Journal of Power Sources 610 (2024) 234668

Contents lists available at ScienceDirect

Journal of Power Sources

journal homepage: www.elsevier.com/locate/jpowsour

Improving diagnostics and prognostics of implantable cardioverter defibrillator batteries with interpretable machine learning models

Giacomo Galuppini [a,c], Qiaohao Liang [a], Prabhakar A. Tamirisa [b], Jeffrey A. Lemmerman [b], Melani G. Sullivan [b], Michael J.M. Mazack [b], Partha M. Gomadam [b], Martin Z. Bazant [a], Richard D. Braatz [a,*]

[a] Massachusetts Institute of Technology, Cambridge, MA 02139, USA
[b] Medtronic Energy and Component Center, Brooklyn Center, MN, USA
[c] University of Pavia, Pavia, PV, Italy

HIGHLIGHTS

- ICD battery reliability is ensured with life-test experiments spanning multiple years.
- ML provides accurate prediction of life-test experiments based on production data.
- Interpretable ML fosters the development of battery design and physics-based models.
- Approach is validated on 21 datasets, analysed for the first time in the literature.

ARTICLE INFO

Keywords:
Batteries
Defibrillators
Machine learning
Generalized additive models
Diagnostics
Prognostics

ABSTRACT

Medtronic Implantable Cardioverter Defibrillators (ICDs) and Cardiac Resynchronization Therapy Defibrillators (CRT-Ds) rely on high-energy density, lithium batteries, which are manufactured with a special lithium/carbon monofluoride (CF_x)–silver vanadium oxide (SVO) hybrid cathode design. Consistently high battery performance is crucial for this application, since poor performance may result in ineffective patient treatment, whereas early replacement may involve surgery and increase in maintenance costs. To evaluate performance, batteries are tested, both at the time of production and post-production, through periodic sampling carried out over multiple years. This considerable amount of experimental data is exploited for the first time in this work to develop a data-driven, machine learning approach, relying on Generalized Additive Models (GAMs) to predict battery performance, based on production data. GAMs combine prediction accuracy, which enables evaluation of battery performance immediately after production, with model interpretability, which provides clues on how to further improve battery design and production. Model interpretation allows to identify key features from the battery production data that offer physical insights to support future battery development, and foster the development of physics-based model for hybrid cathode batteries. The proposed approach is validated on 21 different datasets, targeting several performance-related features, and delivers consistently high prediction accuracy on test data.

1. Introduction

Lithium-ion batteries are the leading technology for energy storage for a wide range of applications, due to having high energy densities, long lifetimes, and low production cost [1,2]. For many applications, access to early, accurate predictions of primary performance and Remaining Useful Life (RUL) of the battery unlocks new opportunities in battery production, use, and optimization. These include acceleration of the cell development cycle, rapid validation of new manufacturing processes, and process optimization over large parameter spaces [3,4]. End-users can also benefit from accurate performance and RUL predictions to reduce battery waste [2,3,5,6]. Specifically, this work focuses on predictive modelling of lithium/carbon monofluoride (CF_x) – silver vanadium oxide (SVO) hybrid cathode technology batteries that power Implantable Cardioverter Defibrillators (ICDs) and Cardiac Resynchronization Therapy – Defibrillators (CRT-Ds) [7]. The CF_x–SVO battery design allows for high energy density, to ensure longevity and sufficient rate capability to provide high power pulses for treating abnormal heart rhythms. In this scenario, poor battery performance may result

* Corresponding author.
E-mail address: braatz@mit.edu (R.D. Braatz).

https://doi.org/10.1016/j.jpowsour.2024.234668
Received 16 March 2024; Received in revised form 24 April 2024; Accepted 3 May 2024
Available online 17 May 2024

in ineffective patient treatment. Therefore, an accurate evaluation of the battery performance is crucial to ensure reliability of ICDs and CRT-Ds [8,9]. To this end, the batteries are tested, both at the time of production ("burn-in experiments") and post-production through periodic sampling of production batteries ("life-test experiments"). For this class of batteries, past modelling efforts were focused on the prediction of the voltage-capacity behaviour with resistance-based models,including time- and temperature-dependence of resistance [8–10]. These models offer a fairly general description of the battery dynamics, but their predictive accuracy is still insufficient to replace the actual life-test experiments. When compared to experimental data, the models overpredict voltage for high discharge rates, and underestimates performance for long horizons (over five years). Both effects are particularly undesirable in this application. On one hand, voltage overprediction for high discharge rates may cause the powered device to deliver weak pulses, which may not be sufficient to effectively restore a normal heart rhythm. On the other hand, performance underestimation over long horizons may lead to an early replacement of the battery powering the device. In case of subcutaneous ICDs, this operation may involve surgery and lead to a considerable increase in the maintenance costs. Moreover, the available resistance-based models only offer population-level predictions, whereas ([8] used Monte Carlo simulations to understand impact of production variability on performance, but only focused on a few parameters of the first-principles model), and may not be straightforwardly specialized to provide battery-specific predictions.

As the development of predictive algorithms is an active field of research, several modelling methodologies have been applied to this problem in the scientific literature. First-principles models are available for describing battery dynamics associated with reaction, diffusion, and conduction [11–13], but the modelling of degradation mechanisms is an open research area [14–16], including thermal effects, Solid-Electrolyte Interface (SEI) growth, lithium plating, active material loss, and impedance increase [17–19]. Moreover, accurate first-principles models typically have poor parameter identifiability [20,21]. On the other hand, data-driven modelling has gained popularity in recent years, despite difficulties associated with the generation of informative datasets (battery lifetime may span from months to years) [22–24]. The value of data-driven approaches for battery diagnosis and prognosis from small to large datasets of lithium-ion batteries has been demonstrated [25–32], including for construction of *interpretable* models [26,33–36]. The ability to select relevant predictors among a wide set of candidates, and provide insight into the contribution of each selected predictor on the model output, makes this class of models particularly interesting as this information can be exploited by manufacturers to refine battery production processes [37]. For instance, Elastic Net (EN) [38] has been used to build an interpretable model for predicting the remaining battery lifetime, based on domain-specific features constructed from the raw data [26], and to predict battery capacity based on voltage relaxation data [39]. The performance of other common regularized regression and latent variable algorithms is assessed in [33], which also proposes a new set of capacity-based features. The analysis is extended in [36] to improve prediction accuracy under extremely fast-charging conditions. EN and Support Vector Regression models are developed in [39], based on statistical features obtained from the voltage relaxation curve. Machine learning models have also been successfully employed to identify the dominant ageing mechanism [40], and for State of Charge and State of Health prediction [32]. For the CF_x–SVO batteries analysed in this paper, the limitations of the available resistance-based models, and the availability of a considerable amount of experimental data resulting from over 10 years of tests, motivates a shift to the data-driven modelling paradigm. Preliminary modelling efforts have been carried out by relying on several machine-learning algorithms, including Algebraic Learning Via Elastic Net (ALVEN) [41,42], and Group Sparse Neural Networks (GSNNs) [43]. Both methods combine model training and feature selection. In particular, ALVEN is a nonlinear regression model learning methodology which is specifically designed to build interpretable, accurate, and robust models from manufacturing data. However, the predictive performance of ALVEN was not sufficient for this application. GSNNs address the task of simultaneously optimizing the weights of a neural network, the number of neurons for each hidden layer, and the subset of active input features. However, GSNNs also delivered conservative performance predictions, and did not provide interpretability of the results. The machine learning approach adopted in this work employs then Generalized Additive Models (GAMs) [38,44], which we show are able to accurately learn and predict life-test data from burn-in data. Their additive structure further allows high interpretability, by distinguishing the contribution that each model input makes on the model output. Due to the huge number of highly correlated, candidate predictors, a feature selection strategy based on the Maximum Relevance Minimum Redundancy (MRMR) algorithm is employed [45,46]. Feature selection via MRMR, and GAMs training, are combined with a nested cross-validation approach, enabling a rigorous selection of the best subset of predictors, and best hyperparameter values for the GAMs [38,42]. The results obtained for 21 different datasets highlight that GAMs can achieve very high prediction accuracy, evaluated as Mean Squared Error (MSE) and coefficient of determination R^2 [38], by relying on a small (< 10) subset of all available predictors (≈ 300).

2. Methods

This section provides information about data and data preprocessing, as well as about the machine learning algorithms used in this work.

2.1. Dataset description

Batteries powering ICDs require high energy density to ensure longevity, and sufficient rate capability to provide high power pulses for treating abnormal heart rhythms. Lithium/carbon monofluoride (CF_x)-silver vanadium oxide (SVO) multi-active material porous electrode batteries [47,48] leverage the excellent energy density of CF_x and power density of SVO through the use of a CF_x-SVO hybrid cathode. This battery design allows years of correct functionality of ICDs under low-rate background monitoring (~ 11 μA, equivalent to 8×10^{-6} C or 1.3×10^{-7} A cm^{-2}), and is able to provide high-rate defibrillation pulses ($\sim$3.5 A, equivalent to ~ 2.5C or ~ 0.04 A cm^{-2}) on demand.

Medtronic internally manufactures the CF_x-SVO hybrid cathode batteries used to power ICDs and CRT-Ds. The available data comprise electrical background, pulse discharge data, and manufacturing data, including electrical burn-in data, of batteries manufactured since 2012. Production samples are collected for long-term discharge and pulse performance characterization, after manufacturing and burn-in tests are completed in the factory. Based on measurements carried out during manufacturing, more than 50 candidate predictors are available, including cathode thickness (at various sampling locations along cathode length), mass of cathode, weight of battery plus electrolyte (after electrolyte filling), and electrolyte weight. Moreover, timestamps are assigned to starting and termination of a number of production phases, and are made available as candidate predictors.

During burn-in, a battery is discharged at a constant current (drain phase) of 10 mA to remove the specified amount of capacity ($\approx$20 mAh). Then, it recovers at open circuit, and eventually goes through a series of four high-current pulses (1.16 A) with a short, open-circuit rest between each pulse. After pulsing, a battery recovers again at open circuit. An example of a burn-in test is in Fig. 1(b). Based on measurements carried out during burn-in, more than 200 additional candidate predictors are available, including maximum/minimum/initial/final voltages, currents, temperatures and resistances recorded during each of the aforementioned burn-in phases. Statistics (average and standard

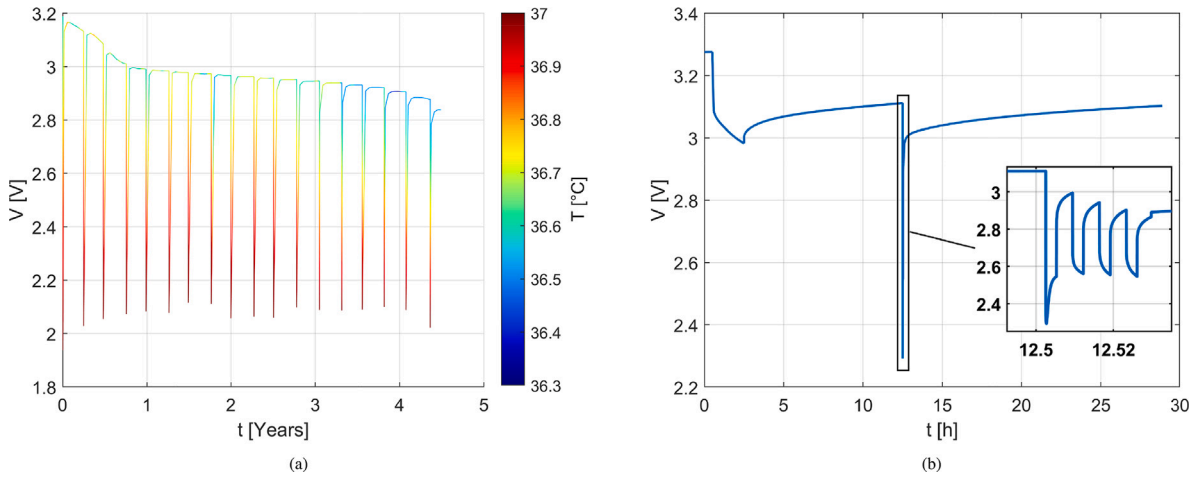

Fig. 1. Examples of voltage V [V] during (a) life-test and (b) burn-in experiments. For life-test, the line colour represents cell temperature T [°C].

deviations) for voltages, currents, temperatures and resistances are computed at the lot level, for a number of burn-in measurements. Finally, timestamp values are also available for several burn-in phases.

In order to collect life-test data, batteries are tested at 37 °C under a background resistive load of 270 kΩ, and subjected to constant energy pulses at a quarterly interval under a resistive load of 0.65 Ω. Depending on the battery, 3 different pulse regimes are applied: either 1 pulse of 64 J, 4 pulses of 64 J each, or 1 pulse of 32 J. Pulses are applied in a train of four pulses with 10 s between pulses in the 4 pulse test. Pulses are continued until either the desired energy is delivered, or a 60-s pulse time is reached, or the pulse load voltage drops below 1 V. Additionally, the cell temperature during background discharge is recorded and made available as a candidate predictor during the modelling phase. An example of life-test experiment is in Fig. 1(a).

In this work, we develop predictive models for four life-test quantities:

- *Pulse Average Voltage* (PulseAvgV), the battery average voltage during a pulse.
- *Pulse Duration* (PulseDuration), the duration of a pulse, expressed in seconds.
- *Pulse Minimum Voltage* (PulseMinV), the battery minimum voltage during a pulse.
- *Smoothed Background Voltage* (SmoothedBckgndV), the battery voltage during background discharge Due to the measurement protocol, raw background voltage data are affected by spike noise. A preliminary smoothing operation is carried out by means of robust local regression [38,49].

For each of the above quantities, and based on the pulsing regime, a total of 21 datasets are constructed (see Fig. 2). In the reminder of this article, each dataset name (PulseAvgV, PulseDuration, PulseMinV or SmoothedBckgndV) is completed by a code identifying the pulsing regime and, if required, the pulse number. The encoding convention is Datasetname*xx*JP*yy*, where *xx* denotes the energy of pulses, expressed in J (32 or 64), and *yy* denotes the number of pulses in each pulse train (1 or 4). In addition, for pulse-related quantities, the pulse number is also included with the convention Datasetname*xx*JP*yy*_np*zz*, where *zz* in the pulse number in the pulse train (1 for 1 pulse trains, 1 to 4 for 4 pulse trains). The scatter plots in Fig. 2 also highlight a different variability in the four life-test quantities. In particular, SmoothedBckgndV appears to be the least dispersed variable (data point stay within tenths of mV around the average SmoothedBckgndV vs. Capacity trend), regardless of the life-test experimental conditions. PulseAvgV and PulseMinV show a higher dispersion, in the order of hundreds of mV, whereas PulseDuration shows a dispersion in the order of hundreds of milliseconds. Variability arises from tolerances in the production process, as well as from variability in the raw materials used in the manufacturing process. The machine learning models developed in this work aim to improve the prediction accuracy over the simple, average behaviour (which can be obtained via e.g., local regression or smoothing algorithms [38], relying on capacity as the sole predictor), to capture – and possibly explain – variability in the target variable. Further details about the size of the available datasets are reported in Table 1.

2.2. Machine learning approach

This section describes the overall modelling methodology adopted in this work for the development of interpretable machine learning models based on GAMs. The methodology involves a preliminary data preprocessing step, including removal of unreliable or incomplete records, and removal of constant or incomplete features. Then, a set of GAMs is trained using a different number of relevant predictors, which are selected using the MRMR. Based on cross-validation results, the best GAM is selected and its performance assessed on a set of fresh data, to avoid overoptimistic evaluation of GAM's predictive capabilities. The flowchart in Fig. 3 summarizes the procedure for the identification of GAMs. First, the dataset $\mathcal{D}$ is split into training set $\mathcal{D}_{tr}$ and test set $\mathcal{D}_{te}$ using a *Grouped, Hold 15% Out* cross-validation approach. The training set $\mathcal{D}_{tr}$ is used to optimize parameters and hyperparameters of the model (including the predictor subset), and the test set $\mathcal{D}_{te}$ is used to evaluate the predictive performance of the model on fresh data. A *Grouped, Repeated Hold 30% Out* cross-validation approach is used to determine the best predictor subset: For a number p of predictors, ranging from 1 to $N_p = 10$, GAM training is repeated $N_r = 20$ times. During each repetition, (with active training set is $\mathcal{D}_{tr,tr,n}$, and validation set $\mathcal{D}_{tr,val,n}$, with $n = 1, \dots, N_r$), predictors are first ranked according to the MRMR algorithm, and the first p are made available as GAM inputs. GAM hyperparameters are optimized using Bayesian Optimization (BO) [50] and *Grouped K Fold* cross-validation approach with $K = 5$ folds ($\mathcal{D}_{tr,val,n,k}$, with $k = 1, \dots, K$). The best hyperparameter values are chosen according to the minimum validation MSE, averaged over the 5 folds. Once all training repetitions are carried out, the best predictor subset is then chosen using the OSE rule, based on the validation MSE, averaged over the 20 repetitions. A final GAM training – using the best predictor subset, and the best hyperparameters values – is carried out on $\mathcal{D}_{tr}$. The final model is tested on $\mathcal{D}_{te}$. For the final GAM, GOF is quantified using MSE and R^2.

Further details about data preprocessing, MRMR algorithm, GAMs, cross-validation, and goodness-of-fit (GOF) scores are in the reminder of this section.

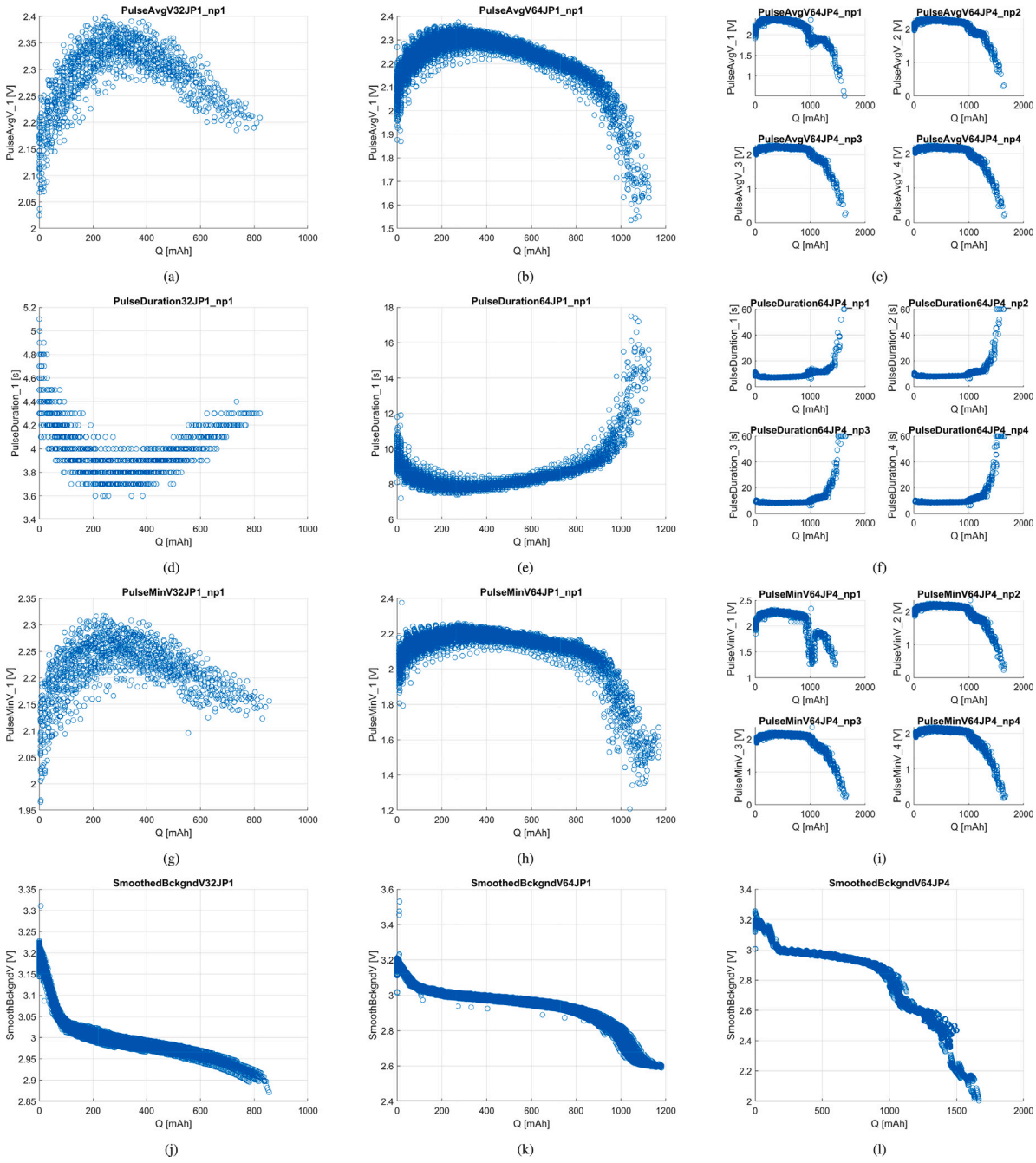

Fig. 2. Scatter plots of voltage vs. capacity (a–c, g–l) and duration vs. capacity for (d–f) of the life test-related datasets explored in this work.

2.3. Dataset preprocessing

Several steps are taken to preprocess each dataset:

- *Conversion of timestamp features to numeric values*. The dataset includes several timestamps associated to various production phases. Timestamps are recorded in Date/Time format (day-month-year hour:minute:second) and are converted to numeric values, expressed in minutes. For each record, the FILL_Weight_Post-Weight_1DateTime timestamp is chosen as baseline, and converted timestamp features represent the elapsed time from such value.
- *Removal of unreliable records*. Records considered as unreliable – due to known production issues and/or measurement errors – are removed from the dataset.
- *Removal of outliers*. Based on physical considerations, records including outlier values are removed from the dataset.
- *Removal of constant features*. Features with variance $\leq 10^{-10}$ are considered constant and are removed from the dataset (a single constant term is included in the modelling phase, if necessary).
- *Removal of incomplete features*. Features with $> 25\%$ of missing data are removed from the dataset.
- *Removal of incomplete records*. Records with missing data are removed from the dataset.

2.4. Minimum redundancy maximum relevance algorithm

The MRMR algorithm [45,46,51] finds an optimal set of features that is mutually and maximally dissimilar and can represent the response variable effectively. Its goal is to find an optimal set X of

Table 1
Number of available cells and data points for each dataset (after preprocessing).

Dataset	Number of cells	Number of data points
PulseAvgV32JP1_np1_Prediction_v3_Pyc	65	1143
PulseAvgV64JP1_np1_Prediction_v3_Pyc	394	7130
PulseAvgV64JP4_np1_Prediction_v3_Pyc	126	2053
PulseAvgV64JP4_np2_Prediction_v3_Pyc	126	2094
PulseAvgV64JP4_np3_Prediction_v3_Pyc	126	2094
PulseAvgV64JP4_np4_Prediction_v3_Pyc	126	2094
PulseDuration32JP1_np1_Prediction_v3_Pyc	65	1143
PulseDuration64JP1_np1_Prediction_v3_Pyc	394	7131
PulseDuration64JP4_np1_Prediction_v3_Pyc	126	2098
PulseDuration64JP4_np2_Prediction_v3_Pyc	126	2098
PulseDuration64JP4_np3_Prediction_v3_Pyc	126	2098
PulseDuration64JP4_np4_Prediction_v3_Pyc	126	2098
PulseMinV32JP1_np1_Prediction_v3_Pyc	66	1209
PulseMinV64JP1_np1_Prediction_v3_Pyc	324	6839
PulseMinV64JP4_np1_Prediction_v3_pyc	74	1367
PulseMinV64JP4_np2_Prediction_v3_Pyc	132	2345
PulseMinV64JP4_np3_Prediction_v3_Pyc	132	2345
PulseMinV64JP4_np4_Prediction_v3_Pyc	132	2344
SmoothedBckgndV32JP1_Prediction_v3_Pyc_Lifetest	84	9646
SmoothedBckgndV64JP1_Prediction_v3_Pyc_Lifetest	208	39 396
SmoothedBckgndV64JP4_Prediction_v3_Pyc_Lifetest	35	6303

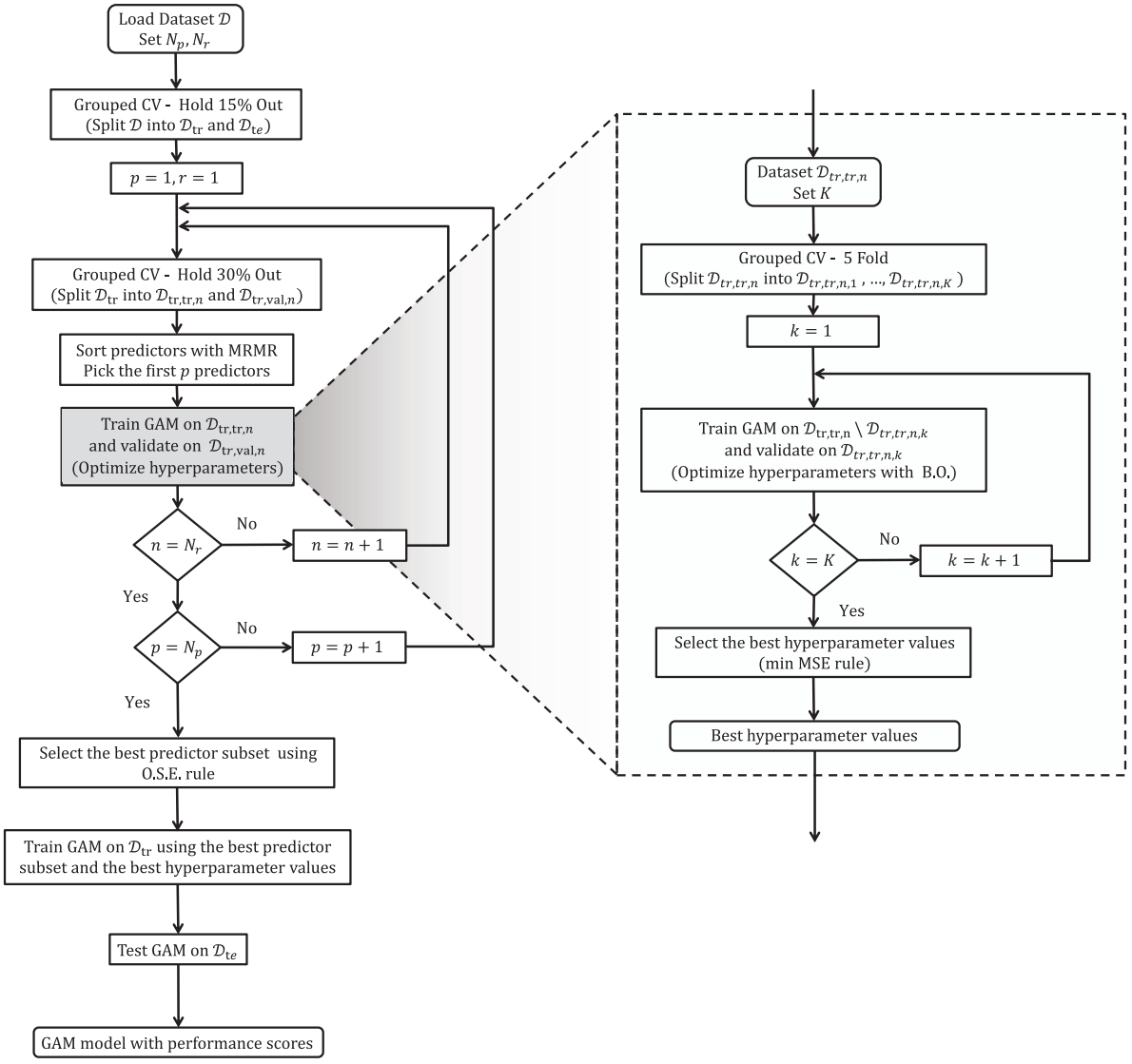

Fig. 3. Flowchart summarizing the overall machine learning methodology for the development of GAMs.

features that maximizes the relevance of X with respect to a response variable y,

$$V_X = \frac{1}{|X|} \sum_{x \in X} I(x, y), \tag{1}$$

and minimizes the redundancy of X,

$$W_X = \frac{1}{|X|^2} \sum_{x,z \in X} I(x, z), \tag{2}$$

where $|X|$ is the number of features in X and

$$I(A, B) = \sum_{a_i \in \mathcal{A}, b_j \in \mathcal{B}} P(A = a_i, B = b_i) \log \frac{P(A = a_i, B = b_i)}{P(A = a_i) P(B = b_i)} \tag{3}$$

is the mutual information of two (discrete) random variables (A, B) taking values over the space $\mathcal{A} \times \mathcal{B}$, with $\mathcal{A} = \{a_1, a_2, \ldots, a_n\}$ and $\mathcal{B} = \{b_1, b_2, \ldots, b_m\}$.

Finding an optimal set X requires considering all $2|\Omega|$ combinations, where Ω is the entire feature set. Instead, the MRMR algorithm ranks features through the forward addition scheme, which requires $\mathcal{O}(|\Omega| \cdot |X|)$ computations, by using the value of the mutual information quotient,

$$\mathrm{MIQ}_x = \frac{V_x}{W_x}, \tag{4}$$

where V_x and W_x are the relevance and redundancy of a feature, respectively:

$$V_x = I(x, y), \tag{5}$$

$$W_x = \frac{1}{X} \sum_{z \in X} I(x, z). \tag{6}$$

A large MIQ score value indicates that the corresponding predictor is important. Algorithm 1 summarizes the procedure, where $\overline{X}$ is the complementary set of $X \subseteq \Omega$ in Ω.

Algorithm 1 MRMR Algorithm

$X = \emptyset$
$x^{MaxR} \leftarrow \arg\max_{x \in \Omega} V_x$
add x^{MaxR} to X
$X^{NZR-zr} \leftarrow \{x \in \overline{X} \text{ s.t. } V_x > 0,\ W_x = 0\}$
while $X^{NZR-zr} \neq \emptyset$ **do**
    $x^{MaxR} \leftarrow \arg\max_{x \in X^{NZR-zr}} V_x$
    add x^{MaxR} to X
    remove x^{MaxR} from X^{NZR-zr}
end while
$X^{NZR-nzr} \leftarrow \{x \in \overline{X} \text{ s.t. } V_x > 0,\ W_x > 0\}$
while $X^{NZR-nzr} \neq \emptyset$ **do**
    $x^{MaxMIQ} = \arg\max_{x \in X^{NZR-nzr}} MIQ_x$
    add x^{MaxMIQ} to X
    remove x^{MaxMIQ} from $X^{NZR-nzr}$
end while
$X^{ZR-nzr} \leftarrow \{x \in \overline{X} \text{ s.t. } V_x = 0,\ W_x > 0\}$
while $X^{ZR-nzr} \neq \emptyset$ **do**
    pick a feature x^r randomly from X^{ZR-nzr}
    add x^r to X
    remove x^r from X^{ZR-nzr}
end while

2.5. Generalized additive models

A generalized additive model (GAM) [38,44,52,53] is an interpretable model that explains a response variable $y \sim \mathcal{N}(\mu, \sigma^2)$ using a sum of univariate and bivariate (interactions) shape functions of the predictors,

$$\hat{y} = \mu = c + \sum_{i=1}^{N_p} f_i(x_i) + \sum_{i,j=1}^{N_p} g_{i,j}(x_i, x_j), \tag{7}$$

where $\hat{y}$ is the GAM prediction (corresponding to the expected value of y), c is a constant, $f_i(x_i)$ is an univariate shape function of the ith predictor, and $g_{i,j}(x_i, x_j)$ is a bivariate shape function of the ith and jth predictors. Common choices for f_i and $g_{i,j}$ are splines or boosted trees [38,44,52,53]. Similarly, GAMs can also be trained to predict the standard deviation σ of y.

This work uses a Matlab implementation of GAMs, which adopts a set of boosted trees to learn each shape function. Further details related to the implementation can be found in the documentation of Matlab Statistics and Machine Learning Toolbox [54]. GAM training is carried out using the least-squares boosting algorithm [38]. At each iteration of the algorithm, a new set of trees is built by training one tree at a time. Every tree is trained to learn the difference between the observed response and the aggregated prediction of all trees trained previously. To control the boosting learning speed, tree predictions are weighted by a learning rate $\eta \in (0; 1]$. Every time a new tree is trained in the least-squares boosting algorithm, the overall model prediction $\hat{y}^+$ is updated as

$$\hat{y}^+ = \hat{y} + \eta \hat{y}_{\text{new}} \tag{8}$$

where $\hat{y}$ is the current prediction, and $\hat{y}_{\text{new}}$ is the model prediction contribution from the latest tree. The updated residual e^+ is computed as

$$e^+ = e - \eta \hat{y}_{\text{new}} \tag{9}$$

where e is the current residual.

The algorithm starts training a set of trees for univariate shape functions. At each iteration, the latest set of trees is included in the model if it improves the Mean Squared Error (MSE) by a value larger than a specified tolerance. Otherwise, the algorithm stops the iterations for univariate shape functions, and starts the iterations for bivariate shape functions. The algorithm stops when the MSE is not sufficiently improved by training trees for bivariate shape functions.

To prevent overfitting, a maximum number of trees per predictors can be set for univariate and bivariate shape functions. Different learning rates can also be assigned for univariate and bivariate shape functions.

2.6. Best predictors subset and hyperparameter selection via cross-validation

Cross-validation is the most widely applied method for predictor subset and hyperparameter selection [42]. Cross-validation estimates the expected out-of-sample prediction error by holding out a portion of data when training the model, and evaluating the model performance based on the holdout dataset. The predictor subset and/or hyperparameters that give the smallest validation error is selected, and the final model is rebuilt on all of the data [38,42]. A more robust choice can be carried out according to the *One-Standard-Error* (OSE) rule, which selects the most parsimonious model whose error is smaller than one standard deviation above the error of the best model [38,42]. Depending on data availability, several cross-validation strategies can be applied. With simple held-out validation, the dataset is simply split in training and validation folds. This choice is only recommended when there are enough sample points. When the amount of data is limited, K-fold cross-validation can be applied. The dataset is split into k folds. The model is trained using $k - 1$ folds and validated on the remaining one. Model training is repeated k times, and the performance metrics averaged. A possible alternative is Monte Carlo cross-validation. In this case, data are shuffled and then split into training and validation datasets. The procedure is repeated n times, and the performance is averaged over all the validation errors [38,42].

Remark. Independence and Identical Distribution of data (I.I.D.) is a common assumption behind many model identification algorithms, and the data splitting for cross-validation should ensure independence of the data subsets. When dealing with data originating from manufacturing processes, the I.I.D. assumption can be violated if the process has an underlying group structure. In this case, *grouped* cross-validation [42] avoids an overly optimistic estimation of the true error and prevents fitting possible system biases. With grouped cross-validation, data splitting is carried out at the group level: all of the data belonging to one group should be assigned to the same data subset. In this work, grouping is based on battery production batch (i.e., data from batteries belonging to the same batch are not split across different dataset partitions).

Remark. The implementation of GAMs discussed above requires the definition of several hyperparameters, including maximum number of trees, learning rates, and use of bivariate terms. In this work, *Grouped K-fold* cross-validation [38,42] is used to determine the best value of GAM hyperparameters, as those resulting in the minimum average validation MSE. When optimizing for bivariate terms, the number of possible combinations grows very rapidly with the number of predictors. Therefore, a grid search approach over the hyperparameter may be computationally intractable. In this case, BO can be effectively adopted as an efficient optimization strategy [55,56] and is adopted in this work.

2.7. Model performance assessment with goodness-of-fit scores

Let y_n be the nth observation, $\hat{y}_n$ be the corresponding model prediction, and N be the number of available observations.

The Mean Squared Error (MSE) is defined as

$$\mathrm{MSE} = \frac{1}{N}\sum_{n=1}^{N}(y_n - \hat{y}_n)^2. \tag{10}$$

The Coefficient of Determination (R^2) is defined as

$$R^2 = 1 - \frac{\mathrm{RSS}}{\mathrm{TSS}} \tag{11}$$

where the Residual Sum of Squares (RSS) and the Total Sum of Squares (TSS) are defined as

$$\mathrm{RSS} = \sum_{n=1}^{N}(y_n - \hat{y}_n)^2, \tag{12}$$

$$\mathrm{TSS} = \sum_{n=1}^{N}(y_n - \overline{y})^2, \tag{13}$$

with

$$\overline{y} = \frac{1}{N}\sum_{n=1}^{N} y_n. \tag{14}$$

2.8. Model interpretation with partial dependence plots

Partial dependence [38,57] represents the relationships between a predictor variable and the predicted response in a trained model, and can be computed by marginalizing over the other predictor variables. Consider partial dependence on a subset X^S of the whole predictor variable set $X = x_1, x_2, \ldots, x_m$. A subset X^S includes either one variable or two variables: $X^S = \{x^{S_1}\}$ or $X^S = \{x^{S_1}\ x^{S_2}\}$. A predicted response $f(X)$ depends on all variables in X,

$$f(X) = f(X^S, \overline{X}^S), \tag{15}$$

where $\overline{X}^S$ is the complementary set of X^S in X. The partial dependence of predicted responses on X^S is defined by the expectation of predicted responses with respect to $\overline{X}^S$,

$$f^S(X^S) = \int f(X^S, \overline{X}^S) P(\overline{X}^S) d\overline{X}^S, \tag{16}$$

where $P(\overline{X}^S)$ is the marginal probability of $\overline{X}^S$,

$$P(\overline{X}^S) \approx \int P(X^S, \overline{X}^S) d\overline{X}^S. \tag{17}$$

Assuming I.I.D. observations, and weak dependence between X^S and $\overline{X}^S$ and the interactions of X^S and $\overline{X}^S$, partial dependence can be estimated as

$$f^S(X^S) = \frac{1}{N}\sum_{i=1}^{N} f(X^S, \overline{X}_i^S) \tag{18}$$

where N is the number of observations and $X_i = (X_i^S, \overline{X}_i^S)$ is the ith observation.

3. Results

The predictive performances of the modelling approach are summarized in Fig. 4, which compares model predictions to the test regression line for all the 21 datasets explored in this work. Table 2 provides a quantification of performances, as training[1] and test GOF scores (MSE and R^2). As expected from the huge number of available data, and due to the cross-validation approach, the GOF is well-aligned between training and test partitions, for all datasets, with a minor degradation when moving from train to test subsets. Test values of R^2 are consistently high ($R^2 > 0.9$), with the only exception being the PulseMinV32JP1_np1 and PulseDuration64JP4_np2 datasets, which result in slightly lower scores ($R^2 = 0.87$). This performance can also be appreciated in Fig. 4, where model predictions are closely aligned with the regression line. The analysis of residuals, reported in the Supplementary Material, supports the correctness of results: for all datasets, train and test residuals are nearly zero-mean. Train and test residual distributions closely resemble Gaussian distributions for high probability densities, while distribution tails may deviate from the Gaussian. The latter may be due to the presence of particularly low/high values of the target variables, which were considered as reliable measurements during the dataset preprocessing. In a few cases, the distribution of residuals shows some skewedness, which suggests that these extreme measurements concentrate either in the lower or upper range of admissible target values. The variance of residuals typically shows low correlation with the target data, which is consistent with the hypothesis of homoscedastic measurement noise.

Tables 3 and 4 summarize the selected predictors for pulse- and background-related datasets, respectively (see Table 5 for predictor meanings). Predictors are sorted according to the MRMR algorithm, in descending order. For GAMs including predictor interactions, the predictors are reported in Supplementary Tables 1 to 12.

As highlighted by the scatterplots in Fig. 2, all target variables show a strong dependence on capacity, which is always present in the GAMs as the most important predictor. For some datasets from the 64JP4 set, (PulseAvgV64JP4_np1, PulseAvgV64JP4_np2, PulseDuration64JP4_np1, PulseDuration64JP4_np2, PulseDuration64JP4_np3, PulseDuration64JP4_np4, and PulseMinV64JP4_np1), capacity is chosen as the only relevant predictor. Each corresponding GAM is therefore a simple smoother of the target variable vs. capacity scatterplot (see Fig. 2), whose variability cannot be properly explained in terms of the available candidate predictors from production and burn-in data. In all other cases, GAMs rely on multiple predictors, whose impact on the prediction of the target variable can be investigated by means of *Partial Dependence Plots* (see Method Section). In general, correlation between random variables does not necessarily imply causation [38]. However, for some of the selected predictors, a clear trend in the PDP, combined with domain expertise, can point to interesting directions to be further investigated. For all datasets, all PDPs are provided in the Supplementary Material. As discussed earlier, capacity has the strongest impact of the target variable, which is also highlighted by

[1] scores are computed on the whole training set $\mathcal{D}_{tr}$

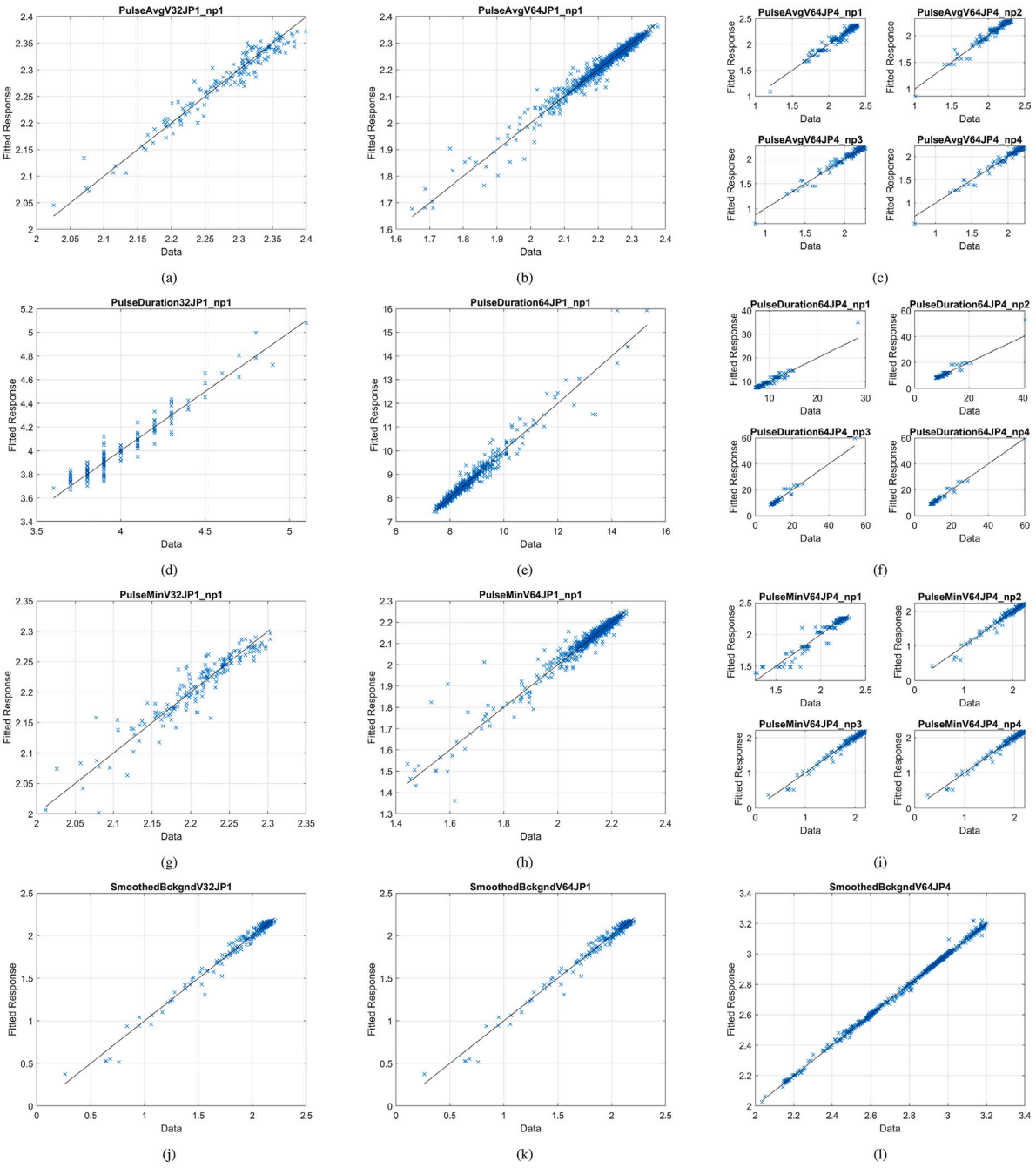

Fig. 4. Model predictions and regression lines for the datasets explored in this work. Results are related to the test partition of each dataset.

the related PDPs. Other interesting relationships include linear (see e.g., Supplementary Figs. 4, 12, 15, 41, 47, 72, 80, 82, 91, 97, 104, 114, 123, and 128), quadratic (Supplementary Fig. 120), and exponential (Supplementary Fig. 1) dependencies between the target variable and selected predictors.

The above analysis suggests that resistance- and voltage-related predictors from burn-in pulse train can predict variations of pulse-related features, such as average pulse voltage, minimum pulse voltage, and pulse duration, observed during life-test studies. For example, as seen in Supplementary Fig. 41, the last voltage measured at the end of the 2nd pulse during burn-in (BB_1_16_A_Pulse_VPt4), can already reflect the battery's performance (PulseDuration32JP1_np1) under high current pulses; from the PDP, lower voltages during the second burn-pulse can mean higher pulse duration, reflective of higher resistance or lower pulse average voltage. Similarly, Supplementary Fig. 47 shows the partial dependence of pulse duration for the 1st pulse in a 4 pulse train of 64 J pulses (PulseDuration64JP1_np1) on the initial burn-in pulse resistance of the 4th pulse (BB_1_16_A_Pulse_RP4Init); higher burn-in pulse resistance values are correlated to higher pulse duration later in life. The effect of pointwise cathode thickness measurements also produces noticeable trends in both pulse- and background-related target variables. Variability in cathode thickness within the same cell is usually neglected in pseudo-2D first-principles models [10,13]. However, that variability may introduce variances in geometric properties of the cathode across its length, which not only affects density, porosity, and particle size distribution in the cathode, but also reaction rate and diffusion coefficient in the electrode and overall dynamics of the battery. Domain expertise also confirms the effect of cathode mass/weight on pulse and background voltages. Interestingly, both cathode mass and thickness appear as terms interacting with capacity in

Table 2
Train and test GOF scores (MSE and R^2) of GAMs.

Dataset	Train MSE	Test MSE	Train R^2	Test R^2
PulseAvgV32JP1_np1	0.00014575	0.000285102	0.961037915	0.941054075
PulseAvgV64JP1_np1	7.07652E−05	0.000289954	0.992465167	0.965906866
PulseAvgV64JP4_np1	0.001554356	0.001474513	0.969504759	0.959513788
PulseAvgV64JP4_np2	0.001105054	0.001170778	0.977983493	0.961582935
PulseAvgV64JP4_np3	0.001002586	0.001041193	0.981803	0.96838532
PulseAvgV64JP4_np4	0.001081184	0.001058037	0.98204262	0.970620462
PulseDuration32JP1_np1	0.001645879	0.005186101	0.967352187	0.924653552
PulseDuration64JP1_np1	0.006045291	0.029516279	0.992732313	0.957175809
PulseDuration64JP4_np1	0.894232341	0.285256022	0.928652973	0.929367551
PulseDuration64JP4_np2	0.649248717	0.836235949	0.97398745	0.876140736
PulseDuration64JP4_np3	0.756731717	0.49232884	0.977308994	0.958115483
PulseDuration64JP4_np4	1.064863786	0.293577897	0.974007537	0.980820311
PulseMinV32JP1_np1	0.000122527	0.000390717	0.955905967	0.872117225
PulseMinV64JP1_np1	0.000116913	0.000654668	0.990946076	0.943698587
PulseMinV64JP4_np1	0.004228281	0.003221383	0.926405426	0.939599677
PulseMinV64JP4_np2	0.000768641	0.001828769	0.988123303	0.974109714
PulseMinV64JP4_np3	0.00067271	0.001534866	0.990463651	0.980420011
PulseMinV64JP4_np4	0.000244664	0.00088715	0.996790055	0.989879493
SmoothedBckgndV32JP1	2.8203E−06	1.07534E−05	0.9993221	0.997477596
SmoothedBckgndV64JP1	9.57996E−06	3.10012E−05	0.998875171	0.996254048
SmoothedBckgndV64JP4	3.05659E−05	9.53002E−05	0.999486832	0.998346696

Table 3
Summary of selected predictors for Pulse Datasets (Minimum Voltage, Average Voltage, Pulse Duration)

PulseMinV32JP1_np1	PulseMinV64JP1_np1	PulseMinV64JP4_np1	PulseMinV64JP4_np2	PulseMinV64JP4_np3	PulseMinV64JP4_np4
Q_PulseMinV_1	Q_PulseMinV_1	Q_PulseMinV_1	Q_PulseMinV_2	Q_PulseMinV_3	Q_PulseMinV_4
BB_1_16_A_Pulse_RP4Init	BB_UPCoT_Perform_Calculations_VP4Vsigma	Intercept	BB_1_16_A_Pulse_RP3Final	BB_1_16_A_Pulse_RP3Init	BB_UPCoT_Perform_Calculations_VpsOCVsigma1
FILL_Weight_Post-Weight 1	BB_1_16_A_Pulse_IONMin		Intercept	BB_1_16_A_Pulse_IONMax	CMI_Blackwell Wt thick_Thickness 5
bb_1_16_a_pulse_rp1diff	CathodeMass			Intercept	BB_1_16_A_Pulse_VPt2DateTime
Intercept	BB_1_16_A_Pulse_RP1Final				Intercept
	Intercept				

PulseAvgV32JP1_np1	PulseAvgV64JP1_np1	PulseAvgV64JP4_np1	PulseAvgV64JP4_np2	PulseAvgV64JP4_np3	PulseAvgV64JP4_np4
Q_PulseAvgV_1	Q_PulseAvgV_1	Q_PulseAvgV_1	Q_PulseAvgV_2	Q_PulseAvgV_3	Q_PulseAvgV_4
BB_1_16_A_Pulse_RP4Final	BB_UPCoT_Perform_Calculations_VP4Vsigma	Intercept	Intercept	Electrolyte FillEndDateTime	Electrolyte FillEndDateTime
FILL_Weight_Post-Weight 1	BB_10_mA_drain_VMIN			Intercept	Intercept
BB_UPCoT_Perform_Calculations_VpsOCVsigma1	fill_weight_diff-weight 1				
Intercept	BB_1_16_A_Pulse_IONMax				
	BB_10_mA_drain_TFIN				
	BB_UPCoT_Perform_Calculations_pPERCENT				
	BB_PostSoak_TFIN				
	BB_UPCoT_Perform_Calculations_VP4Vavg				
	Intercept				

PulseDuration32JP1_np1	PulseDuration64JP1_np1	PulseDura-tion64JP4_np1	PulseDuration64JP4_np2	PulseDuration64JP4_np3	PulseDuration64JP4_np4
Q_PulseDuration_1	Q_PulseDuration_1	Q_PulseDuration_1	Q_PulseDuration_2	Q_PulseDuration_3	Q_PulseDuration_4
BB_1_16_A_Pulse_VPt4	PostSoakOCV_V	Intercept	Intercept	Intercept	Intercept
BatHROCVDataCollectionEndDateTime	CathodeMass				
Intercept	BatHROCVDataCollectionEndDateTime				
	BB_UPCoT_Perform_Calculations_pPERCENT				
	BB_1_16_A_Pulse_RP4Init				
	Intercept				

Table 4
Summary of selected predictors for Background Voltage Datasets.

SmoothedBckgndV32JP1	SmoothedBckgndV64JP1	SmoothedBckgndV64JP4
Q_BckgndV	Q_BckgndV	Q_BckgndV
BB_10_mA_drain_VMAX	fill_weight_diff-weight 1	FILL_Weight_Post-Weight 1
BB_1_16_A_Pulse_RP1Init	bb_1_16_a_pulse_rp3diff	T_BckgndV
T_BckgndV	T_BckgndV	CMI_Blackwell Wt thick_Thickness 5
BB_UPCoT_Perform_Calculations_pPERCENT	BB_PostSoak_TFIN	Intercept
BB_10_mA_drain_TFIN	BB_10_mA_drain_TFIN	
WhenCreated	BB_10_mA_drain_tFIN	
Intercept	Intercept	

several datasets (e.g., PulseDuration64JP1_np1, PulseMinV64JP1_np1, PulseMinV64JP4_np4, and SmoothedBckgndV32JP1), suggesting that they could play an important role in defining the behaviour of the cell during life tests. Lastly, cell temperature during background discharge was expected to influence background voltage, based on thermodynamic and kinetic considerations, which is confirmed by the modelling results. At this stage, it must be recalled that any physics-based interpretation of data-driven models should be in principle validated by suitable experiments performed on the process, e.g. by adjusting one design parameter at a time, and verifying whether this results in the expected change in the measured process output. For ICD cells and the life-test protocol, however, this procedure would require several

Table 5
Meaning of all features selected by GAMs as predictors.

Feature name	Measurement unit	Meaning
BB_10_mA_drain_TFIN	°C	Temperature at the end of 10 mA drain phase of burn-in
BB_10_mA_drain_VMAX	V	In process maximum voltage during the 10 mA drain phase of burn-in
BB_10_mA_drain_VMIN	V	In process minimum voltage during the 10 mA drain phase of burn-in
BB_10_mA_drain_tFIN	hrs	DateTime at which the load was on during the 10 mA drain phase of burn-in
BB_1_16_A_Pulse_IONMax	A	Maximum current measured during the 1.16 A pulse
BB_1_16_A_Pulse_IONMin	A	Minimum current measured during the 1.16 A pulse
BB_1_16_A_Pulse_RP1Final	Ohm	Resistance at the end of the first burn-in pulse
BB_1_16_A_Pulse_RP1Init	Ohm	Resistance at the beginning of the first burn-in pulse
BB_1_16_A_Pulse_RP3Final	Ohm	Resistance at the end of the third burn-in pulse
BB_1_16_A_Pulse_RP3Init	Ohm	Resistance at the beginning of the third burn-in pulse
BB_1_16_A_Pulse_RP4Final	Ohm	Resistance at the end of the fourth burn-in pulse
BB_1_16_A_Pulse_RP4Init	Ohm	Resistance at the beginning of the fourth burn-in pulse
BB_1_16_A_Pulse_VPt2DateTime	DateTime	DateTime of the first burn-in pulse voltage minimum
BB_1_16_A_Pulse_VPt4	V	Last Voltage at the second burn-in pulse
BB_PostSoak_TFIN	°C	Final temperature for Post soak OCV step (OCV step prior to 10 mA drain step)
BB_UPCoT_Perform_Calculations_VP4Vavg	V	Mean of the 4th pulse voltage at the lot level
BB_UPCoT_Perform_Calculations_VP4Vsigma	V	Standard deviation for the 4th pulse voltage at the lot level
BB_UPCoT_Perform_Calculations_VpsOCVsigma1	V	Standard deviation of VEPR (End Process Voltage measurement in the Post Soak OCV step) with a passing VEPR
BB_UPCoT_Perform_Calculations_pPERCENT	Percent	Lot minimum percentage pass
BatHROCVDataCollectionEndDateTime	DateTime	DateTime at the OCV collection step prior to burn-in
CMI_Blackwell Wt thick_Thickness 5	inch	Cathode thickness at sampling location 5 (along cathode length)
CathodeMass	g	Mass of cathode
Electrolyte FillEndDateTime	DateTime	DateTime of electrolyte filling termination
FILL_Weight_Post-Weight 1	g	Weight of battery plus electrolyte after electrolyte filling
PostSoakOCV_V	V	Voltage of the Post soak OCV step (OCV step prior to 10 mA drain step)
Q_BckgndV	mAh	Capacity of background discharge voltage samples
Q_PulseAvgV_1	mAh	Capacity before the first life-test pulse voltage samples (PulseAvgV datasets)
Q_PulseAvgV_2	mAh	Capacity before the second life-test pulse voltage samples (PulseAvgV datasets)
Q_PulseAvgV_3	mAh	Capacity before the third life-test pulse voltage samples (PulseAvgV datasets)
Q_PulseAvgV_4	mAh	Capacity before the fourth life-test pulse voltage samples (PulseAvgV datasets)
Q_PulseDuration_1	mAh	Capacity before the first life-test pulse voltage samples (PulseDuration datasets)
Q_PulseDuration_2	mAh	Capacity before the second life-test pulse voltage samples (PulseDuration datasets)
Q_PulseDuration_3	mAh	Capacity before the third life-test pulse voltage samples (PulseDuration datasets)
Q_PulseDuration_4	mAh	Capacity before the fourth life-test pulse voltage samples (PulseDuration datasets)
Q_PulseMinV_1	mAh	Capacity after the first life-test pulse voltage samples (PulseMinV datasets)
Q_PulseMinV_2	mAh	Capacity after the second life-test pulse voltage samples (PulseMinV datasets)
Q_PulseMinV_3	mAh	Capacity after the third life-test pulse voltage samples (PulseMinV datasets)
Q_PulseMinV_4	mAh	Capacity after the fourth life-test pulse voltage samples (PulseMinV datasets)
T_BckgndV	°C	Temperature of background discharge voltage samples
WhenCreated	DateTime	DateTime of battery production termination
bb_1_16_a_pulse_rp1diff	Ohm	Difference of BB_1_16_A_Pulse_RP1Final and BB_1_16_A_Pulse_RP1Initial
bb_1_16_a_pulse_rp3diff	Ohm	Difference of BB_1_16_A_Pulse_RP3Final and BB_1_16_A_Pulse_RP3Initial
fill_weight_diff-weight 1	g	Electrolyte weight (Difference of FILL_Weight_Post-Weight_1 and FILL_Weight_Pre-Weight_1)
FILL_Weight_Post-Weight_1DateTime	DateTime	DateTime of production at which weight of battery is measured (after electrolyte fill)

months to years, and would therefore be extremely time consuming. An alternative approach would then be relying on the development of detailed physics-based models, in which cell design parameters could be more straightforwardly adjusted. The combination of physics-based and data-driven modelling is discussed in the following section.

4. Discussion

The value of the observations introduced in the previous section is twofold. On one hand, they can be used to support the development of mechanistic models, by suggesting possible dependencies between predictors and target variables. For example, variability in cathode thickness and cathode mass could be included in pseudo-2D first-principles models, as discussed above. Thinner cathodes have shorter charging times and are less likely to exhibit transport limitations and phase-transformation fronts [58], which lead to mechanical stresses [59], concentration polarization in the electrolyte [12], and dissipated heat, all of which can affect battery performance and accelerate degradation [60]. Smaller active particles usually result in faster reaction kinetics, due to higher internal surface area, and as well as faster diffusion in both in bulk and particle volume [61]. Moreover, any heterogeneities in thickness, internal area, porosity or other properties across one cell cause parallel variations in internal resistance [62], which are known to accelerate the degradation in the analogous situation of parallel cells in battery packs, due to inhomogeneous current distributions and localized heating [63]. As a result, our dependency study presented above implies that would be recommended to account for the impact of cathode thickness on key input parameters for first principles porous electrode models. Other possible developments include the explicit combination of first-principles and data-driven models. Within this hybrid framework [64,65], a first-principles model based on porous electrode theory (PET) [10] or its multiphase generalization (MPET) [13,66] for hybrid batteries with multiple active materials can provide baseline, population-level predictions for a wide range of operating conditions, thanks to the high level of generalization of this class of models. The baseline prediction can then be improved by training data-driven models to specialize the prediction at the individual battery level, by leveraging specific measurements of the battery — which may not be directly compatible with the structure of the first-principle model. On the other hand, the above analysis can lead to improvements in the battery production. Since PDPs highlight and quantify the effect each predictor has on the predicted variables in the model, their inspection can point to the production phases requiring tighter tolerances, in order to increase the consistence of battery performance. Based on the results of this work, it may be convenient to invest in tightening production tolerance for the thickness of cathode, which affects pulse minimum (PulseMinV64JP4_np4) and background voltage (SmoothedBckgndV64JP4), respectively, as seen in Supplementary Figs. 104 and 128. Tighter production tolerances on cathode mass may reduce variability in the pulse minimum voltage (PulseMinV64JP1_np1), as seen in Supplementary Fig. 82. Pulse average (PulseAvgV64JP1_np1) and background voltage (SmoothedBckgndV64JP1) could also benefit from improvements in temperature control during the battery burn-in process (BB PostSoak TFIN), as seen in Supplementary Figs. 14 and 120. Other possible developments

include the design of new cell acceptance criteria and new cell grading strategies, based on model predictions. As soon as a battery has completed burn-in, the models constructed in this work can be used to predict the outcome of life-test experiments and evaluate its expected performances.

5. Conclusion

Machine learning is a promising route for diagnostics and prognostics of lithium-ion batteries and enables emerging applications in their development, manufacturing, and optimization. In this work, we developed interpretable, machine learning models to predict primary battery performance of hybrid cathode technology batteries based on production data and on tests carried out immediately after manufacturing. The machine learning approach adopted in this work relies on Generalized Additive Models as the main building block, and on the Maximum Relevance Minimum Redundancy algorithm and cross-validation to select the predictor subset. Our approach provides accurate predictions ($R^2 > 0.87$ on test data) for all of 21 different datasets. Physical intuition supports the data-driven choice of a subset of predictors that may be responsible for the observed variability in the battery primary performances. Broadly speaking, our results confirm the possibility of exploiting a variety of production data, sometimes already available to the manufacturer, to unlock the development of data-driven models that are tailored to the specific product. This in turn enables high prediction accuracy at low cost. Moreover, the use of interpretable data-driven models can provide further insight in the variability of the production process, and give clues on how to refine it.

Data and code availability

All the software code related to this work is available online at https://github.com/GiacomoGaluppini, under CC BY 4.0 licence. All materials and data are proprietary of Medtronic.

Funding

This work was funded by Medtronic.

CRediT authorship contribution statement

Giacomo Galuppini: Writing – original draft, Visualization, Software, Methodology, Conceptualization. **Qiaohao Liang:** Writing – original draft, Validation. **Prabhakar A. Tamirisa:** Writing – review & editing, Data curation. **Jeffrey A. Lemmerman:** Writing – original draft, Data curation. **Melani G. Sullivan:** Data curation. **Michael J.M. Mazack:** Data curation. **Partha M. Gomadam:** Data curation. **Martin Z. Bazant:** Writing – review & editing, Funding acquisition. **Richard D. Braatz:** Writing – review & editing, Supervision, Methodology, Funding acquisition, Conceptualization.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

The data that has been used is confidential.

Appendix A. Supplementary data

Supplementary material related to this article can be found online at https://doi.org/10.1016/j.jpowsour.2024.234668. Supplemental information can be found online at the Journal's website.

References

[1] R. Schmuch, R. Wagner, G. Hörpel, T. Placke, M. Winter, Performance and cost of materials for lithium-based rechargeable automotive batteries, Nat. Energy 3 (4) (2018) 267–278.
[2] V. Ramadesigan, P.W. Northrop, S. De, S. Santhanagopalan, R.D. Braatz, V.R. Subramanian, Modeling and simulation of lithium-ion batteries from a systems engineering perspective, J. Electrochem. Soc. 159 (3) (2012) R31–R45.
[3] P.M. Attia, A. Grover, N. Jin, K.A. Severson, T.M. Markov, Y.-H. Liao, M.H. Chen, B. Cheong, N. Perkins, Z. Yang, P.K. Herring, M. Aykol, S.J. Harris, R.D. Braatz, S. Ermon, W.C. Chueh, Closed-loop optimization of fast-charging protocols for batteries with machine learning, Nature 578 (7795) (2020) 397–402.
[4] B. Jiang, W.E. Gent, F. Mohr, S. Das, M.D. Berliner, M. Forsuelo, H. Zhao, P.M. Attia, A. Grover, P.K. Herring, M.Z. Bazant, S.J. Harris, S. Ermon, W.C. Chueh, R.D. Braatz, Bayesian learning for rapid prediction of lithium-ion battery-cycling protocols, Joule 5 (12) (2021) 3187–3203.
[5] S.B. Peterson, J. Apt, J. Whitacre, Lithium-ion battery cell degradation resulting from realistic vehicle and vehicle-to-grid utilization, J. Power Sources 195 (8) (2010) 2385–2392.
[6] W. Waag, C. Fleischer, D.U. Sauer, Critical review of the methods for monitoring of lithium-ion batteries in electric and hybrid vehicles, J. Power Sources 258 (2014) 321–339.
[7] Medtronic cardiac rhythm products information for healthcare professionals, 2023, https://www.medtronic.com/us-en/healthcare-professionals/products/cardiacrhythm.html. (Accessed: 24 May 2023).
[8] A. Crespi, C. Schmidt, J. Norton, K. Chen, P. Skarstad, Modeling and characterization of the resistance of lithium/SVO batteries for implantable cardioverter defibrillators, J. Electrochem. Soc. 148 (1) (2001) A30–A37.
[9] P.M. Gomadam, J.R. Brown, E.R. Scott, C.L. Schmidt, Predicting charge-times of implantable cardioverter defibrillators, in: Annual International Conference of the IEEE Engineering in Medicine and Biology Society, 2009, pp. 3020–3023.
[10] P.M. Gomadam, D.R. Merritt, E.R. Scott, C.L. Schmidt, P.M. Skarstad, J.W. Weidner, Modeling Li/CF_x-SVO hybrid-cathode batteries, J. Electrochem. Soc. 154 (11) (2007) A1058–A1064.
[11] J. Newman, W. Tiedemann, Porous-electrode theory with battery applications, AIChE J. 21 (1) (1975) 25–41.
[12] R.B. Smith, M.Z. Bazant, Multiphase porous electrode theory, J. Electrochem. Soc. 164 (11) (2017) E3291–E3310.
[13] Q. Liang, M.Z. Bazant, Hybrid-MPET: An open-source simulation software for hybrid electrode batteries, 2023, arXiv preprint arXiv:2305.15599.
[14] S. Paul, C. Diegelmann, H. Kabza, W. Tillmetz, Analysis of ageing inhomogeneities in lithium-ion battery systems, J. Power Sources 239 (2013) 642–650.
[15] S.. Schuster, T. Bach, E. Fleder, J. Müller, M. Brand, G. Sextl, A. Jossen, Nonlinear aging characteristics of lithium-ion cells under different operational conditions, J. Energy Storage 1 (2015) 44–53.
[16] M. Broussely, S. Herreyre, P. Biensan, P. Kasztejna, K. Nechev, R. Staniewicz, Aging mechanism in Li-ion cells and calendar life predictions, J. Power Sources 97 (2001) 13–21.
[17] M.B. Pinson, M.Z. Bazant, Theory of SEI formation in rechargeable batteries: Capacity fade, accelerated aging and lifetime prediction, J. Electrochem. Soc. 160 (2) (2012) A243–A250.
[18] T. Gao, Y. Han, D. Fraggedakis, S. Das, T. Zhou, C.-N. Yeh, S. Xu, W.C. Chueh, J. Li, M.Z. Bazant, Interplay of lithium intercalation and plating on a single graphite particle, Joule 5 (2) (2021) 393–414.
[19] A. Jana, A.S. Mitra, S. Das, W.C. Chueh, M.Z. Bazant, R.E. García, Physics-based, reduced order degradation model of lithium-ion batteries, J. Power Sources 545 (2022) 231900.
[20] M.D. Berliner, H. Zhao, S. Das, M. Forsuelo, B. Jiang, W.H. Chueh, M.Z. Bazant, R.D. Braatz, Nonlinear identifiability analysis of the porous electrode theory model of lithium-ion batteries, J. Electrochem. Soc. 168 (9) (2021) 090546.
[21] G. Galuppini, M.D. Berliner, D.A. Cogswell, D. Zhuang, M.Z. Bazant, R.D. Braatz, Nonlinear identifiability analysis of multiphase porous electrode theory-based battery models: A lithium iron phosphate case study, J. Power Sources 573 (2023) 233009.
[22] X.-S. Si, W. Wang, C.-H. Hu, D.-H. Zhou, Remaining useful life estimation–A review on the statistical data driven approaches, European J. Oper. Res. 213 (1) (2011) 1–14.
[23] L. Wu, X. Fu, Y. Guan, Review of the remaining useful life prognostics of vehicle lithium-ion batteries using data-driven methodologies, Appl. Sci. 6 (6) (2016) 166–177.
[24] J. Zhao, X. Feng, Q. Pang, J. Wang, Y. Lian, M. Ouyang, A.F. Burke, Battery prognostics and health management from a machine learning perspective, J. Power Sources 581 (2023) 233474.
[25] B. Saha, K. Goebel, J. Christophersen, Comparison of prognostic algorithms for estimating remaining useful life of batteries, Trans. Inst. Meas. Control 31 (3–4) (2009) 293–308.
[26] K.A. Severson, P.M. Attia, N. Jin, N. Perkins, B. Jiang, Z. Yang, M.H. Chen, M. Aykol, P.K. Herring, D. Fraggedakis, M.Z. Bazant, S.J. Harris, W.C. Chueh, R.D. Braatz, Data-driven prediction of battery cycle life before capacity degradation, Nat. Energy 4 (5) (2019) 383–391.

[27] G. Lee, J. Kim, C. Lee, State-of-health estimation of li-ion batteries in the early phases of qualification tests: An interpretable machine learning approach, Expert Syst. Appl. 197 (2022) 116817.
[28] K. Liu, X. Hu, H. Zhou, L. Tong, W.D. Widanage, J. Marco, Feature analyses and modeling of lithium-ion battery manufacturing based on random forest classification, IEEE/ASME Trans. Mechatronics 26 (6) (2021) 2944–2955.
[29] K. Liu, M.F. Niri, G. Apachitei, M. Lain, D. Greenwood, J. Marco, Interpretable machine learning for battery capacities prediction and coating parameters analysis, Control Eng. Pract. 124 (2022) 105202.
[30] P. Khumprom, N. Yodo, A data-driven predictive prognostic model for lithium-ion batteries based on a deep learning algorithm, Energies 12 (4) (2019) 660–681.
[31] X. Hu, L. Xu, X. Lin, M. Pecht, Battery lifetime prognostics, Joule 4 (2) (2020) 310–346.
[32] M.-F. Ng, J. Zhao, Q. Yan, G.J. Conduit, Z.W. Seh, Predicting the state of charge and health of batteries using data-driven machine learning, Nat. Mach. Intell. 2 (3) (2020) 161–170.
[33] P.M. Attia, K.A. Severson, J.D. Witmer, Statistical learning for accurate and interpretable battery lifetime prediction, J. Electrochem. Soc. 168 (9) (2021) 090547.
[34] X. Liu, X.-Q. Zhang, X. Chen, G.-L. Zhu, C. Yan, J.-Q. Huang, H.-J. Peng, A generalizable, data-driven online approach to forecast capacity degradation trajectory of lithium batteries, J. Energy Chem. 68 (2022) 548–555.
[35] X. Liu, H.-J. Peng, B.-Q. Li, X. Chen, Z. Li, J.-Q. Huang, Q. Zhang, Untangling degradation chemistries of lithium-sulfur batteries through interpretable hybrid machine learning, Angew. Chem. 134 (48) (2022) e202214037.
[36] S. Tao, C. Sun, S. Fu, Y. Wang, R. Ma, Z. Han, Y. Sun, Y. Li, G. Wei, X. Zhang, G. Zhou, H. Sun, Battery cross-operation-condition lifetime prediction via interpretable feature engineering assisted adaptive machine learning, ACS Energy Lett. 8 (8) (2023) 3269–3279.
[37] J. Schnell, C. Nentwich, F. Endres, A. Kollenda, F. Distel, T. Knoche, G. Reinhart, Data mining in lithium-ion battery cell production, J. Power Sources 413 (2019) 360–366.
[38] T. Hastie, R. Tibshirani, J.H. Friedman, J.H. Friedman, The Elements of Statistical Learning: Data Mining, Inference, and Prediction, vol. 2, Springer, New York, 2009.
[39] J. Zhu, Y. Wang, Y. Huang, R. Bhushan Gopaluni, Y. Cao, M. Heere, M.J. Mühlbauer, L. Mereacre, H. Dai, X. Liu, A. Senyshyn, X. Wei, M. Knapp, H. Ehrenberg, Data-driven capacity estimation of commercial lithium-ion batteries from voltage relaxation, Nature Commun. 13 (1) (2022) 2261–2271.
[40] B.-R. Chen, C.M. Walker, S. Kim, M.R. Kunz, T.R. Tanim, E.J. Dufek, Battery aging mode identification across NMC compositions and designs using machine learning, Joule 6 (12) (2022) 2776–2793.
[41] W. Sun, R.D. Braatz, ALVEN: Algebraic learning via elastic net for static and dynamic nonlinear model identification, Comput. Chem. Eng. 143 (2020) 107103.
[42] W. Sun, R.D. Braatz, Smart process analytics for predictive modeling, Comput. Chem. Eng. 144 (2021) 107134.
[43] S. Scardapane, D. Comminiello, A. Hussain, A. Uncini, Group sparse regularization for deep neural networks, Neurocomputing 241 (2017) 81–89.
[44] T.J. Hastie, Generalized additive models, in: Statistical Models in S, Routledge, New York, 2017, pp. 249–307.
[45] G.A. Darbellay, I. Vajda, Estimation of the information by an adaptive partitioning of the observation space, IEEE Trans. Inform. Theory 45 (4) (1999) 1315–1321.
[46] C. Ding, H. Peng, Minimum redundancy feature selection from microarray gene expression data, J. Bioinform. Comput. Biol. 3 (2) (2005) 185–205.
[47] H. Gan, R.S. Rubino, E.S. Takeuchi, Dual-chemistry cathode system for high-rate pulse applications, J. Power Sources 146 (1–2) (2005) 101–106.
[48] K. Chen, D.R. Merritt, W.G. Howard, C.L. Schmidt, P.M. Skarstad, Hybrid cathode lithium batteries for implantable medical applications, J. Power Sources 162 (2) (2006) 837–840.
[49] Matlab curve fitting toolbox documentation, 2023, https://it.mathworks.com/help/curvefit/. (Accessed: 08 May 2023).
[50] M. Pelikan, D.E. Goldberg, E. Cantú-Paz, et al., BOA: The Bayesian optimization algorithm, in: Proceedings of the Genetic and Evolutionary Computation Conference, vol. 1, 1999, pp. 525–532.
[51] R. Ibraheem, Y. Wu, T. Lyons, G. Dos Reis, Early prediction of lithium-ion cell degradation trajectories using signatures of voltage curves up to 4-minute sub-sampling rates, Appl. Energy 352 (2023) 121974.
[52] Y. Lou, R. Caruana, J. Gehrke, Intelligible models for classification and regression, in: Proceedings of the 18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2012, pp. 150–158.
[53] Y. Lou, R. Caruana, J. Gehrke, G. Hooker, Accurate intelligible models with pairwise interactions, in: Proceedings of the 19th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2013, pp. 623–631.
[54] Matlab statistics and machine learning toolbox documentation, 2023, https://it.mathworks.com/help/stats/. (Accessed: 08 May 2023).
[55] J. Wu, X.-Y. Chen, H. Zhang, L.-D. Xiong, H. Lei, S.-H. Deng, Hyperparameter optimization for machine learning models based on Bayesian optimization, J. Electron. Sci. Technol. 17 (1) (2019) 26–40.
[56] J. Snoek, H. Larochelle, R.P. Adams, Practical Bayesian optimization of machine learning algorithms, in: F. Pereira, C. Burges, L. Bottou, K. Weinberger (Eds.), in: Advances in Neural Information Processing Systems, vol. 25, Curran Associates, Inc., 2012.
[57] J.H. Friedman, Greedy function approximation: A gradient boosting machine, Ann. Statist. (2001) 1189–1232.
[58] T.R. Ferguson, M.Z. Bazant, Phase transformation dynamics in porous battery electrodes, Electrochim. Acta 146 (2014) 89–97.
[59] W. Ai, L. Kraft, J. Sturm, A. Jossen, B. Wu, Electrochemical thermal-mechanical modelling of stress inhomogeneity in lithium-ion pouch cells, J. Electrochem. Soc. 167 (1) (2020) 013512.
[60] C. Liu, L. Liu, Optimal design of Li-ion batteries through multi-physics modeling and multi-objective optimization, J. Electrochem. Soc. 164 (11) (2017) E3254–E3264.
[61] R. Malik, D. Burch, M. Bazant, G. Ceder, Particle size dependence of the ionic diffusivity, Nano Lett. 10 (10) (2010) 4123–4127.
[62] B. Kenney, K. Darcovich, D.D. MacNeil, I.J. Davidson, Modelling the impact of variations in electrode manufacturing on lithium-ion battery modules, J. Power Sources 213 (2012) 391–401.
[63] R. Gogoana, M.B. Pinson, M.Z. Bazant, S.E. Sarma, Internal resistance matching for parallel-connected lithium-ion cells and impacts on battery pack cycle life, J. Power Sources 252 (2014) 8–13.
[64] M. Aykol, C.B. Gopal, A. Anapolsky, P.K. Herring, B. van Vlijmen, M.D. Berliner, M.Z. Bazant, R.D. Braatz, W.C. Chueh, B.D. Storey, Perspective–Combining physics and machine learning to predict battery lifetime, J. Electrochem. Soc. 168 (3) (2021) 030525.
[65] R.G. Nascimento, M. Corbetta, C.S. Kulkarni, F.A. Viana, Hybrid physics-informed neural networks for lithium-ion battery modeling and prognosis, J. Power Sources 513 (2021) 230526.
[66] Q. Liang, G. Galuppini, P.M. Gomadam, P.A. Tamirisa, J.A. Lemmerman, M.J. Mazack, M.G. Sullivan, R.D. Braatz, M.Z. Bazant, Physics-based modeling of pulse and relaxation of high-rate li/cf_x-SVO batteries in implantable medical devices, J. Power Sources (2024).