

Article

Ensemble Surrogate Models for Fast LIB Performance Predictions

Marco Quartulli ^{1,*} , Amaia Gil ¹, Ane Miren Florez-Tapia ¹, Pablo Cereijo ² , Elixabete Ayerbe ² 
and Igor G. Olaizola ¹ 

¹ Vicomtech, Basque Research and Technology Alliance, Mikeletegi 57, 20009 Donostia-San Sebastián (ES), Spain; agil@vicomtech.org (A.G.); amflorez@vicomtech.org (A.M.F.-T.); iolaizola@vicomtech.org (I.G.O.)

² CIDETEC, Basque Research and Technology Alliance, Pº Miramón 196, 20014 Donostia-San Sebastián (ES), Spain; pcereijo@cidetec.es (P.C.); eayerbe@cidetec.es (E.A.)

* Correspondence: mquartulli@vicomtech.org; Tel.: +34-943-309230

Abstract: Battery Cell design and control have been widely explored through modeling and simulation. On the one hand, Doyle's pseudo-two-dimensional (P2D) model and Single Particle Models are among the most popular electrochemical models capable of predicting battery performance and therefore guiding cell characterization. On the other hand, empirical models obtained, for example, by Machine Learning (ML) methods represent a simpler and computationally more efficient complement to electrochemical models and have been widely used for Battery Management System (BMS) control purposes. This article proposes ML-based ensemble models to be used for the estimation of the performance of an LIB cell across a wide range of input material characteristics and parameters and evaluates 1. Deep Learning ensembles for simulation convergence classification and 2. structured regressors for battery energy and power predictions. The results represent an improvement on state-of-the-art LIB surrogate models and indicate that deep ensembles represent a promising direction for battery modeling and design.

Keywords: Li-ion battery; surrogate modeling; deep learning ensembles



Citation: Quartulli, M.; Gil, A.; Florez-Tapia, A.M.; Cereijo, P.; Ayerbe, E.; Olaizola, I.G. Ensemble Surrogate Models for Fast LIB Performance Predictions. *Energies* **2021**, *14*, 4115. <https://doi.org/10.3390/en14144115>

Academic Editor: Marcin Kamiński

Received: 12 June 2021

Accepted: 6 July 2021

Published: 8 July 2021

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2020 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

This paper considers the task of estimating the energy and power density of LIB designs across a range of characteristics and parameters and focuses on the problem of obtaining these results without incurring the considerable computational costs of detailed physical simulations in a recurrent manner. While state-of-the-art contributions such as [1] exploit Deep Learning neural networks trained on simulated data as surrogate models for this objective, to the best of our knowledge, no literature contribution yet has evaluated the performance of so-called ensemble models for this task. We address this gap in the literature by comparing state-of-the-art models with a number of ensemble surrogates.

Even though Lithium-Ion Batteries (LIB) have progressively been improved since their market introduction in 1991 by Sony, their massive deployment requires them to be further optimized in terms of performance, durability, and safety. From the physical point of view, to obtain these results, in the battery design phase, it is important to reduce limitations to ion transport, thus avoiding undesirable cell polarization effects. Consequently, an extensive literature describes the effect each design parameter has on transport mechanisms, and therefore on cell performance. On the one hand, the effects of several parameters (including electrode thickness, particle size, tortuosity and discharge rate) have been investigated by experimental methods. On the other hand, cell design has also been explored through modeling approaches, demonstrating that model-based design can be used to reduce the number of experiments, while accurately describing battery performance and providing guidance for battery characterization.

In general, the battery models presented in the literature mainly fall into two categories: physics-based electrochemical models and empirical ones. In addition, the recent

development of physics-based equivalent circuit models aims to combine the high accuracy of physical models with the reduced computational cost of empirical ones.

Physics-based electrochemical models [2–4] use partial differential equations to describe the phenomena taking place within the battery. They can be used to forecast its electrochemical state, and to provide accurate information about variables such as lithium concentrations and over-potentials, which can be used to understand the phenomena that are limiting performance or durability. For example, the widely used Pseudo-two-Dimensional (P2D) electrochemical model developed by Doyle and Newman [5,6] is based on porous electrode theory, concentrated solution theory, and kinetics equations. Further electrochemical modeling approaches rely on Single Particle (SP) simplifications, in which the properties of the electrolyte are normally ignored [7–9]. Further recent approaches such as SPMe [10] consider electrolyte dynamics in SP models. Other possibilities include Simplified Models building on polynomial profiles [11,12], Galerkin approximations [13], transfer function modeling [14], and the like. Concerning limitations, a drawback of detailed electrochemical models is the significant computational cost of simulations [15]. To overcome this limitation, various reduced-order models have been developed, which are either distributed-parameter models [10,13] or lumped-parameter models [14,16]. Distributed-parameter models are normally expressed in the form of ordinary differential-algebraic equations (DAEs), derived, for example, using the Galerkin projection methods or Proper Orthogonal Decomposition, trying to preserve the physical meaning of all model parameters. Lumped-parameter models, which mimic the output voltage of the battery using electrical components such as resistors and capacitors, are described below.

Empirical models are typically either based on equivalent circuits [16,17] or on data-driven approaches [18,19]. battery control algorithm rather than in cell design optimization applications. They are composed of an open-circuit voltage source connected to a set of electric elements, such as resistors and capacitors, to model the electrical behavior of a battery. While these models are intuitive and relatively simple to use in control system design and implementation, they do not provide insights on the internal behavior of the battery. In this regard, new approaches have recently been explored, such as the developed distributed-parameter ECMs [20,21], in which the models are normally expressed in the form of DAEs, which can be solved rapidly using the proposed method with high accuracy. This represents an improvement on existing physics-based Li-ion battery models, especially in real-time, dynamic environments. Physics-based equivalent circuit models combine the benefits of high accuracy physical models with the lower computational cost of empirical ones, for instance, by combining a concise transmission line structure with partial differential equations for the mass transport processes that describe the concentration distributions and that are solved with the finite difference method, avoiding simplifications or approximations, thus guaranteeing the accuracy of the results [20]. Online estimation and prediction of the Remaining Useful Life also often use data-driven empirical methods which, however, have not been commonly exploited for cell design purposes.

So-called ‘surrogate’ models are obtained by Machine Learning methods from data generated by simulations. Once the training phase is completed, these models can provide estimations of battery performance indicators with a lower computational cost and with an accuracy similar to that of physical simulators. Probabilistic surrogate models are often used for optimization: by running the simulations at a set of points (experimental design), one obtains fast surrogates for otherwise expensive objective functions [22]. In methods such as Upper and Lower Confidence Bound [23], Expected Improvement [24], DYCORS [25], and SOP [26], the optimization jointly performs both exploration and exploitation, looking for an optimum, while, at the same time, sampling by simulation the most uncertain parameter regions. In this sense, surrogate optimization schemes can efficiently leverage the ability of probabilistic surrogate model classes, from Gaussian Processes to Sequential Radial Basis Functions, to jointly generate estimates for both the local value and the local uncertainty of an objective function of interest. Relevant efforts for LIB aging modeling include those based on Gaussian processes by Liu et al. [27–29].

Contributing to the domain of such surrogates, Wu et al. [1] propose to address cell design characterization by combining Machine Learning classifiers and regressors based on deep feed-forward neural network models. While the first neural network is a classifier that predicts whether a set of input design variables would result in a physically realizable cell, the second neural network estimates the specific energy and specific power of the design. Both neural networks are trained and validated using data from finite-element, thermo-electrochemical simulations.

Although ensemble models are well represented in the literature, to the best of our knowledge, no contribution yet has evaluated their performance as surrogates of physical LIB simulators. This paper addresses this gap by comparing a state-of-the-art LIB surrogate model based on deep feed-forward networks with a set of ensemble models integrating deep classifiers and regressors. In this sense, we demonstrate the accuracy of composite surrogate models for the estimation of the density of power and energy of a given LIB parameter set. The input and output parameters taken into consideration for the electrochemical simulations are listed in Table 1. Their values are measured by proprietary electrochemical and physicochemical protocols for a set of proprietary electrodes in CIDETEC, and vary between the minimum and maximum values shown in Table 1.

Table 1. Input and output parameters for both the simulators and the proposed surrogate models. The simulated dataset consists of 1000 records with 11 input and 3 output attributes.

Variable	Type		Units	Range
Anode/Cathode				
thickness	input	L_n, L_p	μm	45–75/60–90
porosity	input	ϵ_n, ϵ_p	-	0.2–0.3/0.18–0.28
particle radius	input	r_n, r_p	μm	3.5–8.5/8–11
Separator				
thickness	input	L_s	μm	15.0–30.0
porosity	input	ϵ	-	0.35–0.45
Electrolyte				
ionic conductivity	input	k_e	S/m	0.1–1.0
initial concentration	input	c_0	mol/m ³	1750–2250
Whole Cell				
applied C-rate	input	C_{rate}	1/h	0.33–3.0
convergence status	output	Boolean	-	N.A.
energy	output	E	Wh	N.A.
power	output	P	W	N.A.

In terms of application, the present paper does not directly focus on optimization. Instead, as also previously done in [1], it considers surrogates that can be used to estimate the performance of battery designs across a wide range of material characteristics and parameters. This applicative objective reduces the importance of using surrogate models that can output estimated values and their uncertainties, for instance by exploiting techniques such as Monte Carlo Dropout [30] and Variational Inference [31]. Consequently, in the rest of our treatment, we limit ourselves to non-probabilistic surrogates.

The methodology we introduce and detail in the sections below is based on the general framework put forward by [32,33], integrating and extending the Pseudo-2D surrogate proposed by [1] to evaluate the performance of composite surrogate models integrating both classification and regression. Through classification, a Machine Learning model evaluates the convergence of the simulator, while, through regression, the algorithms predict the output parameters of the Ragone plot [34] (energy and power, as per Figure 1). We extend the state of the art in [1] by showing the advantage of adopting respectively ensemble methods for convergence classification and structured regression for the estimation of the Ragone parameters, rather than simpler feed-forward networks for both tasks. Furthermore, from the methodological point of view, we apply a quantitative performance evaluation proce-

ture that is based on K-fold validation [35] rather than on simple train/test/validation splitting as in [1]. The accuracy of some of the ensemble surrogate models we introduce compares favorably to that of the state-of-the-art method introduced in [1]. Furthermore, the approach in the reference is extended by using a K-fold cross-validation method to evaluate if the model can generalize the quality of its results.

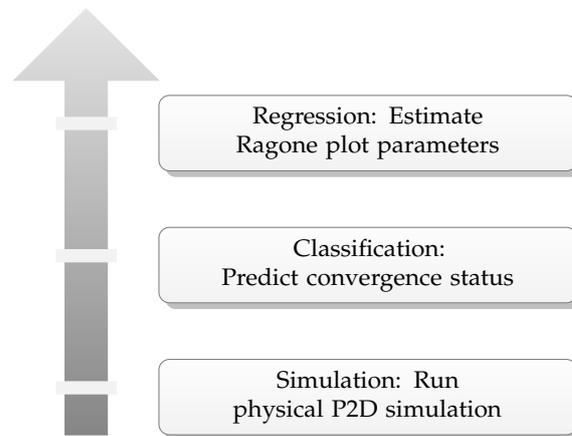


Figure 1. Global workflow. The methodology progresses from physical simulations to simulator convergence classification to the estimation by regression of a specific subset of Li-ion battery parameters of interest.

2. Methods

The construction and validation of the P2D-based surrogate model have involved several steps:

- Selection of the appropriate input and output parameters. In this regard, measurable electrode and electrolyte transport properties are selected as input parameters, while the parameters of the very-well known Ragone plot (energy and power) are selected as outputs to be used for battery characterization purposes, as shown in Table 1, where the minimum and maximum values for each of the parameters are also reported.
- Selection of electrochemical model and Design of Experiments. A proprietary P2D model is used to generate the data set.
- Development of surrogate models. Each surrogate is composed of three models: a binary simulation convergence classifier and two separate regression models that estimate integrated power and energy density values from the input parameters (see Figure 2).
- Model validation, typically carried out similarly as for the data-driven models.

2.1. Electrochemical Model and Design of Experiments

The electrochemical model used for building the data-set is expressed in terms of four conservation relations described in Smith et al. [36] by partial differential equations and their corresponding boundary conditions. The numerical approach considered for solving the system of equations is based on Finite Elements Methods (FEM) for space-discretization as implemented in the open-source FEniCS toolkit (<https://fenicsproject.org/>, accessed on 8 May 2021), and on Implicit Euler methods for time-discretization.

The data set is generated by running the model with the selected input variables and their ranges presented in Table 1.

As in Wu et al. [1], in this paper, we focus on the variables that are controllable during battery manufacturing. Among all the design variables, the electrode thickness, porosity, and particle radius are chosen because they can be controlled within the material or the electrode manufacturing steps. For example, electrode thickness and porosity can be experimentally adjusted during the calendaring step of the cell manufacturing process. Separator thickness and porosity ranges are defined considering commercially available components. In addition, the initial electrolyte concentration and ionic conductivity are

further important tunable variables and are considered as they play an important role in determining thick electrode performance limitations due to the dry-out of the electrolyte. When it comes to the electrolyte, two parameters are selected for the design of experiments. First, the ionic conductivity (k_e) is selected as it can be modulated by doping salts, acids, metals, alkali, etc. to the solvent matrix and the initial salt concentration for a similar reason (i.e., c_0). For the sake of simplicity, we have assumed that the ionic conductivity has no dependence on the electrolyte concentration. In further developments, we envision that this dependency will be taken into account. Finally, the applied C-rate is also selected as an input variable, as this parameter may vary from very low values to very high ones depending on the battery application. Constant-current discharge protocols are used as a baseline, with the aim of the characterization of the cell in terms of energy and power during discharge. Even so, the testing protocol could be adapted to other battery applications. The rest of the parameters needed to simulate battery performance using a P2D model are summarized in Table 2 and Figure 3.

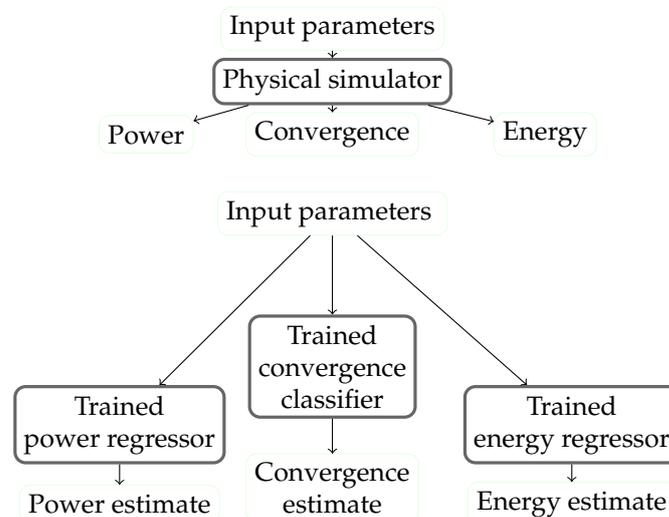


Figure 2. The general structure of the model. A simulator produces values for energy and power density integrals, together with a convergence flag indicating the successful execution of a run, up to a fixed end time. Run-stopping criteria corresponding to non-convergence involve, for instance, physically meaningless negative ion density values. A classifier and two regressors learned from the simulated data can be exploited to generate estimates. The classifier estimates whether a simulation will produce physically unrealizable conditions, and therefore stop without completing a run. The two regressors estimate integrated energy and power for converging runs.

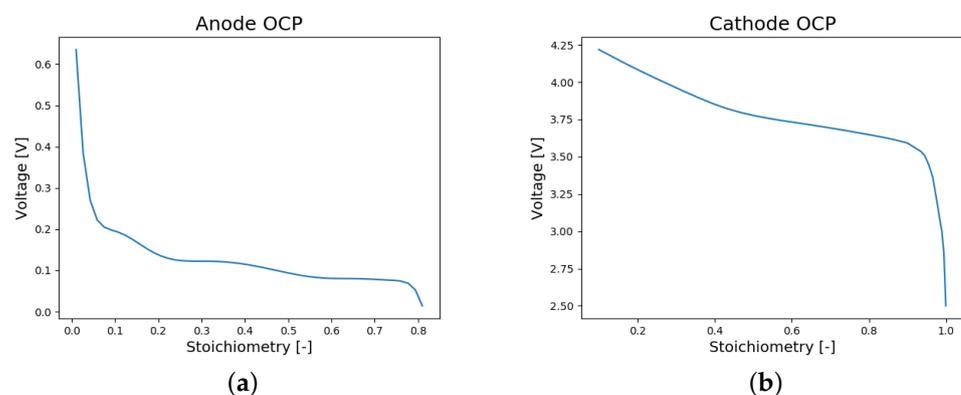


Figure 3. Open circuit potentials for the electrodes of the P2D model simulator. (a) Anode OCP curve: voltage response on each stoichiometry; (b) Cathode OCP curve: voltage response on each stoichiometry.

Table 2. Further parameters for the P2D model simulator.

Variable	Units	Value
Anode/Cathode		
active material volume fraction	%	56/70
effective electrical conductivity	S/m	3.0/2.0
maximum lithium concentration	mol/m ³	73,900.0/28,100.0
initial lithium concentration (100% SOC)	mol/m ³	59,859.0/3653.0
reference reaction rate coefficient	m ^{2.5} /mol ^{0.5} s	6.1 · 10 ⁻⁹ /1.0 · 10 ⁻¹⁰
reference solid diffusion coefficient	m ² /s	1.0 · 10 ⁻¹⁴ /1.0 · 10 ⁻¹³
open circuit potential (OCP)	V	Figure 3a/Figure 3b
Electrolyte		
transference number	-	0.4
diffusion coefficient	m ² /s	4.0 · 10 ⁻⁶
Whole Cell & Constants		
Area	m ²	2.16 · 10 ⁻⁴
Universal gas constant	J/K mol	8.31
Faraday constant	C/mol	96,485.34

Considering the previous inputs, we generate a simulated data set by running a P2D model implemented in Python. As previously mentioned, simulations consider a given set of input parameter values. They propagate a set of variables describing the electrochemical status of the battery up to a well-defined and fixed time limit. Upon both convergence and premature termination of the simulation, Ragone energy and power estimates are integrated as per

$$E = \int_0^{t_d} IV dt \quad [Wh]$$

$$P = \int_0^{t_d} IV/t_d dt \quad [W].$$

where t_d is the discharge time. The Ragone plot [34] is commonly used to illustrate the performance of energy storage devices, and is widely used to compare technologies catering to specific demands. Accordingly, in this work, we generate a Ragone plot for a wide range of electrode/cell design parameters aimed at a variety of future designs of electrochemical energy storage devices for different usage applications. Constant currents for discharge are used as a baseline since the Ragone plot is usually generated with a constant C-rate. Accordingly, we do not consider variable currents.

Reasons for failures in simulating up to the specified end time include reaching physically meaningless or non-realizable situations, such as negative values for ion concentrations. In these cases, the run will stop prematurely, producing only partial estimates for the above Ragone integrals, and appropriately setting a negative convergence status flag.

2.2. Surrogate Modeling

We generate a surrogate model to reproduce the results of the P2D model simulator. Once trained, this empirical model can approximate simulation results in a fraction of the running time of the physical simulator, keeping the approximation errors under control. The input parameters taken into account correspond to the inputs of simulations. As previously mentioned, the output variables include the convergence status of the simulation, and the Ragone plot variables corresponding to energy and power integrals. The proposed surrogate model structure is a composite one: the convergence status is modeled by a binary classifier (Section 2.2.1) and the energy and power output integrals are approximated by regressors (Section 2.2.4).

2.2.1. Simulation Convergence Classification

We consider that, in a Pseudo-2D electrochemical model, the completion status of a simulation can either be full (corresponding to propagation up to the intended complete

temporal extension) or partial (corresponding to early termination). This status can depend on several phenomena with varying degrees of complexity. Those phenomena range from the consumption of chemical species to variations in the separator/electrode interface resistance due to parasitic ion deposition and layering reactions. Modeling the range of phenomena involved requires adding flexibility to the surrogate model. To that end, we combine the prediction capabilities of committees of ML classifiers by using ensemble methods [37].

The key hypothesis is that, by exploiting extended ensembles of classifiers, we can implicitly learn about the different mechanisms that can stop the execution of the simulator. The resulting composite models can then better reproduce the varying degrees of complexity of the physical phenomena that lead a simulation to an early stop, thereby improving on state-of-the-art approaches such as the one in [1].

To evaluate this hypothesis, we compare a single feed-forward network (indicated by the ‘single’ label in subsequent tables and diagrams), corresponding to the state of the art in [1], to a set of ensemble binary classifiers based on different approaches: a voting ensemble composing a small number of tree-based classifiers (‘voting’), an efficient Gradient Boosting ensemble implementation (‘xgboost’), and a stacking ensemble of models including feed-forward networks (‘stacking’). The list of classifiers is in Table 3.

Table 3. Convergence classifier types.

Classification Model Type	Label
Feed-forward deep network	‘single’
Voting ensemble	‘voting’
Gradient boosting ensemble	‘xgboost’
Deep stacking ensemble	‘stacking’

We briefly introduce each model in the present section and detail implementation parameters in Section 3, dedicated to experiments.

2.2.2. Feed-Forward Deep Network

We start by establishing a baseline for the performance of convergence estimators, by considering a deep learning classifier as per the state of the art in [1], as a reference against which the performance of the rest of the models can be quantitatively evaluated. To that end, we define a fully connected network with a single hidden layer, considering a log-entropy loss function suited for classification [38], and an ADAMW optimizer [39] to address possible weight decay issues [40]. The limited structural complexity of the network corresponds to the results published in the literature.

2.2.3. Voting Ensemble of Classifiers

We build a first ensemble classifier by combining three simple models. The combination operates by basic voting among level-1 members, listed in Table 4. We describe them briefly. A pruned decision tree is one in which sections that are not critical to reducing an empirical loss function are iteratively eliminated. The pruning operates by considering the Minimum Description Length principle to reduce the risk of over-fitting [41]. The second type of level-1 model is a decision stump, a basic single-level decision tree [42]. The third type of level-1 model considered is a random forest, in itself an ensemble of decision trees [43] that outputs the mode of the output of the trees that compose it.

Table 4. Voting ensemble instance types.

Voting Ensemble Member Type	Instances
Pruned decision tree	1
Decision stump optimized by AdaBoost	1
Random forest	1

We then consider a Gradient Boosting implementation in XGBoost [44]. This specific model is selected on account of its flexibility and of the robustness of available implementations, which draw on efficient linear model solvers and effective tree learning algorithms. We complete the set of classifiers by taking into account a further model ensemble that includes instances of the feed-forward network. In this last case, the stacking is performed by a second-level classifier that takes as input the outputs of the first-level classification models that compose the committee. In this sense, the predictions of the first-level models are combined by an ensemble bagging/boosting combination mechanism that is based on a ‘stacking’ second-level classifier.

The models considered in the set are listed in Table 5. The three level-1 instances of the feed-forward network in the ensemble are similar in structure yet independent of one another, in the sense that they are independently trained on non-overlapping subsets of the data set.

Table 5. Stacking ensemble instance types.

Stacking Ensemble Member Type	Instances
Feed-forward network	3
Pruned decision tree	1
Decision stump optimized by AdaBoost	1
Random forest	1

Detailed descriptions of the models, together with the results obtained on a test data set, are reported and compared in Section 3.

2.2.4. Ragone Variable Regression

Regression is carried out separately on the integrated energy and power values generated by the simulation. The input data set is filtered considering only the samples that correspond to complete propagation of the model through time. The integrated energy and power values are approximated by regressors whose functional structure has a basis in the physical model. In particular, a polynomial structure is extended to a more general one, by considering an exponential transformation of the ionic conductivity of the electrolyte

$$\hat{V}(\{\vartheta_i\}_{i \in \Theta}) = \sum_{j=0}^N a_j \vartheta_i^j \quad \text{with } a_j \in \mathbb{R}, \quad \vartheta_i \in \{\theta_i, \exp(\theta_i)\}_{i \in \Theta} \quad (1)$$

so that an estimate \hat{V} for one of the integrated Ragone integrals (energy or power) is expressed in terms of a polynomial of order N of the Θ input parameters θ_i of the simulation as well as of their exponential $\exp(\theta_i)$. This inclusion of the exponentiation of the parameters of the simulation is motivated by the structural form of the Butler–Volmer equation, as well as by the solution implied by the conservation and diffusion equations in the model. Note that, for simplicity, neither temperature nor concentration dependence is assumed for parameters such as the ionic conductivity in the present contribution. We plan to address this relation in future extensions of this work.

3. Results

Simulated data consists of 11 input and 3 output attributes, as detailed in Table 1. The outputs are not temporally located, since they represent the integration of the state of a simulation that is successfully extended until a given time limit. Concerning the available data and parameter measurements, we observe that uncertainties from sensors and experimental measurements are not considered at this stage. We believe that sensor noises are more critical in BMS development, as they could have a big impact on the performance of the numerical algorithms used to estimate the state variables. They can be incorporated in treatments such as the present one by e.g., Variational Inference techniques, modifying the output layers of the network architectures considered and the loss functions specified.

3.1. Simulation Convergence Prediction by Classification

We partition the data set into separate training, testing and validation sets by 60/20/20% stratified sampling without replacement. Consequently, the distribution of the simulation convergence status is preserved in the generated sets. The data points from each set are shown in Figure 4. We use K-fold validation [35] as a strategy for the measurement of performance of the classification methods, with the K parameter set to 8. The input variables are normalized by a Z-score transformation in the preprocessing step. The randomized data are organized in batches of 32 samples to speed up the learning [45,46]. While the validation subset is fixed, the train/test split varies by $k \in \{1 \dots K\}$ in the K-fold validation procedure. The values of the input simulation parameters tend to be uniformly distributed in the input data set, with a maximum absolute value of the Pearson correlation between variables around 0.19.

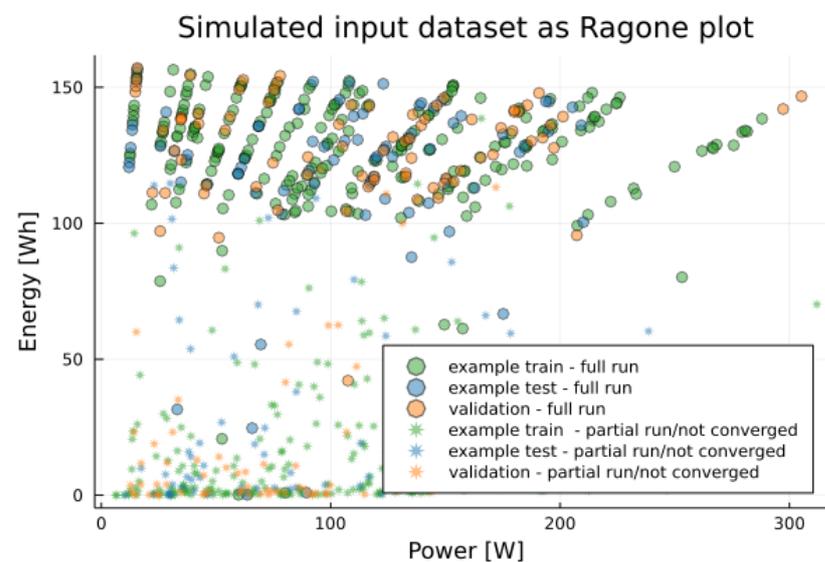


Figure 4. Simulated input data set as Ragone power/energy plot, describing an example partition of the data into model training, testing, and validation sets by stratified sampling without replacement. Asterisks are used to mark partial and non-converging runs. Samples in the 20% validation set are in orange. Samples from an example training/testing 60%/20% split are depicted in green and blue, respectively. While the validation subset is fixed, the train/test split varies by k in the K-fold validation procedure.

To establish baseline results, we first consider a state-of-the-art single fully connected feed-forward network as in [1]. The hyper-parameters for the surrogate are summarized in Table 6. The number of epochs is experimentally limited to 15, observing the convergence plots produced during the learning phase. As a consequence of the limited number of training epochs, training time is limited to five seconds on a single Intel i7-8550U CPU device running at 1.80 GHz. Though a GPU can be used as a learning device, the fast convergence of the learning procedure means that learning directly on the CPU is practically usable. The Confusion Matrix for the single feed-forward network for simulation convergence estimation as trained with the whole training/testing set and measured on the validation set is reported in Table 7. The results for this fully connected deep classifier are included and compared in Figure 5.

Table 6. Feed-forward convergence classifier configuration.

Batch size	32
Optimizer	ADAMW [47]
Number of epochs	15
Loss function	Mean Absolute Error
Hidden layer neurons	$\Theta \times 4$
Number of hidden layers	1
Activation functions	tanh
Dropout	0.2

Table 7. Confusion Matrices for a single run of the simulator convergence classification models: feed-forward network (a), voting ensemble (b), gradient boosting ensemble (c), deep stacking ensemble (d). The results for the single run may not be representative of the full results obtained via the K-fold validation in Figure 5.

(a)				
		True convergence status		Total
		Positive	Negative	
Convergence estimate	Positive	90.0	7.0	97.0
	Negative	9.0	94.0	103.0
	Total	99.0	101.0	200.0
(b)				
		True convergence status		Total
		Positive	Negative	
Convergence estimate	Positive	93.0	10.0	103.0
	Negative	6.0	91.0	97.0
	Total	99.0	101.0	200.0
(c)				
		True convergence status		Total
		Positive	Negative	
Convergence estimate	Positive	95.0	10.0	105.0
	Negative	4.0	91.0	95.0
	Total	99.0	101.0	200.0
(d)				
		True convergence status		Total
		Positive	Negative	
Convergence estimate	Positive	92.0	7.0	99.0
	Negative	7.0	94.0	101.0
	Total	99.0	101.0	200.0

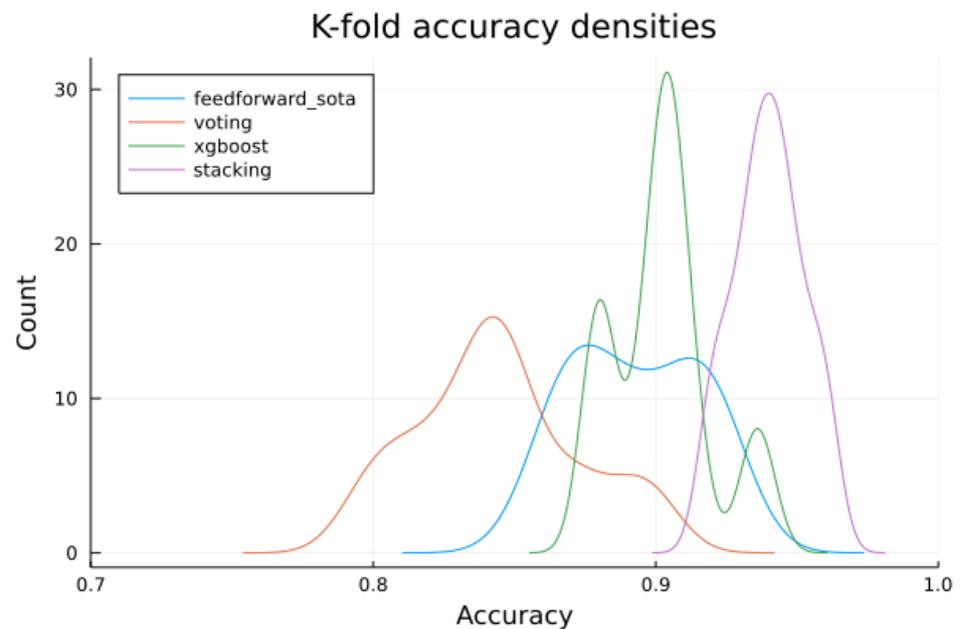


Figure 5. K-fold validation accuracy densities for the simulation convergence classifiers considered in the experiments. The curves indicate that a stacking deep ensemble (‘stacking’) can outperform a state of the art fully connected network (‘feedforward_sota’) as well as Gradient Boosting and voting ensembles (respectively ‘xgboost’ and ‘voting’).

A first ensemble model implements a voting committee with simple tree-based models as described in the above Section 2.2.1. The confusion matrix is reported in Table 7, while K-fold and validation results are summarized and compared in Figure 5.

After that, a gradient boosting ensemble classifier is configured with the hyper-parameter values in Table 8. At each training cycle, an ensemble of decision trees with a predefined maximum depth is improved through adding estimators for residuals to globally optimize a given objective function [48]. In our case, the maximum depth and the number of training cycles for the classifier are both set to two, while the objective taken into account for the loss function minimization is a binary logistic function. The step size shrinkage hyper-parameter used in the updating steps to prevent over-fitting is set to one, to indicate that no shrinking should take place. The Confusion Matrix for the gradient boosting simulation convergence estimator trained with the whole training/testing data and validated on the validation set is reported in Table 7. The resulting performance indicators for the gradient boosting ensemble classification model are included in Figure 5.

Table 8. Boosting ensemble classifier configuration.

Number of rounds	2
Maximum depth	2
Step size shrinkage	1
Objective function	binary logistic.

Finally, we consider an ensemble convergence model operating by a second-level classifier stacking the results obtained by first-level classifiers as per Section 2.2.1. The Confusion Matrix for the stacking ensemble simulation convergence model is reported in Table 7. The performance for the simulator convergence prediction of the stacking ensemble model is again reported in Figure 5.

3.2. Regression

Regression is carried out separately on the integrated energy and power values generated by the simulation. The input data set is filtered considering only the samples

that correspond to the successful propagation of the model through time. For both models, we consider 70% of the samples for training and the remaining 30% of the samples for testing.

3.2.1. Energy

The distribution of the energy integral is shown in Figure 6. Values lower than 80 J are considered as outliers and filtered out. By considering the correlations between the energy and the input variables, a first simple model is proposed to predict the values of the energy integral, \hat{E} , in the form of Equation (2):

$$\hat{E} = a \cdot L_p + b \cdot e^{c \cdot k_e} + d \quad (2)$$

L_p being the thickness of the positive electrode and k_e the ionic conductivity of the electrolyte. The resulting mean squared error and the R^2 score are $MSE = 32.83$ and $R^2 = 0.848$, respectively. In order to improve these results, a more complete model is introduced, as shown in Equation (3):

$$\hat{E} = a \cdot L_p + b \cdot e^{c \cdot k_e} + d \cdot L_n + e \cdot C_{rate} + f \quad (3)$$

where L_n represents the thickness of the negative electrode and C_{rate} is the applied current as a C-rate. Doing so results in a mean squared error of $MSE = 19.30$ and a $R^2 = 0.911$. Both models can be compared as in Figure 7a.

To ensure that the models have the desired accuracy and variance, some cross-validation is needed. To avoid the possibility of high bias in cases of limited data, a K-Fold cross-validation technique is used with $K = 30$, which means that the data set is split into 30 different groups. The value for K is chosen so that each train/test group of data samples is large enough to be statistically representative of the broader data set. The R^2 and the MSE are compared to those obtained when creating the model in Figure 7b,c.

From Figure 7b,c, it can be concluded that, in the Simple Model, around 20 out of 30 times, the R^2 values are higher than 0.8, and the same number of times, the MSE is below 25. For the Complete Model, around 20 of 30 times, the R^2 is higher than 0.9 and the MSE lower than 20. The models behave as expected and are validated by comparing these values with the ones obtained first in the models.

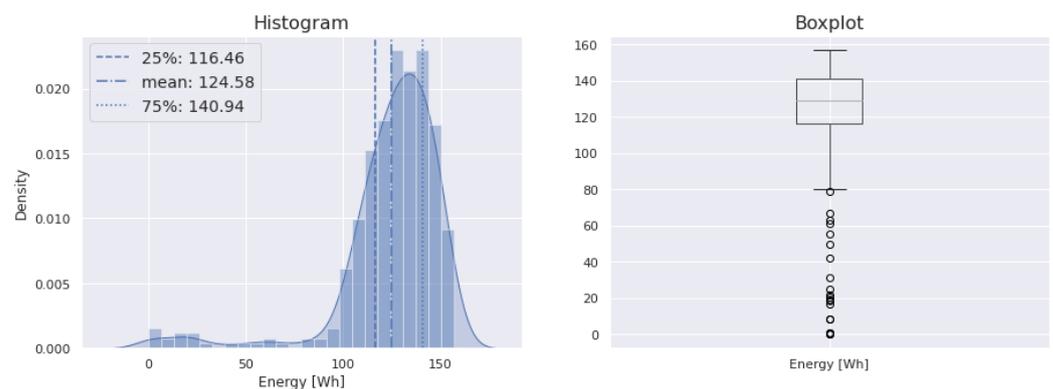


Figure 6. Distribution of the integrated energy variable (output of the simulation). Histogram (left) and box-plot (right) of the variable. As shown in the box-plot, values lower than 80 J are considered as outliers in this distribution.

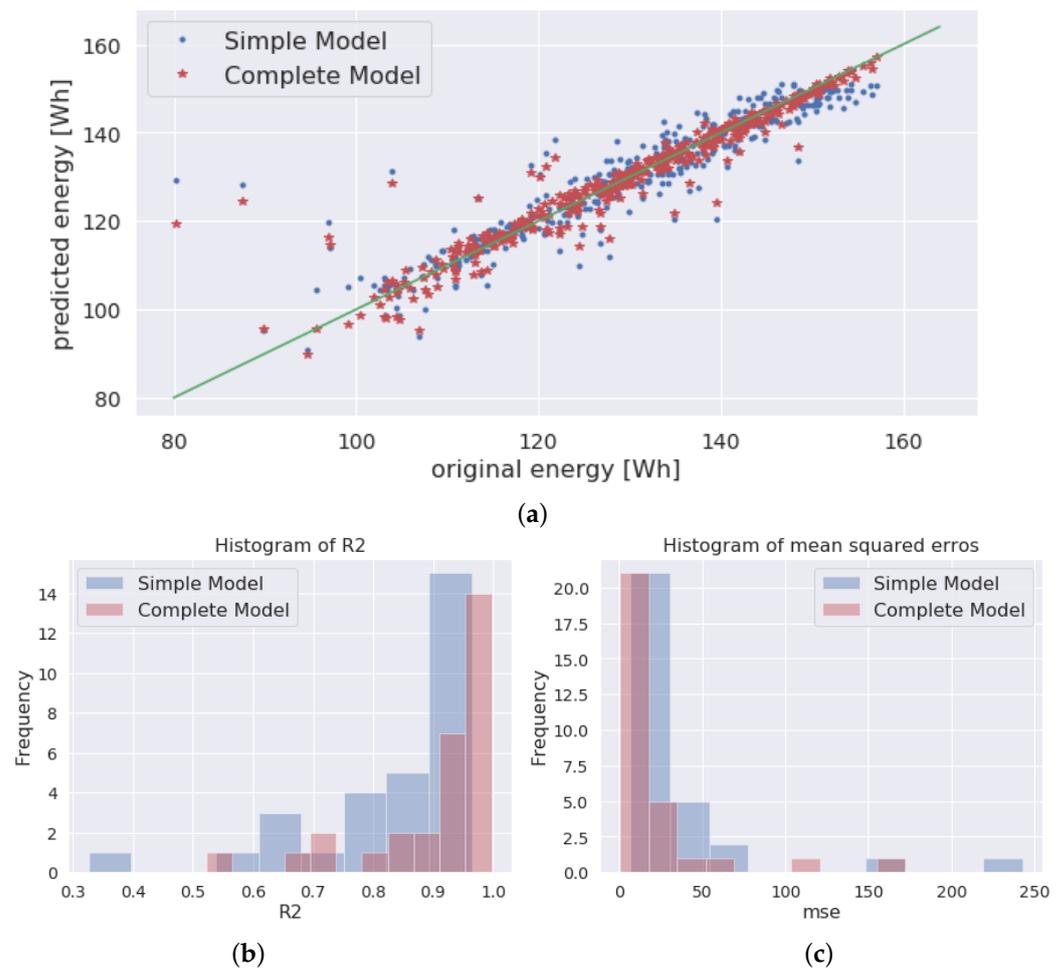


Figure 7. Results of the proposed regression models for the energy output integral (a) and histograms of the MSE (b) and R^2 (c) of 30 K-Folds performed to the model for the energy.

3.2.2. Power

The distribution of the power integral of simulation outputs is shown in Figure 8. Values higher than 285 W can be considered as outliers, and, therefore, the dataset is again filtered for these values. The power integral presents a significantly higher correlation with the applied current and the thickness of the positive electrode, compared to that of other input variables. Thus, a first simple linear regression is proposed to predict the values of the power integral, \hat{P} based on the following equation:

$$\hat{P} = a \cdot C_{rate} + b \cdot L_p + c \quad (4)$$

with C_{rate} the C-rate and L_p the thickness of the positive electrode. The results obtained by optimizing the constants in Equation (4) can be observed in Figure 9a. The resulting mean squared error and the R^2 score would be $MSE = 183.49$ and $R^2 = 0.988$, respectively.

As in the case of energy estimates, K-fold cross-validation is performed in the developed model, with $K = 30$ meaning that the dataset is again split into 30 different groups. R^2 and MSE values are compared to those obtained when creating the model. The results for the power model are shown in Figure 9b,c.

From Figure 9b,c, it can be concluded that, 20 out of 30 times, the R^2 values are above 0.985, and that, although there are instances for which the MSE 's values are high, 20 out of 30 times, this MSE is below 60.

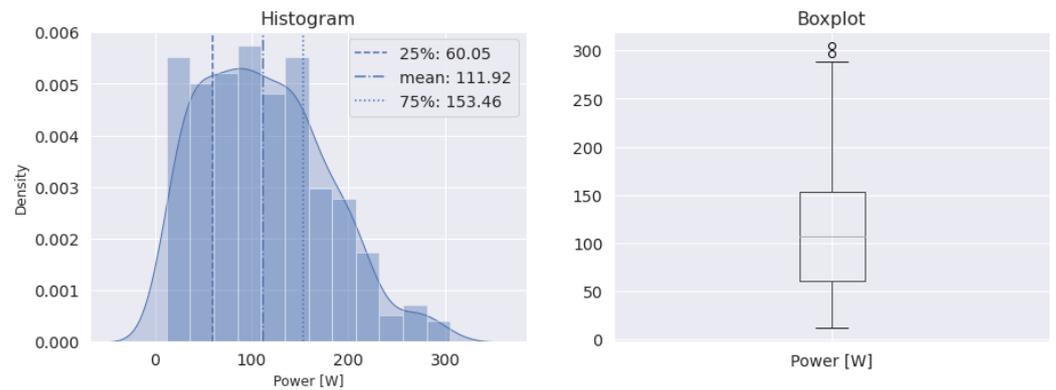


Figure 8. Distribution for the integrated power simulation output.

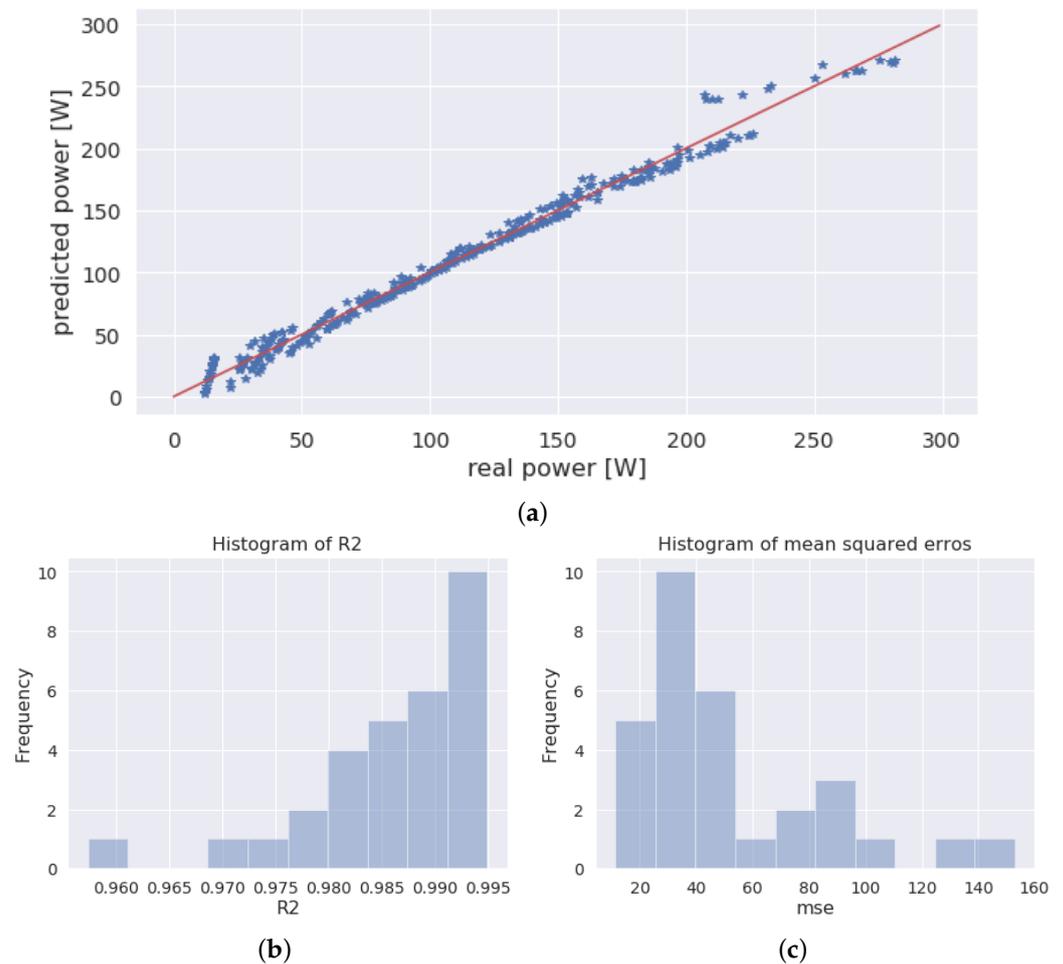


Figure 9. Results of the proposed regression model for the power integral (a) and histograms of the MSE (b) and R^2 (c) of 30 K-folds performed to the model for the power integral estimation.

4. Discussion

The results above indicate that an effective way to speed up the characterization of LIB designs involves complementing simulations with empirical, ML surrogate models learned from simulation results [33]. In terms of computational performance, P2D simulations using CIDETEC's proprietary code take, on average, 20 seconds for a 1C discharge. Once the surrogate models are trained, instead, predictions can be obtained in a few hundredths of a second. Deep ensembles and structured regression models represent an interesting direction for the development of efficient surrogate models capable of rapidly and accurately predicting Ragone plot variables such as energy and power integrals. Those results can

be integrated with design workflows and extended to other fields such as management and control. In the specific case of the prediction of early termination of the simulator by classification, the results seem to indicate that ensemble models can indeed outperform state-of-the-art surrogates based on deep learning fully connected networks, exploiting their flexibility in implicitly learning the multiple mechanisms that can stop the execution of the simulator.

On the other hand, it is perhaps worth mentioning that an important point that is sometimes overlooked [49] is that physical simulations are complemented rather than substituted by such data-based modeling: some of the computational costs are moved in time rather than eliminated, since some simulated data need to be generated first for the learning to take place.

5. Conclusions and Future Work

The present contribution has introduced and evaluated composite surrogate models for the prediction of the performance of Lithium-Ion Batteries which combine classifiers based on deep ensembles and structured regressors for the prediction of energy and power densities. We have compared a single state-of-the-art feed-forward network to three types of ensemble binary classifiers for the prediction of simulation convergence. The quantitative results obtained indicate that ensemble models can indeed outperform state of the art models based on a single deep network. Furthermore, we have quantitatively validated the applicability of structured regressors to the estimation of LIB energy and power. Future work to be considered includes analyzing feature and parameter sensitivities [50] and the use of ML explainability/interpretability methods to ensemble models from an electrochemical perspective. Furthermore, model and data uncertainty management [51,52] should also be analyzed: the quantification of the risk from different sources is valuable for the real-world applicability of the developed methodology.

Author Contributions: Physical simulation, P.C. and E.A.; classification surrogates: M.Q.; regression models, A.G., A.M.F.-T.; funding acquisition, I.G.O. and E.A; text revision: all. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data presented in this study are available on request from the corresponding author. The data are not publicly available due to their proprietary nature.

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

References

1. Wu, B.; Han, S.; Shin, K.G.; Lu, W. Application of artificial neural networks in design of lithium-ion batteries. *J. Power Sources* **2018**, *395*, 128–136. [[CrossRef](#)]
2. Bizeray, A.M.; Zhao, S.; Duncan, S.R.; Howey, D.A. Lithium-ion battery thermal-electrochemical model-based state estimation using orthogonal collocation and a modified extended Kalman filter. *J. Power Sources* **2015**, *296*, 400–412. [[CrossRef](#)]
3. Doyle, M.; Newman, J.; Gozdz, A.S.; Schmutz, C.N.; Tarascon, J.M. Comparison of modeling predictions with experimental data from plastic lithium ion cells. *J. Electrochem. Soc.* **1996**, *143*, 1890. [[CrossRef](#)]
4. Kumaresan, K.; Sikha, G.; White, R.E. Thermal model for a Li-ion cell. *J. Electrochem. Soc.* **2007**, *155*, A164. [[CrossRef](#)]
5. Doyle, M.; Fuller, T.F.; Newman, J. Modeling of galvanostatic charge and discharge of the lithium/polymer/insertion cell. *J. Electrochem. Soc.* **1993**, *140*, 1526. [[CrossRef](#)]
6. Newman, J.; Thomas-Alyea, K.E. *Electrochemical Systems*; John Wiley & Sons: Hoboken, NJ, USA, 2012.
7. Santhanagopalan, S.; Guo, Q.; Ramadass, P.; White, R.E. Review of models for predicting the cycling performance of lithium ion batteries. *J. Power Sources* **2006**, *156*, 620–628. [[CrossRef](#)]
8. Bernardi, D.; Pawlikowski, E.; Newman, J. A general energy balance for battery systems. *J. Electrochem. Soc.* **1985**, *132*, 5. [[CrossRef](#)]
9. Bandhauer, T.M.; Garimella, S.; Fuller, T.F. A critical review of thermal issues in lithium-ion batteries. *J. Electrochem. Soc.* **2011**, *158*, R1. [[CrossRef](#)]

10. Moura, S.J.; Bribiesca Argomedo, F.; Klein, R.; Mirtabatabaei, A.; Krstic, M. Battery State Estimation for a Single Particle Model with Electrolyte Dynamics. UC Berkeley: Energy, Controls, and Applications Lab. Available online: <https://escholarship.org/uc/item/5xt11817> (accessed on 7 July 2021).
11. Subramanian, V.R.; Ritter, J.A.; White, R.E. Approximate solutions for galvanostatic discharge of spherical particles i. constant diffusion coefficient. *J. Electrochem. Soc.* **2001**, *148*, E444. [[CrossRef](#)]
12. Majdabadi, M.M.; Farhad, S.; Farkhondeh, M.; Fraser, R.A.; Fowler, M. Simplified electrochemical multi-particle model for LiFePO₄ cathodes in lithium-ion batteries. *J. Power Sources* **2015**, *275*, 633–643. [[CrossRef](#)]
13. Dao, T.S.; Vyasrayani, C.P.; McPhee, J. Simplification and order reduction of lithium-ion battery model based on porous-electrode theory. *J. Power Sources* **2012**, *198*, 329–337. [[CrossRef](#)]
14. Smith, K.A.; Rahn, C.D.; Wang, C.Y. Control oriented 1D electrochemical model of lithium ion battery. *Energy Convers. Manag.* **2007**, *48*, 2565–2578. [[CrossRef](#)]
15. Jokar, A.; Rajabloo, B.; Désilets, M.; Lacroix, M. Review of simplified Pseudo-two-Dimensional models of lithium-ion batteries. *J. Power Sources* **2016**, *327*, 44–55. [[CrossRef](#)]
16. Nejad, S.; Gladwin, D.; Stone, D. A systematic review of lumped-parameter equivalent circuit models for real-time estimation of lithium-ion battery states. *J. Power Sources* **2016**, *316*, 183–196. [[CrossRef](#)]
17. Seaman, A.; Dao, T.S.; McPhee, J. A survey of mathematics-based equivalent-circuit and electrochemical battery models for hybrid and electric vehicle simulation. *J. Power Sources* **2014**, *256*, 410–423. [[CrossRef](#)]
18. Wang, Y.; Yang, D.; Zhang, X.; Chen, Z. Probability based remaining capacity estimation using data-driven and neural network model. *J. Power Sources* **2016**, *315*, 199–208. [[CrossRef](#)]
19. Wei, J.; Dong, G.; Chen, Z. Remaining useful life prediction and state of health diagnosis for lithium-ion batteries using particle filter and support vector regression. *IEEE Trans. Ind. Electron.* **2017**, *65*, 5634–5643. [[CrossRef](#)]
20. Geng, Z.; Wang, S.; Lacey, M.J.; Brandell, D.; Thiringer, T. Bridging physics-based and equivalent circuit models for lithium-ion batteries. *Electrochim. Acta* **2021**, *372*, 137829. [[CrossRef](#)]
21. Li, Y.; Vilathgamuwa, M.; Farrell, T.; Choi, S.S.; Tran, N.T.; Teague, J. A physics-based distributed-parameter equivalent circuit model for lithium-ion batteries. *Electrochim. Acta* **2019**, *299*, 451–469. [[CrossRef](#)]
22. Jones, D.R. A taxonomy of global optimization methods based on response surfaces. *J. Glob. Optim.* **2001**, *21*, 345–383. [[CrossRef](#)]
23. Forrester, A.I.; Keane, A.J. Recent advances in surrogate-based optimization. *Prog. Aerosp. Sci.* **2009**, *45*, 50–79. [[CrossRef](#)]
24. Zhang, Y.; Han, Z.H.; Zhang, K.S. Variable-fidelity expected improvement method for efficient global optimization of expensive functions. *Struct. Multidiscip. Optim.* **2018**, *58*, 1431–1451. [[CrossRef](#)]
25. Regis, R.G.; Shoemaker, C.A. Combining radial basis function surrogates and dynamic coordinate search in high-dimensional expensive black-box optimization. *Eng. Optim.* **2013**, *45*, 529–555. [[CrossRef](#)]
26. Krityakierne, T.; Akhtar, T.; Shoemaker, C.A. SOP: Parallel surrogate global optimization with Pareto center selection for computationally expensive single objective problems. *J. Glob. Optim.* **2016**, *66*, 417–437. [[CrossRef](#)]
27. Liu, K.; Li, Y.; Hu, X.; Lucu, M.; Widanage, W.D. Gaussian process regression with automatic relevance determination kernel for calendar aging prediction of lithium-ion batteries. *IEEE Trans. Ind. Inform.* **2019**, *16*, 3767–3777. [[CrossRef](#)]
28. Liu, K.; Hu, X.; Wei, Z.; Li, Y.; Jiang, Y. Modified Gaussian process regression models for cyclic capacity prediction of lithium-ion batteries. *IEEE Trans. Transp. Electrif.* **2019**, *5*, 1225–1236. [[CrossRef](#)]
29. Liu, K.; Shang, Y.; Ouyang, Q.; Widanage, W.D. A data-driven approach with uncertainty quantification for predicting future capacities and remaining useful life of lithium-ion battery. *IEEE Trans. Ind. Electron.* **2020**. [[CrossRef](#)]
30. Gal, Y.; Ghahramani, Z. Dropout as a bayesian approximation: Representing model uncertainty in deep learning. In Proceedings of the International Conference on Machine Learning, PMLR, New York, NY, USA, 20–22 June 2016; pp. 1050–1059.
31. Graves, A. Practical variational inference for neural networks. In *Advances in Neural Information Processing Systems*; Neural Information Processing Systems Foundation, Inc. (NIPS): Granada, Spain, 12–15 December 2011; Volume 24, pp. 2348–2356.
32. Ascione, F.; Bianco, N.; De Masi, R.F.; De Stasio, C.; Mauro, G.M.; Vanoli, G.P. Artificial neural networks for predicting the energy behavior of a building category: A powerful tool for cost-optimal analysis. In *Cost-Effective Energy Efficient Building Retrofitting*; Elsevier: Amsterdam, The Netherlands, 2017; pp. 305–340.
33. Ascione, F.; Bianco, N.; De Stasio, C.; Mauro, G.M.; Vanoli, G.P. CASA, cost-optimal analysis by multi-objective optimisation and artificial neural networks: A new framework for the robust assessment of cost-optimal energy retrofit, feasible for any building. *Energy Build.* **2017**, *146*, 200–219. [[CrossRef](#)]
34. Ragone, D.V. *Review of Battery Systems for Electrically Vehicles*; SAE Technical Paper Series; SAE International: Warrendale, PA, USA, 1968.
35. Fushiki, T. Estimation of prediction error by using K-fold cross-validation. *Stat. Comput.* **2011**, *21*, 137–146. [[CrossRef](#)]
36. Smith, K.; Wang, C.-Y. Solid-state diffusion limitations on pulse operation of a lithium ion cell for hybrid electric vehicles. *J. Power Sources* **2006**, *161*, 628. [[CrossRef](#)]
37. Sagi, O.; Rokach, L. Ensemble learning: A survey. *Wiley Interdiscip. Rev. Data Min. Knowl. Discov.* **2018**, *8*, e1249. [[CrossRef](#)]
38. MacKay, D.J.; Mac Kay, D.J. *Information Theory, Inference and Learning Algorithms*; Cambridge University Press: Cambridge, UK, 2003.
39. Loshchilov, I.; Hutter, F. Decoupled weight decay regularization. *arXiv* **2017**, arXiv:1711.05101.
40. Ruder, S. An overview of gradient descent optimization algorithms. *arXiv* **2016**, arXiv:1609.04747.

41. Breslow, L.A.; Aha, D.W. Simplifying decision trees: A survey. *Knowl. Eng. Rev.* **1997**, *12*, 1–40. [[CrossRef](#)]
42. Iba, W.; Langley, P. Induction of one-level decision trees. In *Machine Learning Proceedings 1992*; Elsevier: Amsterdam, The Netherlands, 1992; pp. 233–240.
43. Friedman, J.; Hastie, T.; Tibshirani, R. *The Elements of Statistical Learning*; Springer Series in Statistics New York; Springer: New York, NY, USA, 2001; Volume 1.
44. Chen, T.; Guestrin, C. XGBoost: A Scalable Tree Boosting System. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*; KDD '16; ACM: New York, NY, USA, 2016; pp. 785–794.
45. Keskar, N.S.; Mudigere, D.; Nocedal, J.; Smelyanskiy, M.; Tang, P.T.P. On large-batch training for deep learning: Generalization gap and sharp minima. *arXiv* **2016**, arXiv:1609.04836.
46. Masters, D.; Luschi, C. Revisiting small batch training for deep neural networks. *arXiv* **2018**, arXiv:1804.07612.
47. Lydia, A.; Francis, S. Adagrad—An optimizer for stochastic gradient descent. *Int. J. Inf. Comput. Sci.* **2019**, *6*, 566–568.
48. Friedman, J.H. Stochastic gradient boosting. *Comput. Stat. Data Anal.* **2002**, *38*, 367–378. [[CrossRef](#)]
49. Breen, P.G.; Foley, C.N.; Boekholt, T.; Zwart, S.P. Newton vs. the machine: Solving the chaotic three-body problem using deep neural networks. *arXiv* **2019**, arXiv:1910.07291.
50. Liu, K.; Hu, X.; Zhou, H.; Tong, L.; Widanalage, D.; Marco, J. Feature analyses and modelling of lithium-ion batteries manufacturing based on random forest classification. *IEEE/ASME Trans. Mech.* **2021**. [[CrossRef](#)]
51. Kläs, M.; Vollmer, A.M. Uncertainty in machine learning applications: A practice-driven classification of uncertainty. In *International Conference on Computer Safety, Reliability, and Security*; Springer: Berlin/Heidelberg, Germany, 2018; pp. 431–438.
52. Hüllermeier, E.; Waegeman, W. Aleatoric and epistemic uncertainty in machine learning: An introduction to concepts and methods. *Mach. Learn.* **2021**, *110*, 457–506. [[CrossRef](#)]