount of the battery capacity that has been used, expressed as a percentage of the maximum capacity. A discharge of 80% or more of the total capacity is considered a ‘deep discharge’.

Terminal Voltage (V)—This refers to the voltage between the battery terminals when a load is applied. The terminal voltage changes with the SOC and the discharge or charge current.

Open-Circuit Voltage (OCV)—This is the voltage between the battery terminals when no load is applied. The OCV is influenced by the battery SOC and tends to increase as the SOC increases.

Internal Resistance—This refers to the resistance within the battery, which can vary with the SOC and depending on whether the battery is being charged or discharged. An increase in the internal resistance decreases the battery efficiency and increases the amount of charging energy that is converted into heat, thus reducing the thermal stability.

2.4. Battery Technologies and Circuitual Models

Table 1 presents a summary of some of the most important battery technologies available on the market, namely lead–acid (PbO_2), nickel–metal–hydride (NiMH), lithium nickel manganese cobalt (NMC), lithium nickel cobalt aluminum oxide (NCA) and lithium iron phosphate (LFP). The table embeds an analysis of the upsides and downsides of each battery type to gain a deeper understanding of the characteristics that make a battery type successful in automotive applications. As an example, Li-ion batteries, including NMC, NCA and LFP, possess high energy density and specific power, boast a good cell voltage, have a low self-discharging rate, and exhibit efficient charging and discharging cycles. These characteristics make Li-ion batteries a reliable and powerful option for automotive applications.

Table 1. Battery technologies comparison.

Type	Specific Energy ¹ (Wh/kg)	Energy Density ¹ (Wh/L)	Specific Power ¹ (W/kg)	Self-Discharge Rate ^{1,2,4} (%)	Cycles ¹ (10 ³ Cycles)	η_{disch} ^{1,3} (%)	Cost ¹ (\$/kWh)
PbO ₂	30–70	80–90	150–400	5–15	0.3–0.7	70–95	50–200
NiMH	60–120	80–150	250–400	1–3	0.5–3	70–95	100–250
NMC	100–200	200–400	300–1500	3	1–2	95–98	100–200
NCA	200–300	300–700	1500–2000	3	1–2	95–98	100–250
LFP	90–160	90–120	300–1500	1	>2	95–99	100–150

¹ Based on [48–50]; ² refers to the rate at which it loses its charge over time, mainly due to chemical reactions;

³ measures how effectively the battery is able to convert stored energy into usable energy during discharge;

⁴ per month.

The examination of various battery technologies in Table 1 sets the stage for a deeper understanding of the battery behavior, whereas the circuit models serve as an effective tool for simulating and analyzing the internal physics of the batteries. The simplest circuit model is known as the internal resistance (IR) model and is depicted in Figure 1a. The model consists of an ohmic internal resistance (R_o), the battery output current (i_{batt} , which is negative during charging and positive during discharging), the terminal battery voltage (v_{batt}), and the open circuit voltage (OCV, v_{oc}) that is measured when there is no current flowing through the circuit. The internal resistance and OCV are functions of SOC, SOH, and temperature, providing crucial insights into the battery performance and behavior. However, the IR model is not suitable for accurately estimating the SOC during dynamic operations (non-constant load) as it does not capture the transient behavior of the cells. A more advanced model, known as the one-time constant (OTC) model, can be used to address such issue. In the OTC model, a parallel RC network is added in series to the internal resistance R_o of the IR model to better approximate the dynamic behavior of the battery. The OTC model depicted in Figure 1b consists of three main parts: the voltage source V_{oc} ; the ohmic resistance R_o ; a system constituted by the parallel of a R_{otc} resistance

and a C_{otc} capacity that describes the transient response of the battery during charging or discharging [51].

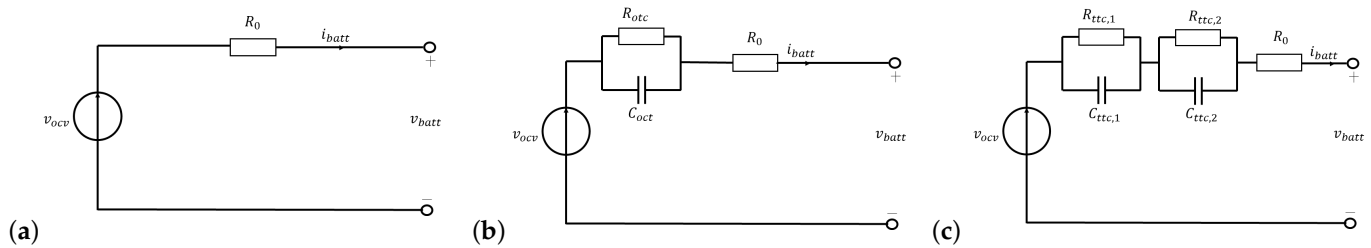


Figure 1. (a) RI circuitual model, (b) OCT circuitual model, (c) TTC circuitual model.

However, the observation of the battery output voltage when no load is applied reveals that the battery has a significant difference in its short-term and long-term transient behavior, making it difficult to accurately represent its dynamic characteristics when solely relying on the OTC model. This issue can be addressed by adding an extra RC network in series to the OTC circuit, thus forming the two time constants (TTC) circuit model [51].

The TTC circuit presented in Figure 1c is composed of four parts: the voltage source V_{oc} ; R_o ; the system made up of $R_{ttc,1}$ and $C_{ttc,1}$ to describe the short-term characteristics; the system made up of $R_{ttc,2}$ and $C_{ttc,2}$ to describe the long-term characteristics.

3. Methodology

The present research moved from data collection and processing using ETAS INCA and ETAS MDA software. These tools were exploited to establish a database of vehicles with varying mileage conditions to gather data on the corresponding OCV. This information was then used to determine the SOH of the battery. The data-collection process consisted of two parts, namely the recording of the OCV values during the vehicle pre-start phase and the collection of the OCV values at different SOC levels after the vehicle discharge and a resting time of two hours. The experiment was repeated to determine the correlation between the SOC level and the battery SOH. The data-collection process covered approximately 30 min for each cycle, and all relevant data (e.g., battery pack temperature, SOC) were tracked during the experiment.

3.1. Background Data Collection and CAN Bus

The previously discussed IR model serves as a useful tool to simulate and analyze the internal physics of the batteries. However, the accuracy of the simulation results, such as temperature evolution, SOC, and SOH analysis, depends on the reliability of the values attributed to the main parameters, e.g., the internal resistance, which is difficult to obtain from the manufacturers. The present section delves into the importance of a proper time analysis in the open-circuit condition of the TTC model in physical battery cell models. The main data were collected through the CAN bus system, which enables communication between the different parts of the vehicle through a network of nodes (represented by electronic control units-ECUs) connected by two wires: CAN low and CAN high. The CAN bus is a multi-master serial communication network that is simple, low-cost, and robust. Communication takes place through CAN frames, which consist of components such as:

1. Start of frame (SOF)—This is a dominant “0”, indicating that a node intends to broadcast information.
2. Identifier (ID)—This is a unique identifier that defines the frame and holds a higher priority for lower ID values.
3. Remote transmission request (RTR)—This indicates whether a node is sending data or requesting dedicated data from another node.
4. Control—This 6-bit field contains the identifier extension bit and the data length code (DLC), which specifies the length of the data bytes to be transmitted.

5. Data payload (Data)—This contains the actual information being communicated.
6. Cyclic redundancy check (CRC)—This ensures data integrity.
7. Acknowledgment (ACK)—This indicates if the node has received and acknowledged the data correctly.
8. End of frame (EOF)—This marks the end of the CAN frame.

The raw CAN bus data are recorded using a CAN bus data logger and decoded into human-readable values using a software tool and a standardized file format known as a “CAN database” or DBC file. ETAS integrated calibration and application tool (INCA) and ETAS measure data analyzer (MDA) allow for raw CAN bus data to be properly recorded and the relevant DBC file to extract human-readable data. However, the decoding rules for CAN bus signals in most assets, including vehicles, are proprietary and only known to the manufacturer, varying across different models and brands.

3.2. Data Collection

The testing procedure was conducted on prototypes of vehicles that have not yet been mass-produced, so the battery pack is either a production model or was assembled with production components and technologies. To maintain the battery condition, the hood was opened and the battery was switched off by laying it down on a support after each use without closing the hood. According to the testing procedure (TP), the vehicle was brought to the pre-start phase by pressing the start button twice without pressing the brake pedal to prevent the closure of the HV battery circuit. Then, the INCA software was initialized, the connections between the CAN data-logger and the vehicle were checked, and the a2l file, which contains information about the ECU, was downloaded. Next, all signals were verified to ensure that they were functioning and visible on the INCA software. The SOC level was then checked as a reference for collecting OCV values. If the signal visualization was successful, the first part of the data collection process (EXP-1) would begin. The testing procedure involved recording the OCV values for approximately 15 s. Afterwards, the vehicle was started by pressing the start button and the brake pedal, which resulted in a voltage drop as the HV battery circuit closed and in a rise in the closed-circuit voltages (CCVs). Once the vehicle had been turned on for the same duration, the recording was stopped, and the data validity was assessed. The validity criteria required the voltage drop and discrepancies between OCV and CCV to be observed in the measurement data analysis (MDA) before and after starting the vehicle. If the data met the validity criteria, they were saved in the appropriate folders to expand the measurement database. If the criteria were not met, the OCV measurement was repeated after an additional hour of waiting.

If the first part of the TP was successfully performed, the second part, which involved the battery discharge, would start. The objective of this process was to reach the desired SOC level for OCV measurement, with the entire data-collection phase being related to the SOC level reached at the end of the discharge cycle. OCV values were collected at different SOC levels, spaced approximately 15% apart, except for the last level, with a safety margin of from 1 to 1.5% remaining before stopping the vehicle to account for any recalculations within the electronic control units (ECUs). This margin was used to ensure that the OCV measurement is as precise as possible. After a designated amount of time of two hours, the TP would be repeated by measuring the remaining OCV values at the SOC level at the end of the previous discharging cycle. Based on the company expertise, two hours of rest provide sufficient time for the high-voltage battery circuit to attain steady-state conditions, specifically concerning the open circuit condition, thereby mitigating residual currents that may flow within the circuit due to the capacitor still releasing a current. It was recommended to keep track of all relevant data during the experiment, which could be used to determine the correlation between SOC level and the remaining driving range. Typically, discharging 15% of the HV battery took around 30 min at a medium-high speed without regenerative braking.

3.3. Data Processing

The analyses were conducted at cell level for each vehicle by deriving the cell energy content from the battery pack energy content and number of cells (n_{cells}). As a result, the battery pack of each vehicle represents a different level of aging. To assess the non-uniformity or non-homogeneity of the cells in a battery pack, the deviation between the voltage measured for each individual cell and its nominal value was analyzed at each SOC. The deviation of each individual cell was calculated by comparing the measured voltage to the expected value based on the cell specifications. The average and maximum deviations were then determined, with the maximum deviation identifying the largest or most significant error in the battery pack, which might be indicative of a problematic cell that needs to be addressed. This step was crucial to ensure the reliability and performance of the battery. On the left-hand side of Figure 2 the deviation point value for each vehicle and each state of charge are reported.

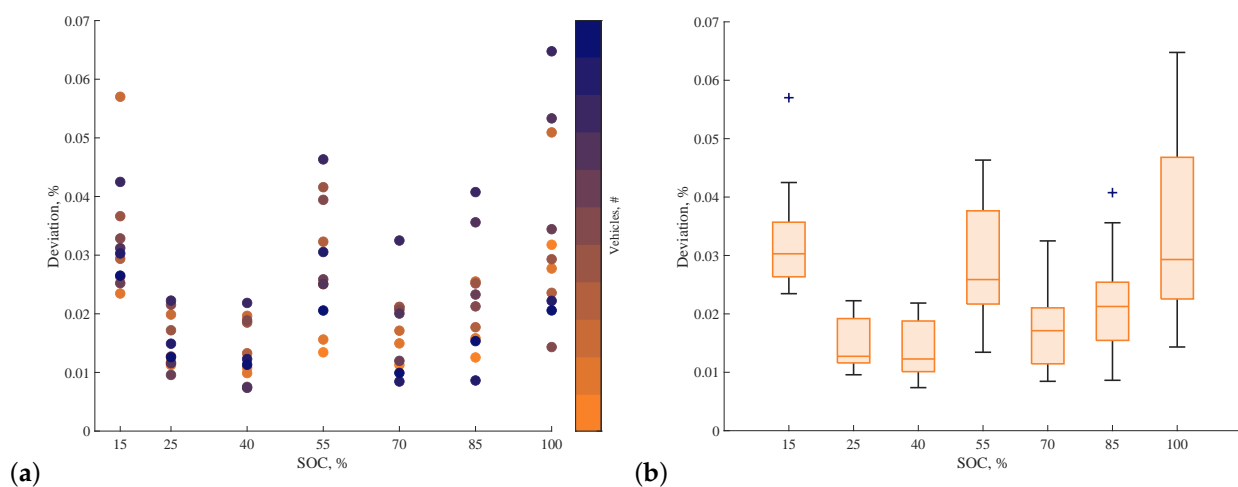


Figure 2. (a) The deviation point value for each vehicle and each state of charge, (b) Deviation boxchart: a visual representation of summary statistics for a sample of data, including the median, the lower and upper quartiles, outliers (denoted by a + sign), and the minimum and maximum values that are not outliers.

On the right-hand side of the figure, the box plot presents summary statistics for each battery SOC level, including the median, lower and upper quartiles, outliers, and non-outlier minimum and maximum values. The box plot serves as a visual representation of key statistical measures for the sample data. The box itself depicts the interquartile range, with the bottom and top edges representing the 25th and 75th percentiles, respectively. The vertical line within the box represents the sample median. The deviation of the median from the center of the box indicates sample skewness, reflecting the asymmetry of data distribution around its mean. This measure assesses the extent to which the data diverge from a normal distribution, specifically in terms of the balance between their left and right tails. The whiskers, represented by lines extending above and below the box, cover a range from the end of the interquartile range to the furthest observation within a length typically equivalent to 1.5 times the interquartile range. Observations lying beyond this range are identified as outliers and are denoted by a + sign [52].

As the mileage conditions vary, the deviation values fluctuate around extremely low values, around 0.07, for all considered SOCs.

3.4. Problem Formulation

It is worth recalling that the OCV (v_{ocv}) is the potential difference between the positive and negative terminals when there is no current flowing and the cell is at rest. The trend of the OCV as a function of the SOC can be visualized by discharging the cell from a 100% SOC down to a defined final SOC 'at rest' with given steps. Measuring the cell 'at rest'

requires an equilibrium to be achieved before measuring the potential difference. Based on these considerations, it was expected to obtain OCV curves between the beginning of life (BOL) and the end of life (EOL) as the battery ages. However, the results from all the measurements and data collected from different vehicles showed that the outcomes were far from what was expected. All the curves more or less overlapped one to the other, showing only some differences for the highest SOC values. Therefore, the idea was to use the OCV values collected from all the vehicles to properly link them to the amount of energy (kWh) per cell E_{stored} . This could be explained by taking into account the energy content represented at each OCV, starting from the amount of energy of the vehicle at each SOC level, which is available from other signals from the same experiment. From an empirical point of view, it was observed that a fully charged battery of 58 kWh has an OCV voltage value of 4.2 V when fully charged and drops to 3.3/3.4 V when the energy drops to 10 kWh.

For each vehicle mileage, the battery SOH can be calculated by dividing the actual battery capacity by the capacity of a new and fully charged battery:

$$SOH = \frac{C_{actual}}{C_{BOL}} \quad (1)$$

In turn, the cell capacity can be evaluated as:

$$C_i = \frac{E_{stored}}{n_{cells} \cdot v_{ocv}} \quad (2)$$

In the present experimental campaign, for the recorded battery SOC levels n (i.e., 15, 25, 40, 55, 70, 85, 100), the capacity is then calculated according to Equation (2) and the capacity at the beginning of life C_{BOL} is evaluated as the average of the capacity at the available SOC levels.

$$C_{BOL} = \frac{1}{n} \sum_{i=1}^n C_i \quad (3)$$

The procedure was carried out under two assumptions: (1) the battery pack has the same voltage per cell as the individual cells and (2) the voltage per cell is equal to the OCV voltage. Referring to the first assumption, the voltage of a battery pack depends on the configuration of the cells. If the cells are connected in series, the voltage of the battery pack will be the sum of the voltages of each individual cell, whereas if the cells are connected in parallel, the voltage of the battery pack will be equal to the voltage of the cell. Therefore, in a battery pack with cells connected in series and in parallel, the battery pack voltage may be higher than the voltage of the individual cell. On the other hand, referring to the second assumption, when assuming that the voltage per cell is equal to the OCV voltage, the C_{BOL} is overestimated. As a matter of fact, whereas the OCV of a single cell is typically measured at rest, under load, the cell voltage drops due to the voltage drop across the internal resistance of the cells and the other components in the battery pack. The difference between the OCV and the voltage under load depends on the load conditions, the battery SOC, the specific chemistry and the OCV measurements accuracy.

3.5. Machine Learning Based SOH Estimation

The best approach to estimate the OCV of individual cells in different vehicles based on their mileage was identified through a comprehensive literature review [25–47]. Several algorithms were evaluated through a careful selection process to determine their suitability. Supervised ML algorithms were selected over unsupervised ones as they are designed to estimate outputs based on input data and known responses, which aligns with the objective of this work. Furthermore, regression algorithms were chosen over classification ones, as the goal is to estimate the continuous value of OCV and C, rather than discrete responses. The selection process involved filtering the algorithms based on the type of data and results that were desired. The remaining algorithms that were investigated, studied,

and applied include linear regression (LR), k-nearest neighbors (KNN), classification and regression tree (CART), random forest (RF), support vector machine (SVM), and dense neural network (DNN).

The input features considered include the vehicle model, type and name, the type of battery, the number of battery cells, SOC levels, battery pack temperature and mileage conditions. Each feature was normalized using the min–max scaler from the Scikit-learn library [53], which helps prevent prioritization phenomena, i.e., it helps to ensure that all features hold an equal weight in the analysis. Parameter optimization was performed for each algorithm. The main methods considered were grid search and random search. The main difference between them is the way they explore the hyperparameter space. Grid search explores a predefined set of hyperparameters in a systematic way by creating a grid of all possible combinations and evaluating each combination. It is suitable when the hyperparameters have a small range of values. Random search explores the hyperparameter space randomly by selecting a set of hyperparameters at random from a predefined range and evaluating them. It is suitable when the hyperparameters have a large range of values. In general, random search can be more efficient than grid search for high-dimensional hyperparameter spaces, as it can cover a wider range of values and is less likely to get stuck in local optima. However, grid search can be more effective for low-dimensional spaces or when the hyperparameters have a clear interdependence. The selection of the optimal hyperparameter combination is based on the evaluation criteria of the root mean squared error (RMSE). The latter measures the difference between estimated and actual values by taking the square root of the average of the squared differences between estimated and actual values.

Linear Regression—LR is a statistical technique used to establish a linear relationship between an independent variable and a dependent variable. Multiple linear regression examines the relationship between multiple independent variables and a dependent variable, whereas simple linear regression focuses on a single independent variable.

K-nearest neighbors—KNN is a non-parametric supervised learning classifier that leverages proximity to perform classifications or estimations regarding the grouping of individual data points. Although it can be utilized for both regression and classification tasks, it is predominantly employed as a classification algorithm. The underlying principle behind KNN is the assumption that data points with similar attributes tend to cluster together in proximity.

Classification and regression tree (CART)—This is used for both classification and regression problems. It is a type of decision tree algorithm, i.e., it creates a tree-like model to make estimations based on the input features. The tree is constructed by recursively dividing the data into smaller subsets based on the feature that provides the best split in terms of minimizing the error. The process continues until the tree reaches a stopping criterion. The tree is used to make estimations by traversing the tree from the root to a leaf node, where the estimation is made based on the class label or regression value assigned to that node.

Random forest (RF)—This is an ensemble learning method, i.e., it combines multiple decision trees to make a final estimation. In RF, many decision trees are created, each using a different subset of the training data and a different subset of the features. These decision trees are then combined to form a single estimation by either taking a majority vote in the case of classification, or by averaging the estimations in the case of regression.

Support vector machine (SVM)—This is based on the idea of finding a hyperplane that best separates the data into different classes or estimates the target variable. In SVM, the data are transformed into a high-dimensional space and a hyperplane that maximizes the margin between the data points and the hyperplane is defined. The margin represents the distance between the closest data points and the hyperplane, and is used to define the boundary between different classes. In the case of regression, the hyperplane is used to estimate the target variable.

Dense neural network (DNN)—A DNN consists of an input layer, hidden layers, and an output layer. The input layer receives the input data, whereas the hidden layers process the data through a series of computations known as activation functions. The output layer provides the final estimation based on the processed data. The hidden layers use weights and biases to transform the data. Such parameters are acquired through a process called backpropagation, which adjusts the weights and biases to minimize the estimation error. DNNs can handle complex non-linear relationships between the features and targets variable and can model high-dimensional data. These characteristics, combined with the relative simplicity of DNNs in comparison to more complex architectures such as alternative backpropagation neural networks, RNNs, and their variants, prompted the authors to specifically opt for its implementation, taking into consideration the specific application and available dataset. Nevertheless, it is important to acknowledge that alternative choices could have been considered and pursued.

3.6. Comparison of ML Algorithms

These algorithms were analysed based on six indexes, namely usability, performance, interpretability, sensitivity to outliers, required data and preprocessing. The main considerations derived from the literature are summarized in Table 2.

Table 2. Comparison of ML algorithms.

Type	Complexity ¹	Performance ²	Interpretability ³	Sensitivity to Outliers	Data Requirements	Preprocessing
CART	Low	Medium	High	Low	Low–medium	Minimal
DNN	High	High	Low–Medium	Medium	High	Complex
KNN	Low	Low to Medium	Low	High	Low–medium	Minimal
LR	Low	Medium	High	High	Low–medium	Minimal
RF	Medium	High	Medium	Low–medium	Low–medium	Minimal
SVM	Medium	High	Low	Medium	Low–medium	Complex

¹ Understanding, implementing, and utilizing the algorithm; ² Trade-off between accuracy and efficiency (i.e., computational complexity, processing times, and resource utilization); ³ Degree of transparency and understandability of the model estimations and decision-making process; this includes the ability to explain the relationship between the input features and the target output, as well as the ability to understand why a particular estimation was made.

Complexity—Complexity refers to the level of difficulty or the amount of resources that need to be implemented and use a given algorithm. It may be determined by a number of factors, including the number of parameters or hyperparameters that need to be set, the amount and quality of data required for the training and testing, the computational resources required, and the level of expertise needed to understand and use the algorithm effectively. LR is considered simple and easy to implement, with a straightforward optimization procedure. KNN is also considered simple, with a low number of hyperparameters that need to be tuned. CART is also considered simple to implement, but may require some fine-tuning of the hyperparameters to achieve optimal results. RF is more complex to implement than LR, KNN, and CART, but still relatively straightforward. SVM is considered relatively complex but highly effective for certain types of data. DNN is considered the most complex of all the algorithms, with a large number of hyperparameters that need to be tuned and a greater need for computational resources. However, the complexity of each algorithm can vary depending on the user’s level of expertise and the specific implementation and configuration of the algorithm.

Performance—Performance refers to the ability to balance accuracy and efficiency, which involves making accurate predictions or classifications while minimizing computational complexity, processing times, and resource utilization. LR is fast to train and make estimations, but its performance may be limited by its linear assumption. KNN is relatively fast for small datasets, but may become slow for large datasets. CART is fast for both

training and estimation, but may overfit for certain types of data. RF is relatively fast for both training and estimation, and often provides high accuracy. SVM is relatively slow to train, but highly accurate for certain types of data. DNN is computationally expensive to train, but can achieve state-of-the-art results for many problems.

Interpretability—Interpretability refers to the degree of transparency and understandability of the model estimations and decision-making process; it includes the ability to explain the relationship between the input features and the target outputs, as well as the ability to understand why a particular estimation was made. LR has a clear and straightforward interpretation, with coefficients representing the importance of each feature. KNN has limited interpretability, but its results can be visually represented and understood. The low interpretability derives from the fact that the model is not expressed as a mathematical equation or set of rules, but instead relies on the distances between points to make its estimations. This means that it can be difficult to understand why a particular estimation is made, or the relationship between the input features and the target output. Additionally, the estimation made by KNN depends on the choice of k (i.e., the number of nearest neighbors) and the weighting function used, making it less transparent and less interpretable than other algorithms. CART has a clear interpretation, with each split in the tree representing a decision based on the input features. RF is more difficult to interpret than CART, but its results can be visualized to some extent. SVM is difficult to interpret, with its decision boundaries being represented by complex mathematical equations. DNN is the most difficult to interpret, with its results represented by a series of weighted connections between nodes.

Sensitivity to outliers—An algorithm is considered sensitive to outliers if its results are significantly impacted by the presence of outliers in the data (the term outlier indicates those points that are significantly different from the rest of the data). LR is generally sensitive to the presence of outliers. Indeed, the algorithm tries to fit a straight line to the data, and outliers can have a significant impact on the slope and intercept of the line. This can lead to a poor fit between the model and the data, resulting in inaccurate estimations. KNN is generally not very sensitive to outliers because the algorithm is based on voting among the k nearest neighbors, so a single outlier would not have a significant impact on the results. Nevertheless, KNN may be sensitive to outliers if k is small and the outlier is close to the data points being classified. RF is robust and insensitive to outliers. Individual decision trees are not significantly impacted by outliers because random forest is an ensemble method that constructs multiple decision trees and aggregates their estimations. However, if there is a large number of outliers, it may be necessary to preprocess the data in order to eliminate or reduce their impact. The CART algorithm is robust and insensitive to outliers. Individual splits are not significantly affected by outliers because CART is a tree-based method that recursively divides the data into smaller subsets based on the values of the features. These measures are designed to be insensitive to the presence of outliers. However, if there is a large number of outliers, it may be necessary to preprocess the data in order to eliminate or reduce their impact. SVM is less sensitive to outliers than linear regression. In fact, SVM tries to find the hyperplane that best separates the data into different classes, and it can effectively ignore outliers in the process. However, if there are many outliers, SVM may still produce inaccurate results. DNNs are less sensitive to outliers compared to linear regression. Neural networks are able to model complex relationships in the data, and they can effectively handle outliers in the data by adjusting the weights of the model. If there are many outliers, it may still be necessary to preprocess the data to remove or mitigate their impact. However, the sensitivity to outliers can vary depending on the specific implementation and configuration of each algorithm, as well as on the characteristics of the considered data.

Data requirements—Data requirements refer to the amount and type of data needed to successfully train a machine learning model. The data requirements can vary depending on the type of algorithm used. Several popular machine learning algorithms have different requirements for the size of the datasets on which they are trained. LR and KNN typically

require small to medium-sized datasets, which can range from a few hundred to a few thousand data points. The size of the dataset needed for KNN may depend on factors such as the number of features, the number of classes, and the distribution of the data, whereas for LR it may depend on the complexity of the problem and the number of features in the dataset. Decision trees, such as CART and SVM, also require small to medium-sized datasets, which may depend on the complexity of the problem, the number of features, and the choice of kernel function. RF is similar in this regard, but the number of trees in the ensemble and the depth of each tree may also be relevant. DNNs typically require large datasets with from tens of thousands to millions of data points, depending on the complexity of the network architecture and the choice of activation functions.

Preprocessing—Data preprocessing is the manipulation of data prior to training in order to smooth the learning process of a specific algorithm. Generally, these algorithms are sensitive to inconsistent, missing, and noisy data, which prevents them from identifying the correct relationship between input and output variables. As an example, a duplicate or missing value may result in incorrect data statistics. Data cleaning and transformation are required. Data cleaning entails handling missing values, smoothing noisy data, removing outliers, and resolving inconsistencies. Data transformation entails altering the format, structure, and value of data through the use of procedures such as normalization and standardization. LR requires the data to be clean and properly formatted, which may involve dealing with missing values, handling outliers, and scaling the data if necessary. KNN is a non-parametric algorithm that requires the data to be normalized or scaled so that all features equally contribute to the distance metric used by the algorithm. KNN can be sensitive to noisy or irrelevant features, so that feature selection or engineering may be necessary. CART does not require much preprocessing, but it may be sensitive to noisy or irrelevant features. Feature selection or engineering may be necessary, and they can overfit the data, so regularization techniques may be used to prevent overfitting. SVM requires properly preprocessed data, which may involve scaling, normalization, and handling missing values. SVM can be sensitive to noisy or irrelevant features, so that feature selection or engineering may be necessary. RF requires a similar preprocessing to that of decision trees, such as handling missing values, feature selection, and pruning. RF is relatively robust to noisy or irrelevant features, but scaling or normalization may be necessary. DNNs require a similar preprocessing to that of SVMs and may also require additional preprocessing steps, such as normalization, regularization, and data augmentation. DNNs are also sensitive to the choice of activation functions, which may require experimentation and fine-tuning.

4. Results

The dataset comprises data collected from 12 different vehicles, with 2.5 discharging cycles recorded for each vehicle. The SOC level and battery temperature were recorded at the start of each test, with SOC levels varied across seven different levels ranging from 100% to 15%. Table 3 provides a summary of the characteristics of the vehicles used in this project, focusing on the key features such as mileage, battery size, and battery voltage. This table serves as a quick reference for the reader and provides an overview of the data used in this study. As a remark, for confidentiality reasons, the exact number of cells cannot be disclosed, as it pertains to specific types of batteries.

Table 3. Vehicle database.

	VEH1	VEH2	VEH3	VEH4	VEH5	VEH6	VEH7	VEH8	VEH9	VEH10	VEH11	VEH12
Mileage (10 ³ km)	0.75	1.5	4	6.6	12.5	15.2	16.35	17	17.85	22	130	132
C _{batt} (kWh)	72.6	72.6	77.4	58	72.6	58	72.6	58	72.6	77.4	77.4	58
V _{batt} (V)	653	653	697	522.7	653	522.7	653	522.7	653	697	697	522.7

As an example, Figure 3 shows a plot of the recorded OCV for VEH1 at different SOC levels for all the considered cells. The OCV generally decreases as the SOC decreases, which is consistent with the expected behavior of a lithium-ion battery. Both the OCV values and the cell numbers were properly normalized. More specifically, the OCV values were normalized by dividing each value by the OCV maximum, while the number of cells in each group was divided by the total number of cells in the battery. It should be noted that only slight deviations are observed among the various cells.

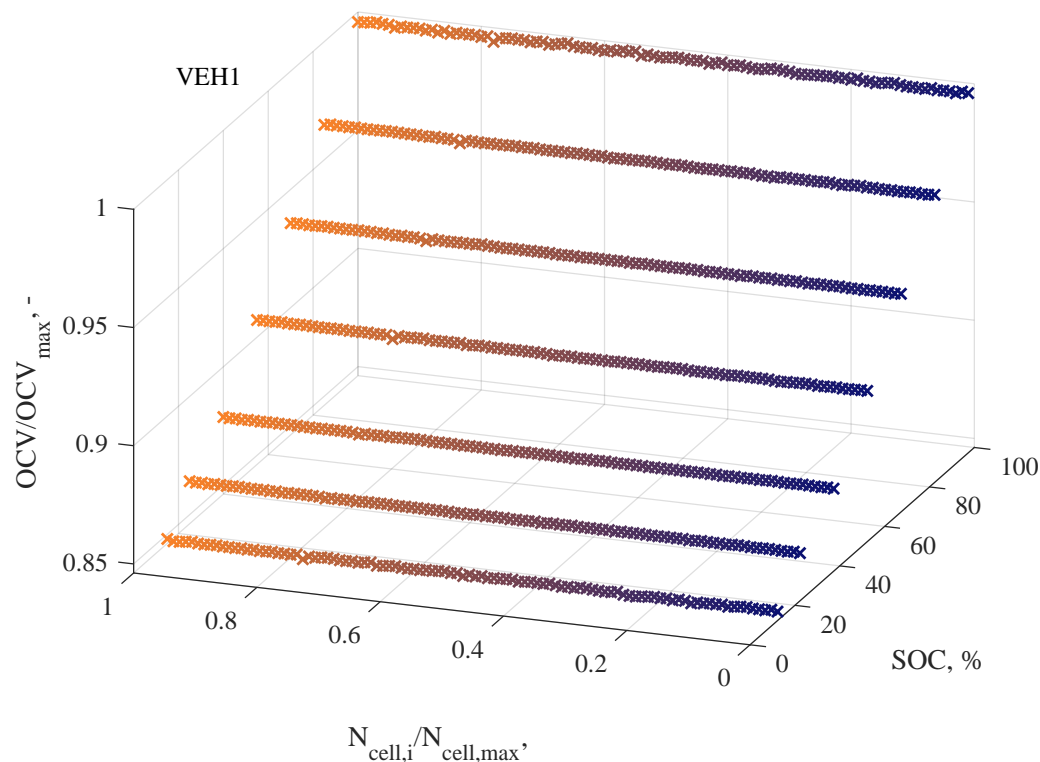


Figure 3. OCV trend VEH1.

The selected algorithms underwent a hyperparametric optimization by analyzing the performance in terms of both training and testing. This comparison was performed to ensure that the selected combination offered good results for the algorithm as a whole and not just due to overfitting on the training data. Overfitting occurs when a statistical model exactly fits its training data, thus impairing the ability of the algorithm to accurately estimate unseen data. In order to avoid overfitting, hyperparameters that result in slightly higher error rates are commonly used. This improves the model's ability to generalize to new data. Figure 4 shows the % relative error resulting from hyperparameter optimization during training and testing for the OCV and C, respectively. In the box chart, it can be observed that both the OCV and C exhibit the highest errors for the LR and SVM algorithms, with a slightly worse performance in the case of C compared to OCV. The DNN algorithm follows, with moderately high errors. On the other hand, the RF, KNN, and CART algorithms provided the best performance among the considered models. In addition, for the KNN, RF, and CART algorithms, except for the outliers, the errors are generally bounded between -0.15 and 0.15 , which is an encouraging result. For the DNN algorithm, the errors are slightly larger, with a range from -0.4 to 0.4 . In contrast, the SVM algorithm exhibits larger errors, with a range from -1.1 to 1.1 , while the LR algorithm shows the largest errors with a range from -2.3 to 2.3 .

For the sake of completeness, a zoom on the OCV trend for all SOC values is reported for the algorithms with the best and worst performance, RF and LR, respectively (Figure 5). It is worth noting that while the RF algorithm demonstrates a comparable performance across all SOC levels, the LR algorithm tends to overestimate the prediction accuracy for

samples that have SOC values close to the mean or central SOC levels. The reason for this may be that the LR model is a linear model that assumes a linear relationship between the input variables and the target variable. Therefore, it might not be able to capture the non-linear relationship between the input variables and the outputs, thus producing biased results.

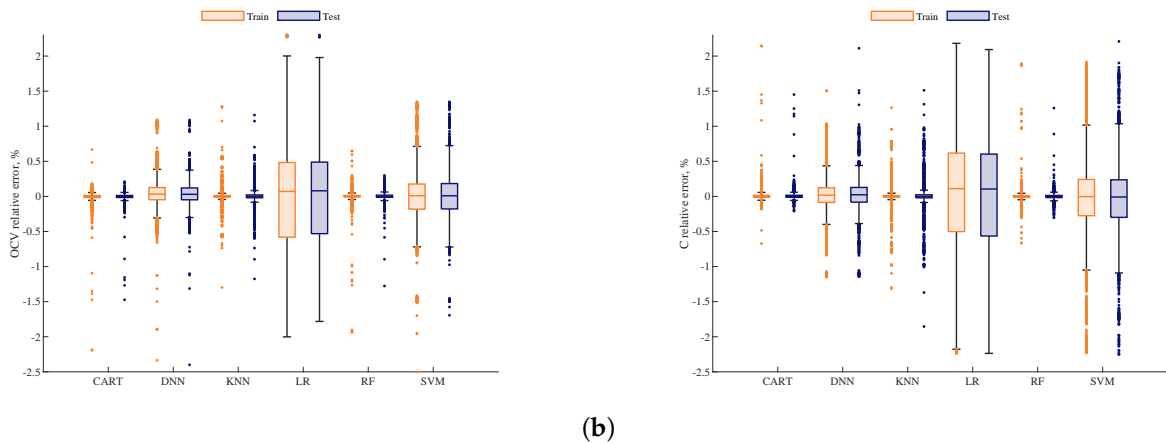


Figure 4. (a) OCV relative error resulting from hyperparameters optimization in train and test for all the considered algorithms, (b) C relative error resulting from hyperparameters optimization in train and test for all the considered algorithms.

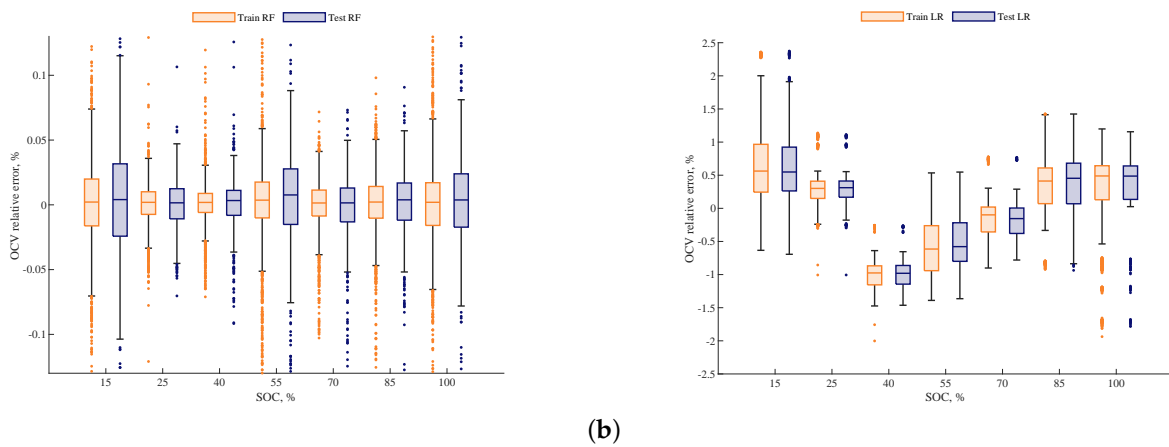


Figure 5. (a) A zoom on the OCV relative error for the RF algorithm analyzed for all SOC values considered (b) A zoom on the OCV relative error for the LR algorithm analyzed for all SOC values considered.

Table 4 presents a comparison of the errors generated by all algorithms. The table shows the average and maximum errors computed from the entire dataset. The results indicate that the random forest (RF) algorithm outperforms the other algorithms in both \bar{e}_{ocv} and \bar{e}_c metrics. Specifically, the RF algorithm achieves the lowest values of both metrics, with only a 0.02% error rate, which is significantly lower than the other algorithms. The DNN algorithm also performs well, achieving the lowest value of $e_{max,ocv}$ with only a 2.4% error rate, but with slightly higher \bar{e}_{ocv} and \bar{e}_c than RF. In contrast, the LR algorithm has the highest \bar{e}_c and $e_{max,ocv}$ values among all algorithms. The SVM algorithm shows a comparable performance with the KNN and CART algorithms in terms of both metrics. The table also includes the percentage improvement of RF over each algorithm. The RF algorithm demonstrates superior performance compared to other machine learning algorithms in various error rate measures. For instance, the RF algorithm achieves an average OCV

error rate improvement of -96.67% , -84.62% , and -92% over the LR, DNN, and SVM algorithms, respectively. Moreover, the maximum C error rate improvement in RF over DNN and SVM is -21.89% and -27.5% , respectively, indicating that RF outperforms these algorithms by a significant amount. Additionally, Figure 6 displays the correlation between the actual and estimated SOH trends together with the corresponding error as a function of the mileage. The error is expressed as the percentage difference between the real and estimated data. Specifically, for each analyzed vehicle, representing a distinct mileage condition, the variation in SOH was calculated. Although it may be disorienting to observe the SOH fluctuating in the first 22×10^3 km, this is due to variations in the SoC levels of the 12 different vehicles considered in Table 3. From 22 to 130×10^3 km, the trend is a clean line due to unavailable intermediate values. These evaluations were then combined into a single graph, along with the error introduced by the presented algorithms as compared to the experimental measurements. The decision to plot all data on a single graph was motivated by readability concerns and the desire to avoid overburdening the reader. Similarly, the same approach is used for the error. The best performance is observed for RF, KNN, and CART, followed by DNN, SVM, and LR, where LR exhibits the poorest performance.

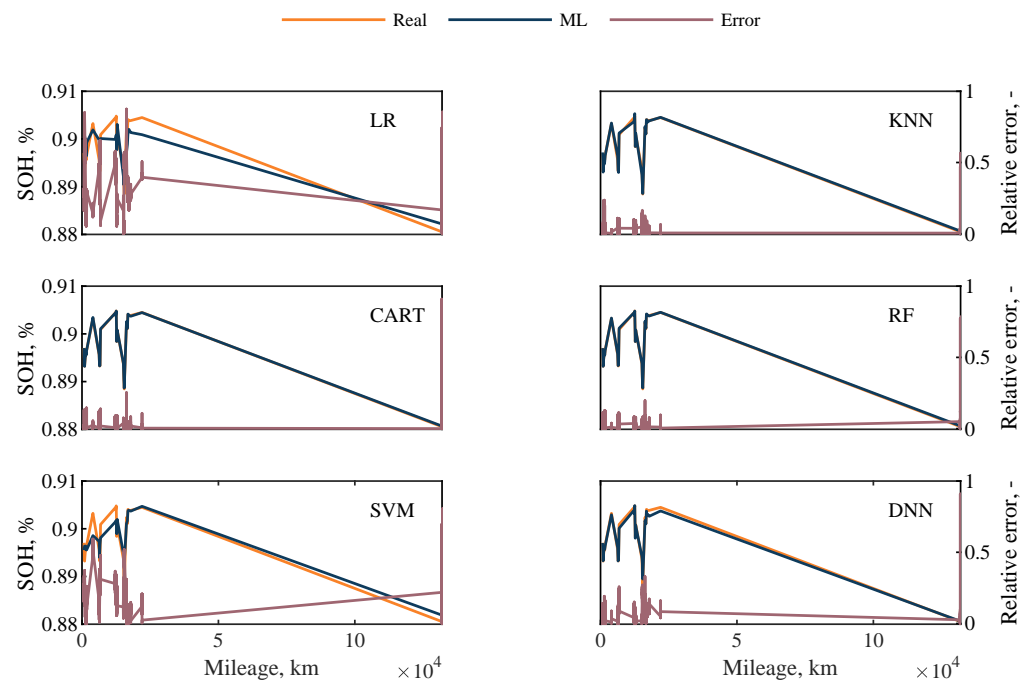


Figure 6. SOH and the error estimation as a function of mileage condition. For each graph, the fluctuations in the first 22×10^3 km are due to variations in the SoC and the 12 different vehicles considered in Table 3; instead from 22 to 130×10^3 km the trend is a clean line due to unavailable intermediate values.

Table 4. Results comparison.

Algorithm	\bar{e}_{ocv}	$e_{max,ocv}$	\bar{e}_c (%)	$e_{max,c}$	RF wrt Al- gorithm ¹	\bar{e}_{ocv}	$e_{max,ocv}$	\bar{e}_C (%)	$e_{max,C}$
LR	0.6	2.37	0.75	3.95	RF_{LR}	-96.67	$+0.42$	-97.33	-41.27
DNN	0.13	2.4	0.16	2.97	RF_{DNN}	-84.62	-0.83	-87.5	-21.89
SVM	0.25	3.00	0.4	3.20	RF_{SVM}	-92	-20.67	-95	-27.5
RF	0.02	2.38	0.02	2.32					
K-NN	0.04	2.62	0.06	2.55	RF_{KNN}	-50	-9.16	-66.67	-9.02
CART	0.02	2.65	0.02	2.60	RF_{CART}	0	-10.19	0	-10.77

¹ Performance comparison of RF with respect to (wrt) other algorithms evaluated as $\frac{x_{RF} - x_{alg}}{x_{alg}} \cdot 100$.

In Figure 7, the performance of the six algorithms are compared using a radar diagram [54], and a score between 1 (worst) and 5 (best) is assigned considering some of the index listed in Table 2, eliminating common parts such as required data and preprocessing. RF outperforms the other algorithms in terms of accuracy and performance, while exhibiting moderate complexity and interpretability. RF, KNN and CART show similar accuracy and performance, with slightly lower complexity and interpretability. DNN shows good performance, but higher complexity and computational effort, as well as lower interpretability. On all performance metrics but interpretability, LR performs worse than other algorithms, while SVM performs worse than other algorithms but better than LR. A larger dataset would be required to obtain a better performance from DNN. LR is unsuitable for complex and high-performance tasks, while KNN's computational effort would increase with larger datasets. For confidentiality reasons, we are unable to disclose the specific details and measurements of the computational efforts. However, the detailed comparison of the algorithms considered in Section 3.6 was introduced to allow for readers to gain an understanding of the computational processes and complexity involved.

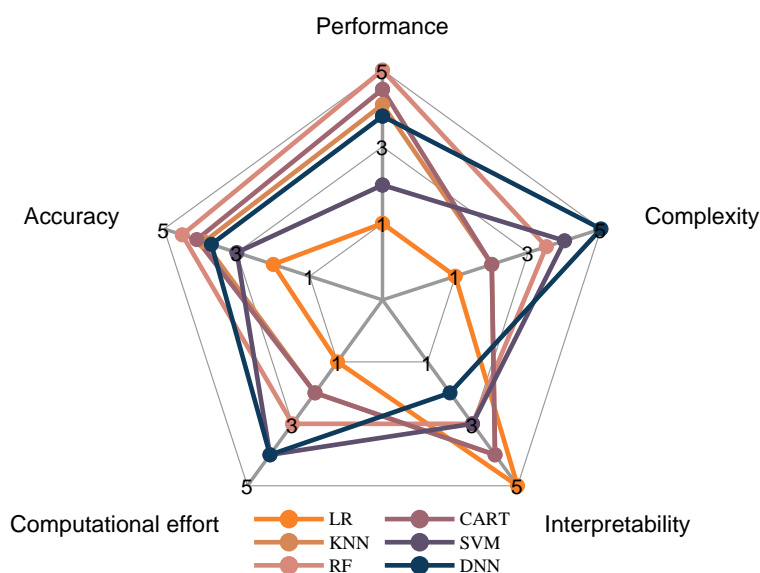


Figure 7. Comparison of the different algorithm results in terms of performance, complexity, interpretability, computational effort and accuracy.

5. Conclusions

The aim of this study was to evaluate the performance of various machine learning algorithms for the estimation of the state of health (SOH) of HV batteries in electric vehicles. The analysis was based on the open-circuit voltage (OCV) and capacity (C) trends over the battery lifespan, obtained from prototype vehicles. Six algorithms were evaluated and compared, including linear regression, k-nearest neighbors, support vector machine, random forest, classification and regression tree, and neural network. The comparison was made in terms of performance, complexity, interpretability, computational effort and accuracy. The results show that the ML algorithms produced generally low error estimates of SOH, with the random forest (RF) algorithm outperforming the others in terms of both average OCV error (\bar{e}_{ocv}) and average C error (\bar{e}_C) metrics, achieving the lowest values of both metrics with only 0.02% error. The DNN algorithm also performed well, achieving the 2.4% maximum OCV error ($e_{max,ocv}$), although it had a slightly higher average OCV and average C error than RF. On the other hand, the LR algorithm had the highest \bar{e}_C and $e_{max,ocv}$ values among all algorithms, as expected. This mainly occurs because a linear relationship is assumed between the input variables and the target ones, thus impairing its ability to capture the non-linear relationship between the input variables and the outputs.

Several factors can contribute to the better performance of RF in terms of SOH estimation and prediction with respect to other ML algorithms. One potential reason is that RF is a highly flexible algorithm that can model complex, non-linear relationships between input variables and outputs. This is important for SOH estimation and for prediction, as the battery behavior is often non-linear and difficult to accurately model using traditional linear techniques. Additionally, RF is less sensitive to overfitting than other algorithms, which can be a problem when working with smaller datasets.

Despite the generally low error estimates, there were still significant fluctuations in the SOH values of the vehicles. This was mainly due to the battery history of the charging and discharging phases, temperature variations, and data limitations caused by the availability of data from tested vehicles used in the experimental campaign. The concentration of data within the first 22×10^3 km of mileage was limited by the availability of vehicles. Clearly, additional research is required to improve the accuracy of SOH estimation in HV batteries. Exploring the use of additional data sources, such as the battery temperature history and charging/discharging cycles, is one potential area for improvement. In addition, it may be beneficial to consider the physical characteristics of the battery, such as the position of the cells within the battery modules, in order to improve SOH estimation accuracy. This would require careful experimental campaigning and data collection to determine the relationship between cell position and SOH. Lastly, expanding the dataset by collecting more vehicle data, especially for higher mileage vehicles, would help to further validate the accuracy of the SOH estimation algorithm.

Author Contributions: Conceptualization, A.M., E.B., D.A.M. and G.B.; methodology, A.M., E.B., D.A.M. and G.B.; software, E.L., A.M. and E.B.; validation, E.L. and E.B.; formal analysis, E.L., A.M. and E.B.; investigation, E.L., A.M. and E.B.; resources, E.L. and E.B.; data curation, E.L., A.M. and E.B.; writing—original draft preparation, A.M.; writing—review and editing, E.B., D.A.M. and G.B.; visualization, A.M.; supervision, E.B., D.A.M. and G.B. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: The data used in this study are not publicly available due to a non-disclosure agreement with the company.

Acknowledgments: The project was carried out within Hyundai Motor Europe Technical Center GmbH, and we would like to thank all the members of the Powertrain department for their substantial contribution to this research. Their technical and administrative support, as well as their expertise and assistance in defining the desired objectives for this application, were instrumental in the successful completion of this project.

Conflicts of Interest: The authors declare no conflict of interest.

Abbreviations

The following abbreviations are used in this manuscript:

ACK	Acknowledgment
ANN	Artificial neural network
BOL	Beginning of life
CART	Classification and regression tree
CCV	Closed circuit voltage
CRC	Cyclic Redundancy Check
DLC	Data Length Code
DNN	Dense neural network
ECU	Electronic control unit
EOL	End of life
EOF	End of Frame
HV	High-Voltage
ID	Identifier

INCA	Integrated Calibration and Application Tool
IR	Internal resistance
KNN	K-Nearest Neighbor
Li-ion	Lithium-ion
LFP	Lithium iron phosphate
LR	Linear regression
LSTM	Long short-term memory
MDA	Measure Data Analyzer
NiMH	Nickel-Metal-Hydride
NCA	Lithium Nickel Manganese Cobalt
NMC	Lithium Nickel Manganese Cobalt
OCV	Open circuit voltage
OTC	One time constant
RF	Random forest
RNN	Recurrent neural network
RTR	Remote transmission request
SOH	State of health
SOC	State of charge
SOF	Start of frame
SVM	Support vector machine
TP	Testing procedure
TTC	Two time constant

References

- Slowik, P.; Pavlenko, N.; Lutsey, N. *Assessment of neXt-Generation Electric Vehicle Technologies*; White paper; International Council on Clean Transportation: Washington, DC, USA, 2016. [\[CrossRef\]](#)
- Fredriksson, G.; Roth, A.; Tagliapietra, S.; Veugelers, R. *Is the European Automotive Industry Ready for the Global Electric Vehicle Revolution?* Bruegel Policy Contribution: Brussels, Belgium, 2018.
- Hybrid Vehicle Market Analysis-Industry Report-Trends, Size & Share. Available online: <https://www.mordorintelligence.com/industry-reports/hybrid-vehicle-market> (accessed on 30 October 2022).
- Electric Vehicle Market Analysis-Industry Report-Trends, Size & Share. Available online: <https://www.mordorintelligence.com/industry-reports/electric-vehicle-market> (accessed on 30 October 2022).
- Lu, L.; Han, X.; Li, J.; Hua, J.; Ouyang, M. A review on the key issues for lithium-ion battery management in electric vehicles. *J. Power Source* **2013**, *226*, 272–288. [\[CrossRef\]](#)
- Vidal, C.; Malysz, P.; Kollmeyer, P.; Emadi, A. Machine Learning Applied to Electrified Vehicle Battery State of Charge and State of Health Estimation: State-of-the-Art. *IEEE Access* **2020**, *8*, 52796–52814. [\[CrossRef\]](#)
- Naguib, M.; Kollmeyer, P.; Emadi, A. Lithium-Ion Battery Pack Robust State of Charge Estimation, Cell Inconsistency, and Balancing: Review. *IEEE Access* **2021**, *9*, 50570–50582. [\[CrossRef\]](#)
- Yao, L.; Xu, S.; Tang, A.; Zhou, F.; Hou, J.; Xiao, Y.; Fu, Z. A Review of Lithium-Ion Battery State of Health Estimation and Prediction Methods. *World Electr. Veh. J.* **2021**, *12*, 113. [\[CrossRef\]](#)
- Mali, V.; Saxena, R.; Kumar, K.; Kalam, A.; Tripathi, B. Review on battery thermal management systems for energy-efficient electric vehicles. *Renew. Sustain. Energy Rev.* **2021**, *151*, 111611. [\[CrossRef\]](#)
- Yang, F.; Zhang, S.; Li, W.; Miao, Q. State-of-charge estimation of lithium-ion batteries using LSTM and UKF. *Energy* **2020**, *201*, 117664. [\[CrossRef\]](#)
- Luciani, S.; Feraco, S.; Bonfitto, A.; Tonoli, A. Hardware-in-the-Loop Assessment of a Data-Driven State of Charge Estimation Method for Lithium-Ion Batteries in Hybrid Vehicles. *Electronics* **2021**, *10*, 2828. [\[CrossRef\]](#)
- Ezemobi, E.; Silvagni, M.; Mozaffari, A.; Tonoli, A.; Khajepour, A. State of Health Estimation of Lithium-Ion Batteries in Electric Vehicles under Dynamic Load Conditions. *Energies* **2022**, *15*, 1234. [\[CrossRef\]](#)
- Hossain Lipu, M.; Hannan, M.; Hussain, A.; Ayob, A.; Saad, M.H.; Karim, T.F.; How, D.N. Data-driven state of charge estimation of lithium-ion batteries: Algorithms, implementation factors, limitations and future trends. *J. Clean. Prod.* **2020**, *277*, 124110. [\[CrossRef\]](#)
- Khodadadi Sadabadi, K.; Jin, X.; Rizzoni, G. Prediction of remaining useful life for a composite electrode lithium ion battery cell using an electrochemical model to estimate the state of health. *J. Power Source* **2021**, *481*, 228861. [\[CrossRef\]](#)
- Anselma, P.G.; Kollmeyer, P.; Lempert, J.; Zhao, Z.; Belingardi, G.; Emadi, A. Battery state-of-health sensitive energy management of hybrid electric vehicles: Lifetime prediction and ageing experimental validation. *Appl. Energy* **2021**, *285*, 116440. [\[CrossRef\]](#)
- Anselma, P.G.; Musa, A.; Maino, C.; Misul, D.; Belingardi, G. Effect of Temperature Distribution on the Predicted Cell Lifetimes for a Plug-In Hybrid Electric Vehicle Battery Pack. *Cell* **2022**. [\[CrossRef\]](#)
- Ungurean, L.; Micea, M.V.; Cârstoiu, G. Online state of health prediction method for lithium-ion batteries, based on gated recurrent unit neural networks. *Int. J. Energy Res.* **2020**, *44*, 6767–6777. [\[CrossRef\]](#)

18. Chiang, Y.H.; Sean, W.Y.; Ke, J.C. Online estimation of internal resistance and open-circuit voltage of lithium-ion batteries in electric vehicles. *J. Power Source* **2011**, *196*, 3921–3932. [[CrossRef](#)]
19. Yang, J.; Xia, B.; Huang, W.; Fu, Y.; Mi, C. Online state-of-health estimation for lithium-ion batteries using constant-voltage charging current analysis. *Appl. Energy* **2018**, *212*, 1589–1600. [[CrossRef](#)]
20. Saldaña, G.; San Martín, J.I.; Zamora, I.; Asensio, F.J.; Oñederra, O. Analysis of the Current Electric Battery Models for Electric Vehicle Simulation. *Energies* **2019**, *12*, 2750. [[CrossRef](#)]
21. Li, J.; Adewuyi, K.; Lotfi, N.; Landers, R.; Park, J. A single particle model with chemical/mechanical degradation physics for lithium ion battery State of Health (SOH) estimation. *Appl. Energy* **2018**, *212*, 1178–1190. [[CrossRef](#)]
22. Marcicki, J.; Canova, M.; Conlisk, A.T.; Rizzoni, G. Design and parametrization analysis of a reduced-order electrochemical model of graphite/LiFePO₄ cells for SOC/SOH estimation. *J. Power Source* **2013**, *237*, 310–324. [[CrossRef](#)]
23. Moura, S.J.; Chaturvedi, N.A.; Krstić, M. Adaptive Partial Differential Equation Observer for Battery State-of-Charge/State-of-Health Estimation Via an Electrochemical Model. *J. Dyn. Syst. Meas. Control* **2014**, *136*, 011015. [[CrossRef](#)]
24. Kim, J.; Cho, B.H. State-of-Charge Estimation and State-of-Health Prediction of a Li-Ion Degraded Battery Based on an EKF Combined with a Per-Unit System. *IEEE Trans. Veh. Technol.* **2011**, *60*, 4249–4260. [[CrossRef](#)]
25. Klass, V.; Behm, M.; Lindbergh, G. A support vector machine-based state-of-health estimation method for lithium-ion batteries under electric vehicle operation. *J. Power Source* **2014**, *270*, 262–272. [[CrossRef](#)]
26. Feng, X.; Weng, C.; He, X.; Han, X.; Lu, L.; Ren, D.; Ouyang, M. Online State-of-Health Estimation for Li-Ion Battery Using Partial Charging Segment Based on Support Vector Machine. *IEEE Trans. Veh. Technol.* **2019**, *68*, 8583–8592. [[CrossRef](#)]
27. Meng, J.; Cai, L.; Luo, G.; Stroe, D.I.; Teodorescu, R. Lithium-ion battery state of health estimation with short-term current pulse test and support vector machine. *Microelectron. Reliab.* **2018**, *88–90*, 1216–1220. [[CrossRef](#)]
28. Chen, Z.; Sun, M.; Shu, X.; Xiao, R.; Shen, J. Online State of Health Estimation for Lithium-Ion Batteries Based on Support Vector Machine. *Appl. Sci.* **2018**, *8*, 925. [[CrossRef](#)]
29. Nuhic, A.; Terzimehic, T.; Soczka-Guth, T.; Buchholz, M.; Dietmayer, K. Health diagnosis and remaining useful life prognostics of lithium-ion batteries using data-driven methods. *J. Power Source* **2013**, *239*, 680–688. [[CrossRef](#)]
30. Mawonou, K.S.; Eddahech, A.; Dumur, D.; Beauvois, D.; Godoy, E. State-of-health estimators coupled to a random forest approach for lithium-ion battery aging factor ranking. *J. Power Source* **2021**, *484*, 229154. [[CrossRef](#)]
31. Yang, N.; Song, Z.; Hofmann, H.; Sun, J. Robust State of Health estimation of lithium-ion batteries using convolutional neural network and random forest. *J. Energy Storage* **2022**, *48*, 103857. [[CrossRef](#)]
32. Yang, D.; Wang, Y.; Pan, R.; Chen, R.; Chen, Z. A Neural Network Based State-of-Health Estimation of Lithium-ion Battery in Electric Vehicles. *Energy Procedia* **2017**, *105*, 2059–2064. [[CrossRef](#)]
33. Zhang, S.; Zhai, B.; Guo, X.; Wang, K.; Peng, N.; Zhang, X. Synchronous estimation of state of health and remaining useful lifetime for lithium-ion battery using the incremental capacity and artificial neural networks. *J. Energy Storage* **2019**, *26*, 100951. [[CrossRef](#)]
34. Bonfitto, A. A Method for the Combined Estimation of Battery State of Charge and State of Health Based on Artificial Neural Networks. *Energies* **2020**, *13*, 2548. [[CrossRef](#)]
35. Ezemobi, E.; Tonoli, A.; Silvagni, M. Battery State of Health Estimation with Improved Generalization Using Parallel Layer Extreme Learning Machine. *Energies* **2021**, *14*, 2243. [[CrossRef](#)]
36. Luciani, S.; Anselma, P.G.; Silvagni, M.; Bonfitto, A.; Tonoli, A. Enabling Rapid State of Health Offline Estimation of a 48 V Lithium-Ion Battery Pack. In Proceedings of the 2022 IEEE Vehicle Power and Propulsion Conference (VPPC), Merced, CA, USA, 1–4 November 2022; pp. 1–6. [[CrossRef](#)]
37. Chemali, E.; Kollmeyer, P.J.; Preindl, M.; Fahmy, Y.; Emadi, A. A Convolutional Neural Network Approach for Estimation of Li-Ion Battery State of Health from Charge Profiles. *Energies* **2022**, *15*, 1185. [[CrossRef](#)]
38. Venugopal, P. State-of-Health Estimation of Li-ion Batteries in Electric Vehicle Using IndRNN under Variable Load Condition. *Energies* **2019**, *12*, 4338. [[CrossRef](#)]
39. Li, P.; Zhang, Z.; Xiong, Q.; Ding, B.; Hou, J.; Luo, D.; Rong, Y.; Li, S. State-of-health estimation and remaining useful life prediction for the lithium-ion battery based on a variant long short term memory neural network. *J. Power Source* **2020**, *459*, 228069. [[CrossRef](#)]
40. Wu, Y.; Xue, Q.; Shen, J.; Lei, Z.; Chen, Z.; Liu, Y. State of Health Estimation for Lithium-Ion Batteries Based on Healthy Features and Long Short-Term Memory. *IEEE Access* **2020**, *8*, 28533–28547. [[CrossRef](#)]
41. Falai, A.; Giuliacci, T.A.; Misul, D.A.; Anselma, P.G. Reducing the Computational Cost for Artificial Intelligence-Based Battery State-of-Health Estimation in Charging Events. *Batteries* **2022**, *8*, 209. [[CrossRef](#)]
42. Bi, J.; Lee, J.C.; Liu, H. Performance Comparison of Long Short-Term Memory and a Temporal Convolutional Network for State of Health Estimation of a Lithium-Ion Battery using Its Charging Characteristics. *Energies* **2022**, *15*, 2448. [[CrossRef](#)]
43. Jo, S.; Jung, S.; Roh, T. Battery State-of-Health Estimation Using Machine Learning and Preprocessing with Relative State-of-Charge. *Energies* **2021**, *14*, 7206. [[CrossRef](#)]
44. Zhao, S.; Zhang, C.; Wang, Y. Lithium-ion battery capacity and remaining useful life prediction using board learning system and long short-term memory neural network. *J. Energy Storage* **2022**, *52*, 104901. [[CrossRef](#)]
45. Khaleghi, S.; Karimi, D.; Beheshti, S.H.; Hosen, M.S.; Behi, H.; Bercibar, M.; Van Mierlo, J. Online health diagnosis of lithium-ion batteries based on nonlinear autoregressive neural network. *Appl. Energy* **2021**, *282*, 116159. [[CrossRef](#)]

46. Chaoui, H.; Ibe-Ekeocha, C.C. State of Charge and State of Health Estimation for Lithium Batteries Using Recurrent Neural Networks. *IEEE Trans. Veh. Technol.* **2017**, *66*, 8773–8783. [[CrossRef](#)]
47. Bamati, S.; Chaoui, H. Lithium-Ion Batteries Long Horizon Health Prognostic Using Machine Learning. *IEEE Trans. Energy Convers.* **2022**, *37*, 1176–1186. [[CrossRef](#)]
48. Dixon, J. Energy storage for electric vehicles. In Proceedings of the 2010 IEEE International Conference on Industrial Technology, Vina del Mar, Chile, 14–17 March 2010; pp. 20–26. [[CrossRef](#)]
49. Ahasan Habib, A.K.M.; Motakabber, S.M.A.; Ibrahimy, M.I. A Comparative Study of Electrochemical Battery for Electric Vehicles Applications. In Proceedings of the 2019 IEEE International Conference on Power, Electrical, and Electronics and Industrial Applications (PEEIACON), Dhaka, Bangladesh, 29 November–1 December 2019; pp. 43–47. [[CrossRef](#)]
50. Townsend, A.; Gouws, R. A Comparative Review of Lead-Acid, Lithium-Ion and Ultra-Capacitor Technologies and Their Degradation Mechanisms. *Energies* **2022**, *15*, 4930. [[CrossRef](#)]
51. He, H.; Xiong, R.; Fan, J. Evaluation of Lithium-Ion Battery Equivalent Circuit Models for State of Charge Estimation by an Experimental Approach. *Energies* **2011**, *4*, 582–598. [[CrossRef](#)]
52. The MathWorks Inc. Boxplot. 2023. Available online: <https://it.mathworks.com/help/stats/boxplot.html> (accessed on 1 February 2022).
53. Sklearn.Preprocessing.MinMaxScaler. Available online: <https://scikit-learn/stable/modules/generated/sklearn.preprocessing.MinMaxScaler.html> (accessed on 30 October 2022).
54. Moses. spider_plot. 2023. Available online: https://github.com/NewGuy012/spider_plot/releases/tag/20.2 (accessed on 20 January 2023).

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.