Journal of The Electrochemical Society, 2021 **168** 020502

Challenging Practices of Algebraic Battery Life Models through Statistical Validation and Model Identification via Machine-Learning

Paul Gasper,[1,z] Kevin Gering,[2] Eric Dufek,[2] and Kandler Smith[1]

[1]*Energy Conversion and Storage Systems Center, National Renewable Energy Laboratory, Golden, CO, 80401, United States of America*
[2]*Energy Storage & Advanced Vehicles Department, Idaho National Laboratory, Idaho Falls, ID, 83415, United States of America*

Various modeling techniques are used to predict the capacity fade of Li-ion batteries. Algebraic reduced-order models, which are inherently interpretable and computationally fast, are ideal for use in battery controllers, technoeconomic models, and multi-objective optimizations. For Li-ion batteries with graphite anodes, solid-electrolyte-interphase (SEI) growth on the graphite surface dominates fade. This fade is often modeled using physically informed equations, such as square-root of time for predicting solvent-diffusion limited SEI growth, and Arrhenius and Tafel-like equations predicting the temperature and state-of-charge rate dependencies. In some cases, completely empirical relationships are proposed. However, statistical validation is rarely conducted to evaluate model optimality, and only a handful of possible models are usually investigated. This article demonstrates a novel procedure for automatically identifying reduced-order degradation models from millions of algorithmically generated equations via bi-level optimization and symbolic regression. Identified models are statistically validated using cross-validation, sensitivity analysis, and uncertainty quantification via bootstrapping. On a $LiFePO_4$/Graphite cell calendar aging data set, automatically identified models utilizing square-root, power law, stretched exponential, and sigmoidal functions result in greater accuracy and lower uncertainty than models identified by human experts, and demonstrate that previously known physical relationships can be empirically "rediscovered" using machine learning.
 [DOI: 10.1149/1945-7111/abdde1]

Manuscript submitted October 29, 2020; revised manuscript received December 30, 2020. Published February 1, 2021.

Supplementary material for this article is available online

The simultaneous decrease of cost and increase of performance of lithium-ion batteries (LIBs) in recent years [1,2] has expanded their use from a longtime market niche of portable electronics to new markets, predominantly electric vehicles and stationary energy storage systems. These new markets require LIBs to achieve long lifetimes while still meeting performance requirements. For example, the profitability of stationary energy storage systems is often dependent on their lifetime; revenue can be generated by delivering energy as long as the batteries are still useful, but LIBs degrade both over time and as a function of energy throughput, eventually requiring replacement, which is capital-intensive and costly. LIB degradation varies substantially based on environmental conditions and use case, making it challenging to estimate the degradation cost and system profitability.[3–6] At the root of estimating degradation cost is estimating the state-of-health (SOH) of a cell, which is some measure of the cell performance relative to its initial state. It is possible to simply measure battery SOH, but this approach is not predictive; a predictive model is required to estimate the future state of the battery from its present state. Predictive models enable battery controllers to optimize for long cell life or for project planners to estimate cost of cell degradation as part of technoeconomic analyses.[3,4,7–9]

Predictive models for battery lifetime can be loosely categorized into empirical, semi-empirical, and electrochemical models. Empirical and semi-empirical models are models that predict capacity fade without any simulation of the internal physics; this includes reduced-order models,[10–24] which are simply algebraic equations, as well as machine-learning based approaches.[25–28] For this application, the key difference between these approaches is their interpretability; low-dimensional algebraic models are straightforward to interpret, as observed behaviors can be attributed to the various components of an equation, while machine learning models can be challenging to draw physical interpretations from Refs. 29, 30. The main draw-back of any empirical model, reduced-order or machine-learning based, is that they risk poor extrapolation to new conditions. This would not be a problem if all relevant conditions for a system could be tested, but the parameter space of battery aging studies is much too large to cost-effectively test. Semi-empirical models attempt to address the challenge of identifying algebraic models that are safe to extrapolate when trained on very small data sets by incorporating physically informed equations, usually simple chemical or electrochemical reaction models such as the Arrhenius equation,[17,18,23] incorporating domain knowledge to counteract a lack of data; recent work, however, has shown that this widely used approach may not be analytically or statistically valid.[10] Electrochemical models, which numerically or analytically simulate the internal physics of a battery, are inherently interpretable but are usually too computationally intensive to predict degradation over tens of thousands of hours. Recent work by Reniers et al. demonstrated that numerical electrochemical models can be fast-computing while modeling both short-term battery dynamics as well as long-term degradation effects.[3,31] However, these models are still much more computationally intensive than the algebraic expressions used by reduced-order models, and are challenging to properly parameterize,[32–34] making them powerful tools for experts, but difficult to apply quickly when experimental data is limited. The flexibility to handle both small and large training data sets, fast computation, and ease-of-implementation inherent to algebraic reduced-order models naturally lends them to use many complex optimizations, where implementing numerical electrochemical models may be computationally prohibitive.[4,7–9]

The main challenge of utilizing a reduced-order battery lifetime model is in identifying an algebraic expression that predicts cell behavior accurately and extrapolates safely. This process is challenging for even the simplest battery degradation mechanisms. The most straightforward degradation data to model for LIBs is calendar degradation, i.e., the degradation observed when a cell is left to age without cycling. This is because the capacity fade for common LIBs during calendar aging is dominated by a single degradation mechanism, the growth of the solid-electrolyte-interphase (SEI) at

[z]E-mail: pauljgasper@gmail.com

the graphite electrode, and the only experimental variables are time, ambient temperature, and cell state-of-charge (SOC). This source of capacity loss is often categorized as the loss of lithium inventory (LLI). Even for this simple case, a litany of models exists. Often, model identification is informed by known physical relationships, such as individual cell fade models assuming $t^{0.5}$ behavior, which models SEI growth when it is rate-limited by the diffusion of neutral lithium through the SEI,[35–37] or SEI growth rate models based on Arrhenius or Tafel equations. There are several reasons that researchers use physically informed reduced-order models:

1. Obvious starting point for identifying reduced-order models of complex systems
2. Inherent interpretability of model behavior when utilizing equations derived from first principles analysis of simple systems
3. Model behavior when extrapolating to untested conditions is known a-priori

However, as demonstrated in a recent article by Attia, Chueh, and Harris,[10] physically informed models may not be statistically validated: they have not been carefully investigated through a statistical lens, extrapolation has not been tested or validated, and model optimality has not been demonstrated through comparison to other possible models. Additionally, because reduced-order models are being used to model complex systems with competing internal processes, rather than the simple systems used to derive most physically informed models, these models may not even be analytically supported. This is not to suggest that physically informed models cannot be utilized, only to highlight the need for statistical validation of any reduced-order model. And, in lieu of effective physically informed equations, a statistically rigorous methodology for empirically deriving reduced-order models needs to be presented. Empirically derived and physically informed models can then be compared to clarify the advantages or disadvantages of both approaches; presenting such a method and comparing the results with prior best-practices is the purpose of this work.

In this work, a new procedure for the automatic identification of reduced-order capacity fade models for LIBs is presented, implementing a variety of statistical techniques to interrogate and validate models. Multiple model types to describe cell capacity fade are compared to identify an optimal model: square-root, power law, stretched exponential, and sigmoidal models. For each model, several variations are tested, where for each variation, certain parameters are optimized locally to each individual cell of a data set and others are optimized globally across the entire data set. This is done simultaneously via bi-level optimization, which is also be referred to as nested optimization. Sub-models that predict the values of local parameters as functions of cell operating conditions are then automatically identified via symbolic regression, which selects a low-dimensional sub-model from over one million possible combinations utilizing LASSO regularization[38] on an algorithmically generated library of possible descriptors. This sub-model then replaces its corresponding local parameter, constructing a global model, which is a single equation describing the behavior of the entire data set. Global model quality is investigated using cross-validation, convergence analysis, sensitivity analysis, uncertainty quantification, and extrapolation.

On a standout experimental data set monitoring calendar aging of 22 commercial $LiFePO_4$ (LFP)/Graphite cells for about 8 months,[18] a range of automatically identified models outperform the physically informed $t^{0.5}$ model structure commonly used in the literature, with substantially lower error, less evidence of systematic deviation between model predictions and data, and lower uncertainty. For this data set, the most accurate model identified is a sigmoidal model that predicts both the degradation rate as well as the curvature of each data series. Automatically identified models for the degradation rate consistently converged on Arrhenius and Tafel-like terms, "rediscovering" these well-understood physical relationships, while also predicting the degradation rate of each cell more accurately than previously documented models. Variation of the curvature between data series reveals non-diffusion-limited SEI growth for this cell, with the power exponent of time shifting from 1 (kinetic-limited SEI growth) when capacity fade is very small to 0.45 (slower than diffusion-limited) when capacity fade is more substantial. Predictions by various models of the capacity fade after 20 years of aging imply that $t^{0.5}$ models may be overpredicting capacity degradation by almost 100% at long times, when compared with more accurate power-law or sigmoidal models.

The approach shown here for automatically identifying reduced-order models can hopefully be used to explore the implications of predictive model accuracy on the conclusions of technoeconomic models or battery controllers, rapidly develop accurate models to predict and compare the degradation behaviors of multiple Li-ion technologies, and also accelerate learning from new experimental data sets. This work builds on a growing amount of literature utilizing compressed sensing methods, such as symbolic regression, for harnessing the advantages of machine-learning in a scientific context. MATLAB code for implementing the bi-level optimization and symbolic regression techniques used in this study has been publicly shared at https://github.com/NREL/AI-Batt to improve the reproducibility of this work and accessibility of these methods.

Reduced-Order Models in Literature

Reduced-order models of individual-cell capacity fade.—New research on calendar fade mechanisms in Li-ion batteries, specifically on batteries using graphite intercalation anodes, has brought issues to light with the historical approach of modeling the calendar fade of individual cells using a $t^{0.5}$ dependence for capacity loss. This $t^{0.5}$ dependence is born from assuming that LLI due to SEI growth on graphite is rate-limited by diffusion of neutral lithium through the SEI after the first few formation cycles have been completed.[36,37] However, recent fundamental studies have shown that this single rate-limiting mechanism does not describe SEI growth in cases where the SEI is very small, or during lithiation or delithiation of the electrode.[36,37,39,40] Of note from these fundamental works is that during calendar aging, the power exponent of time is near 1 when the SEI is very thin, then decays towards 0.5 at long times.[37] The $t^{0.5}$ model is also undoubtedly an oversimplification that overlooks the impact of multiple aging processes at the electrode surfaces that may occur at different and sometimes interdependent rates or be affected by surface heterogeneity. Thus, the $t^{0.5}$ model is not necessarily validated as a general model from an analytical standpoint. This conclusion is doubly true for studies modeling calendar fade in full cells, which experience separate cathode effects during aging, or calendar fade models for LIBs using other anode materials.[41] Empirical analysis also contests the suitability of $t^{0.5}$ models for predicting calendar fade of lithium-ion batteries with graphite anodes. Attia et al. determined that the $t^{0.5}$ model was not statistically validated on several published data sets by comparing fits from multiple models, reporting confidence intervals, and analyzing model residuals.[10] For example, many data series showed "high" R^2 values of 0.99 or greater when fit with a $t^{0.5}$ model, but only 1 out of 17 data series investigated contained 0.5 within the 95% confidence interval for the power exponent of time when fit with a power law model (t^z). Residuals analysis of the $t^{0.5}$ models generally displayed obvious heteroscedasticity, indicating that the models were not effectively predicting system behavior.

Attia et al.[10] also convincingly showed that the model structure substantially effects predictive accuracy; for some data series, models required an intercept term, while for others, an intercept term did not substantially improve the result. In many other recent works, though, the functional structure of calendar fade models is only occasionally explored within each study, though a wide variety of models have been proposed across all studies, clearly indicating that model structure matters. Some articles compare the results of a

few models, choosing one or several for further study based on either qualitative comparison[42] or fit metrics.[12,23,24,43] This includes the square-root of time model, various power law models, logarithms, stretched exponential, and sigmoidal models. Logarithm models are useful for modeling a wide variety of systems but are difficult to physically interpret in this context and are not considered in this work. Stretched exponential models have been used to model decay of the dielectric constant of polymers and glasses with time,[44,45] and in the context of LIBs, have been used to model both calendric and cyclic capacity loss.[42,46,47] Sigmoidal type models have been used to model bacterial growth/survival[48] and more recently chemical reaction kinetics,[49–52] and were proposed for modeling LIB capacity fade by Gering.[13] Various proposed local cell degradation models from literature are reported in Table I. All equations have been rearranged to have the structure $q = \beta_0 - \ldots$, where q is the relative discharge capacity of the battery, and β_0 is the relative discharge capacity at time $t = 0$. This intercept term, β_0, is often neglected, assuming the intercept to be equal to 1.

Reduced-order models of rate dependence on cell operating conditions.—Predictive models for battery degradation are most useful for control algorithms or technoeconomic analysis when they predict battery performance across a wide range of environmental conditions and use cases. One path to constructing such a model is to predict the variation of parameters from individual cell models, i.e., local models, as a function of each cell's operating conditions. For example, consider a local model fitting the relative capacity loss vs time for calendar aged LIBs at various temperatures and SOCs:

$$\boldsymbol{q}_n(t) = \beta_{1,n} \cdot \boldsymbol{t}_n^{0.5} \qquad [1]$$

where $\boldsymbol{q}_n$ is the relative capacity loss vector of cell n as a function time, $\boldsymbol{t}_n$, and $\beta_{1,n}$ is the capacity fade rate of cell n, which in this example is dominated by the SEI growth rate at the graphite electrode. Local models can be constructed with any time-varying experimental variables as the input (time) or the predicted response (relative capacity). After applying Eq. 1 to each two-dimensional data series within a data set, the values of β_1 from each cell can be combined into a one-dimensional vector, $\boldsymbol{\beta}_1$. A global model is then formed by identifying some function that predicts the variation of β_1 across the data set as a function of any time-invariant experimental variables (temperature and SOC). Identifying this function, or sub-model, is a dimensional reduction, compressing unique sets of parameters for each cell into a single set of parameters shared by all cells. Dimension reduction is a required step when fitting a cell test data set without a pre-determined global model, as local data series are not necessarily of the same size. The identified global model can then be optimized to all data series simultaneously and extrapolated both forward in time and to unseen conditions. Battery behavior in real-world applications, which may not have any constant variables, can be predicted by deriving the instantaneous derivative of the global model and integrating over discrete timesteps by determining scalar values for each input variable at each timestep.[11,17,19,20]

The most common approach in literature for modeling the SEI growth rate (β_1 in Eq. 1) is to use Arrhenius- and Tafel-like descriptors.[14–19,22,23,53,56] Arrhenius relationships are used to model the rate of chemical reactions, which for simple cases that can neglect reactant and product concentrations only requires consideration of the reaction temperature. Tafel relationships are used to model the rate of electrochemical reactions, which require consideration of the interactions between electrochemical potential and temperature. A common model structure using Arrhenius and Tafel descriptors is Refs. 17, 53:

$$\boldsymbol{\beta}_1(\boldsymbol{\gamma}, \boldsymbol{T}, \boldsymbol{U_a}) = \gamma_0 \exp\left(-\frac{\gamma_1}{R}\left(\frac{1}{\boldsymbol{T}} - \frac{1}{T_{ref}}\right)\right) \times \exp\left(\frac{\gamma_2 n_e F}{R}\left(\frac{\boldsymbol{U_a}}{\boldsymbol{T}} - \frac{U_{a,ref}}{T_{ref}}\right)\right) \qquad [2]$$

where $\boldsymbol{\gamma}$ is the vector of parameters for the sub-model, of which γ_i is the ith parameter, R is the universal gas constant, $\boldsymbol{T}$ is the vector of experimental temperatures, n_e is the number of electrons of the rate limiting reaction, usually assumed to be equal to 1, F is Faraday's constant, $\boldsymbol{U_a}$ is the vector of experimental anode-to-reference potentials, and T_{ref} and $U_{a,ref}$ are reference values for temperature and anode-to-reference potential, respectively. These reference values aid interpretability but have no impact on model performance; β_1 is equal to γ_0 at $T = T_{ref}$ and $U_a = U_{a,ref}$. The anode-to-reference potential, U_a, is a function of SOC, determined using graphite/Li half-cell measurements or full-cell measurements with a reference electrode.[60] The Arrhenius and Tafel descriptors can be simplified by removing constants and reference values:

Table I. Proposed reduced-order local cell degradation models from literature.

References	Description	Equation
Various authors[15,23]	Linear	$q = 1 - \beta_1 t$
Schmitt, 2017[12]	Linear	$q = \beta_0 - \beta_1 t$
Various authors[10,11,18,19,23,35,43,53–57]	Square-root	$q = 1 - \beta_1 t^{0.5}$
Various authors[10,12,15,17,58]	Square-root	$q = \beta_0 - \beta_1 t^{0.5}$
Various authors[43,57]	Square-root + linear	$q = 1 - \beta_1 t^{0.5} - \beta_2 t$
Schmalstieg, 2014[14]	Power law	$q = 1 - \beta_1 t^{0.75}$
Grolleau, 2014[22]	Power law	$q = 2 - (1 + \beta_1 t)^{\beta 2}$
Various authors[7,10,59]	Power law	$q = 1 - \beta_1 t^{\beta 2}$
Various authors[10,12,47]	Power law	$q = \beta_0 - \beta_1 t^{\beta 2}$
Attia, 2020[10]	Power law	$q = 1 - \beta_1 (t + \beta_3)^{\beta 2}$
Attia, 2020[10]	Power law	$q = \beta_0 - \beta_1 (t + \beta_3)^{\beta 2}$
Attia, 2020[10]	Logarithm	$q = 1 - \beta_1 \ln(t + \beta_2)$
Various authors[10,43]	Logarithm	$q = \beta_0 - \beta_1 \ln(t)$
Ecker, 2012[43]	Square-root + log	$q = \beta_0 - \beta_1 t^{0.5} - \beta_2 \ln(t)$
Attia, 2020[10]	Logarithm	$q = \beta_0 - \beta_1 \ln(t + \beta_2)$
Various authors[42,46,47]	Stretched exponential	$q = \exp(-(\beta_1 t)^{\beta 2})$, $\beta_2 \in \left\{1, 0.745, \frac{3}{5}, \frac{1}{3}\right\}$
Gering, 2017[13]	Sigmoidal	$q = \beta_0 - 2\beta_1\left(\frac{1}{2} - \frac{1}{1 + \exp((\beta_3 t)^{\beta 2})}\right)$

$$\beta_1(\gamma, \boldsymbol{T}, \boldsymbol{U_a}) = \gamma_0 \exp\left(\gamma_1 \frac{1}{\boldsymbol{T}}\right) \exp\left(\gamma_2 \frac{\boldsymbol{U_a}}{\boldsymbol{T}}\right) \quad [3]$$

This model is a two-dimensional model, where the input features T and U_a are used to construct two descriptors, $1/T$ and U_a/T. Literature review in the following paragraph is conducted by simplifying models from literature in a similar manner. It is worth noting that the multiplicative model shown in Eq. 3 can be reframed as a linear model (Eq. 4) by taking the natural log of β_1, so that rather than using a non-linear optimization algorithm, linear optimization algorithms can be utilized.[47] Not all multiplicative models can be transformed into linear models.

$$\log(\beta_1(\gamma, \boldsymbol{T}, \boldsymbol{U_a})) = \log(\gamma_0) + \gamma_1 \frac{1}{\boldsymbol{T}} + \gamma_2 \frac{\boldsymbol{U_a}}{\boldsymbol{T}} \quad [4]$$

Various other models have been proposed for predicting the calendar fade rate of LIBs with graphite anodes in the literature. These models are shown in Table II. Even when only modeling temperature effects, various models have been proposed (Eqs. 5 and 6). For incorporating the additional impact of cell SOC on degradation rate, several different input features have been proposed: U_a (Eqs. 7 and 8), SOC (Eqs. 9–12), and the full cell potential V (Eqs. 13 and 14). These models also propose widely varying structures, such as multiplicative models (Eqs. 3, 5, 11, 14), linear combinations of complex descriptors (Eqs. 9, 12, 13), and non-linear models (Eqs. 6, 7, 8, 10). Most models use exponential functions to predict non-linearities, though one model also includes a polynomial descriptor (Eq. 12), and another (Eq. 14) uses power terms.

The argument against using physically informed Arrhenius and Tafel descriptors is essentially the same as that described previously against modeling the capacity fade of local cells using $t^{0.5}$: the physically informed model is very accurate for ideal systems in specific conditions, but may not be statistically or analytically validated. Given the wide variety of models present in the literature, varying from simple modifications of the Arrhenius and Tafel descriptors in Eq. 2 to models without either descriptor present at all (Eqs. 8, 13, 14), it is obvious that model structure is crucial to model performance, and model structure is most readily justified via statistical methods. A good example of justifying the structure of an empirically identified parameter sub-model is shown in work by Mathieu et al.,[47] who used p-values to determine that all but one of several possible descriptors were statistically significant. Similarly, Rahimian et al. [53] used a t test to evaluate the significance of parameters in their identified model. However, without thorough statistical validation, it is not possible to reach a detailed understanding of model behavior or predictive accuracy compared to other potential model structures.

Approach for Statistically Validating Model Selection

As this study aims to generalize the work by Attia et al.[10] from individual cell models to reduced-order models of cell aging data sets, it is worth repeating their approach for determining a statistically valid reduced-order model, with some editorializing and the addition of one further step concerning cross-validation, which helps to evaluate whether models are over or under parameterized for a given data set.

1. Compare multiple models for each data series to investigate model structure: Multiple models of varying complexity and structure need to be investigated. The relative optimality of a model is only known by comparison to other models.
2. Include confidence intervals to evaluate model uncertainties: Confidence intervals quantify the uncertainty of model predictions and are necessary for investigating model behavior. Confidence intervals can be calculated by a variety of methods, including using F tests, bootstrap resampling, jackknife resampling, Bayesian regression approaches, and others. Bootstrap resampling, which optimizes a model many times using data sets constructed by randomly resampling with replacement from the original data set for each iteration, is an effective method for accurately measuring model uncertainty and determining underlying model behavior for fast optimizing models.[61] Another rigorous method for determining parameter distributions, and thus confidence intervals, is to use a Bayesian approach, which assumes a prior distribution for the values of the model parameters. Compared with bootstrap resampling, which is a "frequentist" approach, Bayesian methods may be sensitive to the selection of the priors but give more information on the relationships between dependent parameters and may be more likely to find a global optimum when there are local minima present in the loss function.[62,63] In any case, both approaches have merits and drawbacks, and it is up to the experience of the researcher to pick a method, and consider their results carefully based on the merits and drawbacks of their chosen methodology.
3. Include various fit metrics to investigate model quality: A wide variety of fit metrics can be utilized, each with their own qualities. The R^2 metric accurately reports the collinearity of the data and a model, but it does not consider the extent of model

Table II. Proposed calendar fade rate models from literature.

Reference	Equation	Eq. #
Ploehn, 2004[35]	$\beta_1(\gamma, \boldsymbol{T}) = \gamma_1 \exp\left(\gamma_2 \frac{1}{\boldsymbol{T}}\right)$	[5]
Belt, 2011[15]	$\beta_1(\gamma, \boldsymbol{T}) = \exp\left(\gamma_1 + \gamma_2 \frac{1}{\boldsymbol{T}}\right)$	[6]
Schimpe, 2018[18]	$\beta_1(\gamma, \boldsymbol{T}, \boldsymbol{U_a}) = \gamma_0 \exp\left(\gamma_1 \frac{1}{\boldsymbol{T}}\right)\left(\gamma_2 + \exp\left(\gamma_3 \frac{\boldsymbol{U_a}}{\boldsymbol{T}}\right)\right)$	[7]
Rumberg, 2020[16]	$\beta_1(\gamma, \boldsymbol{T}, \boldsymbol{U_a}) = \gamma_0 \exp\left(\frac{\gamma_1 + \gamma_2 \boldsymbol{U_a}}{\boldsymbol{T}}\right)$	[8]
Grolleau, 2014[22]	$\beta_1(\gamma, \boldsymbol{T}, \boldsymbol{SOC}) = \gamma_0 \exp\left(\gamma_1 \frac{1}{\boldsymbol{T}}\right) \cdot \boldsymbol{SOC} + \gamma_2 \exp\left(\gamma_3 \frac{1}{\boldsymbol{T}}\right)$	[9]
Redondo-Iglesias, 2017[23]	$\beta_1(\gamma, \boldsymbol{T}, \boldsymbol{SOC}) = \gamma_0 \exp(\gamma_1 \cdot \boldsymbol{SOC}) \exp\left(\frac{\gamma_2 + \gamma_3 \boldsymbol{SOC}}{\boldsymbol{T}}\right)$	[10]
Sarasketa-Zabala, 2015[56]	$\beta_1(\gamma, \boldsymbol{T}, \boldsymbol{SOC}) = \gamma_0 \exp\left(\gamma_1 \frac{1}{\boldsymbol{T}}\right) \cdot \gamma_2 \exp(\gamma_3 \cdot \boldsymbol{SOC})$	[11]
Naumann, 2018[19]	$\beta_1(\gamma, \boldsymbol{T}, \boldsymbol{SOC}) = \gamma_0 \exp\left(\gamma_1 \frac{1}{\boldsymbol{T}}\right)(\gamma_2 + \gamma_3 \cdot (\boldsymbol{SOC} - 0.5)^3)$	[12]
Schamlstieg, 2014[14]	$\beta_1(\gamma, \boldsymbol{T}, \boldsymbol{V}) = \exp\left(\gamma_3 \frac{1}{\boldsymbol{T}}\right)(\gamma_1 + \gamma_2 \cdot \boldsymbol{V})$	[13]
Ecker, 2012[43]	$\beta_1(\gamma, \boldsymbol{T}, \boldsymbol{V}) = \gamma_1^{\boldsymbol{T}} \gamma_2^{\boldsymbol{V}}$	[14]

parameterization, leading to a preference for overparameterized models. Other fit metrics, such as R^2_{adj}, mean-squared error (MSE) (in the context of model regression, referring to the squared residuals divided by the degrees-of-freedom (DOF)), mean signed difference (MSD), mean absolute error (MAE), and mean absolute percent error (MAPE) are also useful for evaluating model quality, among many others. For example, the MAE is useful as it reports the average error in the same units as the response variable, but when predicting a response variable that varies over several orders of magnitude, the MAE will not be influenced by the relative error at low magnitudes. The MAPE, by reporting the average relative error of each data point, more fairly represents the predictive quality of a model in such a case. The MSD preserves the sign of the error, making it a useful metric for identifying confidence intervals, or determining whether the model is consistently under- or over-predicting the response variable. Mathematical expressions used in this work to calculate R^2, R^2_{adj}, DOF, MSD, MAE, MAPE, and MSE are defined in the Appendix.

4. Include residuals analysis to reveal systematic trends in model behavior: Residuals analysis is critical for revealing systematic trends in the fit of a model, as the behavior of models that are highly collinear with their data (high R^2) often cannot be interrogated by simply plotting the data and the fit because model error might be several orders of magnitude smaller than the model predictions. Additionally, calculation of fit metrics reduces the dimensionality of the data, obscuring systematic trends of the residuals.
5. Include cross-validation to estimate model predictive quality: Cross-validation is conducted by iteratively training a model on portions of the data set and calculating the prediction error on the held-out data. Cross-validation is a first step when evaluating the predictive quality of a model, and is complemented by other methods demonstrated in this work: testing models against unseen validation or extrapolation data sets, quantifying model sensitivity to its inputs, and quantifying model uncertainty.

These statistical validation steps are important when models are developed manually, using known physical behaviors to inform model structure, and that importance is heightened when model structure is determined automatically, as is done in this work. The behavior of automatically identified models can reflect unforeseen trends in the data set or behave non-physically in unexpected ways, especially when trained on small data sets, so diligent interrogation of automatically identified models is a necessity.

Procedure for Identifying Optimal Reduced-Order Degradation Models

A schematic of the optimization and analysis procedure used in this work is shown in Fig. 1. Three steps are involved for optimizing each model: 1) Use bi-level optimization to simultaneously regress both local (β) and global parameters (α) to each cell's data series across a data set, 2) automatically identify and regress sub-models for each local parameter, and 3) assemble the global model by replacing local parameters with their respective sub-models and regressing. Any interesting or promising models are then interrogated in step 4) via sensitivity analysis, bootstrap resampling, and extrapolation over long times and to unseen validation data.

Bi-level optimization.—Bi-level optimization was conducted to simultaneously optimize global and local parameters. This was done by nesting n local optimization loops, within which a model with N free local parameters ($\beta_{1:N}$) is optimized, within each iteration of a global optimization loop. The values of each of M global parameters ($\alpha_{1:M}$) are fed down from the global optimization loop into each cell's local optimization loop. The local cell model predictions are then concatenated into a single vector to calculate the error across the whole data set for the global optimization loop. Both optimization loops were solved by unconstrained non-linear least squares (NLLS) optimization using the Levenberg-Marquardt algorithm by the MATLAB function *nlinfit*. Cross-validation error was calculated using leave-one-out cross-validation on a cell-by-cell basis, essentially cross-validating the global parameters. If there are no global parameters, the optimization algorithm is only single-leveled, and cross-validation was conducted using hold-one-out cross-validation on each cells' data series independently. The cross-validation error from each cell was then summed to determine the error on the entire data set.

Identification of local parameter sub-models.—Local parameter sub-models are used to predict the value of local parameters as functions of any time-invariant input features. Sub-models were identified by first generating a library of descriptors from the input features, then down selecting to an optimal set of descriptors from these, a process known as symbolic regression.[64] Several approaches exist for conducting symbolic regression, notably genetic programming [65–69] and compressed sensing (also known as sparse regression) methods.[70–77] This work takes the latter approach, utilizing LASSO ℓ_1-norm regularization to identify subsets of descriptors from the descriptor library. LASSO is not necessarily the best algorithm for this approach; Ouyang et al. [72] review the benefits and limitations of various sparse regression methods, and demonstrate that several may be more effective than LASSO. The main drawback of LASSO is that it is not guaranteed to converge on an optimal solution for small data sets when the number of generated descriptors is very large (for example, there are $\sim 10^{12}$ possible 5-dimensional models in a library of 1000 descriptors).[71,72] LASSO was chosen in this work over other approaches because it is natively implemented in MATLAB, the generated descriptor libraries are small ($\sim$100 descriptors), and the procedure resulted in models that substantially improved upon physically informed models.

Generation of the descriptor libraries was conducted according to a procedure inspired by prior works.[71,75] For modeling the calendar aging of lithium-ion batteries with graphite electrodes, the salient input features are ambient temperature, SOC, and U_a, which were separated into two groups: temperature related, and voltage related. Next, several operators were defined for generating descriptors. Unary operators are operators that are applied to a single feature or descriptor and generate a new descriptor. The unary operators used in this work to modify any feature X are $X^{0.5}$, X^2, X^3, X^{-1}, and e^X. Binary operators combine two features or descriptors from different groups and create a new group. The only binary operator used in this work is element-wise multiplication of two groups, $X_A \cdot X_B$.

Two separate descriptor libraries were generated for identifying linear and multiplicative models. The procedure for applying operators to generate each descriptor library is shown in Table III. As noted previously, multiplicative models can be identified by fitting a linear model to the natural log of a local parameter, and then taking the exponential of the identified linear model. This procedure can identify models with multiplication between exponential or power terms. The transformation of a linear model to a multiplicative model is shown for an arbitrary sub-model in the following equations:

$$\exp(\log(\boldsymbol{\beta})) = \exp(\gamma_0 + \gamma_1 \boldsymbol{X}_1 + \gamma_2 \log(\boldsymbol{X}_2) + \ldots) \quad [15]$$

$$\boldsymbol{\beta} = \exp(\gamma_0)\exp(\gamma_1 \boldsymbol{X}_1) \cdot \boldsymbol{X}_2^{\gamma_2} \cdot \ldots \quad [16]$$

No power terms were identified in this work to keep the descriptor library small, and because accurate model predictions were achieved without any. Note that this procedure cannot be used to generate models with linear combinations of arbitrary power terms (e.g., $\boldsymbol{X}_1^{\gamma_1} + \boldsymbol{X}_2^{\gamma_2}$), as this requires a non-linear optimization, and LASSO applies only to linear models. By identifying more operators and repeatedly applying them to create tiers of increasingly complex descriptors, this procedure can be used to rapidly generate many thousands of possible descriptors from a small set of input

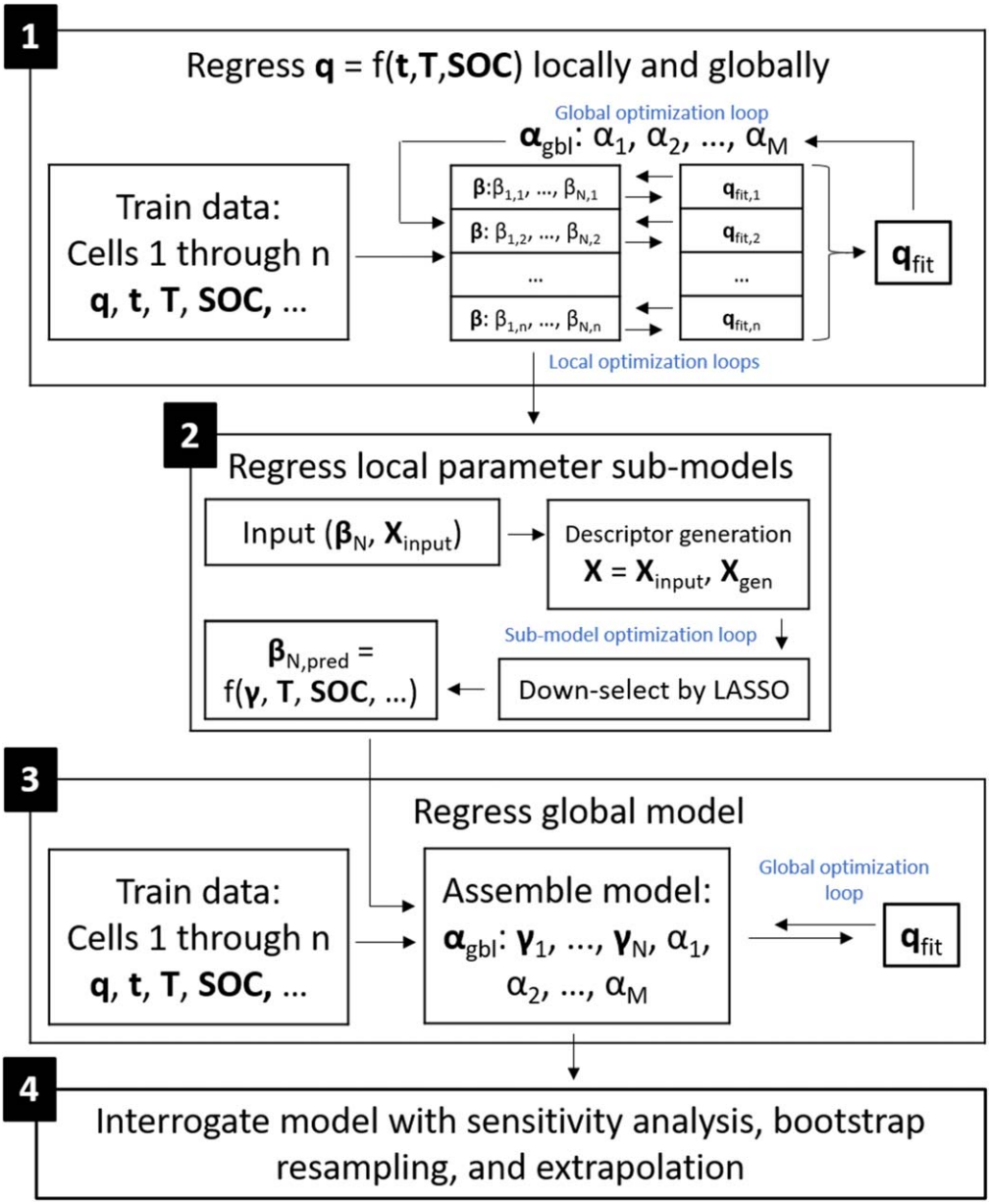

Figure 1. Optimization and analysis procedure.

features. Even with only 110 possible descriptors, there are 215,820 potential 3-dimesional models, so searching for the optimal sub-model for only a handful of model structures can easily exceed over one million possible equations; if sub-models can have 5 or 6 descriptors, the number of potential equations becomes enormous. LASSO does not require a limit on the number of optimal model descriptors, however, it was observed during the implementation of this method that no sub-models converged with more than 6 descriptors.

Down selection of descriptors from the generated libraries was conducted using the MATLAB function *lasso*, which implements LASSO ℓ_1-norm regularization. The LASSO algorithm requires the input of a hyperparameter, λ, which sets the magnitude of the penalty imposed on the number of non-zero model parameters. In this work, λ was optimized by a grid-search over logarithmically spaced values with 200 points between 10^{-6} and 10^{0}. Model error at each λ was quantified by the MSE after cross-validation (MSE_{CV}) using 4-fold cross-validation. LASSO was used not to find the model with the lowest possible MSE_{CV}, but rather to find the model with the fewest parameters that has MSE_{CV} equal to the lowest MSE_{CV} plus one standard deviation, referred to as the "1SE" model in the MATLAB documentation.[78] This was done because the lowest MSE_{CV} model often contains many parameters, but lower-dimensional models are preferred for this work to aid interpretability and avoid over parameterization. The MATLAB function *lasso* can also implement the elastic-net regularization algorithm, which can help balance model sparsity with tolerance to noise or correlated input data; elastic-net was not used here due to the complexity of finding the optimal values for two hyperparameters, but will be considered in the future, along with other sparse regression algorithms.

After identification, the behavior of local parameter sub-models was investigated by cross-validation and bootstrap resampling. NLLS optimization utilizing the Levenberg-Marquardt algorithm via the MATLAB function *nlinfit* was used for calculation cross validation error and quantifying model uncertainty via bootstrap resampling. Hold-one-out cross-validation was conducted to calculate MSE_{CV}. Model uncertainty was quantified by 1000 iterations of bootstrap resampling and model optimization. Confidence intervals were identified from the bootstrap iterations using the MSD, which is simply the sum of all residual errors divided by the number of data

Table III. Procedure for generating descriptor libraries used for automatic identification of local parameter sub-models.

Step	Description	Formula	Example	# descriptors
1	Input features	$X_A(T),X_B(SOC,U_a)$	T,SOC,U_a	3
2	Square-root, square, and cube of input features	$X^{0.5}$, X^2., X^3	T^2, U_a^3, $SOC^{0.5}$	10
3	Inverse of all descriptors	X^{-1}	$1/T$, $1/SOC^3$	20
4	Multiplication of features between groups	$X_1 \cdot X_2$	U_a^2/T, $T^2/SOC^{0.5}$	84
5 (omitted for mult. models)	Exponential of all descriptors	e^X	$\exp(T)$, $\exp(U_a^{0.5})$, $\exp(SOC^2/T^3)$	168
6	Remove any descriptors with infinite or NaN values			110 (linear), 64 (mult.)

points, giving the under- and over-predictions at the desired percentiles. Unless otherwise specified, 90% confidence intervals are used instead of 95% confidence intervals, as is often reported, because random sampling from small data sets can result in a higher proportion of divergent models that do not aid interpretation of model behavior.

Global model optimization and interrogation.—A global model is created by replacing each local parameter with its respective sub-model, forming a single equation that predicts the capacity fade of for the entire data set. Global models were optimized using NLLS fitting utilizing the Levenberg-Marquardt algorithm via the MATLAB function *nlinfit*. Cross-validation error was calculated using leave-one-out cross-validation on a cell-by-cell basis. Sensitivity analysis was conducted by adding Gaussian random noise to each input vector, optimizing the model on the noisy data, and normalizing the MSE of the model trained on noisy input data by the MSE of the unperturbed model. Random noise with magnitudes of 0.001, 0.01, 0.03, 0.05, 0.1, 0.13, and 0.3 times the standard deviation of the input vector being perturbed was used here. Because random noise can occasionally cause extremely poor fit results, the median MSE from 10 iterations is reported.

Confidence intervals for the global model prediction were determined by 1000 iterations of bootstrap resampling and model optimization. 90% confidence intervals of model predictions were identified for each cell's data series separately, using the MSD to calculate percentiles. Determining confidence intervals must be done on a cell-by-cell basis because the set of parameter values that correspond to the confidence intervals at one experimental condition will likely not correspond to the confidence intervals at another. To put it another way, the vectors of bootstrapped parameters that result in the greatest under- or over-prediction for the entire data set do not result in the greatest under- or over-prediction for each cell individually. The results from bootstrapping can also be used to quantify model uncertainty when simulating capacity fade in completely new test conditions in the same manner.

Data

Train, validation, test, and extrapolation data sets.—The data set used in this study was previously published by Schimpe et al.[18] Their work reported calendar aging data from 22 commercial LFP/graphite 26650 format cells (Sony US26650FTC1). Cells were calendar aged at temperatures of 10 °C, 15 °C, 25 °C, 35 °C, 45 °C, and 55 °C, with SOC variation at 15 °C and 45 °C between 0%, 12.5%, 25%, 37.5%, 50%, 67.5%, 75%, 87.5%, and 100%. Cells at other temperatures were stored at 100% SOC. Cell discharge capacity was measured during a reference performance test (RPT) procedure using a CC–CV discharge (1C, C/50 cut-off). After the beginning of life RPT, RPTs were conducted with decreasing frequency as the cells aged, with the first RPT conducted at 7 d, and the duration between later RPTs up to 6 weeks. The average number of RPTs in each cell data series is 10.5 over a period of ~8 months. See their work for other testing details.[18] Data points were extracted from published figures using the WebPlotDigitizer.[79] Data was drawn in duplicate from several data series to estimate error in the data extraction process; across 3 data series, the MAPE between extracted data points was only 0.06%, so noise from data extraction error is considered negligible (see Supplementary Material). The anode-to-reference potential, U_a, at a given SOC was used as an additional input feature. U_a was calculated using the formula by Safari and Delacourt,[60] as used by Schimpe et al.[18] when modeling this data set (reproduced in the Appendix). The relative discharge capacity over time for each of the 22 cells in the data set is shown in Fig. 2. A.mat file and an excel table with all input data used during the modeling process are provided in the Supplementary Material.

The data set is broken up into train, validation, test, and extrapolation sets. Segmenting of the data set into these subsets is not done randomly, but rather to aid interpretation of model behavior. In general, construction of these subsets needs to be done with consideration for the intended use of the model as well as the qualities of the data set; data sets can contain biases, and constructing these subsets completely at random can hide biases within a data set.[29] In this work, the validation set is used to explore the *extrapolation of models with respect to (w.r.t.) time*, which is critical for battery control systems or technoeconomic models. The data in the training sets and validation set overlap, which is unusual, but is done so that models trained on different portions of the data are validated on the same data and can be compared fairly, while still utilizing most of the limited amount of data for training. Comparing the error of all models on the validation set is used to determine the "hyperparameter" of model structure; model structure being the choice of the time dependent equation ($t^{0.5}$, power law, stretched exponential, or sigmoidal), and the selection of each parameter as local or global. The test and extrapolation sets contain data that is unseen during model training. The test set is used to explore the *interpolation of models w.r.t. experimental conditions*, and the extrapolation sets are used to explore the *extrapolation of models w.r.t. experimental conditions.* No data subset explores the *interpolation of models w.r.t. time*, as the time variable is sampled quite heavily, so it is assumed that models will interpolate well w.r.t. time.

To construct these data subsets, test data is first withheld from 6 randomly selected cells that have a varied distribution of temperatures and SOCs: cells 6, 8, 12, 14, 18, and 20. Training sets are then formed from the remaining 16 cells by separating out data points from each cell's data series, using the first 2, 4, 6, 9, and finally all the available RPTs. The validation set is composed of all available RPTs from each of the 16 cells. The convergence behavior of models is observed by comparing model performance on the test set vs the number of RPTs used for training, as well as by training models using data from only 4, 5, 6, 8, or 12 cells of the training set and comparing the prediction error on the test set. Two extrapolation sets are also formed to investigate whether the structure of the identified global models can infer the behavior of unseen experimental conditions with shared physical relationships. The first extrapolation set is composed of six cells from the test set with the highest fade rates: cells 12, 18, 19, 20, 21, and 22. The second extrapolation set is composed of six cells from the test set with low SOCs: cells 2, 3, 4, 13, 14, and 15.

Results

Investigated models.—Four model types (square-root, power law, stretched exponential, and sigmoidal) were used to create fifteen distinct models to investigate in this work, detailed in Table IV. Square-root and power law models are widely used throughout the literature. Stretched exponential models contain one more parameter than power-law models, and have a constant bound at positive infinity, ensuring the model will not extrapolate to infinitely large degradations. Sigmoidal models have the same number of parameters and the same qualitative behavior as stretched exponential models but have constant bounds at both positive and negative infinity, so they may behave differently.

The first investigated model is the square-root model reported in many prior studies,

$$q = 1 - \beta_1 t^{0.5} \quad [17]$$

Then, for each of the power law, stretched exponential, and sigmoidal type equations, several models are defined with increasing number of local parameters. The final model of each type enforces no global parameters. The total number of free model parameters for bi-level optimization is equal to:

$$p = M + n \cdot N \quad [18]$$

where M is the number of global parameters, n is the number of cells in the data set, and N is the number of local parameters. Models with more local parameters will have more total parameters and fewer

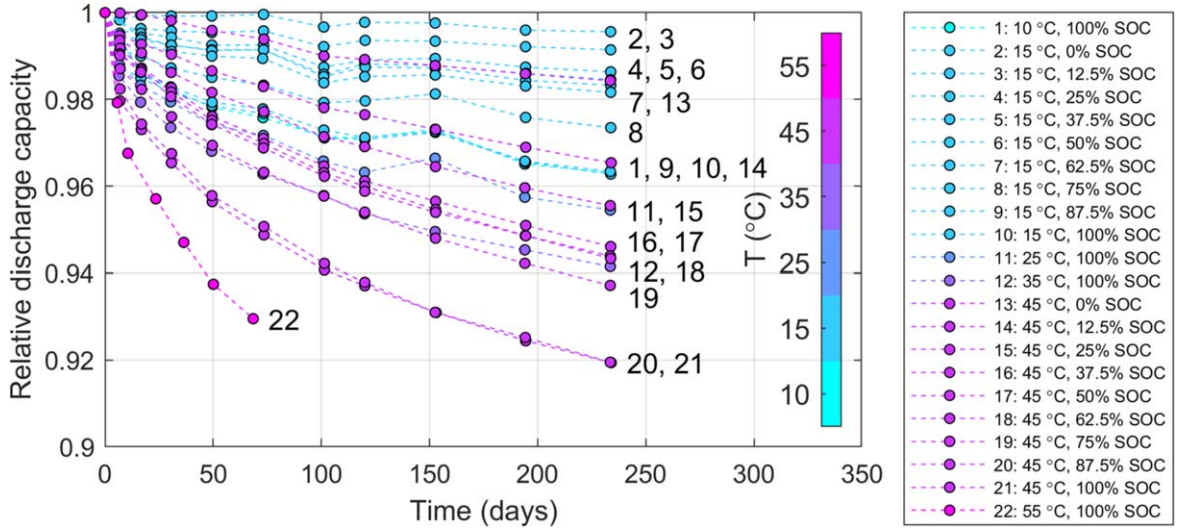

Figure 2. Relative discharge capacity vs time during calendar aging of LFP/graphite cells from Schimpe et al.[18] Cells at high temperature and high SOC degrade much more rapidly than cells at low temperature and low SOC. Several data series at differing conditions result in similar degradation, indicating that multiple temperature and SOC effects impact degradation.

Table IV. Model structures studied in this work.

Model ID	Description	Model equation	# of global params (α)	# of local params (β)	Total # params
1	Square-root	$q = 1 - \beta_1 t^{0.5}$	0	1	16
2	Power law	$q = \alpha_0 - \beta_1 t^{\alpha_2}$	2	1	18
3	Power law	$q = \alpha_0 - \beta_1 t^{\beta_2}$	1	2	33
4	Power law	$q = \beta_0 - \beta_1 t^{\alpha_2}$	1	2	33
5	Power law	$q = \beta_0 - \beta_1 t^{\beta_2}$	0	3	48
6	Stretched exponential	$q = \alpha_0 - \beta_1(1 - 1/\exp((\alpha_2 t)^{\alpha_3}))$	3	1	19
7	Stretched exponential	$q = \alpha_0 - \beta_1(1 - 1/\exp((\beta_2 t)^{\alpha_3}))$	2	2	34
8	Stretched exponential	$q = \alpha_0 - \beta_1(1 - 1/\exp((\alpha_2 t)^{\beta_3}))$	2	2	34
9	Stretched exponential	$q = \alpha_0 - \beta_1(1 - 1/\exp((\beta_2 t)^{\beta_3}))$	1	3	49
10	Stretched exponential	$q = \beta_0 - \beta_1(1 - 1/\exp((\beta_2 t)^{\beta_3}))$	0	4	64
11	Sigmoidal	$q = \alpha_0 - 2\beta_1\left(\frac{1}{2} - \frac{1}{1+\exp((\alpha_2 t)^{\alpha_3})}\right)$	3	1	19
12	Sigmoidal	$q = \alpha_0 - 2\beta_1\left(\frac{1}{2} - \frac{1}{1+\exp((\beta_2 t)^{\alpha_3})}\right)$	2	2	34
13	Sigmoidal	$q = \alpha_0 - 2\beta_1\left(\frac{1}{2} - \frac{1}{1+\exp((\alpha_2 t)^{\beta_3})}\right)$	2	2	34
14	Sigmoidal	$q = \alpha_0 - 2\beta_1\left(\frac{1}{2} - \frac{1}{1+\exp((\beta_2 t)^{\beta_3})}\right)$	1	3	49
15	Sigmoidal	$q = \beta_0 - 2\beta_1\left(\frac{1}{2} - \frac{1}{1+\exp((\beta_2 t)^{\beta_3})}\right)$	0	4	64

DOF, which can lead to over-fitting. Here, DOF refers to the total number of data points in the data set minus the total number of model parameters. The total DOF in the data set is equal to the total number of data points; for the 16 cells of the train/test set, the total DOF is 168.

Individual cell models after bi-level optimization.—The MSE and MSE_{CV} after bi-level optimization (before symbolic regression) are plotted Fig. 3. The plot clearly shows regions of under-parameterized, well-parameterized, and over-parameterized models; models of each type appear in every region, so model structure plays no role in determining whether the model is well-parameterized. Under-parameterized models, such as Models 1, 2, 6, and 11, have only one local parameter each and higher MSE than models with more parameters. Notably, Model 1, which is the $t^{0.5}$ model widely present in the literature, has the largest MSE of any model. Models with a relative DOF of 0.8 appear to be well-parameterized for this data set; their MSE is low, and they show almost no difference between their best fit MSE and MSE_{CV}. For these well-parameterized models, model structure plays a role in the quality of the model fit. For example, consider Models 3 and 4, both power law models with one global parameter and two local parameters, but Model 4 has almost double the MSE of Model 3. Models with a DOF ratio less than 0.8, which have more local parameters, are clearly over-parameterized. Two details make this obvious: adding more parameters has not improved the MSE, and the MSE_{CV} begins to increase dramatically.

The convergence of bi-level optimized models when trained on increasing number of RPTs per cell is shown for the sigmoidal models in Fig. 4. The behavior of the power law and stretched exponential models, shown Fig. S1 (available online at stacks.iop.org/JES/168/020502/mmedia) in the Supplementary Material, follow a nearly identical trend. As models are trained on more data, their error when extrapolating to the validation set decreases, reaching a

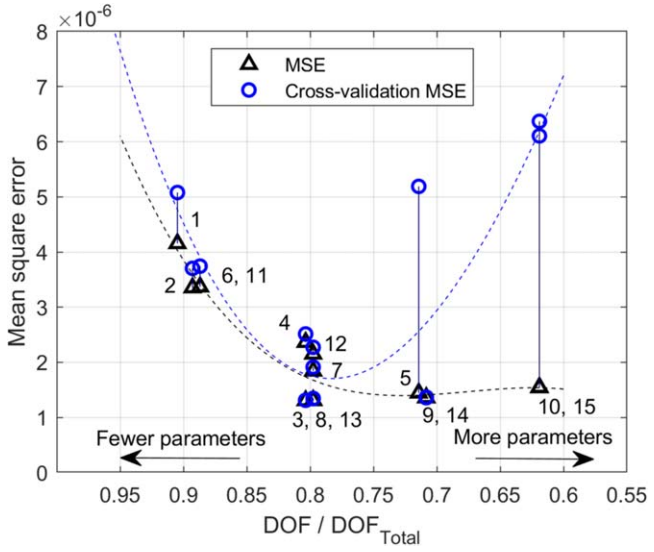

Figure 3. MSE and MSE_{CV} of models after bi-level optimization on the 16 cells of the training set, using all available RPTs. Polynomial trendlines have been added to each data series to guide the eye.

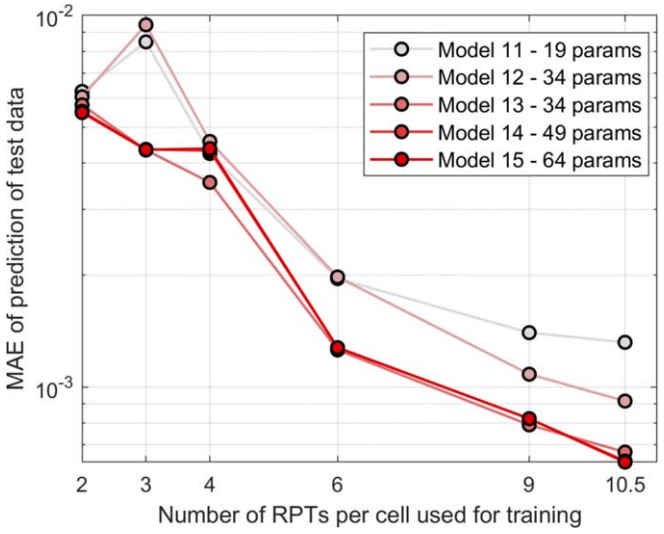

Figure 4. Convergence of bi-level optimized sigmoidal models when trained on increasing number of RPTs per cell.

minimum when trained on all available data (where the training set and the validation set are identical). Under-parameterized models, such as Model 11, never converge to low error. The well-parameterized (Model 13) and over-parameterized (Models 14 and 15) models exhibit a learning behavior between RPTs 4 and 6, where the MAE suddenly decreases. Well- and over-parameterized models cannot be distinguished from Fig. 4, since the cross-validation error is not shown, highlighting the need to cross-validate models.

Global models after symbolic regression.—The MSE and MSE_{CV} after bi-level optimization, symbolic regression, and global optimization are plotted vs. the relative DOF of each model in Fig. 5. Results in Fig. 5 are substantially different than that in Fig. 3, with few obviously over-parameterized models. This is due to the symbolic regression procedure. For models that were obviously over-parameterized after bi-level optimization, the values of the locally fit parameters vary wildly across the data set and cannot be effectively modeled with any combination of proposed descriptors. In this situation, LASSO returns a constant as the optimal model.

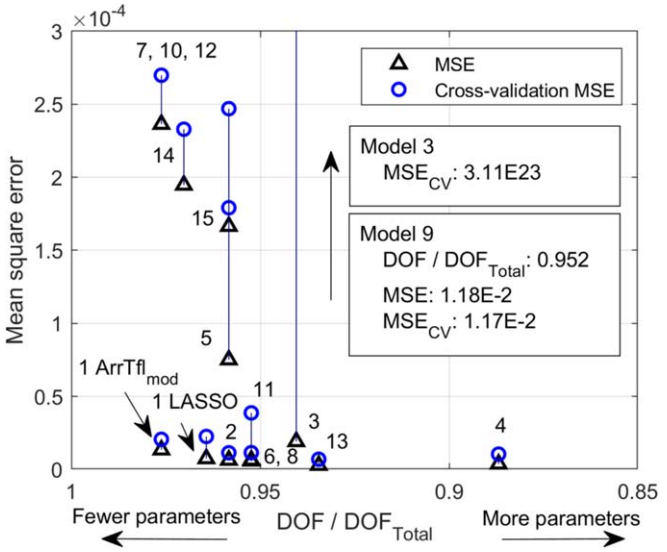

Figure 5. MSE and MSE_{CV} after global optimization on the 16 cells of the training set, using all RPT data. Data points for the MSE_{CV} of Model 3 and the MSE and MSE_{CV} of Model 9 are not shown, as they are orders of magnitude larger than other data points.

This prevents the over-parameterization of the global model, and often results in global models that are under-parameterized (Models 5, 10, 12, and 14). Well-parameterized models are identified for each of the power law, stretched exponential, and sigmoidal model types: Model 2, Model 6, and Model 13, respectively. Of the power law models, Model 2 is preferred over Model 4, because while their error is similar, Model 2 has fewer parameters. Two different global models are investigated for the square-root model, Model 1: "$t^{0.5}$ (LASSO)," which uses an automatically identified β_1 parameter sub-model, and "$t^{0.5}$ ($ArrTfl_{mod}$)," which uses a semi-empirically defined β_1 parameter sub-model consisting of an Arrhenius descriptor and a modified Tafel-like descriptor, which was identified by Schimpe et al. specifically for this data set (Eq. 7).[18] A detailed comparison of these two $t^{0.5}$ models is made in the next section.

The fit quality of some well-parameterized models of the square-root (both LASSO identified and human-expert defined), power law (Model 2), stretched exponential (Model 6), and sigmoidal (Model 13) is shown in Fig. 6. These well-parameterized models are further investigated in the following sections and will be henceforth referred to by their type (e.g., power law). The equations and parameter values for each of these well-parameterized models after training on all RPT data is reported in the Appendix. Immediately apparent is that the sigmoidal model has the best performance, with a MAE of about 0.125% (Fig. 6b) and R^2_{adj} greater than 0.99 (Fig. 6c). The $t^{0.5}$ (LASSO), power law, and stretched exponential models all have similar MAE of about 0.2%, ∼60% more than the sigmoidal model. The $t^{0.5}$ ($ArrTfl_{mod}$) model, which was identified by a human expert, is the worst performing, with a MAE of about 0.25%, about double the MAE of the sigmoidal model. Similar MSE_{CV} between the $t^{0.5}$ ($ArrTfl_{mod}$) and $t^{0.5}$ (LASSO) models (Fig. 6a) suggests that the major source of cross-validation error for both models is the $t^{0.5}$ assumption, rather than the structure of their β_1 parameter sub-models. Histograms of the residual errors from these models are shown in Fig. 6d. The $t^{0.5}$ ($ArrTfl_{mod}$) model has the largest errors, with some errors near ±1%; note that the maximum capacity fade in the training set is only 8%, so this magnitude of error denotes a substantial mismatch between the model and the training data. The $t^{0.5}$ (LASSO) model improves upon the $t^{0.5}$ ($ArrTfl_{mod}$) model, with more data points being fit within a ± 0.25% error margin. The power law model improves upon both iterations of the $t^{0.5}$ models, with most data points being fit within a ± 0.5% margin of error. The stretched exponential model is not shown, as it behaves nearly

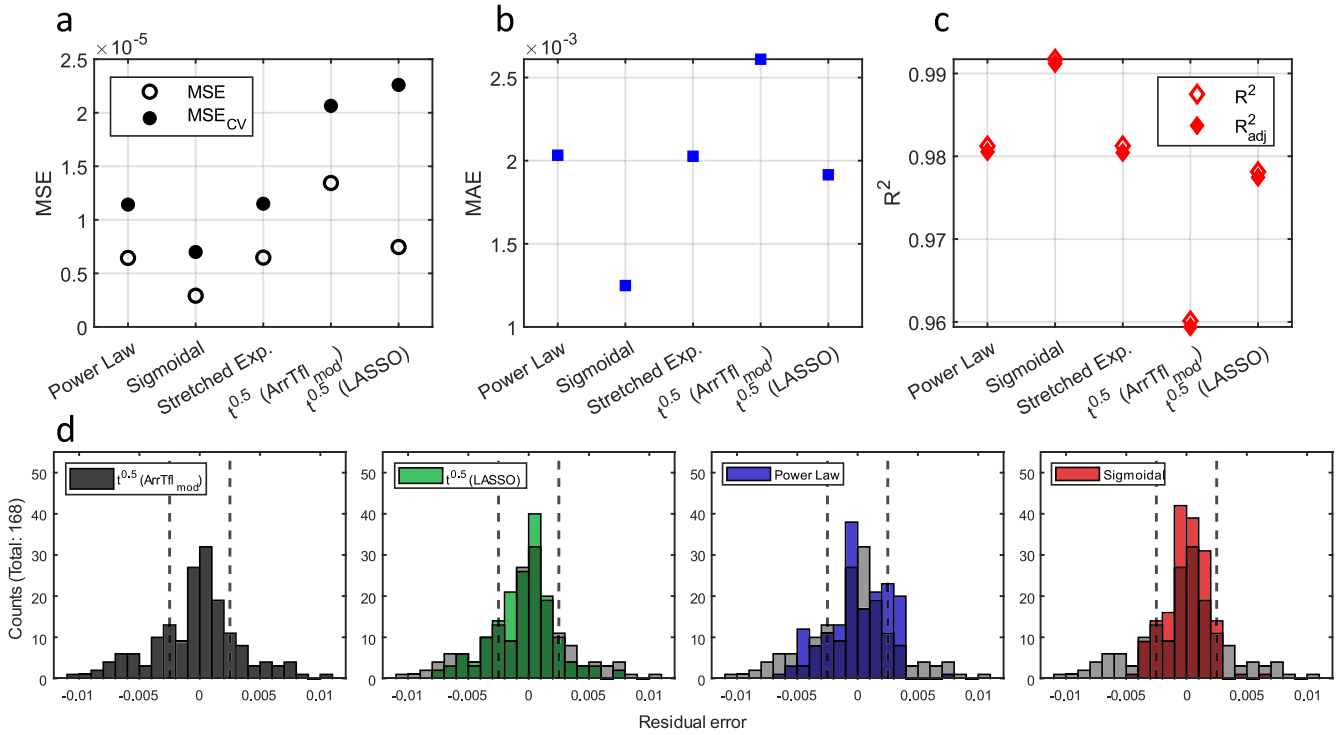

Figure 6. (a) MSE, MSE_{CV}, (b) MAE, (c) R^2, and R^2_{adj} fit metrics for models after global optimization on the 16 cells of the training set using all available RPTs: $t^{0.5}$ model (Model 1), with both LASSO identified (Eq. 19) and the semi-empirically derived $ArrTfl_{mod}$ (Eq. 7) β_1 parameter sub-models, and the best performing power law (Model 2), stretched exponential (Model 6), and sigmoidal models (Model 13). d) Histograms of residual errors for the $t^{0.5}$ ($ArrTfl_{mod}$), $t^{0.5}$ (LASSO), power law, and sigmoidal models. Residual errors from the $t^{0.5}$ ($ArrTfl_{mod}$) model are shown shadowed behind the other models, and vertical dashed lines have been added at ± 0.25% error to aid interpretation.

identically as the power law model. The sigmoidal model clearly shows the best predictive performance, with nearly all data points predicted within a ± 0.25% error margin. Results from the sensitivity analysis (Fig. S2) showed that all models are most sensitive to temperature, followed by U_a. Models with high accuracy (power law, sigmoidal) are more sensitive than the worse performing $t^{0.5}$ model; for this application, more sensitive models are preferred, as it is safe to assume that input features such as temperature can be measured relatively accurately, so more sensitivity to the input temperature is preferred over tolerance to noise.

A comparison of the capacity fade predictions for these models is shown in Fig. 7 (excluding the stretched exponential model, as it behaves nearly identically to the power law model). The sigmoidal model results in the best fit for every case; residual errors of the sigmoidal model show consistent matching of both the slope and magnitude of the relative capacity for each data series, while other models over or under predict degradation (see 15 °C, 100% SOC and 55 °C, 100% SOC) or misjudge the slope of the degradation curve (see 25 °C, 100% SOC and 45 °C, 25% SOC). All models under-predict degradation at 55 °C, 100% SOC, though the power law and sigmoidal models perform substantially better than either $t^{0.5}$ model.

Detailed Analysis

The following subsections analyze models in detail. The impact of using human-expert defined sub-models vs. those identified by symbolic regression is compared using the $t^{0.5}$ model as an example. The behavior of the sigmoidal model, which demonstrates the best performance of any model investigated, is then thoroughly interrogated. The convergence behavior of investigated models is analyzed to determine which features of the data set are learned during training. The impact of input feature selection on the resulting model behaviors is analyzed by comparing power law and sigmoidal models after training with and without U_a as an available feature. Extrapolation of models across test conditions is used to reveal if the automatically identified structure of the models inherently captures physical behaviors, as well as the impact of input feature selection on extrapolation. Finally, extrapolation of models from the ~8 months of training data to 20 years of simulated aging is conducted to magnify differences between models, making obvious the substantial implications of sub-optimal model choice on long term predictions.

Comparison of $t^{0.5}$ global models using LASSO and physics-informed sub-models.—To critically evaluate the impact of automatically identifying local parameter sub-models, the $t^{0.5}$ model is used as a baseline to compare the behavior of sub-models that are derived autonomously (Eq. 19, shown below) vs those derived by human-experts, using both physically informed ("ArrTfl," Eq. 3) and semi-empirical equations ("$ArrTfl_{mod}$," Eq. 7, identified by Schimpe et al. for this data set[18]). The predictions of these sub-models for the local parameter β_1 (Figs. 8a–8c), as well as histograms of the residual errors (Figs. 8d–8f) from each model show somewhat subtle but important differences. The ArrTfl sub-model, being the simplest of the three, captures the qualitative behavior of β_1 but has the worst performance. The residuals plotted in Fig. 8d show that the ArrTfl sub-model predicts only 11 out of the 16 locally fit β_1 values within ± 0.0005 $d^{-0.5}$. The most obvious issue of the ArrTfl sub-model is that as the SOC approaches 0, so does the prediction of β_1, causing substantial error at 0% SOC. The $ArrTfl_{mod}$ sub-model adds a parameter to address this issue, resulting in much better predictions at 0% SOC, and predicts 12 out of the 16 locally fit β_1 values within ±0.0005 $d^{-0.5}$ (Fig. 8e). The LASSO identified sub-model, given by the equation:

$$\beta_1 = \gamma_0 \cdot \exp(\gamma_1 T^2) \exp(\gamma_2 \sqrt{U_a}/T^2) \exp(\gamma_3 T^2/\sqrt{U_a}) \times \exp(\gamma_4/(U_a^2 T^2)) \exp(\gamma_5/(U_a^3 T^2)) \quad [19]$$

shows the best fit quality of the three models, with approximately 70% lower MAPE than the ArrTfl sub-model. The LASSO sub-model

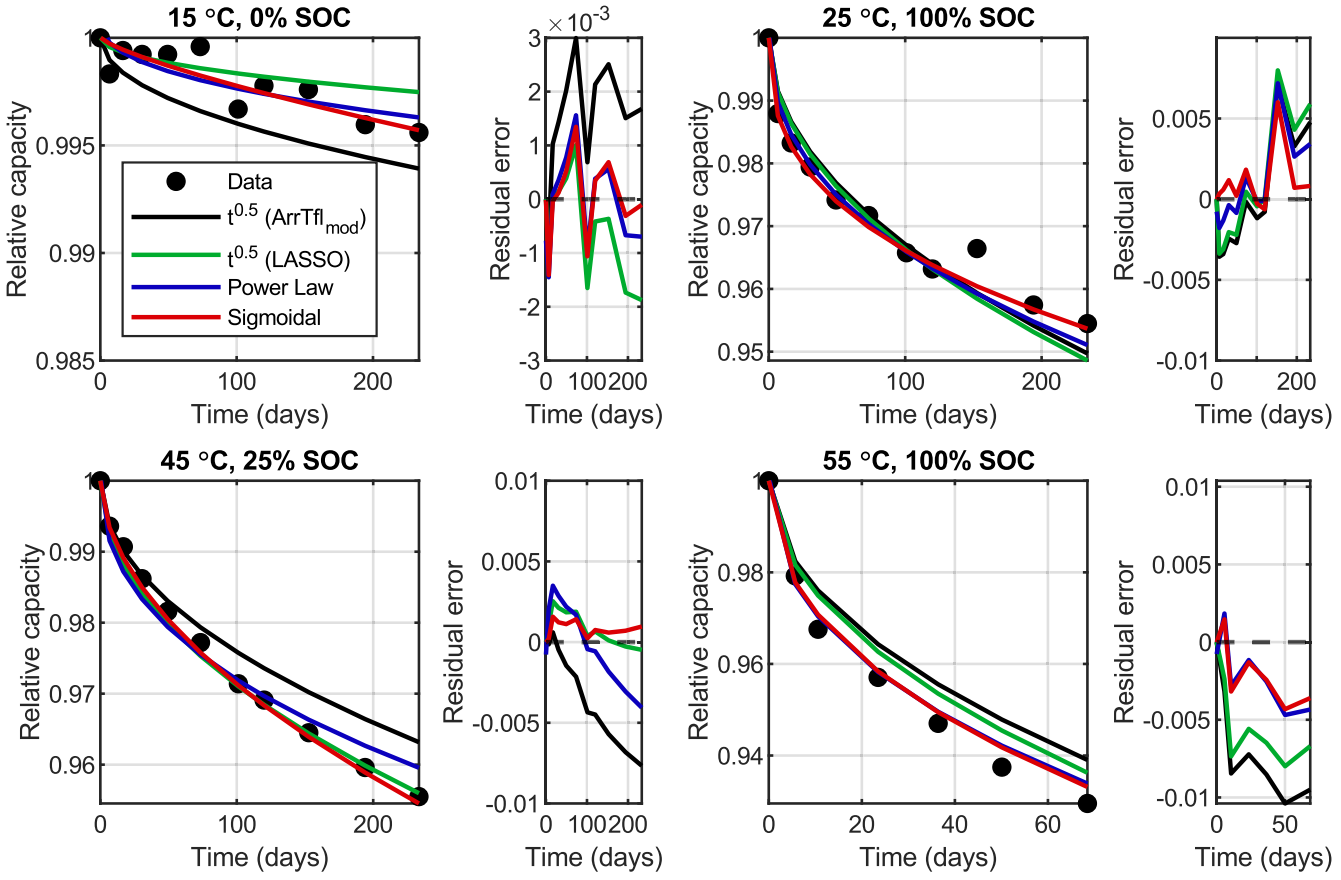

Figure 7. Capacity fade predictions for $t^{0.5}$, power law, and sigmoidal models after global optimization on the 16 cells of the training set, using all available RPTs, plotted on a subset of training cells. The residual errors of each prediction are plotted vs time to the right of each relative capacity plot.

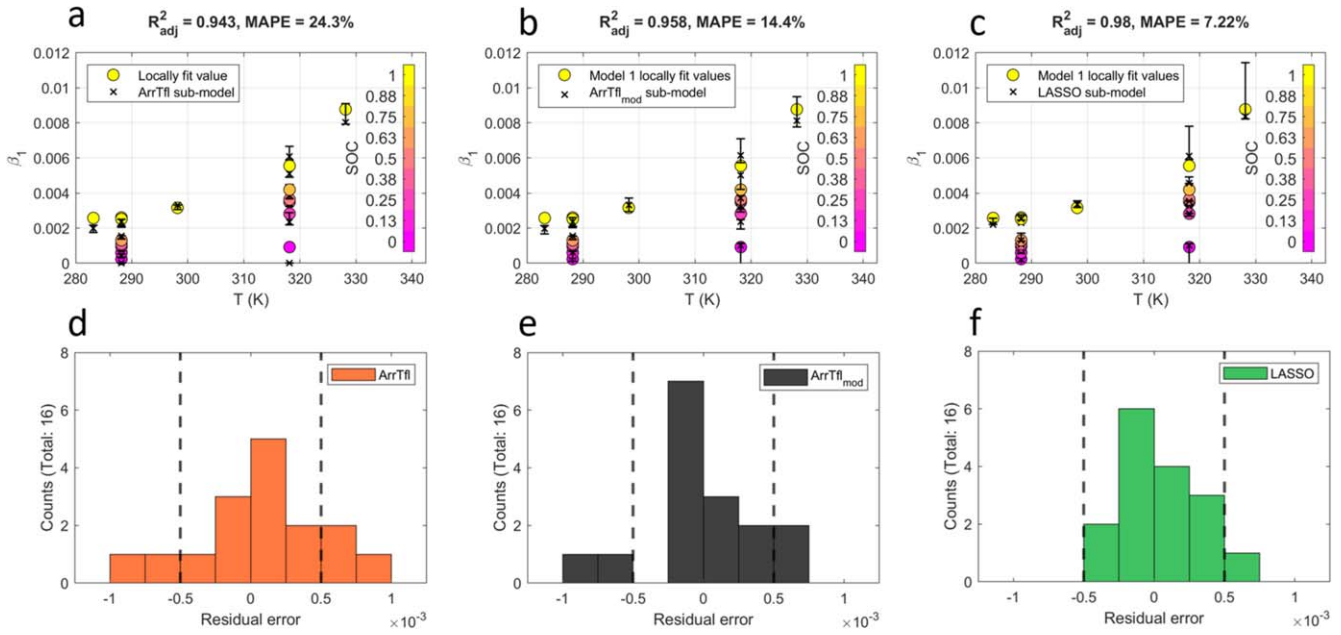

Figure 8. β_1 parameter sub-models for the $t^{0.5}$ model after optimizing on the 16 cells of the training set using all RPT data: (a) ArrTfl sub-model (Eq. 3) and (d) residuals histogram, (b) $\text{ArrTfl}_{\text{mod}}$ sub-model (Eq. 7) and (e) residuals histogram, and (c) LASSO identified sub-model (Eq. 19) and (f) residuals histogram. Circular markers in (a,b,c) are colored according to SOC to help visualize the variation of β_1 with both temperature and SOC. Dashed lines are added in (d)–(f) at ±0.0005 residual error to aid interpretation.

predicts 15 of the 16 locally fit β_1 values within ±0.0005 $\text{days}^{-0.5}$ (Fig. 8f), improving substantially on the performance of the other sub-models.

Structural similarities exist between the LASSO identified β_1 parameter sub-model and the physically informed sub-models. The simplest descriptor identified by LASSO is $\exp(T^2)$, like an

Arrhenius descriptor. Almost all other descriptors identified by LASSO are interactions between temperature and U_a, like the Tafel descriptor used in the physically informed models; remarkably, across all well-performing models, all LASSO identified descriptors containing U_a also contain temperature, demonstrating consistency with known Tafel kinetics that govern the relationship between SEI growth and electrochemical potential. This relationship is automatically identified without any prior knowledge. No descriptor contains SOC; U_a is always preferred, demonstrating the value of providing input features that reflect the behavior of internal processes. This approach of using input features informed by prior knowledge of internal processes has been harnessed to great effect in the field of scientific machine learning, for example, as shown by the hierarchical machine learning approach.[80–83] Examining the 90% confidence intervals for each model, large uncertainty consistently occurs at high fade rates (high β_1 values) and low SOCs for the ArrTfl$_{mod}$ and LASSO identified sub-models; confidence intervals for the prediction at 55 °C and 100% SOC have a width of approximately 2–4·10^{-3} d$^{-0.5}$, while confidence intervals at lower temperatures and SOCs are as small as 1–5·10^{-4} d$^{-0.5}$, simply reflecting that β_1 is more difficult to predict with high confidence at extrema.

The relative capacity predictions of the $t^{0.5}$ (LASSO) and $t^{0.5}$ (ArrTfl$_{mod}$) models are shown with 90% confidence intervals for a few cells of interest in Fig. 9. While not substantially different, given that both models share a $t^{0.5}$ curvature assumption, the $t^{0.5}$ (LASSO) model has lower error, especially at middling SOCs, as seen in the relative capacity prediction at 45 °C and 25% SOC. This is reflected by the fit metrics: the $t^{0.5}$ (LASSO) model has an R^2_{adj} of 0.978 and a MAE of 0.19% relative capacity, while the $t^{0.5}$ (ArrTfl$_{mod}$) model has an R^2_{adj} of 0.959 and a MAE of 0.26% relative capacity, about 37% larger. Both models exhibit significant error in predicting the proper fade rate at 25 °C, 100% SOC, shown by the positive slope of the residual errors of both models at this condition. This error is not due to the prediction of the β_1 value at this condition SOC, which both sub-models predict accurately (Fig. 8), but rather the $t^{0.5}$ assumption. In other test conditions, such as at 45 °C, 25% SOC, the $t^{0.5}$ assumption is more valid, as the $t^{0.5}$ (LASSO) model predicts the slope of the relative capacity fade with only a small amount of error.

Detailed evaluation of the sigmoidal global model.—Figure 10 compares the uncertainty of the capacity predictions for the $t^{0.5}$ (ArrTfl$_{mod}$) and sigmoidal models. Not only are the predictions made by the sigmoidal model more accurate than those made by the $t^{0.5}$ (ArrTfl$_{mod}$) model, but the confidence intervals are substantially narrower for the sigmoidal model. At 25 °C, 100% SOC, the width of the 90% confidence interval at the end of testing for the sigmoidal model is only about 0.3%, and is nearly centered on the experimental data point, while for the $t^{0.5}$ (ArrTfl$_{mod}$) model, the width of the confidence interval is about 0.8%, more than two and a half times larger. Comparison of the confidence intervals is easier when examining the residual error plots. Both models show relatively high uncertainty at low fade rates (15 °C, 0% SOC). The only condition where the uncertainty of the models is similar is at 55 °C, 100% SOC, which has substantially faster capacity fade than any other cell of the training set; this uniqueness and large magnitude of capacity fade results in high uncertainty compared with predictions made for other cells.

The sigmoidal model (model 13) demonstrates the best fit quality, highest sensitivity, and lowest uncertainty of all studied models because it is the only model where the symbolic regression procedure was able to find accurate sub-models for both the rate/extent of capacity fade (β_1 for all models) as well as the power exponent of time (β_3 for the sigmoidal model). For all other models

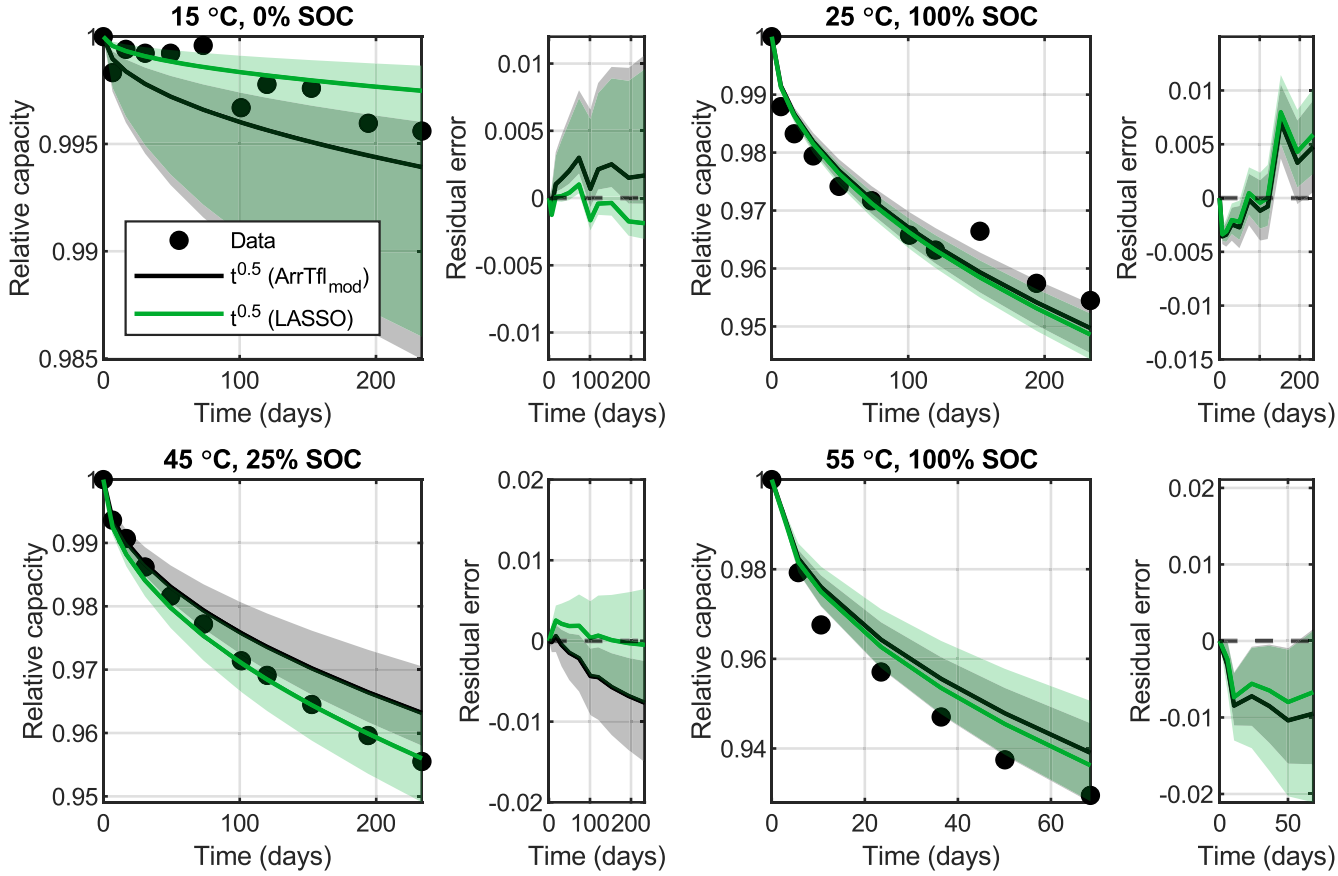

Figure 9. Capacity fade predictions with 90% confidence intervals for model 1 after global optimization on the 16 cells of the training set, using all RPT data, plotted on a subset of training cells. The two $t^{0.5}$ models use the ArrTfl$_{mod}$ β_1 parameter sub-model (Eq. 5) (black), and the LASSO identified β_1 parameter sub-model (green). The residual errors of each prediction are plotted vs time to the right of each relative capacity plot.

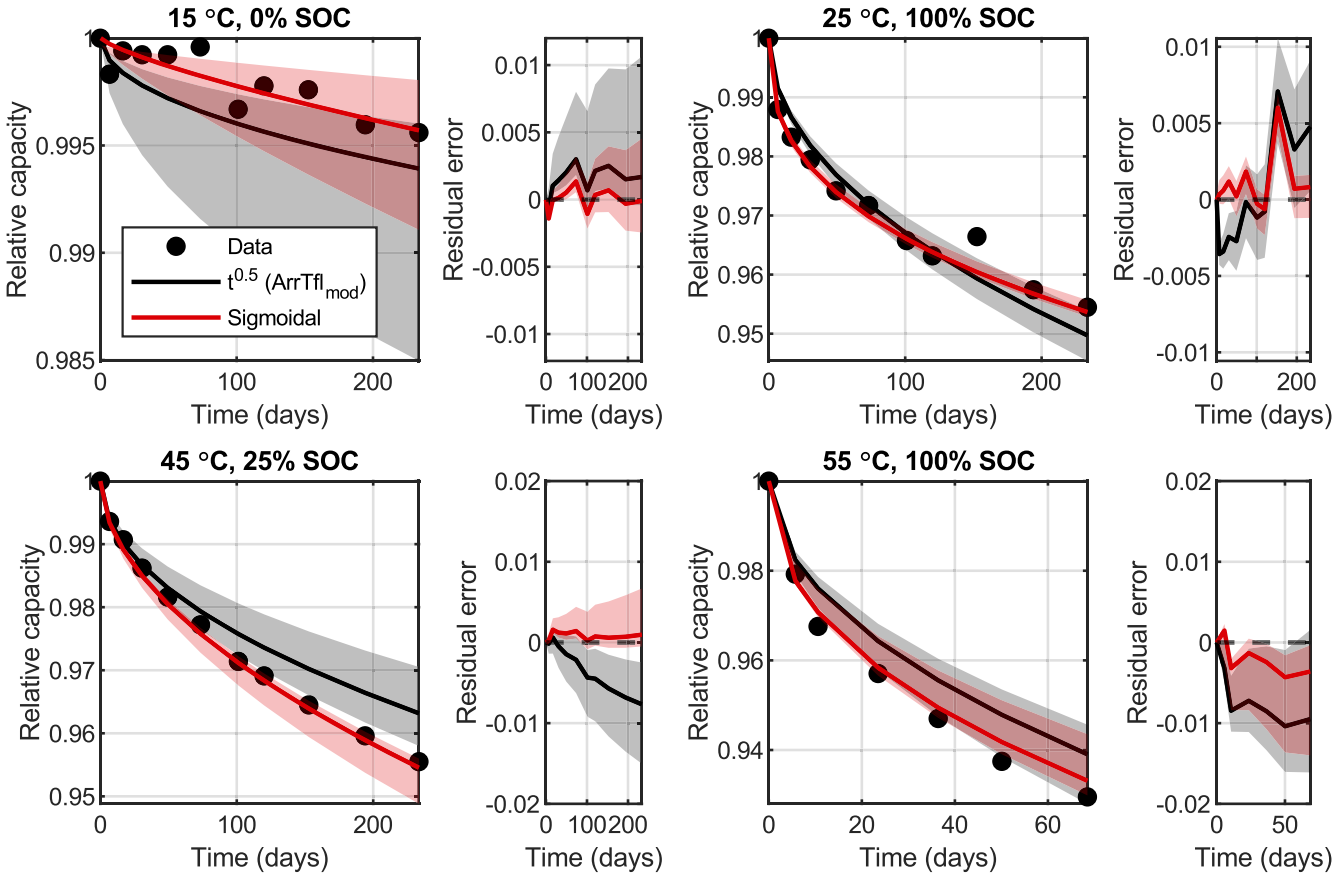

Figure 10. Relative capacity predictions with 90% confidence intervals for the $t^{0.5}$ ($ArrTfl_{mod}$) (black) and sigmoidal (red) models after global optimization on the 16 cells of the training set, using all RPT data, plotted on a subset of the training cells. The residual errors of each prediction are plotted vs time to the right of each relative capacity plot.

that optimized the power exponent of time locally (Models 3, 5, 8, 9, 10, 14, 15, as detailed in Table IV), LASSO was not able to find a predictive model better than a constant. The locally fit β_3 values and the predictions of the LASSO identified sub-model for β_3 for the sigmoidal model are shown in Fig. 11. The LASSO identified β_3 parameter sub-model for the sigmoidal model is:

$$\beta_3 = \gamma_0 \cdot \exp(\gamma_1 T^2 \sqrt{U_a}) \exp(\gamma_2 / (T^2 U_a^2)) \quad [20]$$

While the R^2_{adj} is not extremely high (0.761), the MAPE is reasonably low (14.1%), and the sub-model correctly captures several trends in the variation of the locally fit β_3 values: the convergence of β_3 towards 0.5 at 100% SOC with increasing temperature, and the nearly linear behavior of capacity fade at 0% SOC. The nearly linear behavior of capacity fade at 0% SOC is of particular interest because it reflects the conclusions of recent research on fundamental modeling of graphite SEI growth,[37] which determined that the power exponent of time is near 1 when the graphite SEI is very thin, but then decays very quickly to 0.5 as the SEI thickens. Cells at 0% SOC also may not be self-discharging to the extent of cells at higher SOCs, impacting their aging trajectory. Another possible explanation for this behavior is that at 0% SOC, the capacity fade due to SEI growth is competing with other degradation mechanisms, for example, the influence of the few charge/discharge cycles used in each RPT measurement. An increase of cell capacity during the first tens of cycles has been observed for LFP/graphite LIBs by other researchers.[25] At higher SOCs, the magnitude of capacity fade due to SEI growth begins to eclipse any RPT induced changes to cell capacity. The competition of these two mechanisms may be the cause of the noisy signal observed in the capacity fade at low SOCs, such as at 15 °C and 0% SOC in Fig. 10.

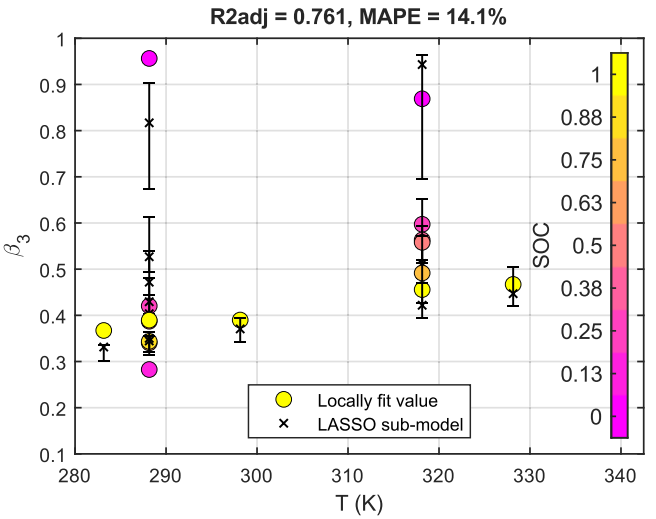

Figure 11. LASSO identified β_3 parameter (power exponent of time) sub-model for the sigmoidal model.

Plotting of the β_1 surface vs temperature and SOC is also a useful way to examine the behavior of the sigmoidal model, as shown in Fig. 12. For sigmoidal type models, β_1 is the limit of the relative capacity loss at infinitely long times. Figure 12 also shows the corresponding uncertainties vs temperature and SOC. The LASSO identified β_1 sub-model predicts the value of β_1 with very low uncertainty, except at low SOCs and high temperatures. This is because no experimental data was measured at low SOCs and high

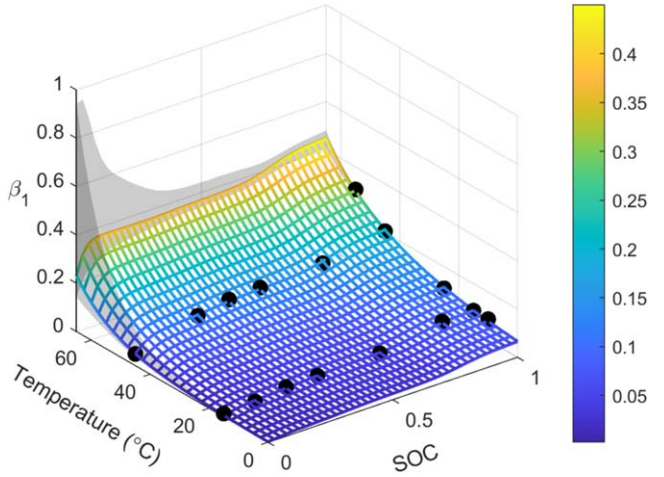

Figure 12. Predicted β_1 values for the sigmoidal model. 90% confidence intervals are shaded on each side face (shading is too narrow to be observed at low temps.). Solid markers denote training data.

temperature, but the predicted degradation rate is relatively high, leading to a wide confidence interval in that region; the limits of the confidence interval range from approximately 0.15 to 0.9, which is a very large range, considering the maximum physically realistic range of β_1 is 0 to 1. Despite this uncertainty, the mean prediction in this region still behaves as expected, showing lower capacity fade at low SOCs. The curvature of β_1 vs SOC reflects the variation of U_a vs SOC, which has several plateaus corresponding to different graphite intercalation stages.[60]

Model convergence behaviors.—The convergence of models to a global minimum when trained on increasing amounts of data is explored here, both in the time dimension as well with respect to the number of test conditions. Convergence of models with respect to test time is investigated by training several model types on increasing numbers of RPTs. Convergence of models with respect to the test conditions is investigated by training the power law model on increasing numbers of cells. In both cases, models using machine-learned sub-model equations are compared directly with models using human-expert defined sub-models.

The convergence of models when trained on an increasing number of RPTs per cell is shown in Fig. 13. At just 2 RPTs of training data, the $t^{0.5}$ ($ArrTfl_{mod}$) model has the lowest prediction error on both the validation and test sets. This is because the $t^{0.5}$ ($ArrTfl_{mod}$) model makes physically informed assumptions about the nature of the capacity fade, reducing the need to learn from the data set. However, after about 70 d of aging (RPT 6), all the models with automatically identified local parameter sub-models exceed the performance of the $t^{0.5}$ ($ArrTfl_{mod}$) model when trained on all available data (~235 d). The sigmoidal model also shows evidence of learning a key feature of the data set at RPT 9, improving substantially upon all other models; this improvement is not reflected in the test set MAE, suggesting whatever features learned from the training set are not reflected in the test set. A simple comparison of the data sets reveals that the test set contains no cells with SOCs less than 25%, so whatever feature of the training data learned by the sigmoidal model at RPT 9 likely improved the predictions for cells at low SOCs. The convergence behavior of all global models is shown in Fig. S3 in the Supplementary Material.

Convergence of the values of global model parameters provides insight into the learning behavior of a model. As an example, the convergence of global power exponent of time from the power law model, α_2, is shown in Fig. 14. The value of the global parameter has converged by 6 RPTs (~70 d of aging) to a value near 0.45, with little change when trained on all available data (~235 d of aging). This result shows that for most cells in this data set, the $t^{0.5}$

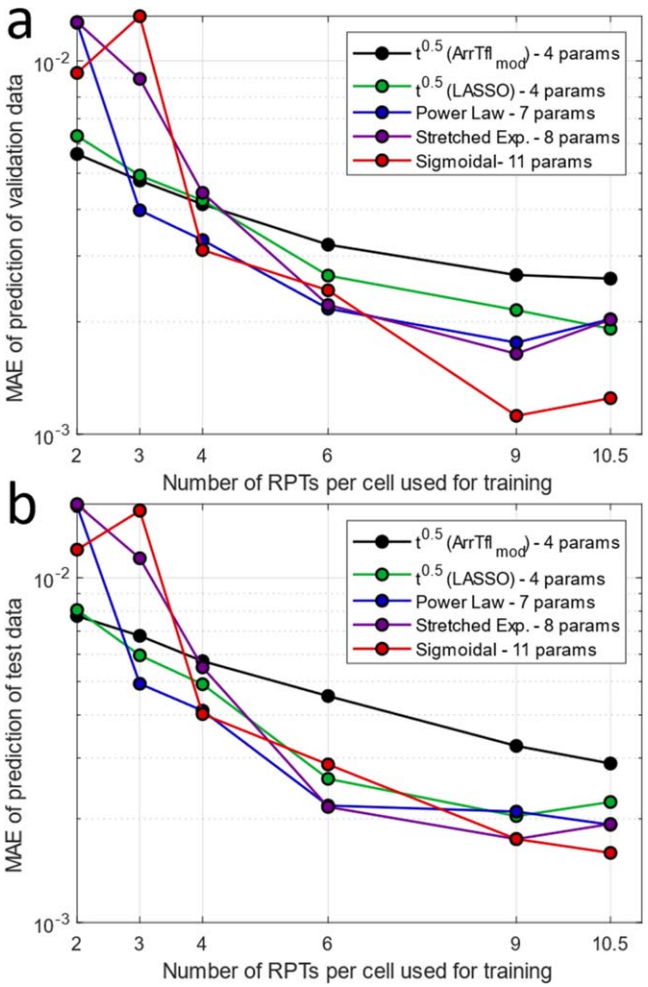

Figure 13. Convergence of global models to the (a) validation data and the (b) test data, which is unseen during the training process.

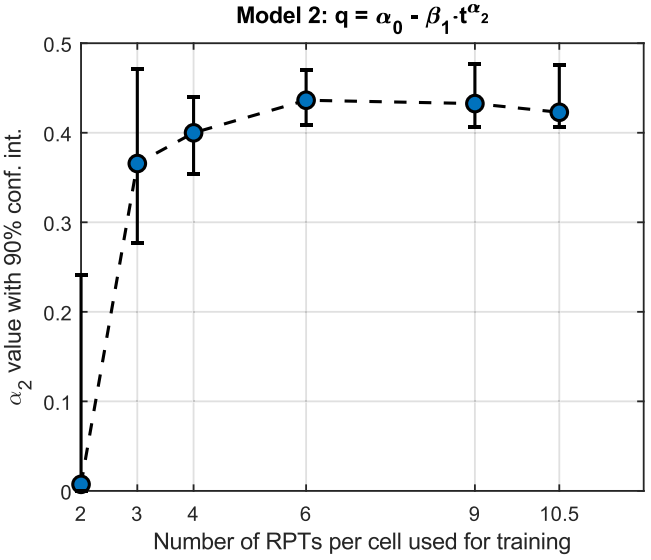

Figure 14. Convergence of global parameter α_2 from the power law model.

assumption does not hold, as 0.5 is not within the 90% confidence interval during any stage of training. Additionally, comparing the value of the optimized power exponent from the power model, 0.45, to the locally fit power exponents from the sigmoidal model

(Fig. 11), it is clear that Model 2 is reaching an optimum that most effectively models the capacity fade of cells at high temperatures (35 °C to 55 °C) and high SOCs; at lower temperature (10 °C to 25 °C) and high SOCs, the locally fit power exponents of the sigmoidal model are between 0.35 and 0.4, and thus the optimized power exponent of 0.45 for the power law model will cause systematic error. This systematic error is clearly observable in the residuals plot of the capacity prediction at 25°C, 100% SOC in Fig. 7.

The convergence of the LASSO identified descriptors for local parameter sub-models can also be investigated to provide insight into model learning behaviors. The first 5 identified descriptors for the β_1 parameter sub-model for the power law model vs the number of RPTs per cell used for training are shown in Table V. When trained with 2 and 3 RPTs, the sub-model predictions have low R^2_{adj} values, indicating that the locally fit values are not varying predictably. At 4 RPTs, the quality of the sub-model prediction increases substantially. At 6 RPTs, the prediction quality again increases, and the chosen descriptors begin to stabilize. The simplest descriptor, $\exp(T^2)$, is repeated when training with 6 RPTs, 9 RPTs, and all RPTs, giving high confidence that this descriptor is reflective of the internal physics that govern the variation of the capacity fade rate. Several Tafel-like descriptors appear throughout the training process; the most common form, $\exp(T^2/\sqrt{U_a})$ or its inverse $\exp(\sqrt{U_a}/T^2)$, appears in every sub-model from RPT 3 onward, demonstrating that this descriptor is also critical for modeling the capacity fade rate. As noted previously, U_a is not present without temperature in any descriptors for well-performing models, indicating that the symbolic regression process has correctly "discovered" the physical relationship between temperature and electrochemical potential known to govern SEI growth behavior without any prior knowledge. Also, no sub-model contains only a single Tafel-like descriptor, instead relying on several similar descriptors. This suggests that none of the proposed descriptors effectively capture the interactions between temperature and U_a. Future work will build larger libraries of possible descriptors to hopefully find more effective models with fewer parameters.

The convergence of the power law model when trained on an increasing number of cells is shown in Figs. 15a, 15c. Cells were chosen to vary the sampled temperature and state-of-charge as much as possible with each added cell; the test conditions of each training set are shown in Fig. 15a. The MAE of the prediction of the test data is plotted vs the number of cells in the training set for the power law models using either automatically identified (LASSO) or human-expert identified (ArrTfl) β_1 sub-models in Fig. 15c. Even with only four cells in the training set, the symbolic regression procedure can identify a sub-model that results in a more accurate global model than a human expert. At five cells, the LASSO approach cannot converge, resulting in high error. At six cells, both models reach an optimum value. As more data is added to the training set, the power law model using the ArrTfl equation gets worse; additional cells at lower SOCs cause the ArrTfl equation to balance underprediction of the degradation rate at low SOCs with overprediction at other conditions (see Fig. 8a), while the LASSO models maintain the same MAE. The stability of the prediction error is reflected by the stability of the identified sub-model descriptors, shown in Table VI, which do not vary significantly from 6 cells onward.

The effect of the sampling strategy on the quality of the power law model using LASSO or ArrTfl sub-models when training on an extremely small data set of only 4 cells is shown in Figs. 15b, 15d. Three sampling strategies are explored, and shown in Fig. 15b: an orthogonal strategy, which varies temperature at 100% SOC and varies SOC at a single temperature, an extrema strategy, which uses data at the "corners" of the parameter space, and a varied strategy, which attempts to get the most unique values of temperature and SOC as possible. The MAE of the prediction error on the validation set of both power law models is shown in Fig. 15d. The error of the models developed using LASSO for the "Extrema" and "Orthogonal" training sets is high, because LASSO is not able to converge on a model and returns a constant as the best-fit. This is because without enough variance in the training set, the symbolic regression procedure cannot find relationships in the data. The impact on the power law model using the ArrTfl equation is not significant, as this model makes prior assumptions about the relationships present in the data.

Impact of U_a vs SOC.—Because all LASSO identified sub-models showed a preference for using U_a as an input feature over SOC, the impact of using U_a as opposed to SOC needs to be investigated. While the transformation of SOC to U_a clearly imparts physically relevant information, causing it to be chosen over SOC, the sampling of U_a across the data set is more imbalanced than the sampling of SOC. The relationship between SOC and U_a is shown in Fig. 16. While SOC is evenly sampled across its range, U_a is not. At low SOCs, U_a increases rapidly, while at higher SOCs, U_a shows several plateaus, which correspond to intercalation stages of the graphite electrode.[60] These plateaus are sampled heavily, so any models constructed using U_a should be accurate when predicting SEI growth rate at SOCs between ~30% and 100%, but the sharp rise in U_a as SOC approaches 0% has only a single sample at 0% SOC, so a model constructed with U_a will have trouble making accurate predictions at low SOCs. To study this hypothesis, the power law and sigmoidal models were re-identified after removing U_a as an input feature, forcing the models to use SOC.

LASSO identified models with and without U_a for the β_1 parameter of the sigmoidal model are shown in Fig. 17 at 45 °C. Both models capture the essential behavior of β_1, which monotonically increases vs SOC. With U_a (Fig. 17a), the sub-model predicts the behavior of β_1 with high accuracy at all sampled SOCs. Without U_a (Fig. 17b), the sub-model does not have enough data to infer the plateaus of U_a vs SOC. This is clear when comparing how the models behave between 40%–70% SOC. The model with U_a predicts that β_1 does not vary much across this voltage plateau, while the model without U_a shows a nearly linear increase of β_1 across this region, leading to underprediction at middling SOCs and

Table V. Convergence of first 5 descriptors identified by LASSO for the β_1 parameter sub-model of the power law model.

# of RPTs	Eq. type	R^2_{adj}	Descriptors				
			1	2	3	4	5
2	Lin.	0.49	$\gamma_1 T^{2s} SOC$	$\gamma_2 T^2/U_a$	$\gamma_3 T^2/\sqrt{U_a}$		
3	Mult.	0.55	$\exp(\gamma_1 T/\sqrt{U_a})$	$\exp(\gamma_2 T^2 U_a)$	$\exp(\gamma_3 T^2/\sqrt{U_a})$		
4	Mult.	0.92	$\exp(\gamma_1 T/\sqrt{U_a})$	$\exp(\gamma_2 T^2 U_a^2)$	$\exp(\gamma_3 T^2/\sqrt{U_a})$	$\exp(\gamma_4\sqrt{U_a/T})$	
6	Mult.	0.97	$\exp(\gamma_1 T^2)$	$\exp(\gamma_2 T^2 SOC)$	$\exp(\gamma_3 T^2\sqrt{U_a})$	$\exp(\gamma_4 U_a/T^2)$	$\exp(\gamma_5\sqrt{U_a}/T^2)$
9	Mult.	0.99	$\exp(\gamma_1 T^2)$	$\exp(\gamma_2 T^2\sqrt{U_a})$	$\exp(\gamma_3 T^2 SOC^2)$	$\exp(\gamma_4\sqrt{U_a/T^2})$	$\exp(\gamma_5/(U_a^2 T^2))$
All	Mult.	0.99	$\exp(\gamma_1 T^2)$	$\exp(\gamma_1\sqrt{U_a}/T^2)$	$\exp(\gamma_3/(U_a^2 T^2))$	$\exp(\gamma_4/(U_a^3 T^2))$	

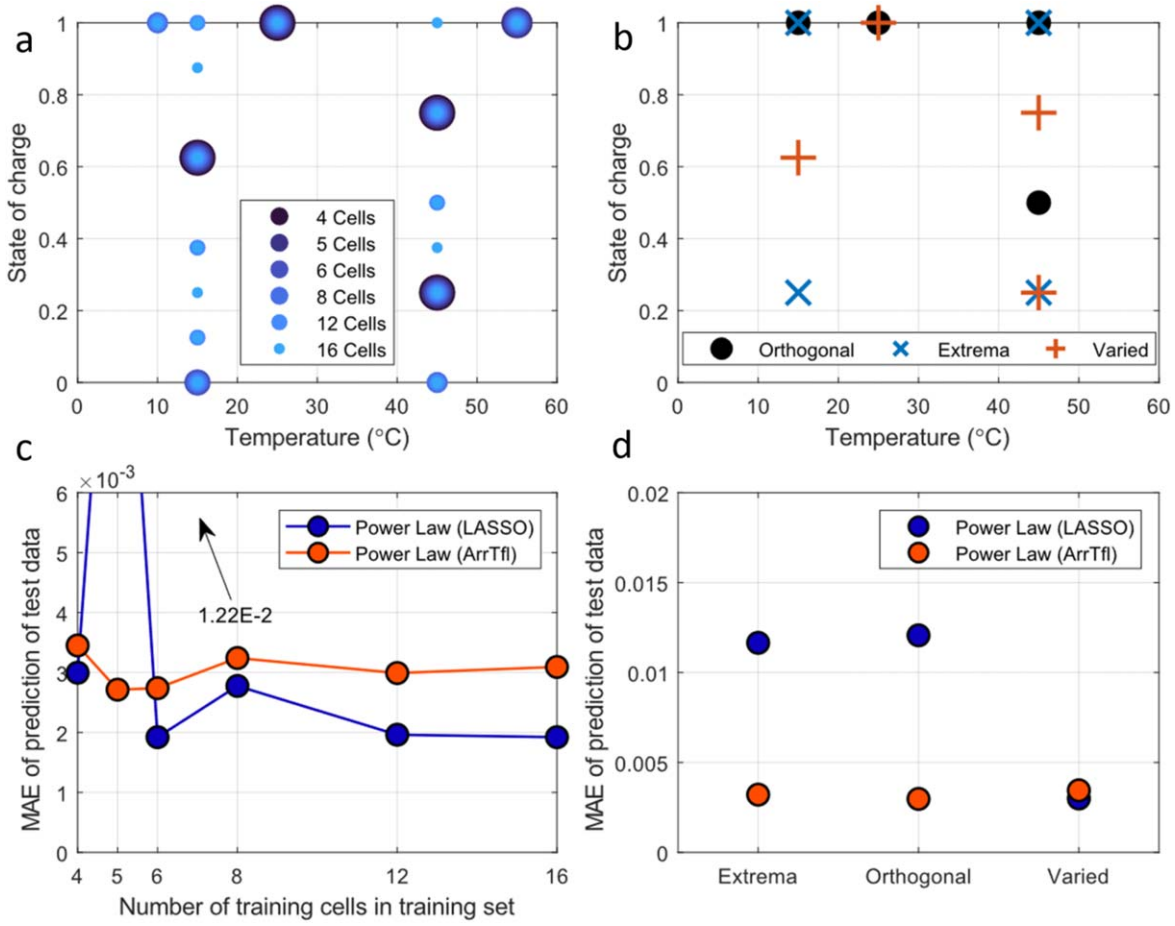

Figure 15. Convergence of the power law model, using both machine-learned (LASSO) and human-expert (ArrTfl) β_1 sub-models, as a function of the cells used for model training. (a) Cell test conditions for each training set, with the (c) MAE of the prediction of the test data from models trained on each training set shown below. (b) Cell test conditions used for exploring the impact of the sampling strategy on model quality for extremely small data sets, with the (d) MAE of the prediction of the test data from models trained on data using various sampling strategy shown below.

Table VI. Convergence of β_1 parameter sub-model descriptors of the power law model when adding cells to the training set.

# of cells	Eq. type	R^2_{adj}	Descriptors				
			1	2	3	4	5
4	Mult.	0.88	$\exp(\gamma_1/T^2)$	$\exp(\gamma_2\sqrt{U_a}/T)$			
5	Mult.	0					
6	Mult.	0.97	$\exp(\gamma_1/T^2)$	$\exp(\gamma_2 T^2/\sqrt{U_a})$	$\exp(\gamma_3\sqrt{U_a}/T^2)$		
8	Mult.	0.99	$\exp(\gamma_1 T^2)$	$\exp(\gamma_2\sqrt{U_a}/T^2)$	$\exp(\gamma_3 SOC^3/T^2)$		
12	Mult.	0.99	$\exp(\gamma_1 T^2)$	$\exp(\gamma_2 SOC/T^2)$	$\exp(\gamma_3\sqrt{U_a}/T^2)$	$\exp(\gamma_4/(U_a^2T^2))$	$\exp(\gamma_5/(U_a^3T^2))$
All	Mult.	0.99	$\exp(\gamma_1 T^2)$	$\exp(\gamma_2\sqrt{U_a}/T^2)$	$\exp(\gamma_3/(U_a^2T^2))$	$\exp(\gamma_4/(U_a^3T^2))$	

overprediction at the high SOCs. At low SOCs, both models show high uncertainty, with only a single data point at low β_1 and low SOC. The model with U_a, which has more curvature, is more difficult to extrapolate without data, and correspondingly has a large confidence interval (width of ~0.15, vs ~0.1 for the model without U_a). The implication here is that if the governing physics of a problem are known, those governing physics should guide experimental design to produce low uncertainty models. For this application, this means that sampling U_a evenly will result in more accurate models than sampling SOC evenly.

The first five descriptors identified by LASSO for the β_1 parameter sub-models of the power law and sigmoidal models with and without U_a are shown in Table VII. All models predict the variation of β_1 with high accuracy. Very similar temperature descriptors are present in each sub-model, with all containing an $\exp(T^2)$ descriptor. The sub-models with U_a have relatively consistent Tafel-like descriptors between then power law and sigmoidal models, with 2 out of the 3 being identical. Without U_a, the identified Tafel-like descriptors are less consistent, sharing only a single binary descriptor in common between the power law and sigmoidal models. This stabilizing effect of U_a on the structure of the automatically identified sub-models demonstrates that U_a is a critical feature for modeling this data set.

Global models built with and without U_a show similar behavior to that of β_1 sub-models discussed previously (Fig. 17). The MSE and MSE_{CV} of the power law and sigmoidal models constructed with and without U_a are shown in Fig. 18. As expected, the models constructed with U_a perform better than models constructed without

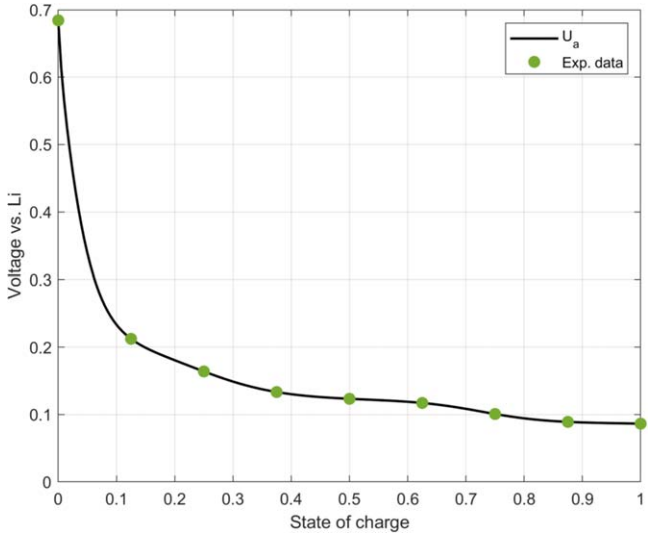

Figure 16. U_a, as modeled by Safari, 2011,[60] plotted vs SOC. The experimentally tested conditions are shown by green markers.

U_a. The sigmoidal model without U_a shows substantially higher MSE_{CV} than other models as well, demonstrating the importance of high value features for creating a stable model. Plots comparing the capacity predictions from the power law and sigmoidal models with and without U_a are shown in Figs. S4 and S5, respectively.

Extrapolation of models across test conditions.—To study the ability of models to learn physical relationships from training data, the extrapolation of some selected models w.r.t. experimental conditions is tested. Two extrapolations tests were conducted, first to cells with high fade rates (cells 12, 18, 19, 20, 21, and 22) and second to cells with SOCs less than or equal to 25% (cells 2, 3, 4, 13, 14, and 15). For each extrapolation test, models which had been identified using all 16 cells of the training set were re-optimized without the extrapolation cells, and then used to predict the performance of the withheld cells; thus, this procedure is testing whether structure of models identified on all available data inherently contain physical insight. Models investigated using this approach are the $t^{0.5}$ ($ArrTfl_{mod}$) and $t^{0.5}$ (LASSO) models, power law models with and without U_a, and sigmoidal models with and without U_a.

The MAE of each model on training and extrapolation data from the two extrapolation tests are shown in Table VIII. In general, extrapolation to low SOCs performs worse than extrapolation to high fade rates when using U_a as an input feature, and vice-versa when using SOC as an input feature. Because this behavior is reflected by all the investigated models, this indicates that selection of input features is more important than model structure for determining model behavior during extrapolation in this study. Models using U_a cannot accurately predict cell behaviors at low SOCs without exposure to low SOC training data, due to the steep curvature of U_a at low SOCs; while the relationship between SOC and U_a is known, this information is not incorporated into the models, so models only "learn" this sharp increase in U_a when provided data from low SOC cells. Models without U_a do not accurately predict cell behaviors at high fade rates, as they cannot infer the plateaus of U_a vs SOC given the available data.

Improved extrapolation to high fade rates when using U_a, and improved extrapolation to low SOCs without U_a are shown by comparing the capacity predictions of the sigmoidal and power law models with and without U_a in Fig. 19. Examining the behavior of the sigmoidal model with and without U_a when extrapolated to high fade rates (Fig. 19a), it seems that both models perform well at 100% SOC, though the model with U_a has a narrower confidence interval (width of ∼0.1 with U_a, and ∼0.15 without U_a) and performs much better at 62.5% SOC. However, models with U_a cannot accurately extrapolate to low SOCs. The power law model with U_a dramatically overpredicts capacity fade when extrapolating to low SOCs (Fig. 19b), while without U_a, the power law model makes fairly accurate predictions. Capacity fade predictions for the power law and sigmoidal models, with and without U_a, are shown for all six cells in the high fade rate extrapolation test in Figs. S6 and S7, respectively, and for the six cells in the low SOC extrapolation test in Figs. S8 and S9, respectively, in the Supplementary Material.

Extrapolation of models to long times.—Extrapolation of models over long times is useful for exaggerating differences and is representative of their intended use as predictive models for control systems or technoeconomic analyses. The predictions of the $t^{0.5}$ ($ArrTfl_{mod}$), power law, and sigmoidal models over 20 years of aging at 50% SOC and temperatures of 10 °C, 25 °C, and 40 °C is shown in Fig. 20; *confidence intervals of the predictions* are shown as opposed to *prediction intervals* to show the impact of the reliability of parameter estimation on model predictions. Prediction intervals, which are generally wider than confidence intervals, may be more appropriate for estimating the reliability of predicting future measurements, and will be used in future analyses. The $t^{0.5}$ ($ArrTfl_{mod}$) model predicts the largest magnitude of capacity fade for all the simulations, while the predictions of the power law and sigmoidal models show different behaviors across the temperatures. The difference in the capacity fade predictions between the models is the most drastic when the capacity fade rate is low: at 10 °C, the $t^{0.5}$ ($ArrTfl_{mod}$) model predicts the cell will reach 5% capacity fade within 4 years, while the sigmoidal model predicts the same amount of degradation will require almost 11 years. This is despite a difference in the MAE of these models of only 0.125% on the training set (MAE for the $t^{0.5}$ ($ArrTfl_{mod}$) model is ∼0.25%, MAE for the sigmoidal model is ∼0.125%). While this is clearly an example picked to exaggerate differences between these models, the implications are significant nonetheless; small differences in model predictions on the training data, when simulated to long times relevant to real-world use, may result in substantial disagreements between models. In the absence of validation data from long-term experiments, the predictions of the sigmoidal model are the most trusted, as it captures the capacity fade behavior across the entire

Table VII. β_1 parameter sub-models identified after global optimization on the 16 cells of the training set using all RPT data for the power law and sigmoidal models, with and without U_a as an input feature. All sub-models are multiplicative.

Model	R^2_{adj}	Descriptors				
		1	2	3	4	5
Power law	0.99	$\exp(\gamma_1 T^2)$	$\exp(\gamma_2 \sqrt{U_a}/T^2)$	$\exp(\gamma_3/(U_a^2 T^2))$	$\exp(\gamma_4/(U_a^3 T^2))$	
Power law, no U_a	0.98	$\exp(\gamma_1 T^2)$	$\exp(\gamma_2 T^2 SOC^2)$	$\exp(\gamma_3 \sqrt{SOC}/T^2)$		
Sigmoidal	0.99	$\exp(\gamma_1 T)$	$\exp(\gamma_2 T^2)$	$\exp(\gamma_3 T^2/\sqrt{U_a})$	$\exp(\gamma_4 \sqrt{U_a}/T^2)$	$\exp(\gamma_5/(U_a^2 T^2))$
Sigmoidal, no U_a	0.99	$\exp(\gamma_1 T^2)$	$\exp(\gamma_2 \sqrt{SOC}/T)$	$\exp(\gamma_3 SOC/T^2)$	$\exp(\gamma_4 \sqrt{SOC}/T^2)$	

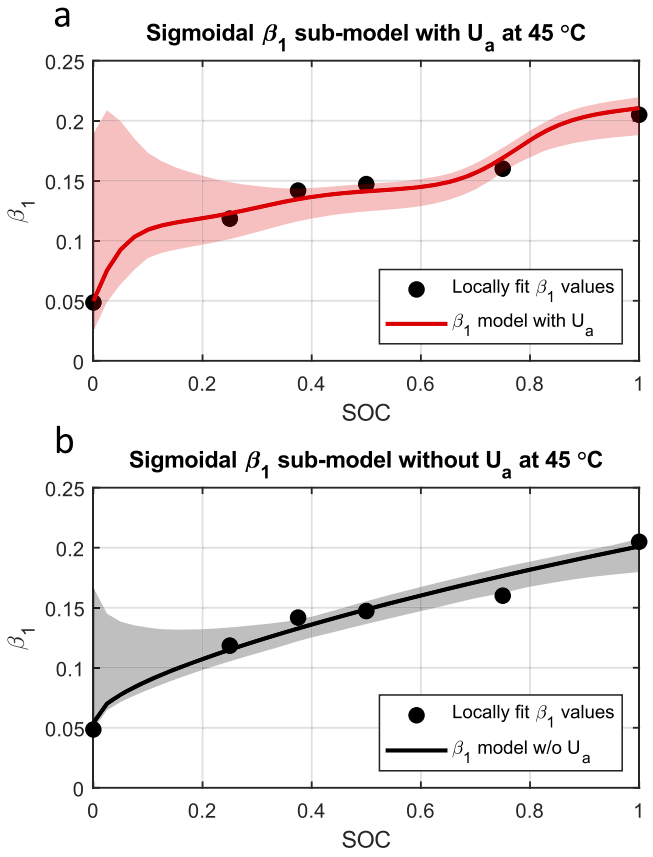

Figure 17. LASSO identified sub-models of the β_1 parameter of the sigmoidal model (a) with and (b) without U_a, calculated vs SOC at 45 °C with 90% confidence intervals.

data set with the best accuracy, indicated by its low error on all subsets of the data and its relatively low uncertainty across all test conditions.

A practical implication of more accurate extrapolations at long times is that other degradation mechanisms can be detected as they emerge during cell aging. LLI can itself have several contribution mechanisms (SEI growth, lithium-metal deposition, accelerated SEI growth due to cell charging, new SEI growth due to particle fracturing), and more accurate degradation models enable researchers to deconvolve multiple degradation mechanisms. For example, modeling of the aging of cycled cells often requires tracking both LLI as well as the loss of active material (LAM) in the positive or negative electrodes, and isolating the contribution of LAM to overall cell capacity requires extrapolation an LLI model (trained on cells/data points known to be dominated by LLI) to cells with competing LLI/LAM degradation modes. LIBs with other chemistries, such as those with layered-oxide cathodes, may also necessitate disambiguation of multiple degradation modes. Incorporating other input features, such

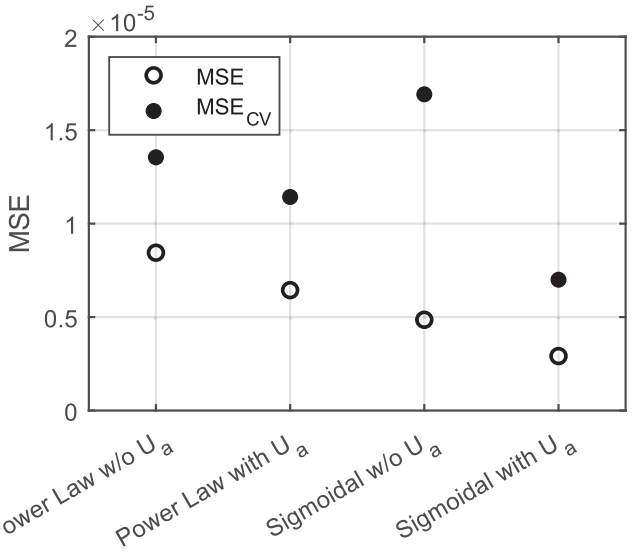

Figure 18. MSE and MSE_{CV} of the power law and sigmoidal models constructed with and without U_a as an input feature.

as information extracted from $dQ \cdot dV^{-1}$ measurements, into the model identification process may greatly assist the disambiguation of competing degradation modes.

Conclusions

In this work, reduced-order models used to predict the capacity fade of LIBs during calendar aging were reviewed to reveal common assumptions and practices. While not true for all studies, many works utilized physically informed models, such as $t^{0.5}$ for predicting the behavior of individual cells, or Arrhenius and Tafel equations to predict SEI growth rate as a function of temperature and SOC. To repeat from the introduction, there are several reasons that researchers typically use physically informed equations when identifying reduced-order models:

1. Obvious starting point for developing reduced-order models of complex systems
2. Inherent interpretability of model behavior when utilizing equations derived from first principles analysis of simple systems
3. Model behavior when extrapolating to untested conditions is known a-priori

The obvious drawback of using physically informed reduced-order models is that they are based on first principles analysis of simplified systems, and do not necessary predict the behavior of complex systems. Moreover, many researchers proposed modifications to the physically informed descriptors (semi-empirical models), or used empirically informed descriptors, implying they could not effectively model their data set using common physically informed

Table VIII. MAE of training and extrapolation to test sets of cells that exhibit high fade rates and cells at low SOCs for a selection of models.

Model	Extrapolation to high fade rates		Extrapolation to low SOCs	
	MAE (Train)	MAE (Test)	MAE (Train)	MAE (Test)
$t^{0.5}$ ($ArrTfl_{mod}$)	0.24%	0.43%	0.26%	0.84%
$t^{0.5}$ (LASSO)	0.17%	0.34%	0.20%	2.97%
Power law	0.17%	0.45%	0.19%	3.08%
Power law, no U_a	0.20%	0.45%	0.25%	0.17%
Sigmoidal	0.11%	0.19%	0.13%	3.25%
Sigmoidal, no U_a	0.14%	0.34%	0.17%	0.14%

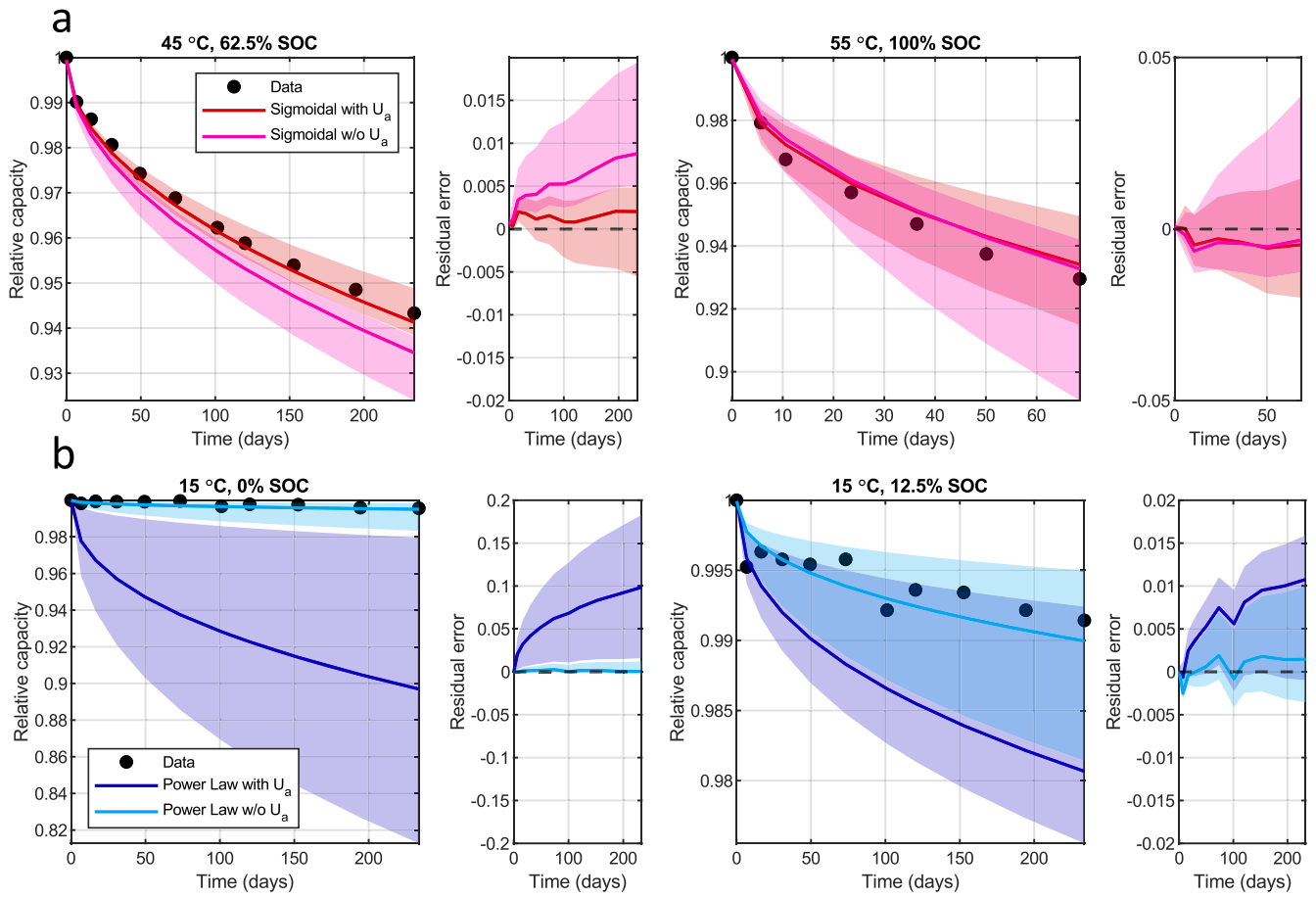

Figure 19. Capacity fade predictions with 90% confidence intervals with and without U_a as an input feature for the (a) sigmoidal model extrapolating to high fade rates and the (b) power law model extrapolating to low SOC. Results are plotted on a subset of cells from each extrapolation test that demonstrate characteristic behaviors. The residual error of each prediction is shown to the right of each plot.

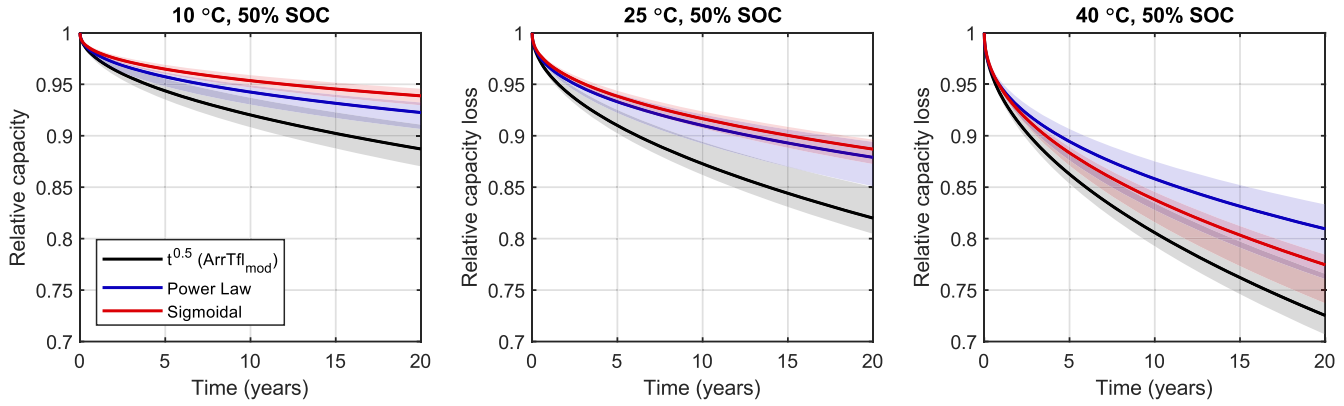

Figure 20. Predictions of 20 years of calendar fade at several temperature and 50% SOC with 90% confidence intervals for the $t^{0.5}$ ($ArrTfl_{mod}$), power law, and sigmoidal models.

equations. These empirically determined equations require statistical validation to ensure safe extrapolation, a step which is often neglected. Even with a physically informed model, careful statistical validation is a necessary first step to proving that model predictions are trustworthy and improves understanding of model behavior.

Thus, inspired by recent work of Attia et al.,[10] recommendations for statistically validating reduced-order capacity fade models were presented, and these recommendations were demonstrated by comparing physically informed and semi-empirical models found in literature to those constructed empirically by an automatic identification procedure utilizing bi-level optimization followed by symbolic regression. Models using power law, stretched exponential, and sigmoidal type equations were trained on a standout calendar aging data set of LFP/graphite cells published by Schimpe et al.[18] Statistical techniques of cross-validation, convergence analysis, sensitivity analysis, uncertainty quantification by bootstrap resampling, and extrapolation were utilized to interrogate the behavior of the studied models. The best fitting model, a sigmoidal model, predicted the relative capacity fade of the training data with approximately half the MAE of a semi-empirical model identified by a human-expert. The sigmoidal model reported here also extrapolated more accurately to validation data and showed much

lower predictive uncertainty. The automatic identification procedure also found models with square-root, power law, and stretched exponential structures that had improved performance relative to human expert models were also identified, demonstrating the flexibility of this method for identifying robust models regardless of the chosen structure.

Through careful interrogation, the behavior of automatically identified models can be interpreted and justified. Investigation of the studied models revealed that the optimization procedure proposed here "rediscovered" two known physical behaviors without any presupposition: the bi-level optimization procedure identified variation in the power exponent of time for calendar fade that reflects recent fundamental research on SEI growth,[37] and the symbolic regression procedure consistently identified descriptors that paired U_a or SOC with temperature, reflecting well-understood Tafel kinetics that govern the relationship between SEI growth rate and electrochemical potential. Additionally, it was shown the model extrapolation to high fade rates or low SOCs was dependent on the selection of input features, rather than on model structure, a clear demonstration that model behavior is dependent not only on its structure, but also its input data.

Extrapolating models forward in time revealed that even minor disagreements between model predictions on the training data (8 months of aging) resulted in substantial deviations between predictions after long times (20 years of aging). Models trained on the same data, each of which predicted the relative capacity fade of the training data with a MAE of less than 0.25% relative capacity, differed in their predictions of the capacity fade after 20 years by a factor of 100% or more (up to 10% disagreement of the relative capacity, or 50% of the practical life of the battery if end of lifeis defined at 80% relative capacity). This result has substantial implications for the use of reduced-order models in battery control systems and technoeconomic models, and highlights why it is necessary to statistically validate reduced-order degradation models for LIBs.

Moreover, utilizing the methodology developed here can greatly accelerate the model development process, and enable fair comparison between degradation models trained on different aging data sets. This is something that is traditionally difficult to do, because it is not possible to assert that models which have been manually identified on different data sets are getting the most out of their respective data, and there exists no reduced-order model general enough to model an arbitrary data set effectively. Models identified using the machine learning empowered approach shown here are more readily comparable, as the search for best model is algorithmically defined, and thus long-term predictions can be analyzed with confidence that comparisons between models trained on different data sets are fair. Regardless of the model structure, quantification of uncertainty can be used to guide the design of future experiments, and rigorous statistical interrogation of models can speed the discovery of underlying physics of complex systems by ensuring that as much information as possible is learned from the available data. And while the focus of this article is on capacity loss governed by a single degradation mechanism, the approach herein is applicable to other battery health metrics such as cell impedance rise, as well as to more complex data sets that demonstrate multiple degradation mechanisms.

Including other features, such as those extracted from detailed electrochemical measurements like $dQ{\cdot}dV^{-1}$, as inputs for battery life models would likely improve the predictive accuracy of these models. This is especially true for data that exhibits multiple degradation mechanisms, or heterogenous aging across test replicates; in the case of calendar aging, as shown here, cell-to-cell heterogeneity is usually reported as very small, and capacity fade is dominated by a single degradation mechanism, so the data can be modeled accurately without including additional features.

The results shown here clearly demonstrate that using physically informed assumptions to predetermine model structure does not necessarily improve either fit to training data or extrapolation to unseen conditions. Looking forward, it is hoped that this work can be a steppingstone for future studies utilizing machine learning methods to automatically identify models for electrochemical systems, building on the growing body of work utilizing symbolic regression techniques in other scientific domains.

Acknowledgments

This work was supported by the Assistant Secretary for Energy Efficiency and Renewable Energy, Office of Vehicle Technologies of the U.S. Department of Energy through the Advanced Battery Development, System Analysis, and Testing program. The National Renewable Energy Laboratory is operated by Alliance for Sustainable Energy under Contract No. DE-AC36-08GO28308 for the U.S. Department of Energy. Idaho National Laboratory is operated by Battelle Energy Alliance under Contract No. DE-AC07-05ID14517 for the U.S. Department of Energy. The views expressed in the article do not necessarily represent the views of the DOE or the U.S. Government. The U.S. Government retains and the publisher, by accepting the article for publication, acknowledges that the U.S. Government retains a nonexclusive, paid-up, irrevocable, worldwide license to publish or reproduce the published form of this work, or allow others to do so, for U.S. Government purposes.

Appendix

Calculation of anode-to-reference potential (U_a) from state-of-charge (SOC).—This equation has been reproduced from Schimpe et al.,[18] and is originally attributed to Safari and Delacourt.[60] U_a is determined first by calculated the relative lithiation of the graphite, x_a, as a function of SOC, and the calculating U_a as a function of x_a. The equation for x_a is a simple linear interpolation of the stoichiometry of lithium in the graphite anode, as measured by anode half-cell measurements:[18]

$$x_a(SOC) = 0.0085 + SOC \cdot (0.78 - 0.0085)$$

U_a is then calculated as a function of x_a by the equation from Safari and Delacourt:[60]

$$\begin{aligned} U_a(x_a) = {} & 0.6379 + 0.5416\exp(-305.5309 \cdot x_a) \\ & + 0.044\tanh\left(-\frac{x_a - 0.1958}{0.1088}\right) \\ & - 0.1978\tanh\left(\frac{x_a - 1.0571}{0.0854}\right) \\ & - 0.6875\tanh\left(\frac{x_a + 0.0117}{0.0529}\right) - 0.0175\tanh\left(\frac{x_a - 0.5692}{0.0875}\right) \end{aligned}$$

Fit metrics.— *Coefficient of determination (R^2)*

$$R^2 = 1 - \frac{\sum_1^n (y_n - y_{pred,n})^2}{\sum_1^n (y_n - \bar{y})^2}$$

where y is the response data (data being fit/predicted), n is the number of data points in y, $\bar{y}$ is the mean of the response data, and y_{pred} is the predicted response.

Adjusted coefficient of determination (R^2_{adj}):

$$R^2_{adj} = 1 - (1 - R^2) \cdot \frac{DOF}{n - 1}$$

where the degrees of freedom (DOF) is equal to:

$$DOF = n - p$$

where p is the number of model parameters.

Mean squared error (MSE):

$$MSE = \frac{\sum_1^n (y_n - y_{pred,n})^2}{DOF}$$

Mean signed difference (MSD):

$$MSD = \frac{\sum_1^n (y_n - y_{pred,n})^2}{n}$$

Mean absolute error (MAE):

$$MAE = \frac{\sum_1^n |y_n - y_{pred,n}|}{n}$$

Mean absolute percent error (MAPE):

$$MAPE = \frac{\sum_1^n \left| \frac{y_n - y_{pred,n}}{y_n} \right|}{n}$$

Model Equations.—For all the following reduced-order model equations, q is relative discharge capacity, t is time in days, T is temperature in Kelvin, SOC is state of charge, and U_a is the anode-to-lithium-reference potential, calculated as a function of SOC using the model by Safari and Delacourt.[59]

Model 1—$t^{0.5}$ ($ArrTfl_{mod}$).— *Model equations*

$$q = 1 - \beta_1 t^{0.5},$$

$$\beta_1 = \gamma_0 \cdot \exp\left(\gamma_1 \frac{1}{T}\right) \cdot \left(\gamma_2 + \exp\left(\gamma_3 \frac{U_a}{T}\right)\right)$$

Model parameters.—β_1 sub-model: $\gamma_0 = 0.00157509920559553$, $\gamma_1 = 21855.2495911229$, $\gamma_2 = 0.391258956773426$, $\gamma_3 = 0.342598196618906$

Model 1—$t^{0.5}$ (LASSO).— *Model equations*

$$q = 1 - \beta_1 t^{0.5},$$

$$\beta_1 = \gamma_0 \cdot \exp(\gamma_1 T^2) \cdot exp(\gamma_2 \sqrt{U_a}/T^2) \cdot \exp(\gamma_3 T^2/\sqrt{U_a}) \cdot \exp(\gamma_4/(U_a^2 T^2)) \cdot \exp(\gamma_5/(U_a^3 T^2))$$

Model parameters.—β_1 sub-model: $\gamma_0 = 5.66055159867028e-05$, $\gamma_1 = 6.78495316614697e-05$, $\gamma_2 = -6.73196811705621e-06$, $\gamma_3 = -392313.347878663$, $\gamma_4 = 780.754335526516$, $\gamma_5 = 12.0073255348081$

Model 2—power law.— *Model equations*

$$q = \alpha_0 - \beta_1 t^{\alpha_1},$$

$$\beta_1 = \gamma_0 \cdot \exp(\gamma_1 T^2) \cdot \exp(\gamma_2 \sqrt{U_a}/T^2) \cdot \exp(\gamma_3/(U_a^2 T^2)) \cdot \exp(\gamma_4/(U_a^3 T^2))$$

Model parameters.—Global: $\alpha_0 = 1.00077162305817$, $\alpha_1 = 0.422870911936737$

β_1 sub-model: $\gamma_0 = 0.000131359732087792$, $\gamma_1 = 4.22464201471607e-05$, $\gamma_2 = -229637.590117639$, $\gamma_3 = 18.0867012887458$, $\gamma_4 = 34.7563919770849$

Model 2—Power law without U_a.— *Model equations*

$$q = \alpha_0 - \beta_1 t^{\alpha_1},$$

$$\beta_1 = \gamma_0 \cdot \exp(\gamma_1 T^2) \cdot \exp(\gamma_2 T^2 SOC^2) \cdot \exp(\gamma_3 \sqrt{SOC}/T^2)$$

Model parameters.—Global: $\alpha_0 = 1.00071226034570$, $\alpha_1 = 0.423245474433938$

β_1 sub-model: $\gamma_0 = 2.48515307384225e-06$, $\gamma_1 = 6.40511004337107e-05$, $\gamma_2 = -1.13531789772891e-06$, $\gamma_3 = 175790.934229002$

Model 6—stretched exponential.— *Model equations*

$$q = \alpha_0 - \beta_1(1 - 1/\exp((\alpha_1 t)^{\alpha_2})), ,$$

$$\beta_1 = \gamma_0 \cdot \exp(\gamma_1 T^2) \cdot \exp(\gamma_2 \sqrt{U_a}/T^2) \cdot \exp(\gamma_3/(U_a^2 T^2)) \cdot \exp(\gamma_4/(U_a^3 T^2))$$

Model parameters.—Global: $\alpha_0 = 1.00075169420981$, $\alpha_1 = 4.24534842040504e-05$, $\alpha_2 = 0.440370572271614$

β_1 sub-model: $\gamma_0 = 0.0106924644709047$, $\gamma_1 = 4.22593522686259e-05$, $\gamma_2 = -230010.659845844$, $\gamma_3 = 18.1022718823750$, $\gamma_4 = 34.7558667980410$

Model 13—sigmoidal.— *Model equations*

$$q = \alpha_0 - 2\beta_1\left(\frac{1}{2} - \frac{1}{1 + \exp((\alpha_1 t)^{\beta_2})}\right),$$

$$\beta_1 = \gamma_{0,\beta_1} \cdot \exp(\gamma_{1,\beta_1} T) \cdot \exp(\gamma_{2,\beta_1} T^2) \cdot \exp(\gamma_{3,\beta_1} T^2/\sqrt{U_a}) \cdot \exp(\gamma_{4,\beta_1} \sqrt{U_a}/T^2) \cdot \exp(\gamma_{5,\beta_1}/(U_a^2 T^2)),$$

$$\beta_2 = \gamma_{0,\beta_2} \cdot \exp(\gamma_{1,\beta_2} T^2 \sqrt{U_a}) \exp(\gamma_{2,\beta_2}/(T^2 U_a^2))$$

Model parameters.—Global: $\alpha_0 = 1.00001376140538$, $\alpha_1 = 2.263154527192 18e-07$,

β_1 sub-model: $\gamma_{0,\beta 1} = 2.87660289775572e-05$, $\gamma_{1,\beta 1} = 0.00194139757136067$, $\gamma_{2,\beta 1} = 0.000159242066031295$, $\gamma_{3,\beta 1} = 1.67382362333744e-05$, $\gamma_{4,\beta 1} = 114553.532911380$, $\gamma_{5,\beta 1} = 1111.32287038007$

β_2 sub-model: $\gamma_{0,\beta 2} = 0.806779018023099$, $\gamma_{1,\beta 2} = 4.21480492252815e-06$, $\gamma_{2,\beta 2} = -23212.9725808128$

Model 13—Sigmoidal without U_a.— *Model equations*

$$q = \alpha_0 - 2\beta_1\left(\frac{1}{2} - \frac{1}{1 + \exp((\alpha_1 t)^{\beta_2})}\right),$$

$$\beta_1 = \gamma_{0,\beta_1} \cdot \exp(\gamma_{1,\beta_1} T^2) \cdot \exp(\gamma_{2,\beta_1} \sqrt{SOC}/T) \cdot \exp(\gamma_{3,\beta_1} SOC/T^2) \cdot \exp(\gamma_{4,\beta_1} \sqrt{SOC}/T^2),$$

$$\beta_2 = \gamma_{0,\beta_2} \cdot \exp(\gamma_{1,\beta_2}/T^2) \exp(\gamma_{2,\beta_2} \sqrt{SOC}/T^2)$$

Model parameters.—Global: $\alpha_0 = 0.999498200258108$, $\alpha_1 = 3.08669713448935e-05$,

β_1 sub-model: $\gamma_{0,\beta_1} = 1.63661734646738e-05$, $\gamma_{1,\beta_1} = 0.000118768550860168$, $\gamma_{2,\beta_1} = -3171.62791793208$, γ_{3,β_1} 36779.9530850153, $\gamma_{4,\beta_1} = 893445.736891740$

β_2 sub-model: $\gamma_{0,\beta_2} = 1.14308657074168$, $\gamma_{1,\beta_2} = -28136.5950354692$, γ_{2,β_2} −73372.3713767496.

ORCID

Paul Gasper https://orcid.org/0000-0001-8834-9458

References

1. M. S. Ziegler and J. E. Trancik, (2020), arXiv (https://arxiv.org/abs/2007.13920).
2. BloombergNEF, *Electric Vehicle Outlook 2020* (2020), (https://about.bnef.com/electric-vehicle-outlook/).
3. J. M. Reniers, G. Mulder, S. Ober-Blöbaum, and D. A. Howey, *J. Power Sources*, **379**, 91 (2018).
4. A. Gailani, M. Al-Greer, M. Short, and T. Crosbie, *Electron*, **9**, 90 (2020).
5. M. Beuse, *Nat. Energy*, **3**, 363 (2018).
6. C. R. Birkl, M. R. Roberts, E. Mcturk, P. G. Bruce, and D. A. Howey, *J. Power Sources*, **341**, 373 (2017).
7. M. Petit, E. Prada, and V. Sauvant-Moynot, *Appl. Energy*, **172**, 398 (2016).
8. S. Englberger, H. Hesse, D. Kucevic, and A. Jossen, *Energies*, **12**, 955 (2019).
9. F. Yang, Y. Xie, Y. Deng, and C. Yuan, *Nat. Commun.*, **9**, 1 (2018).
10. P. M. Attia, W. C. Chueh, and S. J. Harris, *J. Electrochem. Soc.*, **167**, 090535 (2020).
11. E. V. Thomas, I. Bloom, J. P. Christophersen, and V. S. Battaglia, *J. Power Sources*, **206**, 378 (2012).
12. J. Schmitt, A. Maheshwari, M. Heck, S. Lux, and M. Vetter, *J. Power Sources*, **353**, 183 (2017).
13. K. L. Gering, *Electrochim. Acta*, **228**, 636 (2017).
14. J. Schmalstieg, S. Käbitz, M. Ecker, and D. U. Sauer, *J. Power Sources*, **257**, 325 (2014).
15. J. Belt, V. Utgikar, and I. Bloom, *J. Power Sources*, **196**, 10213 (2011).
16. B. Rumberg, B. Epding, I. Stradtmann, M. Schleder, and A. Kwade, *J. Energy Storage*, **30**, 101510 (2020).
17. K. Smith, A. Saxon, M. Keyser, B. Lundstrom, Z. Cao, and A. Roc, *Proc. Am. Control Conf.*, 4062 (2017).
18. M. Schimpe, M. E. von Kuepach, M. Naumann, H. C. Hesse, K. Smith, and A. Jossen, *J. Electrochem. Soc.*, **165**, A181 (2018).
19. M. Naumann, M. Schimpe, P. Keil, H. C. Hesse, and A. Jossen, *J. Energy Storage*, **17**, 153 (2018).
20. M. Naumann, F. Spingler, and A. Jossen, *J. Power Sources*, **451**, 227666 (2020).
21. E. Sarasketa-Zabala, I. Gandiaga, E. Martinez-Laserna, L. M. Rodriguez-Martinez, and I. Villarreal, *J. Power Sources*, **275**, 573 (2015).
22. S. Grolleau, A. Delaille, H. Gualous, P. Gyan, R. Revel, J. Bernard, E. Redonde-Iglesias, and J. PeterSIMCAL Network, *J. Power Sources*, **255**, 450 (2014).
23. E. Redondo-Iglesias, P. Venet, and S. Pelissier, *J. Energy Storage*, **13**, 176 (2017).
24. M. Ecker, N. Nieto, S. Käbitz, J. Schmalstieg, H. Blanke, A. Warnecke, and D. U. Sauer, *J. Power Sources*, **248**, 839 (2014).
25. K. A. Severson et al., *Nat. Energy*, **4**, 383 (2019).
26. R. R. Richardson, M. A. Osborne, and D. A. Howey, *J. Energy Storage*, **23**, 320 (2019).
27. R. R. Richardson, M. A. Osborne, and D. A. Howey, *J. Power Sources*, **357**, 209 (2017).
28. X. Hu, L. Xu, X. Lin, and M. Pecht, *Joule*, **4**, 1 (2020).
29. B. Efron, *J. Am. Stat. Assoc.*, **115**, 636 (2020).
30. W. J. Murdoch, C. Singh, K. Kumbier, R. Abbasi-asl, and B. Yu, *Proc. Natl. Acad. Sci. U. S. A.*, **116**, 44 (2019).
31. J. M. Reniers, G. Mulder, and D. A. Howey, *J. Electrochem. Soc.*, **166**, A3189 (2019).
32. A. Aitio, S. G. Marquis, P. Ascencio, and D. Howey, (2020), arXiv (http://arxiv.org/abs/2001.09890).
33. A. Pozzi, G. Ciaramella, S. Volkwein, and D. M. Raimondo, *Ind. Eng. Chem. Res.*, **58**, 1286 (2019).
34. S. Park, D. Kato, Z. Gima, R. Klein, and S. Moura, *J. Electrochem. Soc.*, **165**, A1309 (2018).
35. H. J. Ploehn, P. Ramadass, and R. E. White, *J. Electrochem. Soc.*, **151**, A456 (2004).
36. F. Single, A. Latz, and B. Horstmann, *ChemSusChem*, **11**, 1950 (2018).
37. L. von Kolzenberg, A. Latz, and B. Horstmann, (2020), arXiv (http://arxiv.org/abs/2004.01458).
38. R. Tibshirani, *J. R. Stat. Soc. Ser. B Stat. Methodol.*, **58**, 267 (1996).
39. S. Das, P. M. Attia, W. C. Chueh, and M. Z. Bazant, *J. Electrochem. Soc.*, **166**, E107 (2019).
40. P. M. Attia, S. Das, S. J. Harris, M. Z. Bazant, and W. C. Chueh, *J. Electrochem. Soc.*, **166**, E97 (2019).
41. M. Dubarry, N. Qin, and P. Brooker, *Curr. Opin. Electrochem.*, **9**, 106 (2018).
42. E. Cuervo-Reyes and R. Flückiger, *J. Electrochem. Soc.*, **166**, A1463 (2019).
43. M. Ecker, J. Gerschler, J. Vogel, S. Käbitz, F. Hust, P. Dechent, and D. U. Sauer, *J. Power Sources*, **215**, 248 (2012).
44. G. Williams and D. C. Watts, *Trans. Faraday Soc.*, **66**, 80 (1970).
45. J. C. Phillips, *Rep. Prog. Phys.*, **59**, 1133 (1996).
46. I. Baghdadi, O. Briat, J. Y. Delétage, P. Gyan, and J. M. Vinassa, *J. Power Sources*, **325**, 273 (2016).
47. R. Mathieu, I. Baghdadi, O. Briat, P. Gyan, and J. M. Vinassa, *Energy*, **141**, 2108 (2017).
48. R. Xiong, G. Xie, A. E. Edmondson, and M. A. Sheard, *Int. J. Food Microbiol.*, **46**, 45 (1999).
49. C. Chen, W. Miao, C. Zhou, and H. Wu, *Bioresour. Technol.*, **225**, 48 (2017).
50. A. Ware and N. Power, *Renew. Energy*, **104**, 50 (2017).
51. Z. P. Xun, J. X. Li, Y. Jiao, R. T. Li, and G. Tang, *Phys. A Stat. Mech. its Appl.*, **540**, 122998 (2020).
52. G. D. Mumbach, J. L. F. Alves, J. C. G. Da Silva, R. F. De Sena, C. Maragoni, R. A. F. Machado, and A. Bolzan, *Energy Convers. Manag.*, **200**, 112031 (2019).
53. S. Khaleghi Rahimian, M. M. Forouzan, S. Han, and Y. Tang, *Electrochim. Acta*, **348**, 136343 (2020).
54. A. Eddahech, O. Briat, and J. M. Vinassa, *IECON Proc. (Industrial Electron. Conf.)*, 6806 (2013).
55. M. Broussely, S. Herreyre, P. Biensan, P. Kasztejna, K. Nechev, and R. J. Staniewicz, *J. Power Sources*, **97–98**, 13 (2001).
56. E. Sarasketa-Zabala, E. Martinez-Laserna, M. Berecibar, I. Gandiaga, L. M. Rodriguez-Martinez, and I. Villareal, *Appl. Energy*, **162**, 839 (2016).
57. S. Käbitz, J. B. Gerschler, M. Ecker, Y. Yurdagel, B. Emmermacher, D. André, T. Mitsch, and D. U. Sauer, *J. Power Sources*, **239**, 572 (2013).
58. T. Lu, Y. Luo, Y. Zhang, W. Luo, L. Yan, and J. Xie, *J. Electrochem. Soc.*, **164**, A775 (2017).
59. F. Alhaider, T. Klein, and S. Gerhard, *NEIS 2018 - Conf. Sustain. Energy Supply Energy Storage Syst.*156 (2020).
60. M. Safari and C. Delacourt, *J. Electrochem. Soc.*, **158**, A562 (2011).
61. Y. Tian, R. Yuan, D. Xue, Y. Zhou, X. Ding, J. Sun, and T. Lookman, *J. Applied Phys.*, **128**, 014103 (2020).
62. G. B. King, A. E. Lovell, L. Neufcourt, and F. M. Nunes, *Phys. Rev. Lett.*, **122**, 232502 (2019).
63. Z. R. Kenz, H. T. Banks, and R. C. Smith, *SIAM/ASA J. Uncertainty Quantification*, **1**, 348 (2013).
64. J. R. Koza, *Genetic Programming* (MIT Press, Cambridge, MA) (1992).
65. D. P. Searson, *Handbook of Genetic Programming Applications*, ed. A. H. Gandomi, A. H. Alavi, and C. Ryan (Springer, Berlin: New York, NY) (2015).
66. S. Silva, http://gplab.sourceforge.net/index.html (2018).
67. M. Schmidt and H. Lipson, *Science (80-.).*, **324**, 81 (2009).
68. J. Bongard and H. Lipson, *Proc. Natl. Acad. Sci. U. S. A.*, **104**, 9943 (2007).
69. D. L. Ly and H. Lipson, *J. Mach. Learn. Res.*, **13**, 3585 (2012).
70. L. M. Ghiringhelli, J. Vybiral, S. V. Levchenko, C. Draxl, and M. Scheffler, *Phys. Rev. Lett.*, **114**, 1 (2015).
71. L. M. Ghiringhelli, J. Vybiral, E. Ahmetcik, R. Ouyang, S. V. Levchenko, C. Draxl, and M. Scheffler, *New J. Phys.*, **19**, 023017 (2017).
72. R. Ouyang, S. Curtarolo, E. Ahmetcik, M. Scheffler, and L. M. Ghiringhelli, *Phys. Rev. Mater.*, **2**, 1 (2018).
73. S. L. Brunton, J. L. Proctor, and J. N. Kutz, *Proc. Natl. Acad. Sci. U. S. A.*, **113**, 3932 (2016).
74. B. M. De Silva, D. M. Higdon, S. L. Brunton, and J. N. Kutz, *Front. Artificial Intell.*, **3**, 1 (2020).
75. N. M. Mangan, J. N. Kutz, S. L. Brunton, and J. L. Proctor, *Proc. R. Soc. A*, **473**, 20170009 (2017).
76. A. Cortiella, K.-C. Park, and A. Doostan, (2020), arXiv (https://arxiv.org/abs/2005.13232).
77. K. Champion, P. Zheng, A. Y. Aravkin, S. L. Brunton, and J. N. Kutz, *IEEE Access*, **8**, 169259 (2020).
78. MathWorks, https://mathworks.com/help/stats/lasso.html (2020).
79. A. Rohatgi, https://automeris.io/WebPlotDigitizer (2020).
80. C. M. Childs and N. R. Washburn, *MRS Commun.*, **9**, 806 (2019).
81. A. Menon, C. Gupta, K. M. Perkins, B. L. DeCost, N. Budwal, R. T. Rios, K. Zhang, B. Póczos, and N. R. Washburn, *Mol. Syst. Des. Eng.*, **2**, 263 (2017).
82. A. Menon, C. M. Childs, B. Poczós, N. R. Washburn, and K. E. Kurtis, *Adv. Theory Simulations*, **2**, 1800164 (2019).
83. A. Menon, J. A. Thompson-Colón, and N. R. Washburn, *Front. Mater.*, **6**, 1 (2019).