mathematics

MDPI

Review

Advanced Data-Driven Diagnostics and Prognostics for Next-Generation Battery Systems

Mohad Tanveer and Heung Soo Kim *

Department of Mechanical, Robotics and Energy Engineering, Dongguk University-Seoul, 30 Pildong-ro 1-gil, Jung-gu, Seoul 04620, Republic of Korea; mohadtanveer@dgu.ac.kr
* Correspondence: heungsoo@dgu.edu

Abstract

Battery diagnostics and prognostics support safe electrified transport and grid storage, yet comparison across model families is hindered by inconsistent notation, validation, and uncertainty reporting. This review organizes state-of-health, state-of-charge, and remaining-useful-life estimation in a common mathematical framework, drawing primarily on lithium-ion evidence and examining conditions for transfer to sodium-ion, lithium–sulfur, and solid-state systems. Deterministic degradation laws, Wiener and Gamma processes, reduced-order electrochemical models, statistical learning, deep architectures, Kalman and particle filters, and physics-informed methods are expressed in common notation and compared by assumptions, identifiability, data requirements, computational cost, and uncertainty treatment. The synthesis shows that stochastic and Bayesian formulations are well suited to probabilistic life prediction, feature-based models remain competitive in small-data regimes, and physics-informed approaches improve extrapolation only when governing models and parameters remain valid. Cross-chemistry transfer requires target-system evidence that the model assumptions and input features remain appropriate. Priorities include cell-wise and cross-condition validation, calibrated prediction intervals, pack-level inference, interpretable models, and privacy-preserving multi-fleet learning.

Keywords: battery prognostics; state-of-health estimation; remaining useful life; stochastic degradation modeling; deep learning; uncertainty quantification

MSC: 62M20

1. Introduction

Electrified transport and grid-scale storage have made rechargeable batteries a critical infrastructure component, and the reliability of that infrastructure now rests on the ability to infer internal conditions from external measurements [1–3]. Lithium-ion cells dominate present deployment on account of high energy density, low self-discharge, and long cycle life [4,5], while sodium-ion, lithium–sulfur, and solid-state formats are entering service with degradation signatures that differ materially from the lithium-ion case. Performance in all of these systems decays irreversibly through coupled electrochemical, thermal, and mechanical processes governed by cycling history, temperature, and load [6–8]. Failure to track those processes leads to premature capacity loss, unanticipated outage, and in severe cases thermal runaway with safety and economic consequences [9].

Diagnostics and prognostics address this problem from opposite ends of the time axis. Diagnosis estimates the present condition of a system from current and historical

Check for updates

Academic Editors: Lin Lin and Dan Liu

Received: 23 September 2026
Revised: 16 October 2026
Accepted: 23 October 2026
Published: 26 October 2026

Copyright: © 2026 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license.

measurement; prognosis projects that condition forward to a defined failure threshold and returns the residual life together with its uncertainty [10,11]. For battery systems, the two are realized through three quantities: the state of health (SOH), which measures capacity or power capability relative to the pristine cell [12]; the remaining useful life (RUL), which counts the cycles or the operating time remaining before an end-of-life criterion is met [13]; and the state of charge (SOC), which expresses the available charge as a percentage of the present maximum capacity of the cell [14].

Methodologically, the field has organized itself around a physics-based and a data-driven pole [15]. Physics-based descriptions, from full-order porous electrode theory to lumped equivalent circuits, encode the governing transport and reaction relations explicitly [16–19]. They can extrapolate beyond the calibration range when their governing form remains valid and their parameters are re-identified as the cell ages. Data-driven descriptions estimate the degradation map directly from measurement without a mechanistic hypothesis, which makes them computationally convenient at inference and suitable for online execution; transfer across chemistry or duty cycle nevertheless requires representative target-domain data, retraining, or explicit domain adaptation, and validity otherwise remains confined to the observed operating envelope [20–23].

Statistical and machine learning methods have accordingly become the dominant computational instrument [24–27]. Kernel machines, ensembles of trees, and Gaussian process models map engineered health indicators onto degradation targets and, in the probabilistic cases, return model-based predictive intervals whose calibration must be verified empirically [28–32]. Deep architectures learn the representation and the regression jointly from raw voltage, current, and temperature streams, which removes the dependence on manual feature design [33–37]. Convolutional, recurrent, and attention-based models capture the local and long-range structure of cycling data, and generative models synthesize degradation trajectories where measured data is scarce [38–41].

Three obstacles persist. Deep models require large labeled corpora, and run-to-failure data is expensive to acquire and correspondingly rare [42]. Most reported validation is performed under laboratory cycling that does not reproduce field variability, so accuracy degrades when the deployment distribution departs from the training distribution [43]. The internal logic of high-capacity models resists physical interpretation, which is a binding constraint in regulated safety-critical sectors [44]. A recent assessment argues further that innovation has concentrated on architectural refinement while degradation mechanisms, material evolution, and multi-physics coupling remain comparatively underexplored [45].

This review responds to those obstacles by treating diagnostics and prognostics as a single estimation problem and by presenting each method family through its governing mathematics. A consistent notation, applied to ninety-four numbered equations and consolidated in Tables A1 and A2 of Appendix A, allows the modeling assumptions, the identifiability conditions, the sample complexity, and the computational cost of competing approaches to be compared on their own terms rather than through reported error values alone. The scope is deliberately broader than lithium-ion, since the mathematical structure of the estimation problem is shared across chemistries even where the underlying degradation physics is not; Section 4.4 makes that separation explicit for the sodium-ion, lithium–sulfur, and solid-state formats. The evidence base behind those formats is nevertheless uneven: the empirical results reviewed in Sections 5–10 are drawn predominantly from lithium-ion cells, while Section 4.4 summarizes the empirical studies available for each of the other three, which are growing in number for sodium-ion, few but methodologically informative for lithium–sulfur, and very limited for solid-state. The title should therefore be read as naming the systems the mathematics is intended to serve and not as a claim that each is equally well evidenced.

This review is distinct from, and complementary to, the authors' recent treatment of trustworthy machine learning for lithium-ion state-of-health estimation [46] and from the most closely related external reviews [24,47,48]. The prior review is organized by learning formulation and confined to SOH in lithium-ion cells; the external reviews are organized, respectively, by model fidelity, by machine-learning workflow stage, and by probabilistic modeling and uncertainty quantification, and all three are likewise confined to lithium-ion batteries. The present review is organized by the mathematics of the underlying estimation problem and extends this literature in four directions not treated jointly in those reviews: remaining useful life through selected first-passage lifetime laws under stated assumptions, developed in Section 4.2; reduced-order electrochemical and thermal descriptions, in Section 4.3; fault diagnosis and safety prognosis, in Section 9; and conditions for transfer to sodium-ion, lithium–sulfur, and solid-state systems, together with the empirical literature now available for each of them, in Section 4.4. Table 1 sets out the comparison explicitly.

Table 1. Scope of the present review compared with the authors' most closely related prior review [46] and with three representative external reviews [24,47,48].

Dimension	Sohail et al. [46]	Martí-Florences et al. [47]	Khaleghi et al. [24]	Thelen et al. [48]	Present Review
Target state	SOH	SOC and SOH	SOH and RUL	SOH and RUL	SOH, SOC, and RUL
Chemistry	Lithium-ion	Lithium-ion	Lithium-ion	Lithium-ion	Lithium-ion (evidence base); conditions for transfer to, and the empirical literature on, sodium-ion, lithium–sulfur, and solid-state systems (Section 4.4)
Organizing axis	Learning formulation	Model fidelity	ML workflow stages	Probabilistic ML and UQ	Mathematics of the estimation problem
First-passage life distribution	Not treated	Not treated	Not treated	Probabilistic RUL, not first passage	Central; inverse Gaussian and Gamma survival, Equations (13)–(16)
Electrochemical and thermal models	Background only	Primary focus	Limited	Limited	Butler–Volmer, Fickian transport, energy balance, Equations (20)–(24)
Sequential Bayesian estimation	Not treated	Observer and filter design	Limited	Covered within UQ	Kalman, EKF, UKF, particle filters, Equations (73)–(81)
Fault and safety prognosis	Not treated	Not treated	Limited	Not treated	Section 9
Cross-chemistry transfer	Not treated	Not treated	Not treated	Not treated	Section 4.4, including the empirical literature for each format, and Equation (93)
Notation	Per-method	Per-method	Per-method	Per-method	Consistent notation across 94 equations, with units and a register of reused symbols, Appendix A

Abbreviations: EKF, extended Kalman filter; ML, machine learning; UKF, unscented Kalman filter; UQ, uncertainty quantification.

The paper is organized as follows. Section 2 states the literature-selection protocol, the notation conventions, the scope and nature of the contribution, and the limitations of the synthesis. Section 3 poses the estimation problem formally, derives the health metrics, and sets out the conditioning of state-of-charge estimation. Section 4 develops deterministic, stochastic, and physics-based degradation models, and Section 4.4 examines what changes when the chemistry is not lithium-ion. Section 5 treats signal processing and feature construction. Section 6 presents statistical and machine learning formulations for diagnosis, and Section 7 presents deep architectures for prognosis together with graph-based and self-supervised formulations. Section 8 develops sequential state estimation and uncertainty quantification. Section 9 addresses fault diagnosis and safety prognosis. Section 10 covers

evaluation metrics, the quality of uncertainty estimates, validation protocols, data leakage, and reporting practice; Section 11 identifies open problems and emerging directions; and Section 12 concludes. Appendix A collects the notation, with dimensions and units, together with a register of symbols that carry more than one meaning.

2. Review Methodology and Scope

2.1. Literature Identification and Selection

The literature reviewed here was identified through a structured narrative search, and the selection procedure is described in this subsection for transparency. As the work is a narrative rather than a systematic review, no review protocol was registered, and the database-specific query strings, the dates of the individual searches and the number of records at each screening stage are not reported; the limitations that follow from this are stated in Section 2.4.

Records were sought in Scopus, Web of Science, IEEE Xplore, ScienceDirect, SpringerLink, Wiley Online Library, the MDPI journal collection, and arXiv, with Google Scholar used for forward and backward citation chasing rather than as a primary index. The search covers material published up to and including September 2026, the final update having been made during revision. No lower bound on publication date was imposed. Several of the formulations reviewed here originate in the electrochemical, signal-processing and statistical-learning literature of the 1980s and 1990s and are cited in their original form, whereas the empirical battery material is concentrated in the last seven years; the composition of the reference list is reported below.

Queries were constructed by combining a state term, a system term and a method term, in the pattern (state) AND (system) AND (method). The state terms were state of health, state of charge, remaining useful life, capacity fade, degradation, prognosis and diagnosis. The system terms were battery, cell, lithium-ion, sodium-ion, lithium–sulfur, solid-state and battery pack. The method terms were machine learning, deep learning, neural network, Gaussian process, support vector, ensemble, Kalman filter, particle filter, Bayesian, stochastic process, first-passage, Wiener process, Gamma process, equivalent circuit, single-particle model, physics-informed, transfer learning and domain adaptation. Each combination was re-expressed in the syntax of the index being searched, and the method terms were also used without a state term when the object of the search was a primary methodological source rather than a battery application. Sources for gradient boosting, graph-based and self-supervised learning, calibration and conformal prediction, and validation and data leakage, treated in Sections 6.3, 7.4, and 10.2–10.4, were reached through the generic method terms and by citation chasing rather than by dedicated queries.

A record was included if it satisfied all of the following: it states a mathematical formulation of diagnosis or prognosis for an electrochemical cell, module or pack, or it supplies a dataset, benchmark or primary methodological result on which such a formulation depends; its evaluation, where it reports one, is attributable to a stated dataset; and it is available in English. A record was excluded if it addresses cell design, materials synthesis, electrode manufacturing, thermal-management hardware or charging-protocol optimization without an estimation component; if its formulation duplicates one already represented, in which case the earlier or more complete source was retained; if it is a secondary account of a result available in a primary source; or if a reported result could not be attributed to an identifiable dataset and split protocol.

Screening proceeded in three passes. Titles and abstracts were screened against the inclusion criteria; the records surviving that pass were read in full and screened against the exclusion criteria; and the retained records were then followed in both citation directions until no further distinct formulation appeared. Selection was performed by the first author

and checked by the second. The review was not double-screened by independent assessors and no inter-rater agreement statistic was computed.

The reference list of this review consists of the retained corpus together with reviews and contextual sources cited for background, for the comparison with earlier reviews in Table 1 and for the reproduced figures, which were not screened against the criteria above, and the figures below can be checked against it directly. The list contains 200 records spanning 1985 to 2026. Of the 199 records that carry a publication date, the remaining one being an undated data portal, the median publication year is 2021; 153 (77%) were published in 2020 or later and 77 in 2022 or later, while 20 predate 2015 and are retained because they are the primary sources for the electrochemical, thermal, signal-processing, feature-analysis, statistical-learning and forecast-evaluation formulations of Sections 4.3, 5–7, 10.2 and 11.2. The distribution is therefore weighted toward recent empirical work without discarding the originating mathematical literature.

2.2. Notation and Reporting Conventions

Because the review draws on several bodies of literature, a symbol that is standard in one of them frequently carries a different standard meaning in another. Imposing globally unique symbols would make the equations unrecognizable against the sources from which they are taken, so two conventions are used instead. First, every symbol is defined in the text where the equation that uses it is introduced, and where a symbol that already carries a meaning elsewhere in the review is reused, the meaning in force is stated at that point, so that the text can be followed without recourse to the appendix; all such reuses are also collected in Table A2, together with each meaning and the equation that fixes it. Second, where a reuse would be likely to mislead, one of the meanings is subscripted, given an argument or re-lettered: the shape coefficient and rate parameter of the Gamma process in Equation (16) are written α_Γ and β_Γ to distinguish them from the capacity-fade coefficients α and β of Equation (12); the first-passage time in Equations (14) and (15) is written T_f to distinguish it from the temperature T used elsewhere in Section 4 and to parallel the failure index n_f of Equation (8); the moving-average window length in Equation (27) is written N_w to distinguish it from the failure threshold w of Equations (14) and (15); the Gamma increment density in Equation (16) is written $f_{\Delta X}$ to distinguish it from the regression function f; and the sample index in Equations (36) and (37) is written as an argument, as in Equation (31), so that a subscript on V always names the kind of voltage, as in V_t, and never a sample. On the same principle, the discharge duration in Equation (1) is written t_d, so that T always denotes temperature, and the sample size in the bound of Equation (44) is written N, so that the bare symbol n always denotes the cycle index. Quantities that recur are also written in the same way wherever they appear: the capacity in Equations (2) and (12) as $C_\mathrm{max}(n)$, the continuous terminal voltage and current in Equations (18) and (35) as $V_\mathrm{t}(t)$ and $I(t)$, the Kullback–Leibler divergence in Equations (38) and (60) as D_KL, the hidden-unit count in Equations (38) and (61) as m_h, and, in Section 6, the kernel as $k(\cdot, \cdot)$ with length scale ℓ and the observation-noise variance as ${\sigma_\varepsilon}^2$. The scaled feature in Equation (26) is written $\tilde{x}$ rather than with an asterisk, which marks complex conjugation in Equation (29), and log denotes the natural logarithm throughout. Dimensions and units are given for every symbol listed in Table A1. The assumptions and, for the distributed formulations, the boundary and initial conditions attached to each equation are stated where the equation is introduced.

2.3. Scope and Nature of the Contribution

The organizing principle of the review is the mathematics of the estimation problem rather than a taxonomy of methods or of applications, and its selection of material follows

from that principle. A formulation is included when it contributes a distinct mathematical statement of diagnosis or prognosis, when it is in current use on battery data, or when it is required to make a later formulation intelligible. For approaches that are already well established, representative rather than exhaustive citations are given, and primary sources are preferred to secondary accounts. The material is drawn from the applied mathematics, control, power electronics, electrochemistry, and machine learning literature, together with the public dataset repositories catalogued in Section 5.1. Elementary definitions, such as the scaling, statistical-moment, activation and error-metric relations, are included so that the notation is closed under the operations used later, and not as contributions in their own right.

The scope is delimited as follows. The review covers cell-level and, where evidence permits, pack-level estimation of state of health, state of charge, and remaining useful life for lithium-ion, sodium-ion, lithium–sulfur, and solid-state systems. It covers deterministic, stochastic-process, reduced-order electrochemical, statistical-learning, deep-learning, sequential-Bayesian, and physics-informed formulations. It excludes cell design, materials synthesis, thermal-management hardware, and charging-protocol optimization except where these bear directly on the estimation problem.

Two kinds of claims should be distinguished at the outset, because reviews of mathematical material can blur them. This review contributes a synthesis and an organizing framework: the estimation problem is stated once, in Remark 1, and each method family is then exhibited as a specialization of that single statement, so that families ordinarily described in incompatible vocabularies become comparable based on assumptions, identifiability, data requirement, cost, uncertainty treatment and extrapolation behavior, as set out in Section 12.1. It does not contribute new theorems, new estimators, or new experimental results, and nothing below is presented as a previously unpublished finding. Where a formulation is given in a form that differs from the source from which it is taken, the difference is a correction or a clarification of an existing statement rather than a new result, and it is explained at the point where it occurs; the discretely normalized smoothing kernel of Equation (28), the dimensional reading of the thermal balance of Equations (23) and (24), and the explicit statement of the conditions under which the inverse Gaussian life distribution of Equation (15) follows from Equation (13) are the three instances.

2.4. Limitations of the Review Design

Five limitations follow from this design and are stated at the outset. First, the synthesis is structural rather than quantitative: reported error values are not pooled across studies, because differences in dataset, split protocol, and end-of-life threshold make them incommensurable, and the comparison in Table 2 is therefore made on assumptions and requirements rather than on accuracy. Second, coverage is representative rather than exhaustive, so the absence of a method from this review is not evidence that it is unused. Third, the evidence base is strongly imbalanced across chemistries, so conclusions drawn for lithium-ion batteries rest on far more run-to-failure data than those drawn for the formats discussed in Section 4.4.

Fourth, as is usual for a narrative review, the database-specific query strings, the dates of the individual searches and the number of records identified, screened and excluded at each stage are not reported; the retained corpus is not distinguished entry by entry from the contextual sources in the reference list; and no PRISMA-type flow diagram is provided. Section 2.1 describes the search and selection procedure for transparency, but not at the level of detail that exact repetition would require. Fifth, because the final screening pass was citation chasing rather than exhaustive enumeration, the corpus is representative of the

formulations in current use rather than complete, and a quantitative estimate of coverage of the field is therefore not available.

Table 2. Comparison of physics-based and data-driven modeling paradigms for battery diagnostics and prognostics across the criteria that govern method selection in practice.

Criterion	Physics-Based Models	Data-Driven Models
Basis	Governing electrochemical, thermal, and electrical relations	Statistical relations learned from historical measurement
Prior knowledge	Extensive; requires cell design and material parameters	Minimal; requires representative labeled data
Parameterization	System-specific identification, repeated as the cell ages	Training on a representative dataset
Extrapolation	Potentially more robust outside the calibration range, but only while the model form and the identified parameters remain valid	Unreliable beyond the training distribution
Computational cost	High for full-order models; moderate for reduced-order models	Mostly incurred in training; inference cost depends on the architecture and hardware
Interpretability	High; parameters carry physical meaning	Low for deep architectures; moderate for tree and kernel methods
Uncertainty	Propagated analytically for selected models; otherwise evaluated numerically, for example by sampling or filtering	Native only for probabilistic estimators
Online suitability	Equivalent-circuit and selected reduced-order electrochemical or gray-box models	Well suited once trained
Typical failure mode	Model-form error and parameter drift	Distribution shift between laboratory and field

3. Formulation of the Estimation Problem and Health Metrics

Let the internal condition of a cell at cycle index n be summarized by a latent degradation state $x(n)$ taking values in a state space $X \subseteq \mathbb{R}^d$, and let $z(n) \in \mathbb{R}^p$ denote the vector of accessible measurements, comprising terminal voltage, current, surface temperature, and, where available, impedance. Diagnosis is the recovery of $x(n)$ from the observation history, and prognosis is the projection of x forward to the first index at which a failure criterion is met. Both are inverse problems that can be ill-posed or poorly conditioned. An instantaneous measurement may not determine the internal state uniquely, whereas identifiability from an observation history depends on the dynamics, the excitation and the signals available, and sensitivity to measurement noise is a separate property; physical structure and learned prior information are used to regularize the resulting estimation problems [11]. The remainder of this section makes the associated quantities precise. Notation is defined locally and summarized in Appendix A; where an established symbol is reused, its context-dependent role is stated explicitly.

3.1. Capacity, Health, and Charge

Discharge capacity is obtained by integrating current over the discharge interval, which gives the coulomb counting relation of Equation (1).

$$Q = \int_0^{t_d} I(t)\, dt \tag{1}$$

Here, $I(t)$ is the current in amperes at time t, taken positive on discharge, t_{d} is the total discharge duration, and Q is the delivered capacity. Equation (1) returns ampere-hours only when time is measured in hours; if the acquisition system records time in seconds, as most public corpora do, the integral must be scaled by $1/3600$. This convention is adopted once here and applied throughout, in particular to Equation (5). The capacity-based definition of SOH follows as the ratio of the maximum available capacity at cycle n to the capacity of the pristine cell, given in Equation (2).

$$SOH_C(n) = \frac{C_{max}(n)}{C_0} \times 100\% \tag{2}$$

where C_0 is the initial capacity and $C_{\max}(n)$ is the measured maximum capacity at cycle n. This quantity decreases monotonically in expectation while fluctuating locally through capacity regeneration during rest, a behavior that any estimator must accommodate. Failure is conventionally declared once the fully charged capacity has fallen to between 70% and 80% of the nominal one [49–52].

Capacity is not the sole admissible basis. Growth of internal resistance provides a definition better matched to power-limited duty, given in Equation (3).

$$SOH_R(n) = \frac{R_{EOL} - R(n)}{R_{EOL} - R_{BOL}} \times 100\% \tag{3}$$

with $R(n)$ representing the internal resistance at cycle n; R_{BOL} being its value at beginning of life, distinguished from the ohmic resistance R_0 of Equations (6), (17) and (18); and R_{EOL} meaning its value at the declared threshold. Written this way, the metric equals 100% at beginning of life and zero at the declared threshold, whereas the capacity-based metric of Equation (2) equals the end-of-life capacity fraction, typically 70% to 80%, at its threshold; the two scales therefore differ, and numerical values of the two can be compared only after their reference states and thresholds have been aligned. The two definitions diverge for cells in which power fade outpaces energy fade, so the operative definition should follow the application rather than convention [12,53–55].

SOC is the residual charge available relative to the maximum capacity of the current cycle, given in Equation (4), in which $Q_{\mathrm{avail}}(k)$ is the charge still available at sample k. It is an intra-cycle quantity, so it is indexed by the sample index k within a cycle, or equivalently by continuous time t, and never by the cycle index n, which indexes cycle-level degradation quantities such as SOH(n) and has that meaning throughout the review; the subscripted symbols n_{f}, n_{s} and n_{t} denote distinct quantities, namely the failure index of Equation (8) and the source and target sample counts of Equation (93). The relation between SOC and the measured voltage is nonlinear and depends on temperature and on the charge and discharge rate.

$$SOC(k) = \frac{Q_{avail}(k)}{C_{max}(n)} \times 100\% \tag{4}$$

In online operation, SOC is propagated recursively from an initial value through the ampere-hour integral of Equation (5) [56–58].

$$SOC(t) = SOC(t_0) - \frac{100}{C_{max}(n)} \int_{t_0}^{t} \eta \, I(\tau) \, d\tau \tag{5}$$

where $C_{\max}(n)$ is the maximum capacity at cycle n, the same capacity basis used in Equation (4), and the integration is again over time in hours, so that for time in seconds the coefficient $100/C_{\max}(n)$ becomes $100/(3600\, C_{\max}(n))$; $\eta \in (0, 1]$ is the coulombic efficiency accounting for charge losses, and the current is again taken positive on discharge, so that the integral subtracts charge on discharge and adds it on charge. Where charge

and discharge efficiencies differ materially, as they do for lithium–sulfur cells, in which the polysulfide shuttle consumes part of the charging current, a separate charge efficiency must be defined rather than a single η; η then depends on the direction of the current, and the usual convention is $\eta = 1$ on discharge, so that only the charging current is scaled. Two weaknesses follow directly from the structure of this relation: the integration accumulates sensor bias without bound, and the estimate inherits any error in the initial condition. Both are addressed by the recursive Bayesian schemes of Section 8, which is the principal reason SOC estimation and filtering theory are inseparable in practice.

3.2. State of Charge: Conditioning of the Estimation Problem

Equations (4) and (5) fix what state of charge is and how it is propagated; neither determines how accurately it can be recovered from measurement. Because the weight of attention in the literature, and in Sections 4–7 of this review, falls on cycle-level degradation, the conditions under which this intra-cycle quantity is identifiable are set out here before the degradation formulations are introduced. Three properties separate the problem from the degradation problem, and together they determine which of the later formulations apply to it.

First, where state of charge is inferred from the voltage response, the inversion that recovers it is ill-conditioned wherever the open-circuit-voltage characteristic is flat. Reconstructing the open-circuit voltage from the terminal measurement through Equations (18) and (19), and then inverting the characteristic, gives the first relation of Equation (6); the second relation states, to first order, the conditioning of that inversion.

$$SOC = V_{oc}^{-1}(V_t + IR_0 + V_1), \Delta SOC \approx \Delta V \left(\frac{dV_{oc}}{dSOC} \right)^{-1} \tag{6}$$

Here V_t, R_0 and V_1 are the terminal voltage, the ohmic resistance and the polarization voltage of the equivalent circuit of Equations (17)–(19); the inverse of the open-circuit-voltage characteristic exists only where that characteristic is strictly monotone in state of charge, δV is the error in the reconstructed open-circuit voltage, and δSOC is the resulting error in the inferred state of charge. The second relation is a first-order approximation, valid for a small error δV about an operating point at which the characteristic is differentiable with nonzero slope; the local amplification factor is the reciprocal of that slope and becomes unbounded as the slope approaches zero, and near a flat region the linear approximation itself can fail. For cells with a pronounced slope, the inversion is benign. For lithium iron phosphate, over its central plateau, it is poorly conditioned, and for the lithium–sulfur cells discussed in Section 4.4, the low-voltage plateau is flat enough that state of charge is poorly observable from voltage alone within it; an observability analysis of a zero-dimensional electrochemical model of such a cell shows this directly [59], and the difficulty has motivated estimators that return a state-of-charge interval from a classifier instead of a point value from an inversion [60]. Hysteresis between the charge and discharge characteristics, and the temperature dependence of the characteristic itself, enter as further departures from the single monotone inverse assumed in Equation (6).

Second, the recursion of Equation (5) is open-loop, so its errors do not decay. Writing b_I for a constant bias in the measured current, so that the measured current is $I + b_\mathrm{I}$, and ε_SOC for the error of the integrated estimate, taken as the estimate minus the true value, the error evolves according to Equation (7) when the coulombic efficiency η and the capacity $C_\mathrm{max}(n)$ are constant over the interval from t_0 to t and equal to the values used by the

estimator; if the bias, the efficiency or the capacity varies over that interval, the second term of Equation (7) is replaced by the corresponding time integral.

$$\varepsilon_{SOC}(t) = \varepsilon_{SOC}(t_0) - \frac{100\eta b_I(t - t_0)}{C_{max}(n)} \tag{7}$$

An error in the initial value therefore persists undiminished, and a current-sensor bias produces drift that grows linearly in elapsed time, with nothing in the recursion to correct either. This is the formal content of the two weaknesses noted after Equation (5), and it is the reason ampere-hour integration is used together with a voltage-based correction rather than on its own.

Third, state of charge is an intra-cycle quantity, whereas the degradation state is a cycle-level one, so the two are estimated on different time scales and, in general, from different signals. This difference has a systematic consequence for method selection. State-of-charge estimation is dominated by the sequential formulations of Section 8, in which the ampere-hour relation of Equation (5) and the branch dynamics of Equation (19) supply the transition and the terminal-voltage relation of Equation (18) supplies the measurement. Where the model and the excitation make the state observable, voltage feedback can correct the initial-value error of Equation (7) and limit the drift caused by its bias term; the bias can itself be estimated by appending b_{I} to the state, provided it is identifiable, but feedback cannot restore information that a locally flat voltage response does not carry. State of health and residual life are dominated instead by the regression and sequence formulations of Sections 6 and 7. Direct learned mappings from a window of voltage, current and temperature to state of charge are also used and can be accurate within the conditions represented in training [41]; because they draw on the whole input history rather than on a single voltage inversion, Equation (6) describes only part of their information content, and their reliability outside the training conditions must be established separately. Finally, the two quantities are coupled rather than independent: the capacity basis $C_{\max}(n)$ in Equation (4) is itself the quantity on which the state of health of Equation (2) is defined, so an error in state of health propagates directly into state of charge. Schemes that estimate the two jointly, typically by nesting a fast state-of-charge filter inside a slower capacity filter, are discussed in Section 8.2 [12].

3.3. End of Life and Residual Life

End of life is the index at which the cell ceases to meet its performance requirement. It is most often expressed as a discrete cycle count [61], as in Equation (8), with the residual life following as the difference in Equation (9). Formally $n_f = \inf\{n \geq 0 : SOH(n) \leq SOH_{\mathrm{EOL}}\}$, the first index at which the health metric falls to the declared threshold.

$$EOL = n_f \tag{8}$$

$$RUL = n_f - n \tag{9}$$

where n_f is the failure index of Equation (8) and n is the current cycle. Figure 1 shows a measured health trajectory against a declared threshold, which makes the construction concrete: the curve falls with local recoveries superimposed, and the crossing defines both the end of life and the residual interval. Written this way, RUL is a deterministic quantity. It is more properly a random variable, since the threshold crossing depends on a degradation trajectory that is itself stochastic and on future operating conditions that are unknown at the time of prediction. Section 4.2 develops the first-passage time formulation that makes this explicit and delivers a closed-form life distribution rather than a point estimate [62–64].

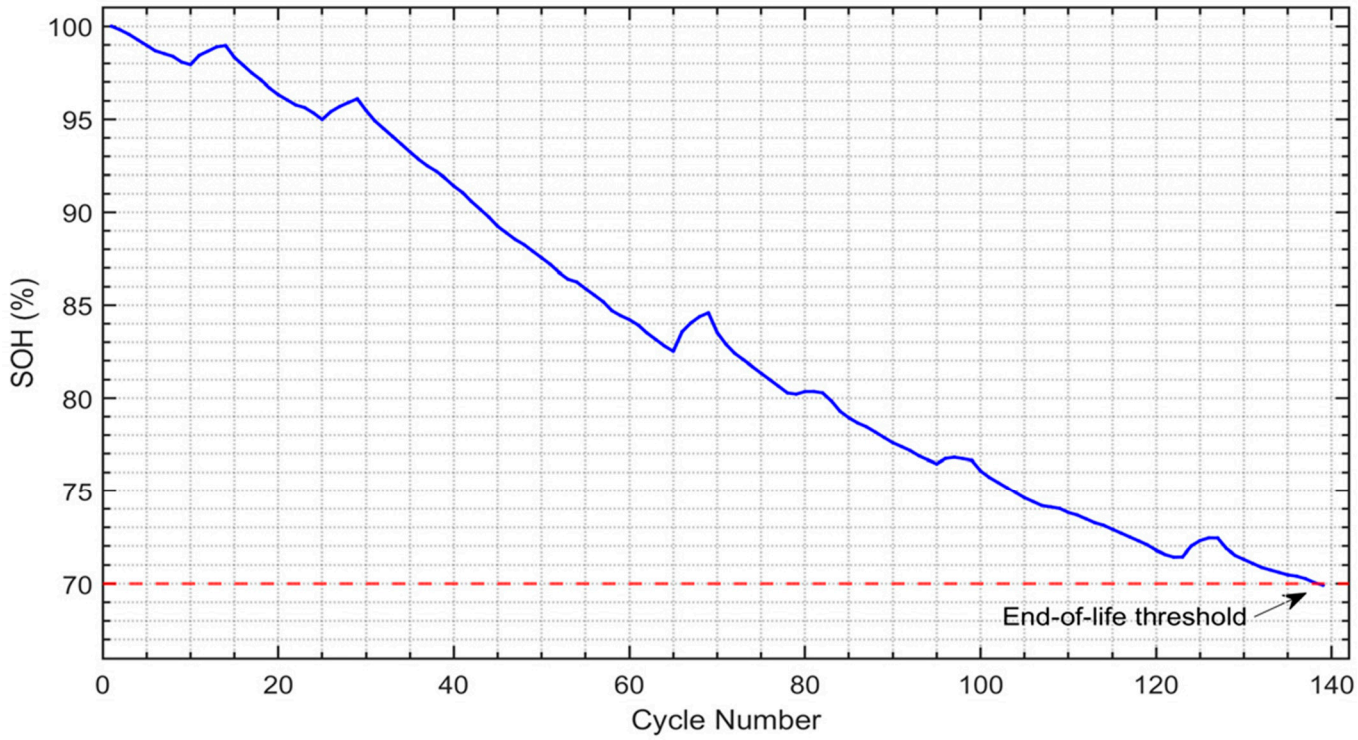


Figure 1. Measured state-of-health trajectory against cycle number for a cell cycled to failure, with the declared end-of-life threshold marked. Local recoveries superimposed on the falling trend are characteristic of capacity regeneration during rest. Reproduced without modification under the terms of the Creative Commons Attribution (CC BY) license from [65].

These quantities are mutually dependent. Capacity falls across the service life while fluctuating within each charge and discharge sequence, so the definition of failure must be stated with respect to a specified measurement condition if it is to be reproducible.

The workflow connecting raw measurement to a maintenance decision proceeds through six stages: data acquisition, signal conditioning, condition monitoring, health assessment, life prediction, and decision support. The reliability of each stage bounds the reliability of those downstream, so an error introduced during conditioning propagates unchecked into the life estimate [11,66].

Remark 1 (unified estimation problem)**.** *With the notation of this section, diagnosis is the estimation of $x(n)$ from the observation history $z(0), \ldots, z(n)$, and prognosis is the estimation of the first-passage index $n_f = \inf\{n \geq 0: SOH(n) \leq SOH_{EOL}\}$ of the latent trajectory, so that $RUL(n) = n_f - n$ is a random variable whose distribution is induced by the law of the trajectory and by the assumed future operating profile. Every formulation in Sections 4–11 is a specialization of this statement: Sections 4.1 and 4.3 fix the dynamics of x; Section 4.2 assigns a probability law to its increments and thereby to n_f; Sections 5–7 replace x by an engineered or learned functional of the measurements; Section 8 propagates the conditional law of x recursively; and Section 11.3 constrains the learned map by the residual of a governing relation.*

4. Degradation Modeling

Three formalisms are available for describing how the latent state evolves, and their historical succession is shown in Figure 2: electrochemical descriptions came first, equivalent circuits followed as a tractable reduction, and data-driven models arrived once cycling corpora became large enough to support them [67]. Deterministic empirical laws fit an algebraic trend to observed fade. Stochastic process models treat the trajectory as a random

function and deliver a life distribution. Physics-based models resolve the transport and reaction processes that generate the fade. The three are complementary, and the hybrid schemes of Section 11 combine them.

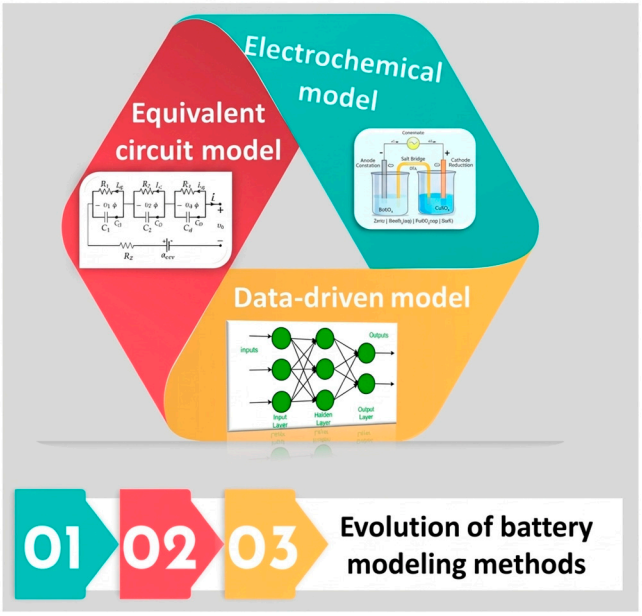


Figure 2. Evolution of battery modeling methods, from electrochemical descriptions through equivalent-circuit reductions to data-driven formulations. Panel (**a**) of the original figure has been adapted by cropping under the terms of the Creative Commons Attribution (CC BY) license from [67].

4.1. Deterministic Empirical Laws

Capacity fade arises from growth of the solid electrolyte interphase, loss of cyclable lithium, loss of active electrode material, lithium plating at low temperature or high charge rate, and mechanical fracture of electrode particles [6,68–70]. The principal pathways and the interfacial processes that drive them are shown in Figure 3. These processes are thermally activated, so their rate constants follow the Arrhenius law of Equation (10) [71].

$$k(T) = A \exp\left(-\frac{E_a}{R_g T}\right) \tag{10}$$

with A a pre-exponential factor, E_a the activation energy, R_g the universal gas constant, and T absolute temperature. Semi-empirical life models extend this by admitting throughput and current rate as explicit stress factors, as in Equation (11).

$$Q_{loss} = B \exp\left(-\frac{E_a + \gamma C_{rate}}{R_g T}\right)(A_h)^z \tag{11}$$

where Q_{loss} is the capacity loss, B is a pre-exponential factor, γ weights the rate dependence of the effective activation energy and is negative in the usual parameterization, so that a higher rate lowers the effective barrier and accelerates loss, C_{rate} is the applied rate, A_h is cumulative ampere-hour throughput, and z is a power-law exponent typically near one half. Because this exponent is not in general an integer, the ampere-hour throughput must be interpreted as a value normalized by a stated reference throughput, so that the bracketed quantity is dimensionless and the pre-exponential factor carries the units of capacity loss; reporting that factor without the reference makes fitted coefficients incomparable between studies. The square-root dependence reflects diffusion-limited interphase growth,

while a linear term captures the later regime in which loss of active material dominates. Superposing the two gives Equation (12).

$$C_{max}(n) = C_0 - \alpha\sqrt{n} - \beta n \tag{12}$$

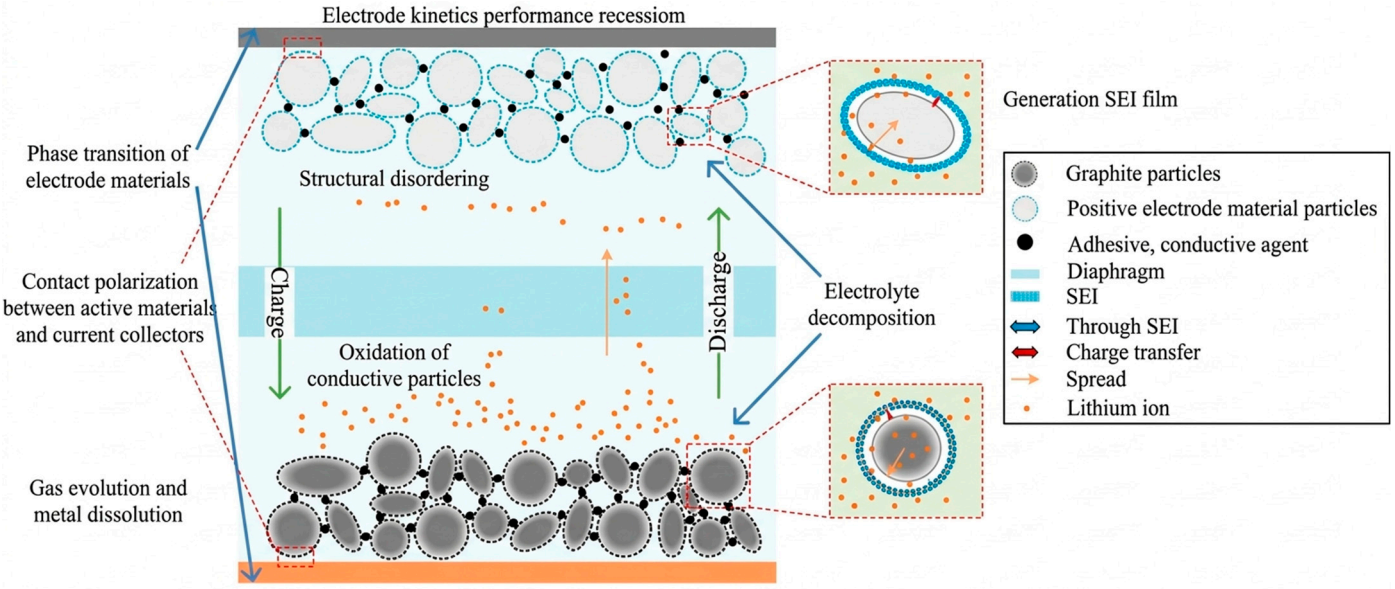


Figure 3. Degradation pathways in a lithium-ion cell, showing solid electrolyte interphase formation, structural disordering of the electrode, contact polarization, electrolyte decomposition, gas evolution, and metal dissolution. Reproduced without modification under the terms of the Creative Commons Attribution (CC BY) license from [71].

In Equation (12), C_0 and $C_{\max}(n)$ are the initial and maximum capacities of Equation (2), and α and β are the diffusion-limited and linear fade coefficients. Relations of this form are attractive because they are identifiable from a handful of parameters and they extrapolate in a physically defensible way. They are nevertheless deterministic, so they cannot represent the cell-to-cell dispersion that is observed even within a single manufacturing batch, and they return no interval around a life prediction.

4.2. Stochastic Degradation Processes

The natural remedy is to model the degradation path as a stochastic process. Let $X(t)$ denote a monotone or near-monotone degradation measure, such as the state of health lost since the beginning of life, so that the threshold w introduced below corresponds to the end-of-life criterion of Section 3.3. The Wiener process with drift, given in Equation (13), is the standard choice where the path is non-monotone, which is precisely the situation created by capacity regeneration [49,62,64,72].

$$X(t) = x_0 + \mu\Lambda(t;\theta) + \sigma_B B(\Lambda(t;\theta)) \tag{13}$$

with initial state x_0, drift coefficient μ, diffusion coefficient σ_B, standard Brownian motion B, and a time transformation $\Lambda(t;\theta)$, monotone increasing with $\Lambda(0) = 0$ and parameter vector θ, that admits nonlinear trends. The value of this construction is that the life distribution follows analytically. If failure is defined as the first passage of X across a threshold w, at the time $T_f = \inf\{t \geq 0: X(t) \geq w\}$, and the linear time scale $\Lambda(t) = t$ is taken, with $x_0 < w$ and

$\mu > 0$, the density of the first-passage time is given by Equation (14), which is the inverse Gaussian density of Equation (15).

$$f_{T_f}(t) = \frac{|w - x_0|}{\sqrt{2\pi\sigma_B^2 t^3}} \exp\left(-\frac{(w - x_0 - \mu t)^2}{2\sigma_B^2 t}\right) \tag{14}$$

$$T_f \sim IG\left(\frac{w - x_0}{\mu}, \frac{(w - x_0)^2}{\sigma_B^2}\right) \tag{15}$$

Equation (15) uses the mean–shape parameterization, with mean $(w - x_0)/\mu$ and shape $(w - x_0)^2/{\sigma_B}^2$, and describes total first-passage life from the initial state on the linear time scale. At a current time t_c, remaining useful life is $T_f - t_c$ conditioned on survival to t_c and the available observations. If $X(t_c) = x_c < w$ is known and the model parameters are fixed, the linear-clock residual-life distribution is obtained by replacing $w - x_0$ with $w - x_c$ in Equations (14) and (15). For a differentiable increasing nonlinear clock, the relevant operational-time increment is $\Lambda(t_c + u; \theta) - \Lambda(t_c; \theta)$, where u is future elapsed time; the corresponding residual-life density is the operational-time first-passage density evaluated at this increment and multiplied by its derivative with respect to u. Uncertainty in the current state or model parameters must be propagated through the filtering posterior. A point prediction and interval may then be obtained from the conditional distribution. Where a monotone latent degradation process is appropriate, the Gamma process in Equation (16) is an alternative with non-negative increments.

$$\Delta X(t) \sim Ga(\alpha_\Gamma \Delta\Lambda(t), \beta_\Gamma), \qquad f_{\Delta X}(x) = \frac{{\beta_\Gamma}^{\alpha_\Gamma \Delta\Lambda}}{\Gamma(\alpha_\Gamma \Delta\Lambda)} x^{\alpha_\Gamma \Delta\Lambda - 1} e^{-\beta_\Gamma x} \tag{16}$$

with $\Delta X(t) = X(t + \Delta t) - X(t)$ being the increment over an interval of length Δt, $\Delta\Lambda(t) = \Lambda(t + \Delta t; \theta) - \Lambda(t; \theta)$ being the corresponding increment of operational time, and $\alpha_\Gamma\ \Delta\Lambda(t)$ being the shape parameter of the increment and β_Γ the rate parameter, so that ${\beta_\Gamma}^{-1}$ is the scale. The subscript distinguishes these parameters from the capacity-fade coefficients α and β of Equation (12), which are unrelated quantities introduced in Section 4.1. The Gamma process does not share the elementary first-passage density of Equation (14); because its paths are monotone, however, the survival function of the life is available directly, for a process started at $X(0) = 0$, as $P(T_f > t) = P(X(t) < w) = \gamma(\alpha_\Gamma \Lambda(t), \beta_\Gamma w)/\Gamma(\alpha_\Gamma \Lambda(t))$, the regularized lower incomplete gamma function, in which $\gamma(\cdot, \cdot)$ is the lower incomplete gamma function and $\Gamma(\cdot)$ the gamma function, from which quantiles follow numerically. Choosing between these processes is an empirical question decided by whether the observed increments admit negative values. Applying a monotone latent model to a capacity signal that shows regeneration requires an observation model that accounts for the recoveries, either as measurement noise or as an explicit recovery term, and the direction of any resulting bias in the life prediction cannot be inferred from monotonicity alone. A state-space formulation of this kind is estimated by the sequential methods of Section 8, among which particle filtering has been applied both to battery life prediction and to state-of-charge estimation [73–75]. With the time transformation fixed, or normalized so that its scale is identifiable, the drift and diffusion coefficients of the Wiener model can be estimated by maximum likelihood from its independent Gaussian increments, and the shape and rate parameters of the Gamma model by the method of moments or maximum likelihood from its independent gamma increments; where measurement noise is not negligible, the likelihood must model it separately from the latent increments. When the clock is estimated jointly with the degradation parameters, identifiability has to be checked explicitly, since a free multiplicative scale of the clock is confounded with the Wiener drift and diffusion

coefficients and with the Gamma shape coefficient, and denser sampling does not remove this ambiguity.

4.3. Physics-Based Descriptions

Physics-based models divide by fidelity. Equivalent-circuit models represent the terminal response with lumped elements and are light enough for onboard execution [15]. The internal resistance model relates terminal to open-circuit voltage through a single ohmic term, Equation (17).

$$V_t(t) = V_{oc}(SOC(t)) - I(t)R_0 \tag{17}$$

This reproduces the instantaneous voltage step and not the relaxation that follows a load change. Adding a parallel resistor–capacitor branch yields the first-order Thévenin model of Equation (18), with branch dynamics given by Equation (19).

$$V_t(t) = V_{oc}(SOC(t)) - I(t)R_0 - V_1(t) \tag{18}$$

$$\frac{dV_1(t)}{dt} = -\frac{V_1(t)}{R_1C_1} + \frac{I(t)}{C_1} \tag{19}$$

where V_t is the terminal voltage, V_{oc} is the open-circuit voltage, R_0 is the ohmic resistance, V_1 is the polarization voltage, and R_1 and C_1 are the polarization resistance and capacitance. Second-order variants add a branch to separate charge-transfer from diffusion time constants at the cost of additional parameters. The parameters of these models carry limited physical meaning and drift as the cell ages, so they must be re-identified online [7,8,16].

Electrochemical models resolve the underlying processes directly. Interfacial kinetics follow the Butler–Volmer relation of Equation (20) and solid-phase lithium transport follows Fickian diffusion in spherical coordinates, shown in Equation (21) [76].

$$j = j_0\left[\exp\left(\frac{\alpha_a F\eta_{ov}}{R_gT}\right) - \exp\left(-\frac{\alpha_c F\eta_{ov}}{R_gT}\right)\right] \tag{20}$$

$$\frac{\partial c_s}{\partial t} = \frac{D_s}{r^2}\frac{\partial}{\partial r}\left(r^2\frac{\partial c_s}{\partial r}\right) \tag{21}$$

with j representing the local current density; j_0, the exchange current density; α_a and α_c, the anodic and cathodic charge-transfer coefficients; F, the Faraday constant; η_{ov}, the surface overpotential; c_s, the solid-phase concentration; r, the radial coordinate; and D_s, the solid-phase diffusivity. The overpotential η_{ov} in Equation (20) is distinct from the coulombic efficiency η of Equation (5). The pseudo-two-dimensional model that couples these relations across the electrode thickness is accurate but computationally heavy, which motivates reduced-order surrogates such as the single-particle model for online use.

Equation (21) is a partial differential equation in the radial coordinate and is not well posed without boundary and initial conditions. For a spherical particle of radius R_p, they are given in Equation (22).

$$\frac{\partial c_s}{\partial r}(0,t) = 0, -D_s\frac{\partial c_s}{\partial r}(R_p,t) = \frac{j}{F}, c_s(r,0) = c_{s,0} \tag{22}$$

The first condition imposes symmetry at the particle center, where the flux must vanish. The second equates the flux at the particle surface to the interfacial reaction rate of Equation (20) through Faraday's law and is the point at which the kinetic and transport descriptions are coupled. The third fixes the initial concentration profile, taken uniform at $c_{s,0}$. The sign convention is that j is positive in the anodic direction of Equation (20),

which for the particle is deintercalation, so that a positive j corresponds to lithium leaving the particle. Three assumptions are carried by this formulation and should be checked before it is applied: the active material is represented by spherical particles of a single representative radius; the solid-phase diffusivity D_s is independent of concentration over the range traversed; and, in the single-particle reduction, the reaction current is distributed uniformly across the electrode thickness, so that one representative particle per electrode suffices and electrolyte-phase gradients are neglected. The last of these fails first, and it fails at high rate, where electrolyte transport becomes limiting and the pseudo-two-dimensional description is required.

Thermal state is coupled to the electrical response through the energy balance of Equation (23), with the cell temperature propagated by the lumped model of Equation (24) [77].

$$\dot{q} = I(V_{oc} - V_t) - I\,T\frac{\partial V_{oc}}{\partial T} \tag{23}$$

$$m\,c_p\frac{dT}{dt} = \dot{q} - h\,A_s(T - T_{amb}) \tag{24}$$

In Equations (23) and (24), $\dot{q}$ is the total cell heat generation rate in watts, since the right-hand side of Equation (23) is a product of current and voltage; a volumetric rate q''' in watts per cubic meter would be obtained as $q''' = \dot{q}/V_{\text{cell}}$, with V_{cell} representing the cell volume, and the thermal balance would then have to be written per unit volume throughout. Here, m is the cell mass, c_p is the specific heat capacity, h is the convective heat transfer coefficient, A_s is the surface area, and T_{amb} is the ambient temperature. The first term of Equation (23) is irreversible Joule and polarization heating and the second is the reversible entropic contribution. The lumped model of Equation (24) likewise assumes a spatially uniform cell temperature, a single convective path to the ambient, and heat-capacity and heat-transfer parameters that do not vary over the temperature range considered; it is therefore a description of cell-average behavior and cannot resolve the internal gradients that govern local degradation or the initiation of thermal runaway. Because temperature simultaneously accelerates degradation and shifts the signals from which health is inferred, thermal and health states are not separable, and estimators that treat them jointly are more reliable than those that do not [78].

Impedance spectroscopy provides the experimental bridge between the circuit and electrochemical descriptions. The Nyquist representation of the spectrum separates three regions: the ohmic series resistance at high frequency, the charge-transfer semicircle at intermediate frequency, and the diffusion-limited response at low frequency. Each maps onto a distinct degradation pathway, which makes impedance features among the most physically interpretable health indicators available [24].

The three formalisms are complementary rather than competing, and the hybrid schemes of Section 11 operate in the region where they overlap. Table 2 compares them across the criteria that govern method selection.

4.4. Transferability of the Formulations Across Chemistries

The formulations of Sections 4.1–4.3 were identified on lithium-ion evidence, and the formats now entering deployment, among them sodium-ion, lithium–sulfur, and solid-state constructions, raise the question of how far they carry over. That question is addressed here in two stages: structurally first, by identifying which of the preceding equations survive a change in the underlying degradation process, and then empirically, by reviewing at the end of the subsection the studies available for each format. Establishing which degradation mechanisms operate in any particular chemistry is a material question requiring chemistry-specific evidence, and no such claim is made below. What is set out instead is which of the

preceding equations retain their functional form when a given property of the degradation process changes, and which do not. The analysis is therefore organized by process property rather than by chemistry, and it identifies what would have to be established, and what would have to be re-identified, before any of these formulations were applied to a new system. The statements in the next three paragraphs are properties of the formulations of Sections 4.1–4.3 and of the assumptions they carry, and they are not claims about any particular chemistry; the empirical literature that does exist for each format is reviewed separately below, where both its extent and its limits are stated.

Three levels of dependence can be distinguished. The health metrics in Equations (2)–(4) are definitions once their capacity, charge and resistance reference values have been specified. Defining failure as the first crossing of a stated degradation threshold is also chemistry-independent, but the process model and resulting life distribution are not: Equation (13) assumes a time-changed Wiener process, while Equations (14) and (15) additionally use a linear clock, positive drift, positive diffusion and an initial state below the threshold. Their transfer therefore requires these assumptions to remain appropriate. The empirical laws in Equations (10)–(12) and equivalent-circuit relations in Equations (17)–(19) may retain their form only when their modeling assumptions remain applicable; coefficients and, where necessary, model order must be re-identified on target-domain data. Re-identifying coefficients alone cannot repair an invalid model structure.

Several assumptions should therefore be checked explicitly. Equation (5) uses a single coulombic-efficiency factor; if efficiency depends materially on state, current direction or operating history, the integration model should include that dependence. Equation (16) requires non-negative independent Gamma increments with the stated shape and rate structure. Multiple degradation mechanisms do not alone invalidate this model, although differing increment laws or dependencies may require a richer state or alternative process. Equation (36) defines incremental capacity, but the interpretation and numerical stability of its peaks depend on the voltage–charge response, measurement resolution and pre-processing; a near-flat voltage response can make numerical differentiation particularly sensitive. Finally, omitted influences can bias the state estimate or compromise identifiability, so transfer should include an observability and identifiability assessment of the target-domain model.

The estimator families of Sections 6 and 7 are not automatically chemistry-agnostic. They inherit the information content of their features and the distribution of their training data. Transfer therefore calls for representative target-system evaluation and, where necessary, retraining or explicit domain adaptation as formalized in Equation (93). These statements define conditions to examine before transfer; they do not establish that those conditions hold for a particular chemistry or that transfer has been empirically demonstrated. The present narrative review does not provide a systematic estimate of cross-chemistry dataset coverage.

The empirical position differs sharply among the three formats. It is set out concretely here, because the structural analysis above is silent on how much evidence exists and the asymmetry governs how far any of these formulations can presently be relied upon.

For sodium-ion cells, the evidence base is now sufficient to test some of the formulations above rather than merely to motivate them. State-of-health forecasting over a 20-cycle horizon has been demonstrated on six custom-made pouch cells with a recurrent network trained on health indicators drawn from the charge, discharge and post-charge relaxation segments, over 1200 cycles to roughly the 80% threshold [79]. State-of-charge estimation has been addressed by hybrid schemes that weight a learned estimate against an equivalent-circuit filter of the type set out in Equations (18), (19) and (73)–(77), there with a second-order circuit and an unscented filter [80]. Impedance-based health estimation has

been reported in which features were selected for insensitivity to temperature so that a model identified at a single temperature could be applied across an operating range [81], which addresses the distribution shift discussed in Section 11.2 by invariant feature selection rather than by the discrepancy minimization of Equation (93). Public cycling data for the chemistry has also begun to appear: the multi-chemistry life-prediction benchmark listed in Section 5.1 is, to the authors' knowledge, the first to release sodium-ion life data alongside lithium-ion data [82]. A comparative review of data-driven health and safety estimation for chemistries other than nickel-manganese-cobalt lithium-ion, organized around sodium-ion and lithium iron phosphate, is also available [83].

For lithium–sulfur cells, the literature is smaller but methodologically informative, because the chemistry directly violates one of the conditions of Section 3.2. The discharge characteristic has an extended low-voltage plateau over which the terminal voltage is almost insensitive to the remaining sulfur inventory, so the inversion of Equation (6) is ill-conditioned there. An observability analysis of a zero-dimensional electrochemical model shows that the state is poorly observable from voltage in that region and derives a reduced-order model with better observability in both plateaus [59]. The responses to this difficulty are informative about which formulations transfer. A dual filter that identifies circuit parameters on a fast time scale and the state on a slow one improves accuracy substantially for partially discharged cells [84]; a classifier that returns a state-of-charge interval, combined with ampere-hour integration for the continuous value, avoids the ill-conditioned inversion rather than attempting it [60]; and an extended Kalman filter applied to the differential-algebraic system of a three-step electrochemical model with polysulfide shuttle recovers the individual species masses from current and voltage alone [85]. Aging has been characterized directly on prototype pouch cells, with health estimated from identified circuit parameters, and with the finding that power fade proceeds far more slowly than capacity fade [86]. That finding has a direct bearing on Section 3.1: the resistance-based metric of Equation (3) and the capacity-based metric of Equation (2) diverge more strongly for this chemistry than for lithium-ion chemistry, so the choice between them is not a matter of convention here.

For solid-state cells, the available evidence is very limited. Only isolated studies were found that estimate state or cycle life on measured solid-state cells. Cycle life has been predicted by symbolic regression on features drawn from voltage and capacity curves for a set of 12 polymer-electrolyte cells whose lives ranged from roughly 70 to just over 200 cycles [87], and state of charge has been estimated from impedance spectra by Gaussian process regression in a manner that remains valid across different states of health [88]. The charge and discharge records of the first of these studies have been made public by its authors, but no larger public cycling or aging dataset for solid-state cells was located. Three consequences follow for this review: the sparse and heterogeneous measurements reviewed here do not yet provide a comprehensive validation of the structural conditions identified above for this format; no identification of the life distributions of Equations (14)–(16) on solid-state run-to-failure data was found; and any statement about the transfer of the learned estimators of Sections 6 and 7 to this format is a conjecture rather than a finding. This is the clearest instance of the third limitation stated in Section 2.4.

The evidence base is therefore strongly asymmetric. Lithium-ion supplies the great majority of run-to-failure data and of the empirical results reviewed in Sections 5–10; sodium-ion now supports direct empirical tests of several of these formulations; lithium–sulfur supplies a small but methodologically informative literature concentrated on identifiability; and solid-state supplies very little. The scope of the review should be read accordingly. It is a review of the mathematics of battery diagnostics and prognostics, evaluated predominantly on lithium-ion evidence, together with an explicit analysis of what would be

required to carry that mathematics to the formats now entering deployment, and with the empirical position for each of those formats reported above. It is not a review of established practice for the non-lithium-ion formats, and no such claim is made.

5. Data Acquisition, Signal Processing, and Feature Construction

The estimator of Section 3 consumes features rather than raw samples, and the quality of those features bounds the attainable accuracy. This section treats their construction as a signal-processing problem.

5.1. Benchmark Data and Acquisition

Standardized repositories underpin algorithmic comparison. The NASA Prognostics Center of Excellence (PCoE) provides cycling data to failure under regulated conditions, of which the B0005, B0006, B0007, and B0018 sets remain the most widely used for state of health and remaining useful life validation [89,90]. The Center for Advanced Life Cycle Engineering (CALCE) supplies data across several chemistries under dynamic stress protocols emulating driving behavior [91,92]. The largest fast-charging corpus was released through a collaboration between the Massachusetts Institute of Technology, Stanford University, and the Toyota Research Institute, comprising 124 lithium iron phosphate cells cycled to failure under 72 distinct charging protocols [93]; the demonstration that cycle life is predictable from the first hundred cycles of voltage data was built on that resource. A subsequent closed-loop campaign extended it by allowing the learning algorithm to select the protocols to be tested [94]. Combined with early prediction, the campaign identified high-cycle-life charging protocols among 224 candidates in 16 days, against more than 500 days for an exhaustive search, so that early prediction converts experimental cost into computational cost. Further sets include Oxford cells cycled under an energy-trading duty profile [95], cell-level and system-level operating measurements from a modular battery system of four series-connected packs at Karlsruhe [96], and a second-life corpus obtained by cycling retired cells under grid-storage conditions [97].

These are summarized in Table 3.

Table 3. Publicly available battery experimental datasets for which a citable source is given in this review. Each row states experiment type, repository, cell count, physical format, nominal capacity, electrode chemistry, recorded parameters, and the reference from which the entry is taken, so that every row can be traced to its origin. Cell counts and formats follow the sources cited in the final column. Datasets without a citable source are not listed.

Experiment Type	Repository	Cells	Cell Format	Nominal Capacity	Chemistry	Recorded Parameters	Source
Cyclic aging	NASA PCoE	34	18,650	2 Ah	Mixed	Capacity, IR, V, I, T	[89,90]
	Toyota Research Institute	124	18,650	1.1 Ah	LFP	Capacity, IR, V, I, T	[93]
	Oxford	6	Pouch (Kokam SLPB78205130H)	16 Ah	NMC	Capacity, V, I, T	[95]
	CALCE CS2, CX2 and PL cohorts	43 (15 CS2 + 12 CX2 + 16 PL)	Prismatic/pouch	1.1/1.35/1.5 Ah	LCO	Capacity, V, I, T	[91,92]
	NASA PCoE (random walk)	28	18,650	2.2 Ah	LCO	Capacity, IR, V, I, T	[89]
Driving cycle	CALCE	2	26,650	2230 mAh	LFP	V, I, T	[91]
Calendar aging	CALCE	144	Multiple	Various	Mixed	Capacity, IR, E, V, I	[91]
Battery-system operation	Karlsruhe Institute of Technology	44 (4 packs of 11 in series)	Prismatic	40 Ah	NMC	Cell V and T; system V and I	[96]

Table 3. *Cont.*

Experiment Type	Repository	Cells	Cell Format	Nominal Capacity	Chemistry	Recorded Parameters	Source
Second life	Stanford (grid storage)	6	Cylindrical 21,700 (LG Chem INR21700-M50T)	4.85 Ah	NMC/graphite–silicon	Capacity, V, I, T; pulse-power tests and EIS	[97]
Aggregated benchmark	BatteryML	383	Multiple	Various	Mixed	Standardized cycling records	[98]
	BatteryLife	About 1000	Multiple	Various	Li-ion, Na-ion, Zn-ion	Standardized cycling records, life labels	[82]

Abbreviations: IR, internal resistance; V, voltage; I, current; T, temperature; E, energy; EIS, electrochemical impedance spectroscopy; LFP, lithium iron phosphate; LCO, lithium cobalt oxide; NMC, lithium nickel manganese cobalt oxide.

The datasets of Table 3 are not interchangeable, and the choice among them determines which estimation task can be addressed and which split protocol of Section 10.3 is available. Table 4 states, for each source, the tasks it supports, the basis for that judgment, and the limitation that bounds its use.

Table 4. Suitability of the public datasets of Table 3 for each estimation task, with the basis for each judgment and the principal limitation of each source. Exploratory pack-level diagnosis is supported only by Karlsruhe battery-system data, which carries no reference fault, SOC or health labels, and the final row records that no listed dataset supports pack-level aging studies.

Dataset	Estimation Tasks it Supports	Basis for the Judgment	Principal Limitation
NASA PCoE (cyclic aging)	SOH and RUL; cross-temperature splits	Capacity, internal resistance, voltage, current and temperature recorded over cycling to an end-of-life criterion at several ambient temperatures	Cells are divided among small groups that differ in discharge and temperature protocol, so a cell-wise split within any one condition has few units
Toyota Research Institute	RUL, early-life prediction and cross-protocol generalization	The largest single-source run-to-failure cohort in Table 3, cycled under many distinct fast-charging protocols	One chemistry, one format and one laboratory; it records fast-charging cycles rather than dynamic load profiles, and the flat open-circuit-voltage characteristic of LFP makes voltage-based SOC inversion poorly conditioned (Section 3.2)
Oxford	SOH under realistic grid duty	Year-long cycling under current profiles derived from day-ahead energy-trading operation, with monthly capacity measurements	Six cells and monthly capacity labels only; the dataset description reports no low-rate open-circuit characterization, so it does not support the differential features of Section 5.3
CALCE CS2, CX2 and PL cohorts	SOH and RUL; cross-cohort comparison	Long cycling records across three cohorts, and among the most widely used benchmarks in the literature	LCO is not representative of current traction cells, and the capacity records contain abrupt outliers that require the conditioning of Section 5.2
NASA PCoE (random walk)	SOC and SOH under non-constant duty	Randomized load profiles rather than constant-current cycling, so the duty resembles field operation	Reference capacity is measured only intermittently, so health labels are sparse
CALCE (driving cycle)	SOC under dynamic load	Measured drive-cycle current profiles rather than laboratory cycling; the LFP plateau makes it a demanding test of SOC estimators (Section 3.2)	Two cells: a cell-wise holdout is possible, but it gives a weak basis for estimating variability across cells
CALCE (calendar aging)	Calendar-aging models and storage SOH	Large cohort with capacity, resistance and energy recorded under storage	No cycling, so unsuitable for the cycle-indexed RUL formulation of Section 3.3

Table 4. *Cont.*

Dataset	Estimation Tasks it Supports	Basis for the Judgment	Principal Limitation
Karlsruhe Institute of Technology	Exploratory pack-level diagnosis: cell-to-cell inconsistency and anomaly screening under system operation	Per-cell voltage and temperature for 44 cells in four series-connected packs, with system voltage and current, over 19 charge, discharge and rest profiles	Operational monitoring only, without aging tests, capacity checks or reference labels for health, faults or SOC; it therefore supports exploratory pack-level analysis, but not supervised fault or SOC benchmarking or pack-level SOH and RUL studies
Stanford (grid storage, second life)	Second-life SOH and impedance-based diagnosis	Pulse-power tests and impedance spectra recorded alongside cycling, supporting the features of Section 4.3	Six cells; grid-storage duty does not transfer to automotive duty
BatteryML	Benchmarking, cross-dataset and external validation	Records from several source datasets expressed under one schema, which makes external validation practicable	Aggregation conceals protocol differences between constituents; a common schema is standardization and not harmonization
BatteryLife	Cross-chemistry transfer, external validation and life prediction	The only public source located that provides sodium-ion life labels alongside lithium-ion, as discussed in Section 4.4	As for BatteryML, aggregation conceals protocol differences between constituents, and the size of each non-lithium cohort should be checked before cross-chemistry claims are made
Pack-level aging records	Not supported by any dataset in Table 3	The Karlsruhe data provide per-cell measurements within packs but no aging or health labels, and no other listed source records modules or packs	Pack-level SOH and RUL formulations, including the graph-based construction of Section 7.4, therefore cannot be benchmarked on the public data listed here

Abbreviations: LCO, lithium cobalt oxide; LFP, lithium iron phosphate; RUL, remaining useful life; SOC, state of charge; SOH, state of health.

Two developments have altered the benchmarking picture. Curated aggregations standardize heterogeneous sources into a single schema, as in BatteryML [98] and the BatteryLife collection, which integrates 16 datasets spanning lithium-ion, sodium-ion, and zinc-ion chemistries [82]. Fleet-scale field corpora have also appeared, including operational records from 300 vehicles over three years [99,100], which begin to close the distance between laboratory cycling and service conditions [43].

Acquisition proceeds through controlled experiment or field monitoring. Laboratory protocols include capacity tests establishing reference values, hybrid pulse power characterization quantifying resistance against SOC and rate, impedance spectroscopy, cycle-life testing, calendar-life testing under storage, and accelerated testing that compresses the degradation timescale by elevating stress. Standard driving profiles are emulated to produce realistic load data. Selection among these procedures follows from the target quantity: capacity tests calibrate the health metric, pulse characterization resolves the resistance terms, and cycling under a driving profile exercises the stress factors that dominate in service. Safety and reliability testing forms a third group, covering abnormal condition, external stress, and extreme environmental protocols that generate the fault signatures discussed in Section 9.

The relationship between what is measured, how it is analyzed, and what degradation quantity is recovered is set out in Figure 4 [101]. Field monitoring captures behavior that laboratory cycling cannot reproduce. Parameters are relayed from the battery management system through a vehicle terminal to a central platform, with positional data supplying operating context. Such records contain missing segments, sensor noise, and incomplete run-to-failure histories, so preprocessing is a precondition rather than a refinement [24,30–32].

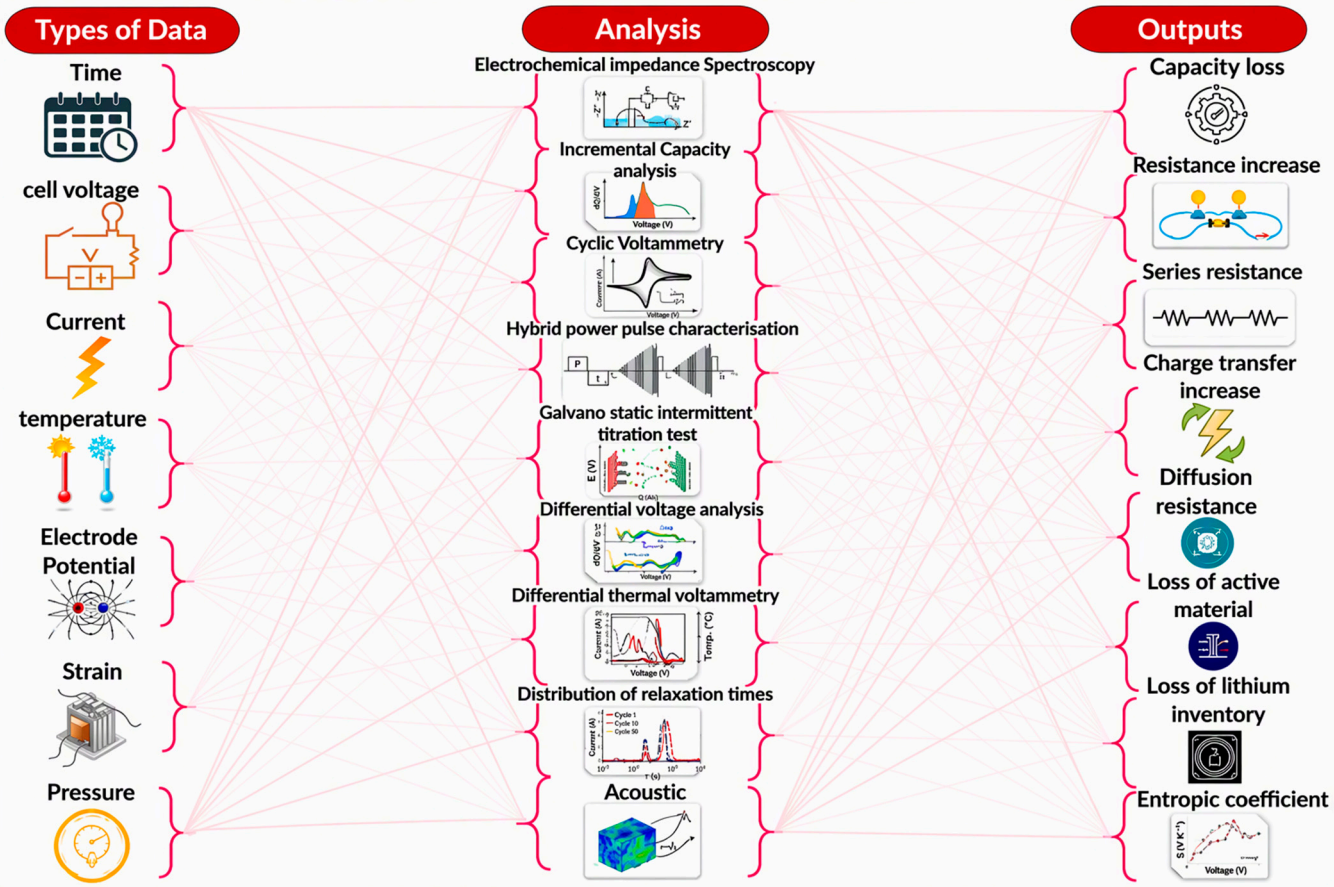


Figure 4. Relationship between measured quantities, analysis techniques, and recovered degradation descriptors, spanning electrical, thermal, mechanical, and acoustic measurement channels. Reproduced without modification under the terms of the Creative Commons Attribution (CC BY) license from [101].

5.2. Conditioning and Filtering

Cleaning unifies formats, removes anomalous samples, and imputes gaps. Linear interpolation of a signal $x(t)$ across a gap bounded by the samples taken at t_k and t_{k+1}, for $t_k \leq t \leq t_{k+1}$, is given in Equation (25).

$$x(t) = x(t_k) + \frac{t - t_k}{t_{k+1} - t_k}(x(t_{k+1}) - x(t_k)) \tag{25}$$

Normalization places features on a common scale, which matters for gradient-based optimization. Min-max scaling maps features to the unit interval and z-score standardization gives zero mean and unit variance; the two relations are collected in Equation (26), in which x is a raw feature value, $\tilde{x}$ is its scaled value, $x_{\min}$ and $x_{\max}$ are the smallest and largest values of the feature, and μ_x and σ_x are its mean and standard deviation.

$$\tilde{x} = \frac{x - x_{min}}{x_{max} - x_{min}}, \tilde{x} = \frac{x - \mu_x}{\sigma_x} \tag{26}$$

These statistics must be computed on the training partition alone. Using full-dataset statistics leaks information across the split and is a recurring source of optimistic bias in reported results.

Noise suppression matters most for the differential quantities of Section 5.3, whose numerical differentiation amplifies high-frequency content. The moving average of

Equation (27) weights a window uniformly, while Gaussian smoothing, Equation (28), weights it with a Gaussian kernel.

$$\bar{x}(i) = \frac{1}{N_w} \sum_{j=i-N_w+1}^{i} x(j) \tag{27}$$

$$x_{GS}(i) = \sum_{j=-L}^{L} g_j x(i-j) \tag{28}$$

In Equation (27), $\bar{x}(i)$ is the moving average at sample i of the sampled signal $x(j)$ and N_w is the window length, written with a subscript to distinguish it from the failure threshold w of Equations (14) and (15); in Equation (28), $x_{GS}(i)$ is the smoothed value and the summation runs over a truncated support of half-width L samples. The weights g_j are defined discretely as the exponential kernel $\exp(-j^2/(2\sigma^2))$, with σ the smoothing width in samples, divided by the sum of that kernel over the same window, so that they sum to unity and the filter has unit gain. Using the continuous Gaussian prefactor does not, in general, normalize a discrete kernel, even with unbounded support; exact unit gain requires division by the sum of the sampled weights. The choice between Gaussian smoothing and the moving average depends on the noise spectrum, the window length and the signal features that must be preserved [24]. The wavelet transform of Equation (29), in which ψ is the mother wavelet, the asterisk denotes complex conjugation, $a \neq 0$ is the scale and b is the translation, localizes the signal jointly in time and frequency.

$$W_x(a,b) = \frac{1}{\sqrt{|a|}} \int_{-\infty}^{\infty} x(t)\, \psi^* \left(\frac{t-b}{a} \right) dt \tag{29}$$

Empirical mode decomposition offers a data-adaptive alternative requiring no basis choice, resolving a signal $x(t)$ into K intrinsic mode functions $\mathrm{IMF}_i(t)$ and a residual $r_K(t)$, as shown in Equation (30) [102].

$$x(t) = \sum_{i=1}^{K} IMF_i(t) + r_K(t) \tag{30}$$

The residual is frequently used alone for capacity estimation, since it follows the underlying trend while excluding the regeneration transients that otherwise corrupt a fit. Applied to an incremental capacity curve, filtering recovers the peak structure that raw differentiation buries under noise impulses. The methods are summarized in Table 5.

Table 5. Summary of data preprocessing methods applied in battery diagnostics and prognostics, with the objective addressed, the governing formulation, and the principal limitation of each technique.

Stage	Method	Equation	Objective	Principal Limitation
Cleaning	Outlier removal and format unification	-	Remove anomalous readings from sensor and communication faults	Aggressive thresholds discard genuine capacity regeneration events
Cleaning	Linear interpolation	Equation (25)	Impute missing samples and restore uniform sampling	Underestimates curvature across long gaps
Scaling	Min-max normalization	Equation (26)	Map features to the unit interval	Sensitive to outliers that define the range
Scaling	Z-score normalization	Equation (26)	Standardize to zero mean and unit variance	Assumes an approximately stationary distribution

Table 5. *Cont.*

Stage	Method	Equation	Objective	Principal Limitation
Filtering	Moving average	Equation (27)	Suppress short-term fluctuation and expose trend	Attenuates peak amplitude in differential curves
Filtering	Gaussian smoothing	Equation (28)	Reduce noise while preserving signal shape	Requires selection of the kernel width
Filtering	Wavelet transform	Equation (29)	Localize the signal in time and frequency	Requires selection of the mother wavelet
Decomposition	Empirical mode decomposition	Equation (30)	Separate capacity regeneration from the underlying trend	Mode mixing and end effects; computationally heavy

5.3. Health Indicator Construction

A health indicator is any scalar functional of the measured signals that correlates with the latent degradation state. An admissible indicator varies smoothly over life, separates beginning-of-life from end-of-life conditions by a wide margin, and maintains a monotone relation with capacity fade [24,103–107].

The raw material from which indicators are built comprises the measured signals within a cycle, the capacity trajectory across cells, and the systematic displacement of the charge and discharge voltage curves with cycle number [71]. Direct indicators are computed from these. Voltage-based quantities are the largest family. The mean voltage falloff of Equation (31) measures the average separation between the nominal voltage V_{nom} and the N terminal-voltage samples $V(i)$ of a discharge window

$$MVF = \frac{1}{N}\sum_{i=1}^{N}(V_{nom} - V(i)) \tag{31}$$

and grows steadily with age. Statistical descriptors over the same window supply complementary information: for N samples $x(i)$ of a signal in the window, with mean μ_{x} and standard deviation σ_{x}, the root mean square of Equation (32) captures magnitude, while skewness and kurtosis, collected in Equation (33), describe distributional asymmetry and tail weight, both of which shift measurably with aging [108–111].

$$RMS = \sqrt{\frac{1}{N}\sum_{i=1}^{N}x(i)^2} \tag{32}$$

$$Skew = \frac{1}{N}\sum_{i=1}^{N}\left(\frac{x(i)-\mu_x}{\sigma_x}\right)^3, \quad Kurt = \frac{1}{N}\sum_{i=1}^{N}\left(\frac{x(i)-\mu_x}{\sigma_x}\right)^4 \tag{33}$$

Sample entropy, Equation (34), quantifies the irregularity of a sequence of length N and was introduced for physiological time series [112]; applied to a voltage sequence, its value as a health indicator, including whether its trajectory over life is monotone, has to be established for the particular signal and operating conditions.

$$SampEn(m, r, N) = -\log\left(\frac{A^m(r)}{B^m(r)}\right) \tag{34}$$

with m being the embedding dimension, r being the similarity tolerance, and $A^m(r)$ and $B^m(r)$ being the numbers of template pairs matching within r at embedding lengths $m + 1$ and m, respectively. Time-based indicators measure durations such as the constant-current charging interval, the time for the discharge voltage to fall by a fixed increment, and the charging voltage saturation time, each of which contracts as capacity is lost and resistance

grows. Temperature-based indicators capture the area under the surface temperature curve or the rate of rise across a fixed voltage increment. Energy-based indicators integrate the product of the terminal voltage $V_t(t)$ and the current $I(t)$ during discharge, between the instants t_1 and t_2 at which the voltage crosses the two bounds of a fixed voltage interval, as in the energy of equal discharge voltage interval (EEDVI) of Equation (35), and correlate strongly with fade [113–115].

$$EEDVI = \int_{t_1}^{t_2} V_t(t)\, I(t)\, dt \tag{35}$$

Statistical indicators divide into basic descriptors of location and spread, impulsive measures such as the crest and clearance factors, and high-order moments. The complete classification is given in Table 6.

Table 6. Classification of battery health indicators by signal domain, with representative features, the governing formulation, and the practical constraint associated with each group.

Group	Representative Indicators	Equation	Practical Constraint
Voltage, statistical	Mean voltage falloff, root mean square, skewness, kurtosis	Equations (31)–(33)	Window length must be fixed across cells for comparability
Voltage, entropy	Sample entropy of the discharge voltage sequence	Equation (34)	Sensitive to the embedding dimension and tolerance
Voltage, geometric	Arc length, velocity, curvature of the terminal voltage curve	-	Requires a consistently defined voltage interval
Voltage, relaxation	Rest voltage value and recovery rate after load removal	-	Requires a sufficiently long rest period
Time-based	Constant-current duration, time for an equal voltage drop, charging voltage saturation time	-	Depends on a fixed charge protocol
Temperature-based	Surface temperature rise rate, area under the temperature curve	-	Confounded by ambient conditions and cooling state
Energy-based	Energy over an equal discharge voltage interval	Equation (35)	Interval bounds must be selected by correlation analysis
Indirect, differential	Incremental capacity and differential voltage peak position, height, and area	Equations (36) and (37)	Requires low current rate and prior smoothing
Indirect, frequency	Median frequency, band power, signal-to-noise ratio, total harmonic distortion	-	Requires a stationary segment of adequate length
Indirect, impedance	Ohmic, charge-transfer, and diffusion resistance from the spectrum	-	Requires dedicated instrumentation and a stable thermal state

Indirect indicators are obtained by transformation. Incremental capacity analysis differentiates the charge $Q(i)$ accumulated up to sample i with respect to the terminal voltage $V(i)$, as shown in Equation (36), exposing the electrochemical phase transitions that correlate with specific aging modes; differential voltage analysis supplies the inverse relation, Equation (37).

$$IC = \frac{dQ}{dV} \approx \frac{Q(i) - Q(i-1)}{V(i) - V(i-1)} \tag{36}$$

$$DV = \frac{dV}{dQ} \approx \frac{V(i) - V(i-1)}{Q(i) - Q(i-1)} \tag{37}$$

Representative curves of both kinds are shown in Figure 5. Peaks in the incremental capacity curve correspond to phase equilibria in the electrode materials, and their position,

amplitude, and area evolve in ways attributable to loss of lithium inventory or loss of active material. Peaks attenuate and shift systematically as the cell ages, which is what makes them diagnostic of the mechanism rather than merely of the extent of fade. Since the curves are formed by numerically differentiating a sampled signal, the smoothing of Section 5.2 is required rather than optional. Frequency-domain indicators are obtained by discrete Fourier or power spectral density transforms, yielding median frequency, band power, signal-to-noise ratio, and total harmonic distortion, all of which reflect impedance change [24].

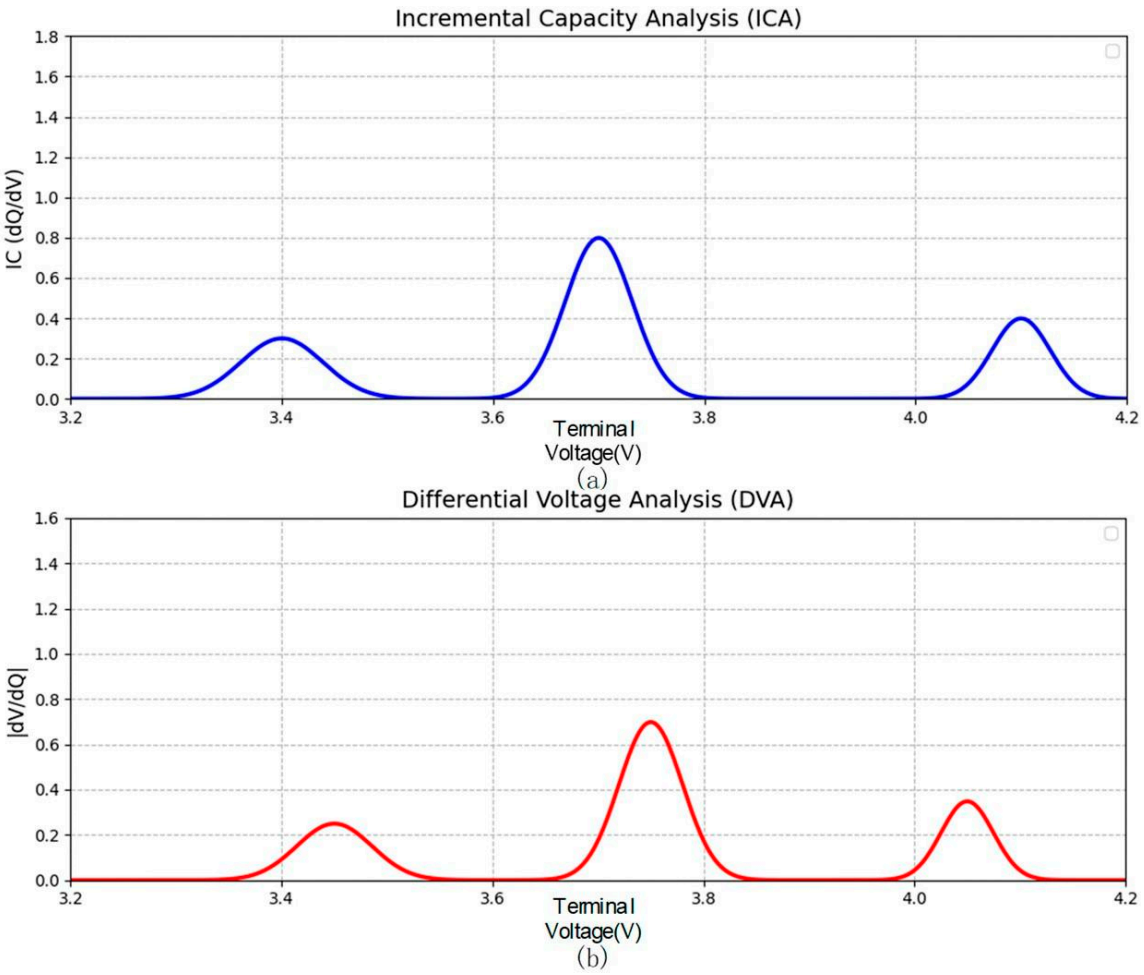


Figure 5. Illustrative incremental capacity analysis (ICA) and differential voltage analysis (DVA) curves: (**a**) ICA and (**b**) DVA, from Tang et al. [71], Figure 3. The panels are not asserted to be matched transforms of a single measured cycle. Reproduced without modification under the terms of the Creative Commons Attribution (CC BY) license from [71].

Where an indicator set is learned rather than designed, a sparsity penalty on the latent code enforces the same parsimony that explicit selection achieves. The sparse autoencoder objective of Equation (38) adds to the reconstruction loss L_{AE} of Equation (59) the Kullback–Leibler divergence $D_{\mathrm{KL}}(\rho\|\hat{\rho}_j) = \rho\log(\rho/\hat{\rho}_j) + (1-\rho)\log[(1-\rho)/(1-\hat{\rho}_j)]$ between Bernoulli distributions whose means are a target activation rate ρ and the observed mean activation $\hat{\rho}_j$ of each of the m_{h} hidden units, weighted by λ, which encourages compact codes; whether the active units correspond to physical degradation modes has to be established independently [116–119].

$$L_{SAE} = L_{AE} + \lambda \sum_{j=1}^{m_h} D_{KL}(\rho \parallel \hat{\rho}_j) \tag{38}$$

5.4. Indicator Selection

Selection reduces dimension and removes redundancy. Filter criteria score candidates independently of any downstream model. For N paired values x_i of a candidate indicator and y_i of the capacity, with sample means $\bar{x}$ and $\bar{y}$, the Pearson coefficient ρ_{XY} of Equation (39) measures linear association, and the Spearman rank coefficient r_s of Equation (40), in which d_i is the difference between the ranks of the two values in the ith pair, measures monotone association without a linearity assumption; the subscripted coefficient ρ_{XY} is unrelated to the activation rate ρ of Equation (38). The closed form of Equation (40) holds when there are no tied ranks; with ties, the coefficient is computed as the Pearson coefficient of Equation (39) applied to the average ranks.

$$\rho_{XY} = \frac{\sum_{i=1}^{N}(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^{N}(x_i - \bar{x})^2}\sqrt{\sum_{i=1}^{N}(y_i - \bar{y})^2}} \tag{39}$$

$$r_s = 1 - \frac{6\sum_{i=1}^{N} d_i^2}{N(N^2 - 1)} \tag{40}$$

Grey relational analysis, Equation (41), ranks each candidate sequence x_i by its grade $\gamma(x_0, x_i)$ of similarity to a reference sequence x_0, both of length N, in which $\Delta_i(k) = |x_0(k) - x_i(k)|$ is the absolute deviation of candidate i from the reference sequence at index k, $\Delta_{\min}$ and $\Delta_{\max}$ are the smallest and largest of these deviations over all candidates and indices, and $\zeta \in (0, 1]$, conventionally 0.5, is the distinguishing coefficient; it is the most widely used criterion in this literature [120]. In Equation (41), x_0 and x_i denote whole sequences, with values $x_0(k)$ and $x_i(k)$, rather than the single values x_i of Equation (39); the reference x_0 and the grade γ are unrelated to the initial state x_0 of Equation (13), to the rate weighting γ of Equation (11) and to the incomplete gamma function of Section 4.2.

$$\gamma(x_0, x_i) = \frac{1}{N}\sum_{k=1}^{N} \frac{\Delta_{min} + \zeta\Delta_{max}}{\Delta_i(k) + \zeta\Delta_{max}} \tag{41}$$

Mutual information, Equation (42), quantifies dependence of arbitrary form and therefore detects relations that correlation coefficients miss. In Equation (42), the sums run over the values x and y taken by the discrete random variables X and Y, $p(x, y)$ is their joint probability mass function, $p(x)$ and $p(y)$ are its marginals, and the logarithm is natural, so that $I(X; Y)$ is in nats.

$$I(X;Y) = \sum_{x \in X}\sum_{y \in Y} p(x,y) \log \frac{p(x,y)}{p(x)p(y)} \tag{42}$$

Wrapper criteria instead score subsets through the performance of a specific estimator, using recursive elimination, genetic search, or stepwise selection. They account for interaction between features and for the inductive bias of the estimator, at the cost of repeated training. The two paradigms differ in where the selection criterion is evaluated: filters score candidates before the estimator is trained, whereas wrappers score them through the trained estimator itself. Deep models bypass explicit selection by learning the representation jointly with the regression, which trades interpretability for convenience.

6. Statistical and Machine Learning Formulations for Diagnosis

Health estimation is a supervised regression problem: given a feature vector constructed as in Section 5, estimate the latent degradation state. Surveys of the estimator families in use place kernel machines, tree ensembles, and neural architectures as the three

dominant groups [121]. This section presents the principal estimator families through their optimization problems, since the differences that matter in deployment follow from the loss, the hypothesis class, and the regularization rather than from implementation detail.

6.1. Risk, Regularization, and the Bias-Variance Trade-Off

Every estimator considered here minimizes an empirical risk over a hypothesis class. For a fixed input x, let the observation y equal $f(x)$ plus a noise term of zero mean and variance ${\sigma_\varepsilon}^2$, independent of the training data; the expected squared error of a predictor $\hat{f}$ fitted on a random training set then decomposes into a squared bias, a variance, and an irreducible noise term, as in Equation (43), in which the expectation and the variance are taken over training sets and noise.

$$\mathbb{E}\left[\left(y-\hat{f}(x)\right)^2\right] = \left(\mathbb{E}\left[\hat{f}(x)\right]-f(x)\right)^2 + \mathbb{V}\left[\hat{f}(x)\right] + \sigma_\varepsilon^2 \tag{43}$$

This decomposition motivates balancing approximation bias against estimation variance when few independent cells are available, which is the usual situation in battery studies; it does not by itself establish that low-capacity estimators outperform deep models. The generalization gap admits the distribution-free bound of Equation (44), in which the expected loss, or risk, $R(\hat{f})$ of a predictor $\hat{f}$ chosen from the hypothesis class $\mathcal{F}$ is bounded by its average loss $\hat{R}_N(\hat{f})$ over the N training samples plus a complexity term and a confidence term.

$$R\left(\hat{f}\right) \le \hat{R}_N\left(\hat{f}\right) + 2\mathfrak{R}_N(\mathcal{F}) + 3\sqrt{\frac{\log(2/\Delta)}{2N}} \tag{44}$$

The bound of Equation (44) is not deterministic: for independent and identically distributed training samples and a loss bounded in the unit interval, it holds simultaneously for all $f \in \mathcal{F}$ with probability at least $1-\delta$ over the draw of the N training samples, where $\mathfrak{R}_N(\mathcal{F})$ denotes the empirical Rademacher complexity of the induced loss class. The inequality is the standard Rademacher complexity bound of statistical learning theory, stated for bounded losses as Theorem 3.3 of [122]. Here, N must count independent samples; cycles of the same cell are dependent and cannot simply be counted as such, so applying the bound to cycling data requires either one sample per cell or a dependence-aware extension. Because the complexity term grows with model capacity while the number of independent cells is limited by the cost of run-to-failure testing, the bound motivates capacity control in this application, but it does not by itself rank model families. Explicit regularization is applied through the penalized objective of Equation (45), in which y is the vector of the N training targets, X the matrix whose rows are the corresponding feature vectors, β the coefficient vector and $\lambda_1, \lambda_2 \ge 0$ the penalty weights; setting $\lambda_1 = 0$ gives ridge regression, setting $\lambda_2 = 0$ gives the least absolute shrinkage and selection operator (lasso), and keeping both positive gives the elastic net.

$$\hat{\beta} = \underset{\beta}{\operatorname{argmin}}\left\{ \| y - X\beta \|_2^2 + \lambda_2 \| \beta \|_2^2 + \lambda_1 \| \beta \|_1 \right\} \tag{45}$$

Before the individual estimator families are developed, it is useful to see where they sit within the wider set of health estimation strategies. Figure 6 divides those strategies into direct estimation from experimental measurement; indirect estimation through a physical, data-based, or hybrid model; and a residual group built on differential and entropy analysis. The formulations of this section occupy the data-based branch, and the physics-based descriptions of Section 4.3 occupy the branch beside it [11].

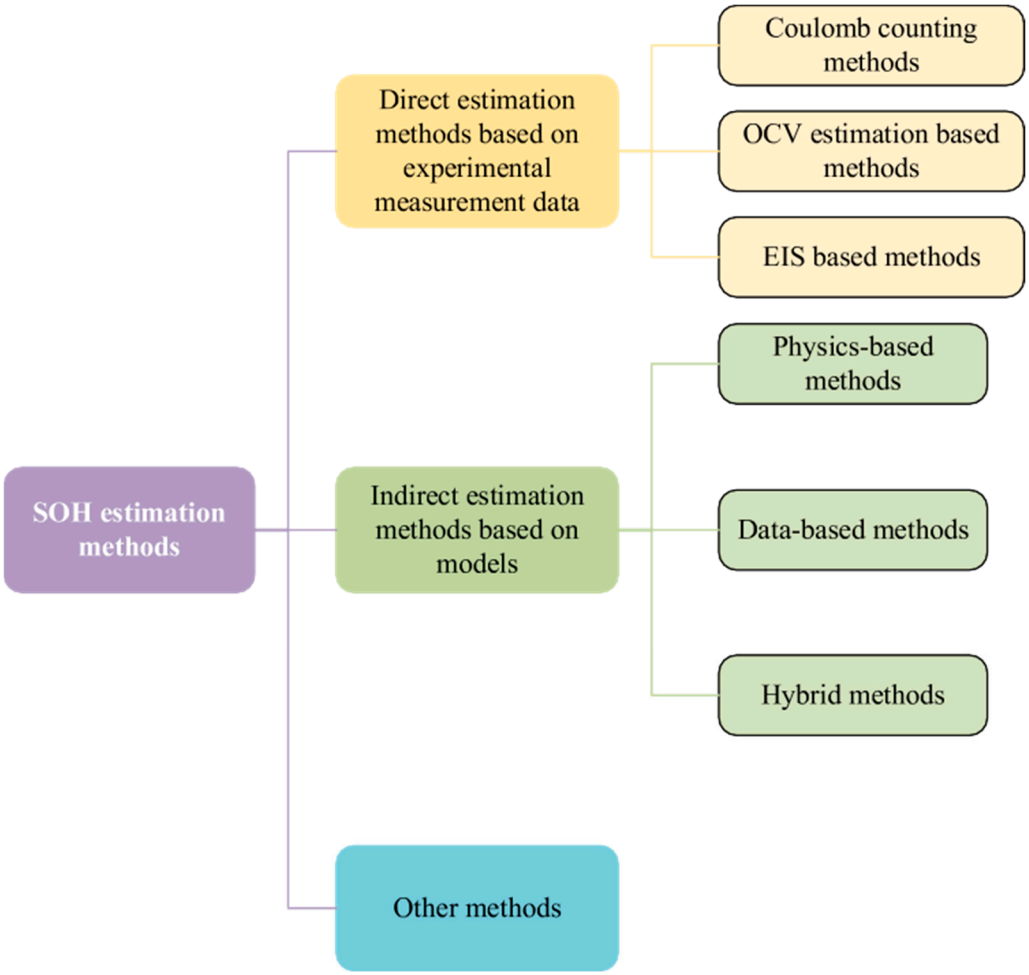


Figure 6. State-of-health estimation strategies, separating direct estimation from experimental measurement, indirect estimation through physical, data-based, and hybrid models, and a residual group of differential and entropy methods. Reproduced without modification under the terms of the Creative Commons Attribution (CC BY) license from [11].

6.2. Kernel Methods

Support vector regression (SVR) maps features through a feature map φ into a reproducing kernel Hilbert space in which the degradation relation is approximately linear [123]. The primal problem is stated in Equation (46), in which w and b are the weight vector and bias of the regression function in that space, and is minimized subject to the ε-insensitive constraints $y_i - \langle w, \varphi(x_i)\rangle - b \leq \varepsilon + \xi_i$, $\langle w, \varphi(x_i)\rangle + b - y_i \leq \varepsilon + \xi_i^*$, and $\xi_i, \xi_i^* \geq 0$ for $i = 1, \ldots, N$, which admit deviations of magnitude up to ε at no cost. Without these constraints, the objective is minimized trivially at the origin, so they are part of the problem rather than an aside.

$$\min_{w,b,\xi} \frac{1}{2} \| w \|^2 + C \sum_{i=1}^{N} (\xi_i + \xi_i^*) \tag{46}$$

with the slack variables ξ_i and ξ_i^* absorbing violations and the penalty parameter $C > 0$ setting the trade-off between flatness and training error. Solving the dual gives the fitted prediction function $\hat{f}$ of Equation (47), in which α_i and α_i^* are the Lagrange multipliers of the two sets of constraints, $k(x_i, x) = \langle \varphi(x_i), \varphi(x)\rangle$ is the kernel, and only the support vectors carry non-zero coefficients.

$$\hat{f}(x) = \sum_{i=1}^{N} (\alpha_i - \alpha_i^*) k(x_i, x) + b \tag{47}$$

The Gaussian radial basis kernel of Equation (48), with length scale $\ell > 0$, is the usual choice, since the degradation response is smooth and locally stationary.

$$k(x_i, x_j) = \exp\left(-\frac{\| x_i - x_j \|^2}{2\ell^2}\right) \tag{48}$$

Kernel machines generalize well from the small samples typical of aging studies, but they scale cubically in sample count in the worst case and they return no native uncertainty estimate. The least squares variant of Equation (49) substitutes a quadratic loss, weighted by the regularization constant $\lambda > 0$, under the equality constraints $y_i = \langle w, \varphi(x_i)\rangle + \mathrm{b} + e_i$, with e_i the residual of sample i; this converts the quadratic program into a linear system at the cost of sparsity, since every sample then becomes a support vector [21,22,124–126].

$$\min_{w,b,e} \frac{1}{2} \| w \|^2 + \frac{\lambda}{2} \sum_{i=1}^{N} e_i^2 \tag{49}$$

6.3. Ensembles of Trees

Random forests average the predictions $h_k(x; \Theta_k)$ of K trees, each grown on a bootstrap resample with randomly restricted feature subsets that are encoded by the random parameter Θ_k, Equation (50) [127].

$$\hat{y}_{RF}(x) = \frac{1}{K} \sum_{k=1}^{K} h_k(x; \Theta_k) \tag{50}$$

Bagging reduces variance without requiring feature scaling and supplies a natural importance ranking, which has made forests a common choice where interpretability of the input attribution matters more than raw accuracy [23,25–27].

Gradient boosting builds the ensemble sequentially, with each stage fitted to the pseudo-residual of the current model. The additive form after M stages is Equation (51), in which F_0 is the initial model, h_m is the base learner fitted at stage m and γ_m is its step size; the target of each new stage is the pseudo-residual r_{im} of Equation (52), the negative gradient of the loss $L(y_i, F(x_i))$ of sample i evaluated at the current model F_{m-1} [128].

$$F_M(x) = F_0(x) + \nu \sum_{m=1}^{M} \gamma_m h_m(x) \tag{51}$$

$$r_{im} = -\left[\frac{\partial L(y_i, F(x_i))}{\partial F(x_i)}\right]_{F=F_{m-1}} \tag{52}$$

with the shrinkage parameter $\nu \in (0, 1]$ regularizing each stage. Boosting attains high accuracy on tabular health indicators and handles missing entries robustly, at the cost of careful control of the learning rate, tree depth, and subsampling fraction to avoid overfitting [29–32].

The implementations in routine use differ in how they grow and regularize the trees rather than in the formulation of Equations (51) and (52). XGBoost adds an explicit penalty on leaf weights and tree complexity to the stage objective and has been applied to state-of-health estimation with a subsequent Markov correction of the regression output [129]. LightGBM grows trees leaf-wise rather than level-wise and bins features into histograms before searching for splits, which makes it the cheaper choice on large indicator sets; it has been used after a manifold-learning reduction of the indicator space [130]. CatBoost uses ordered boosting, in which the residual used to fit each stage is computed, for every observation, from a model trained only on the observations that precede it in a random

permutation; this is designed to remove the target leakage that otherwise arises within training when the same observations are used both to fit a stage and to evaluate its residuals, but it does not replace the separation of training and test data discussed in Section 10.4. The method has been applied to joint health estimation and life prediction from multi-domain features [106]. On tabular health indicators, these ensembles remain competitive with the deep architectures of Section 7 and are the baseline against which those architectures should be compared.

6.4. Gaussian Process Regression and Sparse Bayesian Models

Gaussian process regression (GPR) places a prior over functions specified by a mean function $m(x)$ and a covariance kernel $k(x, x')$, as shown in Equation (53) [131]. Conditioning on the vector y of training targets observed at the inputs X gives the posterior mean μ_* of Equation (54) and the posterior variance ${\sigma_*}^2$ of Equation (55) at a test input x_*.

$$f(x) \sim \mathcal{GP}(m(x), k(x, x')) \tag{53}$$

$$\mu_* = k_*^\top \left(K + \sigma_\varepsilon^2 I\right)^{-1} y \tag{54}$$

$$\sigma_*^2 = k(x_*, x_*) - k_*^\top \left(K + \sigma_\varepsilon^2 I\right)^{-1} k_* \tag{55}$$

Equation (55) gives the variance of the latent function; the predictive variance of a new noisy observation is obtained by adding ${\sigma_\varepsilon}^2$ to it. Here, K is the training covariance matrix, k_* is the vector of covariances between the test input and the training inputs, I is the identity matrix, and ${\sigma_\varepsilon}^2$ is the observation noise variance, which also regularizes the inversion and is the same quantity as in Equation (43). Equations (54) and (55) are written for the zero-mean convention $m(x) \equiv 0$; for a non-zero prior mean, the term $m(x_*)$ is added to Equation (54) and y is replaced by $y - m(X)$. The squared exponential kernel of Equation (56), which is the Gaussian kernel of Equation (48) scaled by the signal variance ${\sigma_\mathrm{f}}^2$, is a common default, with the length scale ℓ controlling the smoothness of the inferred function.

In Equation (56), x and x' are any two inputs and $\|\cdot\|$ is the Euclidean norm, as in Equation (48).

$$k(x, x') = \sigma_f^2 \exp\left(-\frac{\| x - x' \|^2}{2\ell^2}\right) \tag{56}$$

The posterior variance is the reason this family is favored for prognosis: it supplies a model-based predictive interval around the projected trajectory without a post hoc calibration step, although that interval is calibrated only if the kernel and noise model are correct, so its empirical coverage must still be checked against a held-out set using the prediction interval coverage probability (PICP) of Equation (89) reported jointly with interval width [48,132–134]. The cubic scaling in sample count nevertheless restricts direct application to fleet-scale corpora, and performance is dominated by kernel choice. The relevance vector machine (RVM) offers a sparse Bayesian alternative retaining probabilistic output with far fewer retained basis functions [135]. Where several plausible models $\mathcal{M}_1$, …, $\mathcal{M}_M$ are available, Bayesian model averaging of Equation (57) combines their predictive distributions for a new target y_* given the training data $\mathcal{D}$, weighted by the posterior model probabilities, which is more defensible than selecting a single winner on validation error.

$$p(y_* \mid \mathcal{D}) = \sum_{m=1}^{M} p(y_* \mid \mathcal{M}_m, \mathcal{D})\, p(\mathcal{M}_m \mid \mathcal{D}) \tag{57}$$

Table 7 compares the supervised families. The division between probabilistic estimators, which return a posterior, and deterministic estimators, which return a point value, is the distinction that matters most when the output feeds a maintenance decision.

Table 7. Comparison of supervised machine learning methods for battery SOH and RUL estimation, summarizing the operating principle, the associated equations, and the principal advantages and limitations of each family.

Method	Principle	Equations	Advantages	Limitations
Support vector regression	Kernel projection with an epsilon-insensitive loss	Equations (46)–(48)	Strong generalization from small samples; robust to outliers	Cubic scaling in sample count; no native uncertainty; kernel selection required
Least squares SVM	Quadratic loss reducing the problem to linear equations	Equation (49)	Faster training than standard SVR	Loses sparsity; every sample becomes a support vector
Random forest	Bagged ensemble of decision trees	Equation (50)	Low variance; no feature scaling; native importance ranking	Opaque aggregate decision path; high memory footprint
Gradient boosted trees	Sequential fitting to pseudo-residuals	Equations (51) and (52)	High accuracy on tabular health indicators; handles missing data	Prone to overfitting; sensitive to learning rate and depth
Gaussian process regression	Bayesian inference over functions	Equations (53)–(56)	Model-based predictive intervals requiring empirical coverage checks; effective on small datasets	Cubic scaling; kernel choice dominates performance
Relevance vector machine	Sparse Bayesian formulation	-	Probabilistic output with few basis functions; hyperparameters inferred during training	Local optima in training; sensitive to kernel parameters; held-out validation still required

6.5. Unsupervised and Semi-Supervised Formulations

Labels are frequently unavailable in the field, since a reference capacity measurement requires a controlled full discharge. Unsupervised models therefore carry substantial weight in practice. Autoencoders learn a compressed representation by minimizing reconstruction error. The encoder f_θ maps an input x to a latent code z and the decoder g_ϕ maps the code back to a reconstruction $\hat{x}$, Equation (58), where θ and ϕ are the encoder and decoder parameters; the objective, averaged over the N training inputs x_i, is Equation (59).

$$z = f_\theta(x), \qquad \hat{x} = g_\phi(z) \tag{58}$$

$$L_{AE} = \frac{1}{N}\sum_{i=1}^{N} \| x_i - g_\phi(f_\theta(x_i)) \|^2 \tag{59}$$

Sparse variants add a penalty encouraging compact codes, denoising variants train on deliberately corrupted inputs to improve robustness, and stacked variants fuse features across levels. The variational autoencoder treats the latent variable probabilistically, with an approximate posterior $q(z \mid x)$, a likelihood $p(x \mid z)$ and a prior $p(z)$, and is trained by minimizing the loss L_{VAE} of Equation (60), the negative of the evidence lower bound [136].

$$L_{VAE} = -\mathbb{E}_{q(z \mid x)}[\log p(x \mid z)] + D_{KL}(q(z \mid x) \,\|\, p(z)) \tag{60}$$

where the first term penalizes poor reconstruction and the Kullback–Leibler divergence D_{KL} regularizes the approximate posterior toward the prior. This construction has been embedded within particle filters to counter particle degeneracy in life prediction, which is discussed in Section 8.

Deep belief networks stack restricted Boltzmann machines and learn hierarchical features by layer-wise unsupervised pre-training [137]. The energy function of a restricted Boltzmann machine is Equation (61).

$$E(v,h) = -\sum_{i=1}^{m_v} a_i v_i - \sum_{j=1}^{m_h} b_j h_j - \sum_{i=1}^{m_v}\sum_{j=1}^{m_h} v_i w_{ij} h_j \tag{61}$$

with v and h representing the vectors of the m_{v} visible and m_{h} hidden units, a and b representing their biases, and w_{ij} representing the connecting weights. The generative structure supports adaptive feature extraction from raw signals, but the Gibbs sampling required for training imposes a cost that restricts online use.

Generative adversarial networks pit a generator G, which maps noise z drawn from a prior p_z to candidate samples, against a discriminator D, which returns the probability that a sample x was drawn from the distribution p_{data} of measured data, through the minimax objective of Equation (62), in which $V(G, D)$ is the value function [138].

$$\min_G \max_D V(G,D) = \mathbb{E}_{x\sim p_{data}}[\log D(x)] + \mathbb{E}_{z\sim p_z}[\log(1 - D(G(z)))] \tag{62}$$

Training terminates in principle when the discriminator performs no better than chance. Adversarial augmentation has become important where run-to-failure data is scarce, but convergence to the equilibrium is unreliable and mode collapse remains a standing risk. The unsupervised family is summarized in Table 8.

Table 8. Summary of unsupervised and semi-supervised learning methods applied in battery diagnostics and prognostics, contrasting their roles in representation learning, data augmentation, and the handling of unlabeled data.

Method	Role in Diagnosis	Equations	Characteristic Limitation
Autoencoder	Dimensionality reduction and unsupervised feature extraction	Equations (58) and (59)	Compressed data must follow the training distribution
Sparse autoencoder	Extraction of compact representations	Equation (38)	Requires tuning of the sparsity target
Denoising autoencoder	Robust representation from corrupted inputs	-	Corruption level must match the observed noise
Variational autoencoder	Probabilistic latent representation for uncertainty-aware estimation	Equation (60)	Reconstruction quality traded against latent regularity
Deep belief network	Layer-wise generative pre-training of hierarchical features	Equation (61)	High training cost; unsuited to online execution
Generative adversarial network	Synthetic augmentation of sparse degradation datasets	Equation (62)	Unstable convergence; risk of mode collapse
Semi-supervised learning	Training from partially labeled operational records	-	Pseudo-label error propagates through training
Weakly supervised learning	Training under coarse or approximate capacity labels	-	Label granularity bounds achievable accuracy

7. Deep Architectures for Prognosis

Deep models compose many nonlinear transformations and learn the representation jointly with the regression. The forward computation of a fully connected layer is Equation (63), in which a^l is the output of layer l (a^0 being the input feature vector), W^l and b^l are the weight matrix and bias vector of that layer, and σ is the logistic sigmoid of Equation (64), applied elementwise; the rectified linear unit of the same equation is the usual alternative in deep layers. Gradient descent on the training loss L, with learning rate η, updates the network parameters θ at iteration k as in Equation (65).

$$a^l = \sigma(W^l a^{l-1} + b^l) \tag{63}$$

$$\mathrm{ReLU}(z) = \max(0, z), \qquad \sigma(z) = \frac{1}{1 + e^{-z}} \tag{64}$$

$$\theta_{k+1} = \theta_k - \eta \, \nabla_\theta L(\theta_k) \tag{65}$$

Depth allows complex fade trajectories to be represented, while generalization also depends on the hypothesis class, regularization, optimization and the number of independent cells available. Dropout regularizes training by stochastically deactivating units [139] and batch normalization stabilizes optimization by standardizing layer inputs [140]. The extreme learning machine assigns input weights at random and solves only the output layer, through the pseudo-inverse of Equation (66) [141].

$$\beta = H^{\dagger} Y = (H^{\top} H)^{-1} H^{\top} Y \tag{66}$$

with H standing for the hidden-layer output matrix; $H^{\dagger}$, its Moore–Penrose pseudo-inverse; Y, the matrix of training targets; and β, the output-layer weights; the learning rate η of Equation (65) is unrelated to the coulombic efficiency of Equation (5). The second equality in Equation (66) is the pseudo-inverse specialized to the case in which H has full column rank; otherwise, the pseudo-inverse is taken directly. Training is very fast, at the cost of sensitivity to the random draw and a tendency to fit noise in the inputs.

7.1. Convolutional Models

Convolutional networks extract local structure from raw waveforms through the discrete filtering of Equation (67), in which ${a_i}^{l-1}$ is the input to layer l at position i, that is, the activation of layer $l - 1$ or its pooled value (a^0 being the input sequence), the sum runs over the positions j of the kernel support, ${w_j}^l$ are the kernel weights shared across positions and b^l is the bias. An elementwise activation of ${z_i}^l$, typically the rectified linear unit of Equation (64), gives the activations ${a_i}^l$, and the max pooling of Equation (68) returns their maximum ${v_i}^l$ over the window R_i of positions pooled into output i. Strictly, the summation over the kernel index j in Equation (67) is a cross-correlation, the operation that neural-network implementations conventionally call convolution. For two-dimensional inputs, such as the image transformations cited below, both operations carry a second position index.

$$z_i^l = \sum_j a_{i+j}^{l-1} w_j^l + b^l \tag{67}$$

$$v_i^l = \max_{i' \in R_i} a_{i'}^l \tag{68}$$

Weight sharing across the input reduces parameter count substantially relative to a fully connected network, which is precisely the capacity control the sample-size argument of Section 6.1 recommends. Pooling improves efficiency and translation tolerance, but it can discard information relevant to subtle transitions. Applications span capacity estimation

from partial charging segments, impedance-derived features, and image transformations of cycling data [34–39].

7.2. Recurrent Models

Recurrent networks maintain a hidden state h_t accumulating information across the steps t of an input sequence x_t, Equation (69), which matches the conditional structure of cycling history; W_{xh} and W_{hh} are the input and recurrent weight matrices, b_h is the bias and σ is again the logistic sigmoid, for which the hyperbolic tangent is a common alternative. In Equations (69)–(71), t is a discrete step index rather than continuous time.

$$h_t = \sigma(W_{xh}x_t + W_{hh}h_{t-1} + b_h) \tag{69}$$

Gradients vanish or explode over long sequences, limiting the range of dependence that can be captured. Long short-term memory (LSTM) networks resolve this with input, forget, and output gates regulating a memory cell, Equation (70) [142].

$$\begin{aligned} i_t &= \sigma(W_{xi}x_t + W_{hi}h_{t-1} + b_i) \\ f_t &= \sigma\left(W_{xf}x_t + W_{hf}h_{t-1} + b_f\right) \\ o_t &= \sigma(W_{xo}x_t + W_{ho}h_{t-1} + b_o) \\ c_t &= f_t \odot c_{t-1} + i_t \odot \tanh(W_{xc}x_t + W_{hc}h_{t-1} + b_c) \\ h_t &= o_t \odot \tanh(c_t) \end{aligned} \tag{70}$$

with the sigmoid input, forget and output gates i_t, f_t and o_t and the elementwise product $\odot$ controlling the flow of information into and out of the memory-cell state c_t; the weight matrices W and biases b carry the subscripts of the connections and gates they serve. This construction retains degradation signal across many cycles while discarding noise, and it has become the most widely applied architecture for multi-step capacity prediction [40,41,143–146]. Bidirectional variants process the sequence in both directions, which is admissible whenever the degradation record is fully observed offline.

Gated recurrent units (GRUs) reduce the gating to a reset gate r_t and an update gate z_t, Equation (71) [147].

$$\begin{aligned} r_t &= \sigma(W_r \cdot [h_{t-1}, x_t]) \\ z_t &= \sigma(W_z \cdot [h_{t-1}, x_t]) \\ \widetilde{h}_t &= \tanh(W \cdot [r_t \odot h_{t-1}, x_t]) \\ h_t &= (1 - z_t) \odot h_{t-1} + z_t \odot \widetilde{h}_t \end{aligned} \tag{71}$$

Here, $[h_{t-1}, x_t]$ denotes the concatenation of the previous hidden state and the current input; W_r, W_z and W are the weight matrices of the reset gate, the update gate and the candidate state $\tilde{h}_t$; and bias terms are omitted for brevity. The update gate interpolates between the previous hidden state and the candidate state. Fewer parameters shorten training and lower memory demand, which suits smaller corpora, but the simplification limits expressive capacity where degradation behavior is strongly heterogeneous [148–152].

7.3. Attention and Hybrid Architectures

Hybrids that place convolutional layers before recurrent layers combine local feature extraction with temporal modeling; any improvement over either component alone has to be established under matched data and evaluation protocols [153–156]. Figure 7 shows the complete pipeline such a model occupies, from preprocessing of the raw cycling record through the convolutional and recurrent stages to the health estimate and its evaluation [71].

Attention weights the contribution of each position in forming a prediction, through the scaled dot-product of Equation (72) [157].

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^{\top}}{\sqrt{d_k}}\right)V \tag{72}$$

with query, key, and value matrices Q, K, and V formed by linear projection of the input sequence; d_{k} representing the dimension of the key vectors; the softmax applied row by row; and the scaling factor $\sqrt{d_{\text{k}}}$ stabilizing the softmax gradient. These three matrices are local to Equation (72) and are unrelated to the quantities denoted by the same letters elsewhere in the review. Transformer models built on this operation relate arbitrary positions without recurrence and parallelize efficiently, which has made them an established alternative to recurrent networks for long degradation sequences. In the battery setting, this advantage is subject to three qualifications: the cost of attention grows quadratically with sequence length, which matters for long cycling records and for onboard execution; the data demand is high relative to the run-to-failure cohorts of Table 3, a constraint that the transformer-based framework discussed in Section 7.4 addresses by self-supervised pre-training on unlabeled records; and the chronological and cell-wise split controls of Sections 10.3 and 10.4 apply to these models as to any other sequence model. Table 9 compares the properties of these architectures.

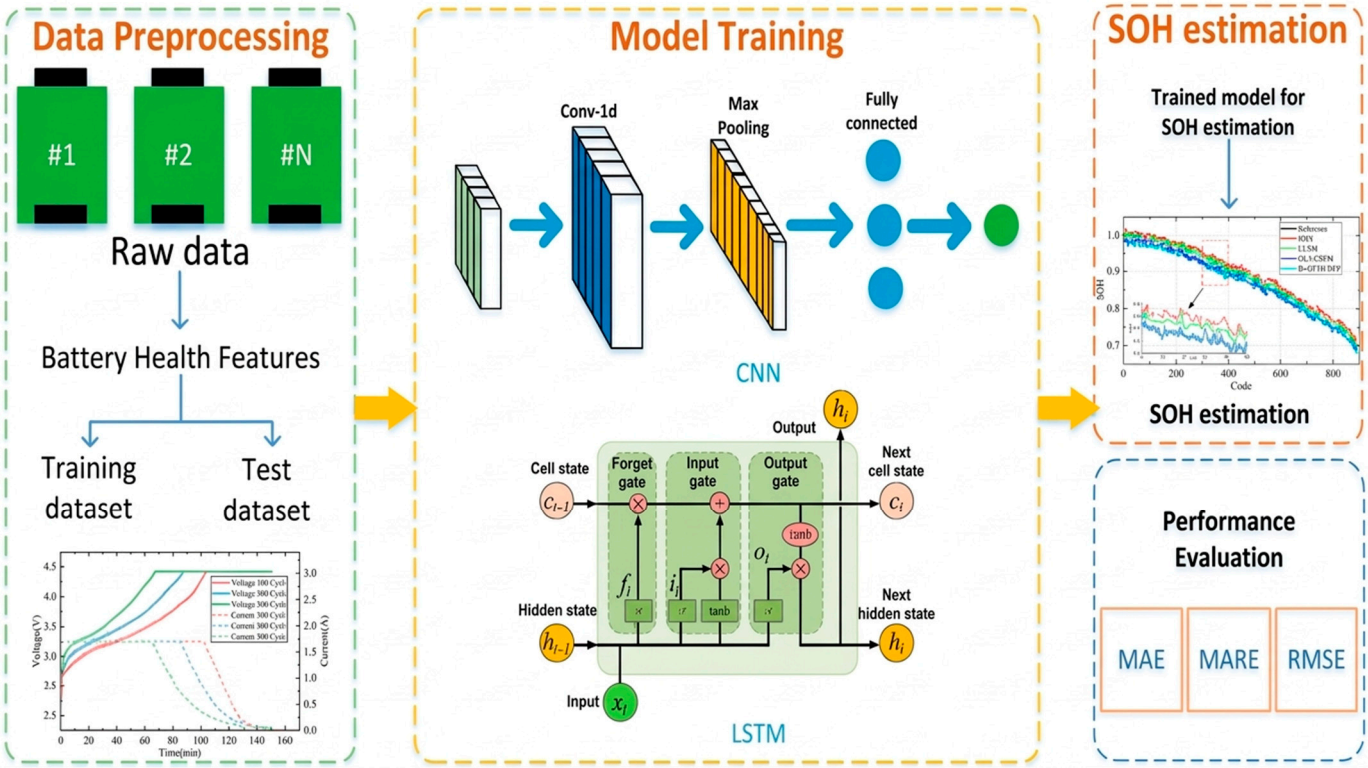


Figure 7. End-to-end pipeline of a convolutional and recurrent hybrid estimator, comprising preprocessing of the raw cycling record, feature extraction by convolutional layers, temporal modeling by recurrent layers, and evaluation of the resulting health estimate. Reproduced without modification under the terms of the Creative Commons Attribution (CC BY) license from [71].

Table 9. Comparison of deep learning architectures applied to battery diagnostics and prognostics, with the associated formulation, the principal strength, and the corresponding limitation of each architecture.

Architecture	Governing Equations	Principal Strength	Limitation
Deep neural network	Equations (63)–(65)	Models complex nonlinear degradation trajectories	Parameter count grows rapidly; prone to overfitting on small samples
Extreme learning machine	Equation (66)	Very fast training; closed-form output weights	Sensitive to random initialization; overfits noisy features
Convolutional neural network	Equations (67) and (68)	Weight sharing reduces parameters; captures local waveform structure	Pooling discards information; weak on long-range dependence
Recurrent neural network	Equation (69)	Natural treatment of sequential cycling history	Vanishing and exploding gradients on long sequences
Long short-term memory	Equation (70)	Retains degradation signal over many cycles	Computationally heavy; sensitive to initialization
Gated recurrent unit	Equation (71)	Fewer parameters and faster training than LSTM	Reduced capacity on highly heterogeneous degradation
Attention and transformer	Equation (72)	Models dependence between arbitrary positions; parallelizable	Large data requirement; quadratic cost in sequence length
Convolutional-recurrent hybrid	Equations (67)–(71)	Combines spatial extraction with temporal modeling	Increased architectural complexity and tuning burden

7.4. Graph-Based and Self-Supervised Formulations

Two families that have entered the battery literature more recently do not fit the convolutional, recurrent or attention categories above, and they are treated here because each addresses a limitation of those categories rather than offering an alternative architecture for the same problem.

Graph formulations replace the fixed grid or sequence over which convolution and recurrence operate with an explicit graph. A layer aggregates over the neighbors of each node and then applies a learned transformation, so that the operator of Equation (67) becomes a sum over an adjacency structure rather than over a fixed kernel support. Two constructions are relevant here. In the first, the nodes are features, measurement channels or segments of the cycling record, and the edges encode which of them are related, for example through mutual information, which allows dependencies that are neither local in time nor uniform across channels to be represented directly; this construction has been used for state-of-health estimation with a graph perceptual neural network whose nodes are health indicators extracted from each cycle and whose edges are retained where their mutual information exceeds a threshold [158]. In the second, the nodes are the cells of a module or pack and the edges are the electrical and thermal couplings between them, which makes the formulation a natural candidate for the pack-level problem discussed in Sections 9.2 and 11.1, where cell-to-cell interaction rather than single-cell degradation is the object of estimation; as Table 4 records, this construction cannot at present be benchmarked on public pack-level aging data. The cost is that the adjacency structure is itself a modeling choice: where it is inferred from the data it becomes an additional estimand, and where it is imposed it carries an assumption that should be stated.

Self-supervised formulations address the labeling constraint of Section 11.1 rather than the representation. The model is first trained on a task constructed from the unlabeled cycling record itself, by masking part of a charge curve and reconstructing it, by predicting a later segment from an earlier one, or by requiring that two augmented views of the same cycle map to nearby representations and views of different cycles to distant ones. The

learned representation is then fine-tuned on the small labeled subset for which reference capacity measurements exist. The attraction in this domain is specific: unlabeled cycling records are comparatively plentiful while reference capacity measurements are expensive and rare, so the pretext task uses exactly the data that the supervised formulations of Section 6 cannot use. Frameworks of this kind have been reported for state-of-health estimation under label scarcity, among them one that pre-trains a transformer encoder on a large unannotated database, incorporates weak labels, and transfers to other cell types after fine-tuning on a small annotated set [159]. Two cautions apply. The pretext task encodes an assumption about which transformations leave health unchanged, and that assumption is rarely tested; and because pre-training consumes the unlabeled portion of the same cells that supply the labeled portion, the split discipline of Section 10.4 must be applied to the pre-training stage as well, or the reported gain will include leakage.

7.5. Comparison of Shallow and Deep Formulations

The choice between shallow and deep estimators is governed by sample size, inference budget, and interpretability requirement. Bibliometric surveys of the field record recurrent architectures as the most frequently applied family, followed by convolutional and kernel methods [160,161]. Controlled comparisons on a common capacity-fade corpus report narrower margins between families than the individual studies suggest, which indicates that dataset and protocol choices account for much of the reported spread [162]. Deep architectures tend to be favored when labeled data is abundant, since they can learn representations that hand-crafted indicators do not capture, whereas shallow estimators are often competitive when few independent cells are available; Section 6.1 explains why capacity control matters in that regime, but no general ranking of the families follows from it. Inference cost depends on the architecture, input size, implementation and hardware, and it should be measured on the target embedded controller where the estimator must execute alongside the control loop. Interpretability remains the sharpest distinction, since the internal logic of a deep network resists mapping onto degradation mechanisms, and this is a binding constraint where regulatory acceptance is required [44]. Table 10 summarizes the comparison.

Table 10. Comparative assessment of traditional machine learning and deep learning for battery diagnostics and prognostics across the dimensions that determine suitability for a given deployment context.

Dimension	Traditional Machine Learning	Deep Learning
Accuracy with abundant data	Moderate; bounded by feature quality	High; learns representations directly from raw signals
Accuracy with scarce data	Often competitive	Prone to overfitting unless strongly regularized or pre-trained
Feature engineering	Required and decisive for performance	Largely automatic
Training cost	Low to moderate	High; scales with depth and dataset size
Inference cost	Usually low; suitable for onboard execution	Depends on architecture and hardware; compression is often needed for embedded use
Interpretability	Moderate for trees and kernels	Low; requires post hoc attribution
Uncertainty quantification	Native for GPR and RVM	Requires ensembling or Bayesian treatment
Robustness to distribution shift	Moderate	Low without domain adaptation

8. Sequential State Estimation and Uncertainty Quantification

Prognosis requires a predictive distribution rather than a point estimate, since a maintenance decision depends on the probability of threshold crossing within a planning horizon. Sequential Bayesian estimation supplies that distribution and simultaneously corrects the drift that afflicts the open-loop integration of Equation (5). This section develops the recursions.

8.1. The State-Space Formulation

Write the degradation and measurement processes in the discrete state-space form of Equation (73), in which k is the step index, counting samples within a cycle when the state is SOC, as in Equation (4), and cycles, in the role of the cycle index n of Section 3, when the state is a degradation quantity.

$$\begin{aligned} x_k &= f_{k-1}(x_{k-1}) + w_{k-1}, \qquad w_{k-1} \sim \mathcal{N}(0, Q_{k-1}) \\ z_k &= h_k(x_k) + v_k, \qquad v_k \sim \mathcal{N}(0, R_k) \end{aligned} \tag{73}$$

with transition map f_{k-1}, observation map h_k, process noise w_{k-1} and measurement noise v_k, the two noises being mutually independent and Gaussian with covariances Q_{k-1} and R_k. In this section w, v, Q and R denote only these noises and covariances, and not the failure threshold, the charge or the resistances of Sections 3 and 4. The estimation objective is the filtering density of the state given all observations to the present index. Where the transition and observation maps are linear, the initial state is Gaussian, and the process and measurement noises are white, Gaussian and independent of each other and of the initial state, this density is Gaussian and is propagated exactly by the Kalman recursions.

8.2. Kalman Filtering and Its Nonlinear Extensions

The prediction step propagates the mean and covariance through the transition model, Equation (74), in which $\hat{x}_{k|k-1}$ and $P_{k|k-1}$ are the mean and covariance of the state at step k given the measurements up to step $k-1$, and F_{k-1} is the transition matrix, with $f_{k-1}(x_{k-1}) = F_{k-1}x_{k-1}$ in the linear case.

$$\hat{x}_{k|k-1} = F_{k-1}\hat{x}_{k-1|k-1}, \qquad P_{k|k-1} = F_{k-1}P_{k-1|k-1}F_{k-1}^{\top} + Q_{k-1} \tag{74}$$

The gain K_k that minimizes the posterior error covariance is given in Equation (75), where H_k is the observation matrix, with $h_k(x_k) = H_k x_k$ in the linear case, and the correction applied on receipt of the measurement z_k is Equation (76), in which I is the identity matrix and $\hat{x}_{k|k}$ and $P_{k|k}$ are the corrected mean and covariance.

$$K_k = P_{k|k-1}H_k^{\top}\left(H_k P_{k|k-1}H_k^{\top} + R_k\right)^{-1} \tag{75}$$

$$\hat{x}_{k|k} = \hat{x}_{k|k-1} + K_k\left(z_k - H_k\hat{x}_{k|k-1}\right), \qquad P_{k|k} = (I - K_k H_k)P_{k|k-1} \tag{76}$$

Battery models are not linear, so the extended filter departs from Equations (74) and (76) in its mean updates: the predicted mean is obtained by evaluating the nonlinear transition map of Equation (73) at the previous posterior mean, and the innovation is the measurement minus the nonlinear observation map evaluated at the predicted mean. The Jacobians of Equation (77), which linearize these maps about the current estimate, are used only to propagate the covariance in Equation (74), to compute the gain in Equation (75) and to update the covariance in Equation (76).

$$F_{k-1} = \left.\frac{\partial f_{k-1}}{\partial x}\right|_{\hat{x}_{k-1|k-1}}, \qquad H_k = \left.\frac{\partial h_k}{\partial x}\right|_{\hat{x}_{k|k-1}} \tag{77}$$

This is adequate where the nonlinearity is mild over the region spanned by the posterior, but it degrades near the extremes of the open-circuit-voltage curve, where the map from SOC to voltage is strongly nonlinear, and the state becomes poorly observable where the curve is nearly flat, as over the central plateau of lithium iron phosphate, because the Jacobian there approaches zero (Section 3.2). Unscented and cubature variants avoid explicit differentiation by propagating deterministically chosen sigma points, which improves accuracy at comparable cost. Joint and dual formulations that estimate state and parameters simultaneously are the standard route to co-estimation of SOC and SOH [63]. The progression of these variants, together with the strengths and limitations that motivate each successive refinement, is set out in Figure 8 [65].

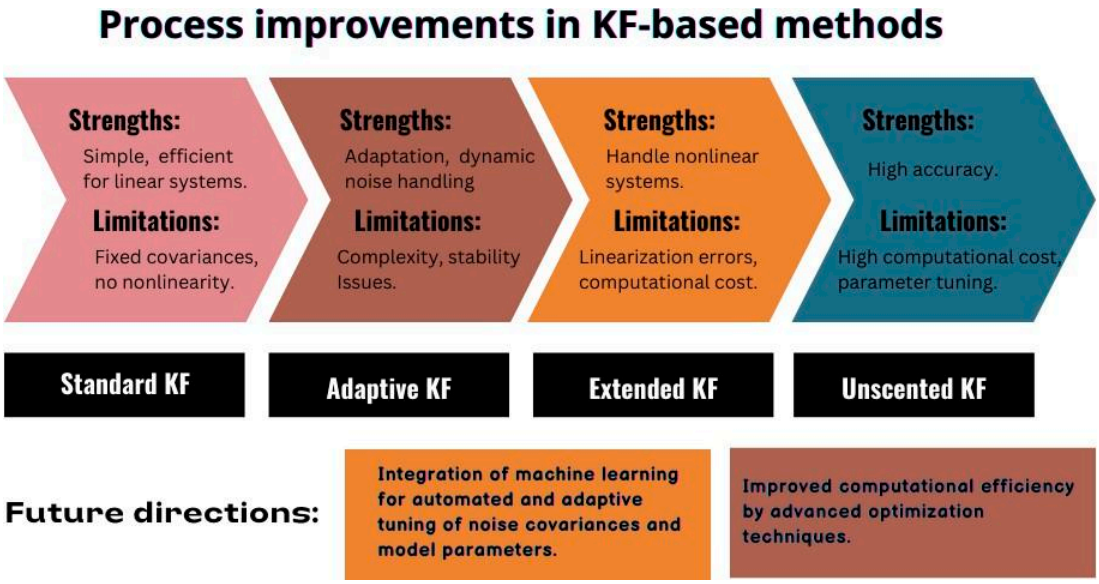


Figure 8. Progression of Kalman filter variants from the standard linear formulation through adaptive, extended, and unscented forms, with the strengths and limitations that motivate each refinement. Reproduced without modification under the terms of the Creative Commons Attribution (CC BY) license from [65].

8.3. Particle Filtering

Where the posterior is multimodal or strongly non-Gaussian, sequential Monte Carlo is a natural choice; capacity regeneration alone does not establish either property, and the appropriate inference method depends on the transition, observation and noise models. The filtering density, given the measurements $z_{1:k} = (z_1, \ldots, z_k)$, is represented by N_p particles $x_k^{(i)}$ with normalized weights $\omega_k^{(i)}$, as shown in Equation (78), in which $\delta(\cdot)$ is the Dirac delta.

$$p(x_k \mid z_{1:k}) \approx \sum_{i=1}^{N_p} \omega_k^{(i)} \, \Delta\left(x_k - x_k^{(i)}\right) \tag{78}$$

with weights updated by importance sampling from a proposal density q, as shown in Equation (79).

$$\omega_k^{(i)} \propto \omega_{k-1}^{(i)} \frac{p\left(z_k \mid x_k^{(i)}\right) p\left(x_k^{(i)} \mid x_{k-1}^{(i)}\right)}{q\left(x_k^{(i)} \mid x_{k-1}^{(i)}, z_k\right)} \tag{79}$$

Weight concentration on a shrinking subset is the characteristic failure mode, diagnosed by the effective sample size of Equation (80), written for weights normalized to sum to unity, and treated by resampling when that quantity falls below a preset fraction of the particle count N_p. Degeneracy and the sample impoverishment that resampling introduces

in its place are the two failure modes the literature addresses, and Figure 9 summarizes the strategies proposed against them [65].

$$N_{eff} = \frac{1}{\sum_{i=1}^{N_p} (\omega_k^{(i)})^2} \tag{80}$$

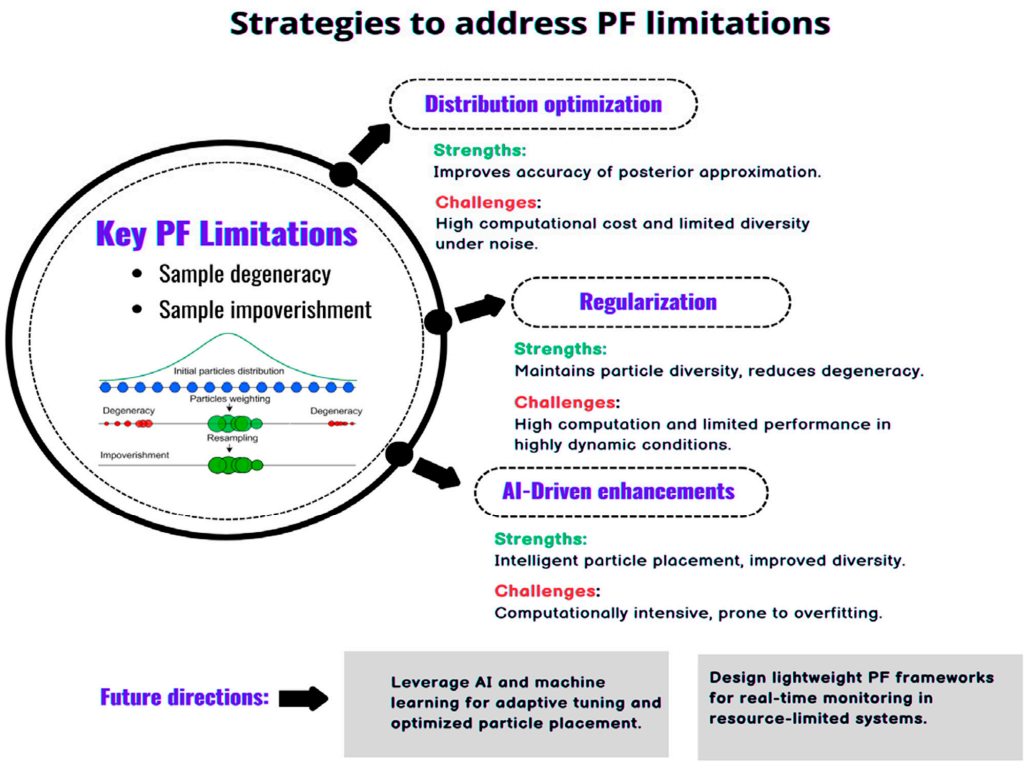


Figure 9. Failure modes of the particle filter and the strategies proposed against them, covering distribution optimization, regularization, and learning-based enhancements. Reproduced without modification under the terms of the Creative Commons Attribution (CC BY) license from [65].

Particle filters are widely combined with the empirical degradation laws of Section 4.1, the physical model supplying the transition and the filter adapting its parameters online, which yields life predictions that remain physically admissible while tracking the individual cell [73,74].

8.4. Bayesian Inference and Uncertainty Calibration

All of the above are instances of the posterior update of Equation (81), in which θ denotes the unknown parameters or states and $\mathcal{D}$ the observed data.

$$p(\theta \mid \mathcal{D}) = \frac{p(\mathcal{D} \mid \theta)\, p(\theta)}{\int p(\mathcal{D} \mid \theta')\, p(\theta')\, d\theta'} \tag{81}$$

Sources of predictive uncertainty separate into aleatoric components arising from measurement noise and intrinsic path variability and into epistemic components arising from limited data and model misspecification. The distinction matters operationally, since epistemic uncertainty is reducible by further data collection while aleatoric uncertainty is not. Deterministic deep models return no uncertainty natively and require ensembling, Monte Carlo dropout, or an explicit Bayesian treatment to produce intervals, and those intervals require calibration checking rather than assumption [48,163–166].

Life prediction under this framework proceeds by propagating the filtered state forward under an assumed future load and recording the first threshold crossing, which

recovers the first-passage time distribution of Section 4.2 as the special case in which the transition model is a Wiener process [167–170]. Where no transition model is available, similarity-based schemes instead match the observed degradation trajectory against a library of run-to-failure histories and read the residual life from the closest matches [171].

9. Fault Diagnosis and Safety Prognosis

Health estimation addresses gradual degradation. Fault diagnosis addresses abrupt departure from expected behavior, and its safety consequences are considerably more severe. Treatments confined to SOH and RUL therefore describe only part of what a deployed system must deliver [11].

9.1. Fault Taxonomy

Faults divide into three classes. External faults originate outside the cell and comprise sensor failures in voltage, current, and temperature channels; thermal management faults; communication and relay faults; and connection faults such as contact oxidation. Internal faults comprise manufacturing defects, overcharge and over-discharge, overheating, internal and micro short circuits, and accelerated degradation. Thermal runaway forms the third class and is triggered by mechanical abuse such as penetration, electrical abuse such as overcharge, or thermal abuse, proceeding through interphase decomposition, separator failure, and successive exothermic reactions at the anode, cathode, and electrolyte [9,172–174]. Figure 10 sets out the taxonomy together with the abuse conditions that trigger the third class.

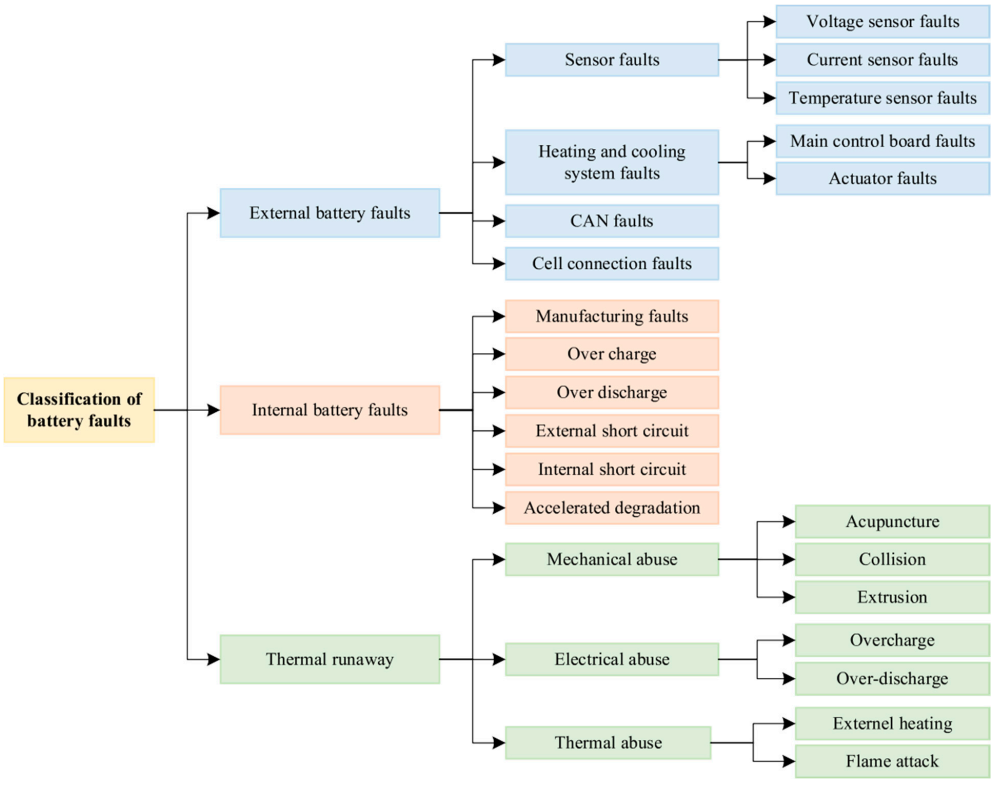


Figure 10. Fault taxonomy for battery systems, distinguishing external faults, internal faults, and thermal runaway, with the mechanical, electrical, and thermal abuse conditions that trigger the third class. Reproduced without modification under the terms of the Creative Commons Attribution (CC BY) license from [11].

9.2. Diagnosis Methods

Model-based diagnosis uses the descriptions of Section 4.3 with state estimation and parameter identification to observe quantities that are not directly measurable, interpreting departures of the identified parameters from their expected trajectories as faults. Experiment-based methods characterize fault signatures directly through abuse testing, which supplies labeled data for conditions that cannot be sampled from normal operation [175–177].

Data-driven methods detect abnormal states without an explicit mechanistic model. Threshold and rule schemes remain common in production because they are transparent and auditable. Statistical schemes exploit inconsistency between cells within a pack, using entropy measures, sliding-window correlation, and modified Z-scores to isolate the deviating cell. Learning-based schemes infer fault signatures directly, and recent work has demonstrated detection that operates reliably on field data from operating vehicles rather than on laboratory abuse tests alone [178–181]. Figure 11 summarizes the available approaches. Read alongside the estimation methods of Figure 6, they divide on the same axis: direct measurement, indirect inference through a physical or learned model, and hybrid combinations of the two.

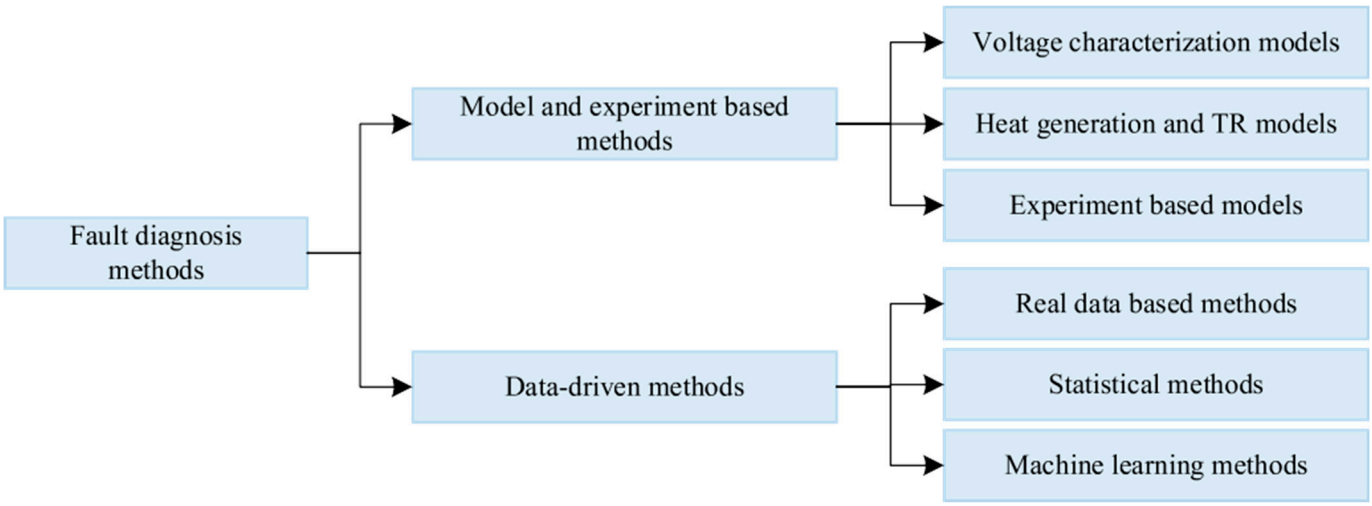


Figure 11. Diagnosis and prediction methods for battery faults, divided into model- and experiment-based approaches on one branch and data-driven approaches on the other. Reproduced without modification under the terms of the Creative Commons Attribution (CC BY) license from [11].

9.3. Early Warning for Thermal Runaway

Warning is more valuable than detection, since the interval between the onset of an internal short circuit and venting can be very short. Warning schemes exploit precursors including abnormal voltage drop, localized temperature rise, impedance change, gas evolution, and cell expansion force [173,174,176,180]. Accuracy can be expected to improve when electrochemical and thermal constraints are embedded in the learning model rather than left implicit, consistent with the evidence on physics-informed learning discussed in Section 11. The barriers to deployment are the cost of the additional sensing that the most accurate schemes require, and the scarcity of labeled failure events, since runaway is rare in a healthy fleet and cannot be induced at scale.

10. Evaluation, Benchmarking, and Reproducibility

10.1. Point-Error Metrics

Comparison across studies requires a common evaluation basis. Throughout this Section 10, $Y(i)$ and $\hat{Y}(i)$ are the reference and predicted values of the target, whether SOH, SOC or RUL, for the ith of the N_s samples of the test set. The absolute error of a single prediction is Equation (82) and the mean absolute error (MAE) over the test set is Equation (83).

$$AE_i = |Y(i) - \hat{Y}(i)| \tag{82}$$

$$MAE = \frac{1}{N_s}\sum_{i=1}^{N_s}|Y(i) - \hat{Y}(i)| \tag{83}$$

The mean squared error of Equation (84) and its root, the root mean square error (RMSE) of Equation (85), weight large deviations more heavily, which makes them sensitive to isolated regeneration events.

$$MSE = \frac{1}{N_s}\sum_{i=1}^{N_s}(Y(i) - \hat{Y}(i))^2 \tag{84}$$

$$RMSE = \sqrt{\frac{1}{N_s}\sum_{i=1}^{N_s}(Y(i) - \hat{Y}(i))^2} \tag{85}$$

The mean absolute percentage error of Equation (86) is defined only when every included target value is nonzero and becomes unstable as the target approaches zero. It should therefore not be the sole metric for RUL trajectories approaching end of life; MAE and RMSE should also be reported. Any exclusion of zero targets must be declared together with the resulting sample count. The maximum absolute error of Equation (87) reports the largest observed error on the evaluated test set, rather than a guaranteed deployment safety bound.

$$MAPE = \frac{1}{N_s}\sum_{i=1}^{N_s}\left|\frac{Y(i) - \hat{Y}(i)}{Y(i)}\right| \times 100\% \tag{86}$$

$$MaxAE = \max_i |Y(i) - \hat{Y}(i)| \tag{87}$$

The coefficient of determination of Equation (88), in which $\bar{Y}$ is the mean of the reference values over the test set, expresses variance explained relative to a mean predictor.

$$R^2 = 1 - \frac{\sum_{i=1}^{N_s}(Y(i) - \hat{Y}(i))^2}{\sum_{i=1}^{N_s}(Y(i) - \bar{Y})^2} \tag{88}$$

10.2. Assessing the Quality of Uncertainty Estimates

Point-error metrics describe the location of the predictive distribution and say nothing about its spread. For an estimator that returns a distribution or an interval, predictive accuracy and the quality of the uncertainty statement are separate properties, and a model may be strong in one and weak in the other: a predictor whose intervals are wide enough to contain almost every outcome may meet or exceed the nominal coverage while conveying very little, and a predictor with small mean absolute error may be systematically overconfident. The two must therefore be reported together, and neither substituted for the other. The appropriate decomposition is into calibration, the agreement between stated and realized frequencies, and sharpness, the concentration of the predictive distribution; the governing principle is to maximize sharpness subject to calibration [182].

At a stated nominal coverage $1 - \alpha$, the prediction interval coverage probability in Equation (89) estimates empirical marginal coverage on the evaluated test population, with $[L_i, U_i]$ the prediction interval for sample I and the indicator function equal to one when $Y(i)$ lies in that interval and to zero otherwise. It should be reported with the nominal target, interval width, test-set size and sampling uncertainty; coverage close to the nominal target alone does not establish conditional calibration. Cell-wise or other appropriate grouped evaluation is needed when observations from the same cell are dependent.

$$PICP = \frac{1}{N_s}\sum_{i=1}^{N_s} 1\{Y(i) \in [L_i, U_i]\} \tag{89}$$

Coverage alone is insufficient, for two reasons that are frequently conflated. The first is that coverage says nothing about width. The mean prediction interval width (MPIW) of Equation (90) summarizes sharpness at the stated nominal level, and must be reported alongside the coverage it accompanies, in the units of the target.

$$\mathrm{MPIW} = \frac{1}{N_s}\sum_{i=1}^{N_s}(U_i - L_i) \tag{90}$$

The second is that coverage at a single nominal level says nothing about the remainder of the predictive distribution. Evaluating the empirical coverage $\hat{c}(\tau_\mathrm{m})$ of the central interval at each of M nominal levels $\tau_1, \ldots, \tau_\mathrm{M}$ spanning the levels of interest, and comparing it with the nominal level, gives the calibration error (CE) of Equation (91). This is the numerical counterpart of the reliability diagram. It is zero when the empirical coverage equals the nominal level at each of the M evaluated levels; such agreement on a finite grid, estimated from a finite test set, does not establish calibration at levels that were not evaluated.

$$\mathrm{CE} = \frac{1}{M}\sum_{m=1}^{M}|\hat{c}(\tau_m) - \tau_m| \tag{91}$$

Marginal calibration is a weaker guarantee than it appears. An estimator can attain the nominal level averaged over a test population while being badly miscalibrated within identifiable subsets of it, which is the situation that matters in deployment, where the operating condition is a single cell, temperature or duty rather than the population average. Partitioning the test set and reporting the worst-group coverage deviation, Equation (92), makes this visible; the natural partitions in this field are by cell, by temperature and by rate.

$$\Delta_{cov} = \max_{g\in\mathrm{G}}\left|\hat{c}_g - (1-\alpha)\right| \tag{92}$$

Here, $\hat{c}_\mathrm{g}$ is the empirical coverage within group g of the set of groups G, $1 - \alpha$ is the nominal coverage, a single level of the kind indexed by τ_m in Equation (91), and the maximum is taken over the groups. Good marginal coverage combined with a large worst-group deviation can indicate subgroup miscalibration, and reporting the former without the latter overstates what the estimator supports. Each group holds a finite and often small number of dependent observations, so the deviation should be reported with the group sizes and its sampling uncertainty, and a small deviation over the groups examined does not establish conditional validity in general. Where a distribution-free guarantee is wanted rather than an empirical check, conformal methods construct intervals whose marginal coverage holds under exchangeability alone, without assumptions on the underlying model [183]. Pooled cycles of battery data are, however, generally not exchangeable, because of their temporal ordering and their grouping by cell, and a construction that respects the grouping must still state the sampling assumption under which its coverage

guarantee holds. Subgroup deviations can be large in battery data. In the study discussed in Section 10.3, residual intervals with 90% nominal coverage, calibrated on the training cells and used as stress tests rather than as conformal intervals, were evaluated on each held-out cell: the mean held-out-cell coverage was 92.6% for Gaussian process regression, at an average interval width of 21.05 state-of-health points, but 76.2% and 76.0% for random forest and extra trees, whose coverage fell below 5% on one cell because their intervals were centered on systematically biased predictions [184]. Calibration reported systematically across cells, temperatures and rates nevertheless remains limited in the evidence reviewed here.

10.3. Validation Protocols

A metric is interpretable only in combination with the split that produced it, and the splits used in this field differ in what they establish. Five protocols should be distinguished, according to the deployment question each one answers.

Cycle-wise splitting partitions cycles of the same cells between training and test. A random cycle-wise split measures interpolation within trajectories the model has already seen, while a chronological split of the same cells tests prediction of the future of a cell already observed, which is the relevant question for online, cell-specific estimation; neither tests transfer to an unseen cell, and the random variant is nonetheless the most commonly reported. Cell-wise splitting holds out entire cells, so that no cycle of a test cell appears in training, and measures generalization to a new unit of the same design under the same conditions. Cross-temperature and cross-rate splitting, which together form the third protocol, hold out whole operating conditions, along with all cells tested under them, so that the model is tested at a temperature or a C-rate absent from training, and test transfer across operating conditions; this is extrapolation only when the held-out condition lies outside the training range, and interpolation otherwise. Cross-condition splitting holds out combinations of the two. External validation evaluates on cells from a different dataset, laboratory or cell design altogether, and tests transfer across that change of source. Which protocol is relevant depends on the intended use: cell-wise splitting answers the deployment question for new cells of a characterized design, and external validation answers it for a population that differs from the training source.

The difference between the first two protocols is large. In a systematic comparison on four cells of a public aging dataset, ten random cycle-wise splits were set against leave-one-cell-out testing for seven regression models. With all discharge diagnostics as predictors, the RMSE under random splitting lay between 1.23 and 2.75 state-of-health points; under leave-one-cell-out the best model attained 4.45 points, and the error of support vector regression rose from 1.84 to 9.19 points. The median ratio between the two protocols was 3.61 across 28 cell-and-model comparisons, with model-level ratios from 1.65 to 5.16, and the rank correlation between the two orderings of the models was zero [184]. The second finding is particularly important, since it shows that a cycle-wise comparison not only overstates accuracy but can also fail to identify the better estimator. The same study reports that richer predictor sets widen the gap rather than closing it, because additional features improve within-cell interpolation far more than transfer to an unseen cell. Results obtained under different protocols are therefore not comparable, and the protocol should be stated wherever an error figure is reported.

Two further requirements follow for prognosis specifically. The prediction starting point must be stated, because accuracy improves trivially as the end-of-life threshold is approached, so an aggregate over starting points concentrated late in life is not comparable with one over starting points early in life. And where several observations from the same cell appear in the test set, they are dependent, so the sampling uncertainty of a pooled

metric is understated unless the grouping is accounted for; per-cell results should be reported alongside the pooled value.

10.4. Data Leakage

Several of the protocols above fail in practice not because the split is chosen badly but because information crosses it elsewhere in the pipeline. Four mechanisms are common in this literature, and each inflates reported accuracy without any error in the model itself.

The first is the split itself. Where the claim concerns unseen cells, assigning cycles of the same cell to both partitions, as random cycle-wise splitting does by construction, permits the model to recognize the cell rather than to estimate its health. Adjacent cycles of one cell are strongly dependent, so the effect can be large: the median ratio of 3.61 reported in Section 10.3 indicates its size on that benchmark, although the benchmark does not rank it against the other three mechanisms.

The second is preprocessing fitted across the split. Normalization constants, the feature mean and standard deviation of Equation (26), filter parameters, and the components of any dimensionality reduction must be estimated on the training partition alone and then applied unchanged to the test partition. Estimating them on pooled data transfers information about the test distribution into the training pipeline. The requirement is stated in Section 5.2 and is repeated here because it is the easiest of the four to commit inadvertently.

The third is feature or model selection performed on the full dataset. Ranking indicators by their association with the target through Equations (39)–(42), or choosing a model order or hyperparameter by test-set performance, uses the test labels to make a modeling decision; the resulting estimate of generalization error is optimistic even though no test sample ever entered a training batch. Selection must be nested within the training partition, using an inner validation split, and the test partition used only once.

The fourth is specific to prognosis and the hardest to detect: the use of future information in constructing features or labels. A health indicator computed over a window extending beyond the prediction time, a trajectory smoothed with a non-causal filter such as the symmetric kernel of Equation (28), a capacity normalized by a maximum attained later in life, and a threshold or normalization chosen with knowledge of the observed end of life of the test cells all give the model access to information unavailable at the moment the prediction would be made. Future failure times may legitimately define the RUL targets of the training cells, since a supervised target is necessarily known only in retrospect; leakage occurs when information unavailable at the prediction time enters the predictors, the preprocessing, the split design or the evaluation conditions. Causality of the feature pipeline with respect to the prediction time should be verified explicitly, because none of the metrics of Section 10.1 can detect its violation.

A reported result is therefore incomplete without three further items: the split protocol, the stage at which each preprocessing and selection decision was fitted, and a statement that the feature pipeline is causal with respect to the prediction time. Where self-supervised pre-training is used, as in Section 7.4, the pre-training corpus must be included in this accounting, since unlabeled records from test cells carry the same identifying information as labeled ones.

10.5. Reporting Practice

Reliability practice outside the battery field offers a template here, in which the evidence required to support a life claim is specified before testing begins rather than selected afterwards. Three practices would materially improve comparability. Reporting several metrics rather than one gives a fuller account of the error distribution. Specifying the prediction starting point is essential for RUL results, for the reason given in Section 10.3.

Evaluating on held-out cells, under the protocols of Section 10.3 and free of the leakage mechanisms of Section 10.4, supports comparison across studies when the prediction targets, cohorts, operating conditions, preprocessing and metrics are also aligned. The metrics are summarized in Table 11.

Table 11. Evaluation metrics used in battery diagnostics and prognostics, with the governing formulation, the aspect of model behavior each metric captures, and the interpretation appropriate to each. The first seven metrics describe predictive accuracy and the last four the quality of the uncertainty statement; the two groups are not substitutes for one another.

Metric	Equation	Captures	Interpretation
Absolute error	Equation (82)	Deviation of a single prediction	Diagnostic inspection of individual cycles
Mean absolute error	Equation (83)	Average deviation with equal weighting	Robust to isolated outliers; reported in physical units
Mean squared error	Equation (84)	Average squared deviation	Emphasizes large errors; sensitive to regeneration spikes
Root mean squared error	Equation (85)	Square root of the mean squared error	Same units as the target; the most widely reported metric
Mean absolute percentage error	Equation (86)	Relative deviation	Use only with declared nonzero targets; unstable near zero
Maximum absolute error	Equation (87)	Worst-case deviation	Largest error observed on the evaluated set; not a deployment guarantee
Coefficient of determination	Equation (88)	Variance explained relative to a mean predictor	Near one indicates a close fit relative to the mean baseline on the evaluated set, not correct specification or generalization
Prediction interval coverage	Equation (89)	Empirical marginal interval coverage	Compare with nominal coverage and report width and sampling uncertainty
Mean prediction interval width	Equation (90)	Sharpness of the intervals at the stated level	Report with coverage and in the units of the target; narrow intervals are informative only if calibrated
Calibration error	Equation (91)	Agreement of stated with realized frequencies across nominal levels	Numerical form of the reliability diagram; zero indicates agreement at the evaluated levels only
Worst-group coverage deviation	Equation (92)	Departure from nominal coverage within the worst subgroup	Audits coverage within chosen groups, such as cells, temperatures or rates; report group sizes and sampling uncertainty

11. Open Problems and Emerging Directions

11.1. Data Availability and Label Scarcity

Run-to-failure data requires months or years of cycling, so it is expensive and rare, and as cell longevity improves the volume obtainable within a fixed program falls further. Labeling is the harder constraint, since a reference capacity measurement is not routinely available in the field. Weak and semi-supervised schemes exploit partially labeled operational records and require further validation. Public corpora also remain predominantly cell-level, whereas pack-level data capturing cell-to-cell dispersion, thermal gradients, and electrical imbalance remain scarce [11,185–188]. Evaluation of pack-level and cross-chemistry prognosis requires datasets that explicitly represent the target system, operating conditions and failure definition. This narrative review does not quantify the overall availability of such datasets.

11.2. Generalization Under Distribution Shift

Estimators generalize poorly beyond the conditions represented in training. Degradation depends on chemistry, cell design, thermal environment, and duty, and a model calibrated on one configuration typically requires retraining on another. Transfer learning and domain adaptation address this by initializing from a source corpus and adapting on limited target data. The discrepancy between the two distributions is commonly quantified by the maximum mean discrepancy of Equation (93), minimized jointly with the supervised loss so that the learned representation becomes approximately invariant across domains [189].

$$MMD^2(\mathcal{D}_s, \mathcal{D}_t) = \| \frac{1}{n_s}\sum_{i=1}^{n_s} \varphi(x_i^s) - \frac{1}{n_t}\sum_{j=1}^{n_t} \varphi\left(x_j^t\right) \|_{\mathcal{H}}^2 \tag{93}$$

with $\mathcal{D}_s$ and $\mathcal{D}_t$ representing the sets of n_s source samples x_i^s and n_t target samples x_j^t, and with φ representing the feature map projecting samples into the reproducing kernel Hilbert space $\mathcal{H}$, in whose norm the discrepancy is measured. Aligning the marginal feature distributions does not by itself preserve the relation between features and target: where that relation differs between source and target, for example, because a different degradation mechanism dominates, alignment can reduce accuracy, a failure known as negative transfer. Labeled target data therefore remains necessary for validation, and the target cells used for adaptation must be kept apart from those used for testing, by the split controls of Section 10.4. Table 12 summarizes the principal robustness challenges together with the corresponding mitigation strategies and the limitations each leaves unresolved.

Table 12. Model robustness challenges in battery diagnostics and prognostics and the corresponding mitigation strategies, with the residual limitation that each strategy leaves unresolved.

Challenge	Manifestation	Mitigation Strategy	Residual Limitation
Distribution shift	Accuracy loss when a laboratory-trained model is applied to field data	Domain adaptation minimizing the maximum mean discrepancy, Equation (93)	Requires representative target data, and labeled target data for validation; marginal alignment does not ensure that the feature-target relation transfers
Chemistry and format transfer	Retraining required for each new cell design	Transfer learning from a pre-trained backbone; foundation models	Gains depend on how far source and target share degradation behavior; negative transfer is possible where they do not
Label scarcity	Reference capacity unavailable in field monitoring	Weak and semi-supervised learning; physics residuals at unlabeled points	Pseudo-label error accumulates over training
Measurement uncertainty	Sensor noise and drift degrade input quality	Filtering and decomposition; probabilistic estimators	Calibration of intervals remains difficult for deep models
Cell-to-cell variability	Pack behavior is not the sum of its cells	Pack-level modeling with inconsistency terms	Pack-level datasets remain scarce
Data confidentiality	Field data held privately by competing operators	Federated learning with parameter exchange	Non-identically distributed clients slow convergence; shared updates can leak information without further safeguards
Embedded resource limits	Deep models can exceed onboard computing power and memory	Pruning, quantization, lightweight architectures	Compression trades accuracy against footprint

The conclusions on generalization, uncertainty treatment and data requirement drawn in Sections 4.4, 5.1, 10 and 11, and restated in Section 12, are of unequal evidential standing:

some rest on quantified comparisons, others on analysis of the formulations or of the available datasets, and one on sparse evidence. Table 13 summarizes them, together with the basis and the limits of each.

Table 13. Evidence underlying the review's qualitative conclusions on generalization, uncertainty treatment and data requirement. Each row states the conclusion as it is made in the text, the form and magnitude of the evidence supporting it, the source of that evidence, and the inference the evidence does not support.

Conclusion as Stated in This Review	Form of the Supporting Evidence	Source	What the Evidence Does Not Establish
Accuracy reported under cycle-wise splitting overstates generalization to unseen cells	Quantified, for the all-diagnostics predictor set: across seven regression models, RMSE of 1.23 to 2.75 SOH points under ten random cycle-wise splits; under leave-one-cell-out the best model attained 4.45 points, the median ratio was 3.61 over 28 cell-and-model comparisons, and model-level ratios ranged from 1.65 to 5.16	[184]	Four cells of one dataset and seven models; the ratio is a finding on that cohort and not a universal constant
The ordering of models obtained under cycle-wise splitting does not transfer to cell-wise evaluation	Quantified, for the all-diagnostics predictor set: rank correlation of zero between the two orderings of the same seven models	[184]	Whether the ordering transfers for models outside those seven, or on larger cohorts
Richer feature sets improve interpolation more than they improve transfer	Quantified: median cell-paired error ratio rising from 1.24 for an age-only predictor to 4.53 for diagnostics combined with age	[184]	That the effect holds for feature sets constructed differently from those tested
Early-cycle data can support life prediction within one chemistry under controlled protocols	Quantified: 9.1% test error for cycle-life prediction from the first 100 cycles, and 4.9% test error for two-group classification from the first 5 cycles, on 124 LFP cells with lives from 150 to 2300 cycles	[93]	Transfer to another chemistry, format or laboratory, or to field duty; all cells were of one type and were cycled in one laboratory under fast-charging protocols
Early prediction converts experimental cost into computational cost	Quantified: high-cycle-life protocols identified among 224 six-step ten-minute candidates in 16 days, against more than 500 days for exhaustive search without early prediction	[94]	That the same reduction is attainable outside the protocol space and chemistry searched
Interval coverage close to the nominal level does not establish conditional calibration	Analytical, following the calibration-and-sharpness decomposition: marginal coverage averages over the test population, so subgroup deviations of opposite sign can cancel, which the worst-group deviation of Equation (92) exposes. Quantified for SOH on four cells with 90% residual intervals: averaged held-out-cell coverage hid cell-level failures, since random forest and extra trees averaged 76.2% and 76.0% but covered less than 5% of one cell, while Gaussian process regression reached 92.6% only with intervals 21.05 SOH points wide	[182,184], Section 10.2	Coverage behavior across temperatures, rates and larger cohorts, or a coverage guarantee for grouped constructions; the quantified evidence covers four cells of one dataset
Run-to-failure labels rather than record volume are the binding data constraint	Structural: among the sources of Table 3, those with the largest cell counts are aggregations of heterogeneous sources or calendar-aging cohorts; the largest single-source cohort cycled to failure has 124 cells; and the only pack-level source carries no health labels	Tables 3 and 4, Section 11.1	A quantitative estimate of the labeling cost, or of how much labeled data a given accuracy requires
Cross-chemistry transfer is unestablished for solid-state systems	Sparse evidence: two isolated studies estimate a state or a cycle life on measured solid-state cells, one of them on 12 polymer-electrolyte cells whose records are public, and no larger public cycling or aging dataset was located	Section 4.4, [87,88]	That the formulations fail to transfer; sparse evidence is not evidence against transfer, and the search of Section 2.1 was not exhaustive

11.3. Hybrid and Physics-Informed Formulations

Hybrid schemes combine the transparency of mechanistic models with the flexibility of learned ones. In the first pattern, a learned component is coupled to a physical model, either correcting the residual of its prediction or estimating and updating its parameters; the particle-filter constructions of Section 8.3, which update the parameters of a physical degradation model online, are model-based estimators of related structure but contain no

learned component. In the second, physical relations are imposed on the learning problem itself. Physics-informed learning adds a residual of the governing relation to the training objective, as shown in Equation (94) [190]. Figure 12 shows a representative coupling of the first kind, in which a long short-term memory network estimates the parameters of a pseudo-two-dimensional electrochemical model from current, voltage, temperature, and state of charge, and the estimated parameters update the physical model [67].

$$L_{PINN} = \frac{1}{N_d}\sum_{i=1}^{N_d} \| \hat{y}_i - y_i \|^2 + \lambda_p \frac{1}{N_c}\sum_{j=1}^{N_c} \| \mathcal{N}[\hat{y}](x_j) \|^2 \tag{94}$$

where $\hat{y}_i$ is the network output for the ith of the N_d labeled samples and y_i is its label, λ_p balances the two contributions, and $\mathcal{N}[\hat{y}](x_j)$ is the residual of the governing relation, written in the form $\mathcal{N}[y] = 0$, evaluated for the network output $\hat{y}$ at the jth of the N_c unlabeled collocation points x_j. The differential operator $\mathcal{N}[\cdot]$ is written as is usual in physics-informed learning and is unrelated to the Gaussian distribution $\mathcal{N}$ of Equation (73). The collocation count N_c is distinct from the particle count N_p in Equations (78)–(80). In the battery setting, formulations of this type have been applied to degradation modeling and prognosis [191] and to state-of-health prognostics from partial charging segments [192]. Table 14 summarizes the hybrid and physics-informed strategies in current use, the mechanism by which each couples physical knowledge to the learned component, and the benefit each provides.

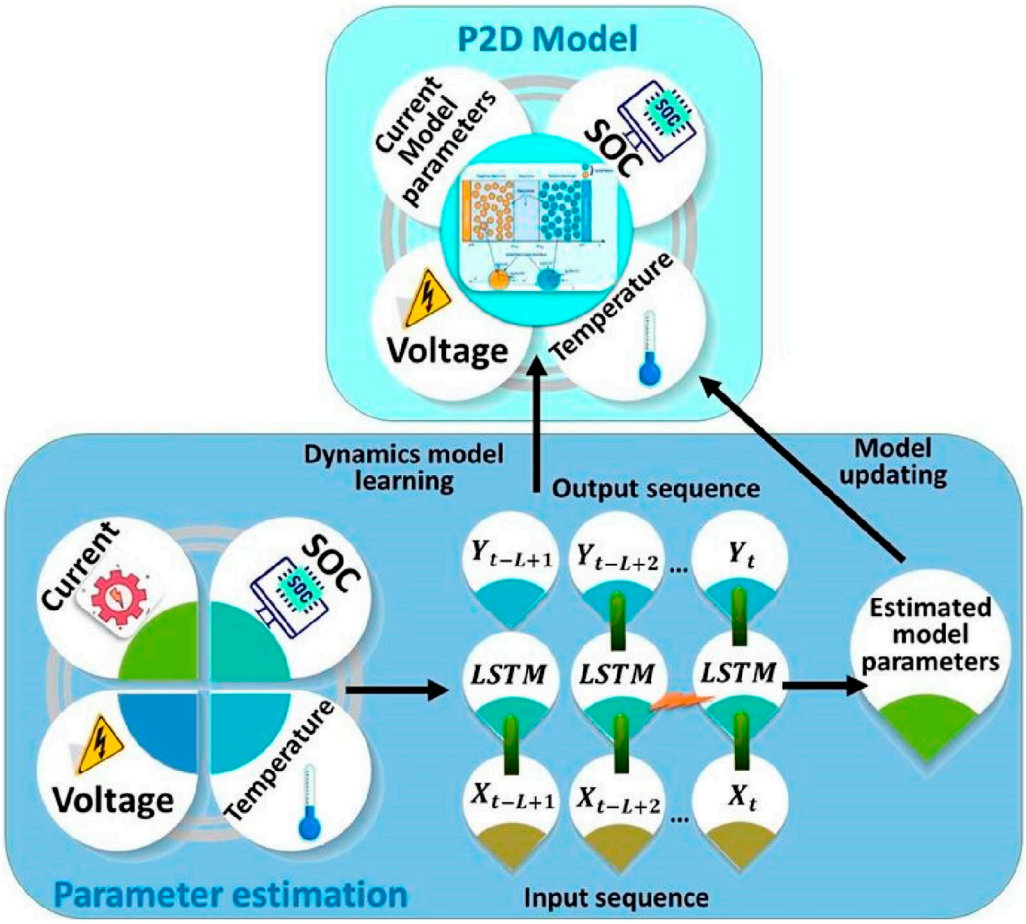


Figure 12. Coupling of a pseudo-two-dimensional (P2D) electrochemical model with a long short-term memory (LSTM) network, in which current, voltage, temperature, and state of charge drive parameter estimation and the estimated parameters update the physical model. Reproduced without modification under the terms of the Creative Commons Attribution (CC BY) license from [67].

Table 14. Hybrid, model-based, data-driven, and physics-informed modeling strategies in battery diagnostics and prognostics, with representative method combinations and the specific benefit each combination provides.

Strategy	Representative Combination	Mechanism	Benefit
Model-based sequential estimation	Empirical degradation function with a particle filter	Physical model captures the trend; the filter updates parameters online	Retains physical plausibility while adapting to the individual cell
Physics-constrained loss	Neural network with an added physics residual, Equation (94)	Governing relation is enforced at unlabeled collocation points	Extracts information from operating data that carries no label
Physics-inspired features	Incremental capacity and impedance features fed to a learner	Feature construction encodes electrochemical structure	Improves interpretability without altering the model class
Data-driven architecture fusion	Convolutional layers with recurrent layers	Spatial extraction followed by temporal modeling	Combines local feature extraction with temporal modeling; any gain over either architecture alone requires a matched baseline comparison
Probabilistic fusion	Gaussian process regression with a state-space filter	Kernel model supplies the trend; the filter supplies recursion	Delivers model-based predictive intervals for multi-step prognosis
Reduced-order coupling	Single-particle model with a neural correction term	Simplified physics is corrected by a learned residual	Approaches full-order accuracy at tractable cost

11.4. Interpretability

High-capacity estimators resist physical interpretation, which impedes adoption where a prediction must be justified. Attribution methods supply model-agnostic explanations by estimating the marginal contribution of each input to a given prediction. Shapley additive explanations derive that contribution from cooperative game theory [193] and local interpretable model-agnostic explanations fit a simple surrogate in the neighborhood of the prediction [194]. Both identify which indicators, charging segments, or temperature ranges drive an estimate, which supports validation against physical expectation. Attention weights offer a native form of interpretation and should be read as a description of where the model attended rather than as a causal account.

11.5. Foundation Models, Language Models, and Federated Training

Three developments postdating the previous generation of reviews are reshaping the agenda. The first is the transfer of the pre-train-and-adapt paradigm to prognostics. Time-series foundation models, pre-trained on large time-series corpora and then adapted to battery data, have reduced the error of long-horizon state-of-health forecasts in cross-condition and cross-chemistry tasks on public datasets, and their generalization has been related to the size of the fine-tuning set [195]. This offers a route to transfer learning that requires evaluation under explicit target-domain and out-of-distribution protocols.

The second is the application of large language models to health forecasting, where few-shot and zero-shot SOH forecasting has been demonstrated from structured prompts without task-specific fine-tuning [196]. The significance lies less in raw accuracy than in the prospect of a general estimator requiring no bespoke model per cell type, together with an interface through which a diagnosis can be interrogated directly.

The third is federated learning, which responds to the fact that the organizations holding the most valuable field data are commercial competitors. Federated schemes train a shared model across distributed private corpora by exchanging model updates rather than raw data, which keeps the records local while drawing on a much larger body of training data [197]. Confidentiality is not guaranteed by this alone, since shared updates can leak information, and depends on the threat model and on the safeguards applied to

those updates. The principal technical obstacle is the non-identically distributed character of the client datasets, since each fleet operates under a different usage regime [198].

11.6. Deployment Architecture

Networked monitoring aggregates data from many operating units and supports fleet-level inference that a standalone controller cannot deliver [199]. Figure 13 shows the resulting architecture, in which the onboard management system and a cloud-resident battery database exchange operating data and derived health information over a communication network [101]. Edge computation complements this by processing locally, which reduces latency and bandwidth. Digital twins maintain a continuously updated computational model of the physical system and offer a route to improved accuracy across the lifecycle [200]. A frequent constraint on both edge and cloud deployment is the computational cost of high-capacity models, which can require pruning, quantization, or architectural simplification for embedded execution.

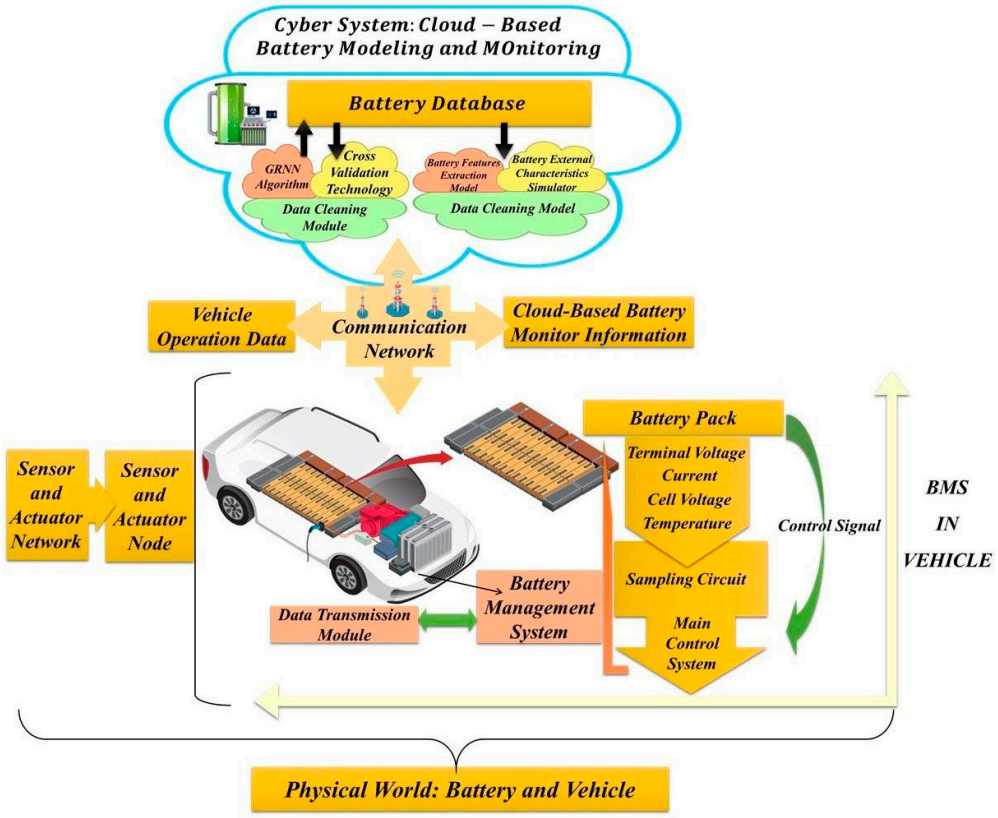


Figure 13. Cloud-based battery modeling and monitoring architecture, in which the onboard management system exchanges operating data and derived health information with a cloud-resident battery database over a communication network. Reproduced without modification under the terms of the Creative Commons Attribution (CC BY) license from [101].

12. Synthesis and Conclusions

12.1. Cross-Family Comparison

Table 15 operationalizes the comparison promised in the abstract. Each family is characterized by the assumption it makes about the degradation process, the condition under which its parameters are identifiable, the data scale at which it becomes usable, its dominant computational cost, how uncertainty is obtained, the condition under which extrapolation is defensible, and the validation protocol that will detect its characteristic

failure mode. Figure 14 places that comparison against the estimation problem of Section 3 and the lifetime laws of Section 4.2.

Table 15. Model families compared on assumptions, identifiability, data requirement, cost, uncertainty, extrapolation condition, and recommended validation. Section numbers give the derivation.

Family	Core Assumption	Identifiability	Data Need	Cost	Uncertainty	Generalization and Applicability Conditions	Recommended Validation
Deterministic laws (Section 4)	Fade follows a fixed algebraic trend	Few parameters; identifiable from one trajectory	Very low	Negligible	None natively; bootstrap over fits	Stress factors stay within the fitted range	Cross-condition holdout
Wiener process (Section 4)	Increments independent Gaussian; monotone drift	Drift and diffusion identifiable from repeated paths	Low to moderate	Low	Closed-form inverse Gaussian life distribution in operational time; usually numerical with random effects or measurement error	Chosen drift, diffusion, time transformation and threshold remain appropriate	Cell-wise split; PICP and interval width
Gamma process (Section 4)	Independent non-negative Gamma increments with the specified shape and common rate structure	Requires informative repeated increments or paths and an identifiable parameterization; monotonicity alone is insufficient	Low to moderate	Low	Survival via incomplete gamma	Latent increments fit the Gamma law; negative observations due to measurement noise are modeled separately	Check increment sign before fitting
Reduced-order electrochemical (Section 4)	Governing transport and kinetics known	Often weakly identifiable; parameters correlated	Low, but needs characterization	Moderate to high	Propagated from parameter posterior	Model form and parameters remain valid	Parameter re-identification as cell ages
Kernel and Bayesian (Section 6)	Smooth map from features to target	Conditional on kernel choice and hyperparameter estimation; not guaranteed by kernel selection alone	Small to moderate	Cubic in sample count	Native for Gaussian processes and RVM	Test features lie inside the training hull	Cell-wise split; leakage audit
Tree ensembles (Section 6)	Piecewise-constant map	Not identifiable; many ensembles fit the data equally well	Moderate	Moderate	Quantile or ensemble spread	Rarely; extrapolation is flat by construction	Cross-condition holdout
Deep architectures (Section 7)	Representation learnable from raw sequences	Over-parameterized; not identifiable	Large labeled corpora	High training; inference cost depends on architecture and hardware	Requires ensembling or Bayesian layers	Deployment distribution matches training	Cell-wise and cross-dataset external validation
Sequential filters (Section 8)	Transition and observation models known	Requires observability of the state	Low; online	Low to moderate	Native recursive posterior	Transition model remains faithful	Innovation whiteness; consistency of the normalized estimation error squared (NEES)
Physics-informed (Section 11)	Governing residual is a valid constraint	Constrained, not guaranteed, by the physics residual; depends on collocation coverage and residual weighting	Moderate	High	By the probabilistic variant used	Governing equation holds over the query domain	Residual audit outside training range

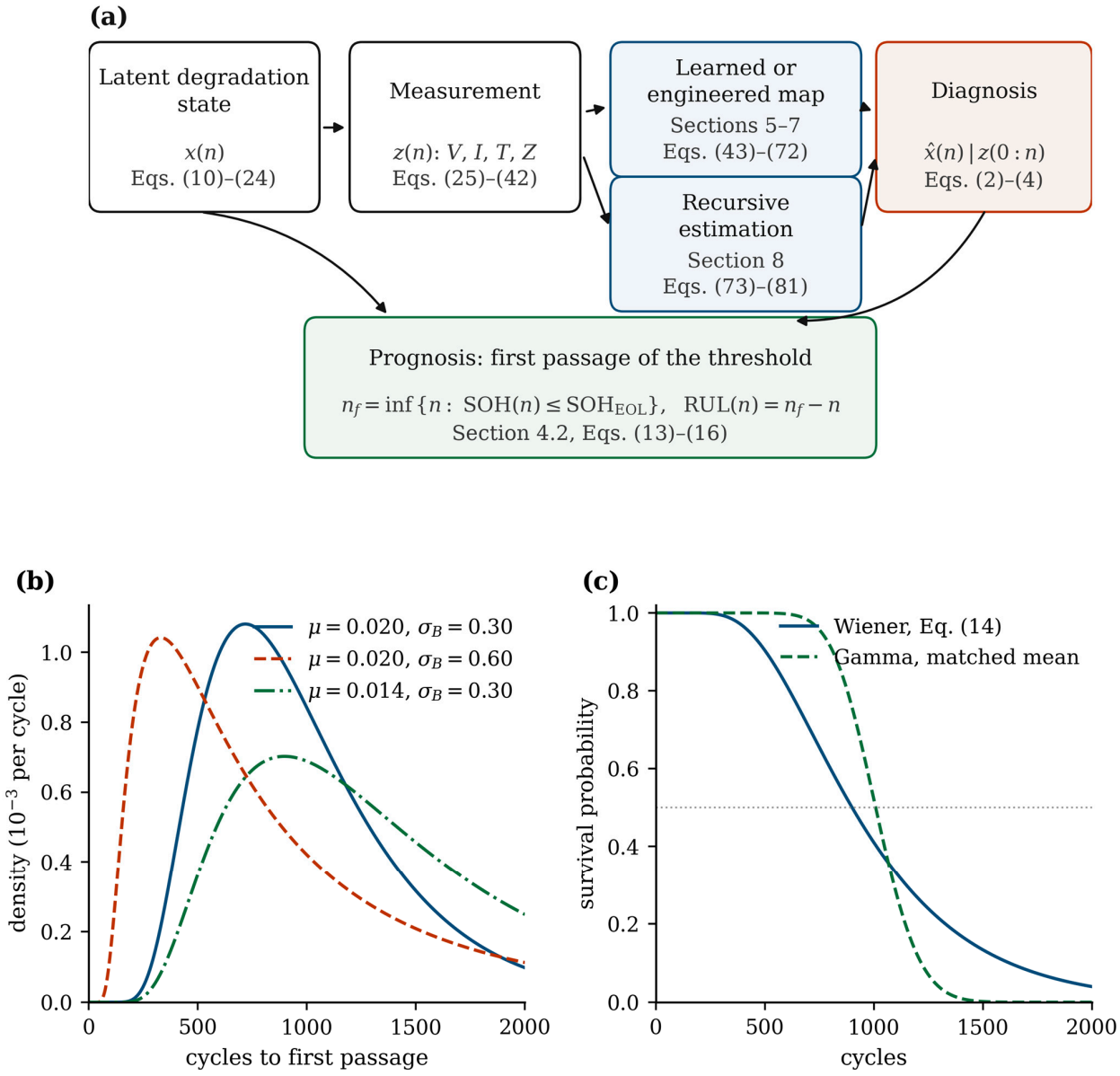


Figure 14. Graphical synthesis of the unified estimation problem and of the lifetime laws it induces, drawn by the authors: (**a**) the latent degradation state is observed only through terminal measurements, from which diagnosis proceeds by an engineered or learned map or by recursive estimation, while prognosis is the first passage of the declared end-of-life threshold; (**b**) inverse-Gaussian first-passage densities of Equation (14) for three drift and diffusion pairs, with the threshold distance held fixed; (**c**) the survival function of the first of these Wiener cases and that of a Gamma process of Equation (16) with the same mean degradation rate, illustrating two selected lifetime distributions whose relative spread is set by the chosen variance parameters rather than by monotonicity alone. Panels (**b**,**c**) are computed from synthetic parameters and are illustrative only; they report no measured data. Both use the linear clock $\Lambda(t) = t$, with t in cycles, the initial state $x_0 = 0$ and the threshold $w = 20$, the degradation measure being expressed in percentage points of state of health lost; the drift and diffusion pairs are given in the legend of (**b**), with μ in points per cycle and σ_B in points per square-root cycle. The Gamma process in (**c**) has $\alpha_\Gamma = 0.04$ per cycle and $\beta_\Gamma = 2.0$ per point, so that its mean increment per cycle, $\alpha_\Gamma / \beta_\Gamma = 0.020$, equals the drift $\mu = 0.020$ of the Wiener case with $\sigma_B = 0.30$, while its variance per cycle, $\alpha_\Gamma / \beta_\Gamma{}^2 = 0.01$, is one ninth of $\sigma_B{}^2 = 0.09$. Integrating the survival functions gives mean lives of 1000 and 1012.5 cycles, and standard deviations of about 474 and 158 cycles, for the Wiener and Gamma cases, respectively.

This review has treated data-driven diagnostics and prognostics for battery systems as a single estimation problem and has presented the principal method families through their governing mathematics. Posing the problem as the recovery of a latent degradation state from partial noisy observation clarifies why the field has converged on the structures it has: capacity control is central to estimator selection because the number of independent cells cycled to failure is limited (124 cells in the largest single-source cohort of Table 3); probabilistic formulations are preferred for prognosis because a maintenance decision requires a distribution; and physics-informed constructions are attractive because the physics residual is computable where labels are not.

12.2. Concluding Remarks

A consistent notational framework, with the symbols collected in Appendix A, has been used to place deterministic, stochastic, physics-based, kernel, ensemble, deep, and sequential Bayesian formulations on a common footing, and Table 15 collects the resulting comparison. On that basis, the trade-offs are explicit. Selected stochastic process models admit analytic lifetime laws under restrictive assumptions; other settings require numerical evaluation of survival probabilities or conditional lifetime distributions. Kernel and ensemble methods can be competitive with limited data, but their computational costs are method-dependent; exact Gaussian process inference, in particular, scales cubically with sample count. Deep architectures learn representations, at a cost in data and interpretation effort that rises in proportion. Sequential filters supply model-based recursive posterior estimates whose calibration must be verified empirically, and they require a transition model whose fidelity bounds the result.

Several directions follow. Extending inference from cells to packs is the most consequential gap, since pack behavior is not the superposition of its constituents. Transfer across chemistry remains a validation question, requiring explicit target-system evidence and reporting of the conditions under which a model can be reused. Evaluation practice requires standardization, in particular: splits matched to the intended use, with cell-wise rather than random cycle-wise splitting wherever generalization to unseen cells is claimed; joint reporting of nominal coverage, empirical interval coverage, interval width, calibration error across nominal levels, worst-group coverage deviation and sampling uncertainty; and a statement of the leakage controls of Section 10.4. Foundation models and federated training address generalization and data access, respectively, and warrant assessment against established baselines rather than bespoke comparisons. Interpretability remains a precondition for regulatory acceptance. Sustained progress will finally depend on renewed attention to degradation mechanisms, material evolution, and multi-physics coupling, rather than on continued refinement of model architecture alone [45].

Author Contributions: Conceptualization, M.T. and H.S.K.; methodology, M.T.; investigation, M.T.; writing, original draft preparation, M.T.; writing, review and editing, H.S.K.; visualization, M.T.; supervision, H.S.K.; project administration, H.S.K.; funding acquisition, H.S.K. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIT), grant number RS-2024-00405691 and this research was also supported by the ANCHOR through the Seoul ANCHOR Center, funded by the Ministry of Education (MOE) and the Seoul Metropolitan Government (2026-ANCHOR-01-007-04).

Data Availability Statement: No new data was generated in this study. The public datasets discussed are listed in Table 3 and cited in Section 5.1; the solid-state cycling records discussed in Section 4.4 are available from the repository given in the cited study.

Acknowledgments: During the preparation of this manuscript, the authors used ChatGPT's GPT-6 Astra (OpenAI) and Claude (Anthropic) to assist with language editing, wording refinement, and consistency checking of the text and reference list. No generative AI tool was used to create, alter, or generate any figure, dataset, experiment, or analysis presented in this work, and no AI tool is listed as an author. The authors reviewed and edited all content and take full responsibility for the content of this publication.

Conflicts of Interest: The authors declare no conflicts of interest. The funders had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript; or in the decision to publish the results.

Appendix A. Nomenclature

Table A1 lists the principal symbols used in the numbered equations, with the meaning and the dimensions or units of each. Because the review draws on several bodies of literature, a number of established symbols carry more than one meaning. Where re-lettering would not obscure a standard equation, one of the meanings has been re-lettered or subscripted, as listed in Section 2.2; every symbol that still carries more than one meaning, including those distinguished only by a subscript or an argument, is stated in the text where it occurs and is collected in Table A2, which states every meaning each such symbol carries, the equation that fixes each meaning, and how the ambiguity is resolved. The conventions behind this treatment are stated in Section 2.2.

Table A1. Symbols used in the governing equations, with the meaning, the dimensions or units, and the equations in which each symbol is defined or principally used. Weights, penalty coefficients and learning rates of the statistical-learning formulations are listed as dimensionless on the convention that inputs and targets are scaled to dimensionless form as in Equation (26); for unscaled variables their units follow from the objective in which they appear. Symbols that carry more than one meaning across the review are collected separately in Table A2.

Symbol	Meaning	Equation
$x(n)$, $z(n)$	Latent degradation state and measurement vector at cycle n. Units: component-dependent, those of the quantities each component represents.	Section 3
n, n_{f}, $\mathrm{SOH}_{\mathrm{EOL}}$	Cycle index; failure index at which the health threshold is first crossed; declared end-of-life value of the health metric. Units: dimensionless (cycle counts); percent.	Equations (8) and (9); Section 3.3
Q, t_{d}, C_0, $C_{\max}$	Delivered capacity; total discharge duration; pristine capacity; maximum capacity at cycle n. Units: A h; h; A h; A h.	Equations (1), (2) and (12)
$C_{\max}(n)$, η	Capacity basis for SOC, as in Equation (4); coulombic efficiency. Units: A h; dimensionless.	Equation (5)
t, t_0, τ	Time; initial time of integration; integration variable; in Equations (69)–(71), t is instead a discrete step index. Units: h in Equations (1), (5) and (7), with the conversion for time in seconds stated after Equation (1); s in the circuit, diffusion and thermal relations of Equations (19), (21) and (24); cycles or hours, as declared, in Section 4.2.	Equations (5) and (69)–(71)
${V_{\mathrm{oc}}}^{-1}(\cdot)$, δV, $\delta\mathrm{SOC}$	Inverse of the open-circuit-voltage characteristic, defined where that characteristic is strictly monotone; error in the reconstructed open-circuit voltage; resulting error in the inferred SOC. Units: V to percent; V; percent.	Equation (6)
$\varepsilon_{\mathrm{SOC}}$, b_{I}	Error of the ampere-hour estimate, taken as the estimate minus the true value; constant bias in the measured current. Units: percent; A.	Equation (7)

Table A1. *Cont.*

Symbol	Meaning	Equation
$R(n)$, R_0, R_1, R_{EOL}, R_{BOL}	Internal resistance at cycle n; ohmic resistance of the equivalent circuit; polarization resistance; end-of-life resistance; beginning-of-life resistance. Units: Ω.	Equations (3), (6) and (17)–(19)
V_{t}, V_{oc}, V_1	Terminal, open-circuit and polarization voltage. Units: V.	Equations (6) and (17)–(19)
C_1	Polarization capacitance, distinct from the capacities C_0 and $C_{\max}$. Units: F (farad).	Equation (19)
A, E_{a}, R_{g}, T	Pre-exponential factor; activation energy; gas constant; absolute temperature. Units: A carries the units of the rate constant k, which follow from the assumed rate law; E_{a} in J mol^{-1}; R_{g} = 8.314 J mol^{-1} K^{-1}; T in K.	Equation (10)
Q_{loss}, B, γ, C_{rate}, A_{h}, z	Capacity loss; pre-exponential factor; rate weighting; C-rate; ampere-hour throughput; fade exponent. Units: Q_{loss} in A h or percent of C_0, and B in the same units; γ in J mol^{-1} per unit C-rate; C_{rate} and z dimensionless; A_{h} normalized by a stated reference throughput, hence dimensionless.	Equation (11)
α, β	Diffusion-limited and linear capacity-fade coefficients. Units: A h cycle$^{-1/2}$ and A h cycle^{-1}, respectively.	Equation (12)
$X(t)$, x_0, w	Degradation path; initial state; failure threshold. Units: those of the declared degradation measure, commonly A h or percent of C_0.	Equations (13) and (14)
μ, σ_{B}, B, $\Lambda(t;\theta)$	Drift; diffusion coefficient; Brownian motion; time transformation. Units: μ in degradation units per unit operational time; σ_{B} in degradation units per square root of operational time; B in square-root operational-time units, since Var $B(s) = s$; Λ in operational-time units.	Equation (13)
t_{c}, x_{c}, u	Current time; degradation state at the current time; future elapsed time. Units: t_{c} and u in cycles or hours as declared; x_{c} in degradation units.	Text after Equation (15)
θ	Parameter vector of the model under discussion: of the time transformation in Equation (13), the encoder in Equations (58) and (59), the network in Equation (65), and the Bayesian model in Equation (81). Units: component-dependent.	Equations (13), (58), (59), (65) and (81)
T_{f}	First-passage (failure) time, the continuous-time counterpart of the failure index n_{f}. Units: cycles or hours, matching the time variable of Equation (13).	Equations (14) and (15)
α_Γ, β_Γ	Gamma shape coefficient and rate parameter. Units: α_Γ in reciprocal operational-time units, so that the Gamma shape is dimensionless; β_Γ in reciprocal degradation units, so that its inverse is the scale.	Equation (16)
j, j_0, α_{a}, α_{c}, F, η_{ov}	Local and exchange current density; transfer coefficients; Faraday constant; surface overpotential, distinct from the coulombic efficiency η of Equation (5). Units: A m^{-2}; dimensionless; F = 96,485 C mol^{-1}; V.	Equation (20)
c_{s}, D_{s}, r	Solid-phase concentration; diffusivity; radial coordinate. Units: mol m^{-3}; m^2 s^{-1}; m.	Equation (21)
R_{p}, $c_{\mathrm{s},0}$	Particle radius; initial solid-phase concentration, taken uniform. Units: m; mol m^{-3}.	Equation (22)
$\dot{q}$, m, c_{p}, h, A_{s}, T_{amb}	Heat generation; mass; specific heat; heat transfer coefficient; surface area; ambient temperature. Units: W; kg; J kg^{-1} K^{-1}; W m^{-2} K^{-1}; m^2; K.	Equations (23) and (24)

Table A1. *Cont.*

Symbol	Meaning	Equation
N_{w}, L, σ	Moving-average window length; Gaussian half-window; Gaussian smoothing width. Units: samples, hence dimensionless.	Equations (27) and (28) and text after (28)
μ_{x}, σ_{x}	Sample mean and standard deviation: of the feature, estimated on the training partition, in Equation (26), and of the signal over the analysis window in Equation (33). Units: those of the feature or signal.	Equations (26) and (33)
m, r, $A^m(\mathrm{r})$, $\mathrm{B}^m(r)$	Embedding dimension; tolerance; template-match counts. Units: m dimensionless; r in the units of the signal, or as a stated multiple of its standard deviation; template-match counts dimensionless.	Equation (34)
ρ, $\hat{\rho}_j$, λ, m_{h}, D_{KL}	Target and observed activation; sparsity weight; hidden-unit count; Kullback–Leibler divergence. Units: dimensionless.	Equations (38), (60) and (61)
$\Delta_{\mathrm{i}}(k)$, $\Delta_{\min}$, $\Delta_{\max}$, ζ	Grey relational deviation; its smallest and largest values over all candidates and indices; distinguishing coefficient. Units: those of the normalized sequence; ζ dimensionless.	Equation (41)
$R(f)$, $\hat{R}_{\mathrm{N}}(f)$, $\Re_{\mathrm{N}}(\mathcal{F})$, δ	Expected and empirical risk; empirical Rademacher complexity of the loss class induced by $\mathcal{F}$; confidence parameter. Units: dimensionless, the loss being bounded in the unit interval.	Equation (44)
λ_1, λ_2	Lasso and ridge penalty weights. Units: dimensionless for inputs and target scaled as in Equation (26); otherwise set by the terms of Equation (45).	Equation (45)
C, ξ_{i}, $\xi_{\mathrm{i}}{}^*$, ε	SVR penalty; slack variables; insensitivity width. Units: C dimensionless for inputs and target scaled as in Equation (26), otherwise set by the objective of Equation (46); slack variables and ε in the units of the target.	Equation (46) and its constraints
w, b, α_{i}, $\alpha_{\mathrm{i}}{}^*$, e_{i}, λ	Weight vector and bias of the support-vector regression function; Lagrange multipliers of the two sets of constraints; residuals and regularization constant of the least-squares variant. Units: dimensionless for inputs and target scaled as in Equation (26); otherwise set by the objectives of Equations (46) and (49).	Equations (46), (47) and (49)
$k(\cdot,\cdot)$, K, k_*, ℓ	Kernel or covariance function; covariance matrix; test covariance vector; length scale. Units: for Gaussian process regression, the kernel values, the covariance matrix and the test covariance vector in the squared units of the target, and dimensionless for the support-vector kernel; the length scale in the units of the input.	Equations (47), (48) and (53)–(56)
z, ϕ	Latent code of the autoencoder; decoder parameters, distinct from the feature map φ of Section 6.2 and Equation (93). Units: dimensionless.	Equations (58)–(60)
$\sigma(\cdot)$	Logistic sigmoid activation, distinct from the smoothing width σ defined after Equation (28) and from the subscripted standard deviations. Units: dimensionless.	Equations (63), (64) and (69)–(71)
η	Learning rate, distinct from the coulombic efficiency η of Equation (5). Units: dimensionless.	Equation (65)
H	Hidden-layer output matrix of the extreme learning machine. Units: dimensionless.	Equation (66)
$w_{\mathrm{j}}{}^{\mathrm{l}}$, R_{i}	Convolution kernel weights of layer l; pooling window of output position i. Units: dimensionless.	Equations (67) and (68)

Table A1. *Cont.*

Symbol	Meaning	Equation
x_k, z_k, f_k, h_k, w_k, v_k, Q_k, R_k	State; measurement; transition and observation functions; process and measurement noise; their covariances. Units: those of the state and of the measurement; the noise covariances in the corresponding squared units.	Equation (73)
F_k, H_k, K_k, P_k	Transition and observation matrices, which are the Jacobians of Equation (77) in the extended filter; Kalman gain; error covariance. Units: the matrices and the gain are ratios of the corresponding quantities; P_k in the squared units of the state.	Equations (74)–(77)
N_p, ω_k, N_{eff}	Particle count; importance weights; effective sample size. Units: dimensionless.	Equations (78)–(80)
$\mathcal{D}$	Observed data in the model-averaged predictive distribution and in the posterior update. Units: component-dependent.	Equations (57) and (81)
N_s, Y, $\hat{Y}$	Test-set size; measured and predicted target. Units: N_s dimensionless; the targets in the units of the estimated quantity.	Equations (82)–(90)
L_i, U_i	Lower and upper prediction-interval bounds. Units: those of the target.	Equations (89) and (90)
MPIW	Mean prediction interval width. Units: those of the target.	Equation (90)
CE, M, τ_m, $\hat{c}(\cdot)$	Calibration error; number of nominal levels; nominal level; empirical coverage of the central interval at a nominal level. Units: dimensionless.	Equation (91)
Δ_{cov}, G, g, $\hat{c}_g$, α	Worst-group coverage deviation; set of groups; group index; empirical coverage within group g; nominal non-coverage level, so that $1-\alpha$ is the nominal coverage. Units: dimensionless.	Equation (92)
φ, $\mathcal{H}$, n_s, n_t	Feature map; reproducing kernel Hilbert space; source and target sample counts. Units: the feature map is dimensionless under a normalized kernel; the sample counts are dimensionless.	Equation (93)
$\mathcal{N}[\cdot]$, N_d, N_c, λ_p	Differential operator; labeled sample count; collocation count; physics-loss weight. Units: the operator returns the units of the governing-equation residual; the counts are dimensionless; λ_p is chosen so that the two loss terms are commensurable.	Equation (94)
$I(t)$, $Q_{avail}(k)$	Current, positive on discharge; available charge at sample k. Units: A; A h.	Equations (1), (4)–(6), (17)–(19) and (23)
N	Number of samples or points in the stated window or training set; distinct from cycle n and evaluation N_s. Units: dimensionless.	Equations (31)–(34), (39)–(41), (44), (46), (47), (49) and (59)
g_j	Discretely normalized Gaussian filter weight. Units: dimensionless.	Equation (28)
ψ, a, b	Mother wavelet, scale and translation; Units: a and b in the units of the independent variable; ψ normalized to unit energy.	Equation (29)
V_{nom}	Nominal reference voltage for mean voltage falloff. Units: V.	Equation (31)
t_1, t_2	Endpoints of the energy-integration interval. Units: s or h as declared.	Equation (35)
d_i	Difference between paired ranks in Spearman's coefficient. Units: dimensionless.	Equation (40)
${\sigma_\varepsilon}^2$	Observation-noise variance, in the bias–variance decomposition and in Gaussian process regression. Units: the squared units of the target.	Equations (43), (54) and (55)
ν	Gradient-boosting shrinkage or learning-rate parameter. Units: dimensionless.	Equation (51)

Table A1. *Cont.*

Symbol	Meaning	Equation
μ^*, σ^{*2}, ${\sigma_f}^2$	GPR posterior mean; latent posterior variance; kernel signal variance. Units: the posterior mean in the units of the target; the variances in its squared units.	Equations (54)–(56)
W, b, a^l, ${z_i}^l$, h_t	Neural layer or recurrent weights; bias; layer activation; pre-activation of a convolutional layer; hidden state. Units: dimensionless.	Equations (63) and (67)–(71)
i_t, f_t, o_t, c_t	LSTM input, forget and output gates; cell state. Units: dimensionless.	Equation (70)
r_t, z_t, $\tilde{h}_t$	GRU reset and update gates; candidate hidden state. Units: dimensionless.	Equation (71)
Q, K, V, d_k	Attention query, key and value matrices; key dimension. Units: dimensionless.	Equation (72)

* denotes complex conjugation.

Table A2 is the reuse register referred to in Section 2.2, and it is intended to be read as a check. Indices that serve only as summation or position indices, dummy variables of integration, and multi-letter labels such as MAE or PICP are not listed; any other symbol absent from Table A2 carries a single meaning throughout, and a symbol present in it should be read with the equation that fixes the intended meaning.

Table A2. Register of symbols that carry more than one meaning across the review. Each row states every meaning the symbol carries, the equation in which each meaning is fixed, and the device by which the meanings are kept distinct. The conventions are stated in Section 2.2.

Symbol	Distinct Meanings and the Equation That Fixes Each	How the Ambiguity is Resolved
A	Pre-exponential factor of the Arrhenius law in Equation (10); ampere-hour throughput A_h in Equation (11); cell surface area A_s for convective heat exchange in Equation (24); template-match count $A^m(r)$ of the sample-entropy statistic in Equation (34).	The throughput and the area carry the subscripts h and s, and the count the superscript m and an argument; the bare symbol is the pre-exponential factor.
a, b	Wavelet scale and translation in Equation (29); layer activation and bias in Equations (63) and (67); activations within the pooling window in Equation (68); visible and hidden biases of the restricted Boltzmann machine in Equation (61); recurrent and gate biases b_h, b_i, b_f, b_o and b_c in Equations (69) and (70); bias of the support-vector and least-squares formulations in Equations (46), (47) and (49); current-sensor bias b_I in Equation (7).	In the equations the activations and biases carry a layer superscript, a positional subscript or an identifying subscript, except the bias b of the support-vector formulations, which is bare and is used only in Section 6.2; the wavelet scale and translation are used only in Section 5.2, where they are defined.
B	Pre-exponential factor of the capacity-loss law in Equation (11); standard Brownian motion in Equation (13); template-match count $B^m(r)$ in Equation (34).	Separated by section and by argument: the pre-exponential factor occurs only in Section 4.1, the Brownian motion only in Section 4.2, where it carries its argument in the equations, and the count only in Section 5.3.
C	Pristine and maximum capacity C_0 and C_{max} in Equations (2), (4), (5), (7) and (12); polarization capacitance C_1 in Equation (19); C-rate C_{rate} in Equation (11); penalty parameter of the support-vector formulation in Equation (46).	The capacities, the capacitance and the C-rate carry a subscript; the bare symbol is the penalty parameter and occurs only in Section 6.2.
c	Solid-phase concentration c_s in Equations (21) and (22); specific heat c_p in Equation (24); cell state c_t of the long short-term memory in Equation (70); empirical coverage $\hat{c}$ in Equations (91) and (92).	Every use carries a subscript or, for the coverage, a circumflex.

Table A2. *Cont.*

Symbol	Distinct Meanings and the Equation That Fixes Each	How the Ambiguity is Resolved
D	Solid-phase diffusivity D_{s} in Equations (21) and (22); Kullback–Leibler divergence D_{KL} in Equations (38) and (60); discriminator of the adversarial objective in Equation (62); observed data $\mathcal{D}$ in Equations (57) and (81); source and target samples $\mathcal{D}_{\mathrm{s}}$ and $\mathcal{D}_{\mathrm{t}}$ in Equation (93).	The data sets are set in script type and the diffusivity in italic; the divergence carries a subscript, and the discriminator occurs only in Section 6.5.
d	Dimension of the state space in Section 3; difference between paired ranks d_{i} in Equation (40); key dimension d_{k} of the attention operator in Equation (72).	The two indexed uses are subscripted and occur in different sections; the state dimension appears only as the exponent of $\mathbb{R}$ in Section 3.
E	Activation energy E_{a} in Equations (10) and (11); expectation operator in Equations (43), (60) and (62); energy function of the restricted Boltzmann machine in Equation (61).	The activation energy is subscripted; the expectation is set in blackboard type with a bracketed argument, and the energy function is written with its two arguments.
e	Base of the natural logarithm in Equations (16) and (64); regression residual e_{i} of the least-squares support-vector formulation in Equation (49).	The residual is subscripted.
F	Faraday constant in Equations (20) and (22); boosted ensemble F_{M} and its stages in Equations (51) and (52); transition matrix or Jacobian F_{k} in Equations (74) and (77); function class $\mathcal{F}$ in Equation (44).	The ensemble is written with its argument or stage subscript, except at the evaluation point $F = F_{m-1}$ of Equation (52), the transition matrix with the time index, and the function class in script type; elsewhere the bare symbol is the Faraday constant.
f	Densities of the first-passage time and of the Gamma increment in Equations (14) and (16); regression function in Equations (43), (44), (47) and (53); state-transition map f_{k} in Equations (73) and (77); encoder f_{θ} in Equations (58) and (59); forget gate f_{t} in Equation (70).	Distinguished by subscript and argument; the densities are subscripted by their random variable, and the bare symbol is the regression function.
G	Generator of the adversarial objective in Equation (62); set of groups in Equation (92).	The generator occurs only in Section 6.5 and the set of groups only in Section 10.2.
g	Discrete Gaussian weights g_{j} in Equation (28); decoder g_{ϕ} in Equations (58) and (59); group index in Equation (92).	The weights and the decoder are subscripted; the bare symbol is the group index and occurs only in Section 10.2.
H	Hidden-layer output matrix of the extreme learning machine in Equation (66); observation matrix or Jacobian H_{k} in Equations (75)–(77); reproducing kernel Hilbert space $\mathcal{H}$ in Equation (93).	The observation matrix carries the time index and the Hilbert space is set in script type.
h	Convective heat-transfer coefficient in Equation (24); base learners h_{k} and h_{m} in Equations (50) and (51); hidden-unit vector of the restricted Boltzmann machine in Equation (61); recurrent hidden state h_{t} in Equations (69)–(71); observation function h_{k} in Equations (73) and (77).	Distinguished by argument and subscript: $h_k(x;\Theta_k)$ is a tree of the random forest, $h_m(x)$ a boosting stage, $h_k(x_k)$ an observation function and h_t a recurrent state; the bare symbol is a coefficient in Section 4.3 and the hidden-unit vector in Section 6.5.
I	Current in Equations (1), (5), (6), (17)–(19), (23) and (35); mutual information in Equation (42); identity matrix in Equations (54), (55) and (76).	The current is written with its time argument or as a factor in the circuit relations; the mutual information with its two random arguments; the identity matrix only inside matrix expressions.
i	Input gate i_{t} of the long short-term memory in Equation (70); otherwise a summation or sample index.	The gate carries the time subscript.
j	Interfacial and exchange current density j and j_0 in Equations (20) and (22); otherwise a summation index.	The current density occurs only in Section 4.3, where j is not used as an index.
K	Number of intrinsic mode functions in Equation (30); number of trees in Equation (50); covariance matrix in Equations (54) and (55); attention key matrix in Equation (72); Kalman gain K_{k} in Equations (75) and (76).	Distinguished by subscript and section; the kernel function itself is written in lower case, as $k(\cdot,\cdot)$.

Table A2. *Cont.*

Symbol	Distinct Meanings and the Equation That Fixes Each	How the Ambiguity is Resolved
k	Sample index within a cycle in Equations (4) and (25); rate constant $k(T)$ in Equation (10); kernel or covariance function in Equations (47), (48) and (53)–(56) and test covariance vector in Equations (54) and (55); time step of the sequential estimators in Equations (73)–(80); sequence-point, tree and iteration index in Equations (41), (50) and (65).	The rate constant and the kernel are written with their arguments and the test covariance vector with the subscript *; the remaining uses are indices, fixed by the section in which they occur.
L, l	Gaussian half-window L in Equation (28); loss functionals in Equations (38), (52), (59), (60), (65) and (94); lower prediction-interval bound L_{i} in Equations (89) and (90); layer index l in Equations (63), (67) and (68); kernel length scale ℓ, set in script type, in Equations (48) and (56).	The loss functionals carry an identifying subscript or an argument; the interval bound is indexed by sample; the length scale is set in script type; the layer index appears only as an unparenthesized superscript, the parenthesized superscript (i) being reserved for the particle index of Equations (78)–(80).
M	Number of boosting stages in Equation (51); number of candidate models in Equation (57), in which the models themselves are written $\mathcal{M}_{\mathrm{m}}$ in script type; number of nominal levels in Equation (91).	Separated by section: Sections 6.3, 6.4 and 10.2, respectively; the models are set in script type and carry the subscript m.
m	Cell mass in Equation (24); embedding dimension of the sample-entropy statistic in Equation (34); stage index of the boosting expansion in Equations (51) and (52); prior mean function $m(x)$ of the Gaussian process in Equation (53); model index in Equation (57); visible and hidden unit counts m_{v} and m_{h} in Equations (38) and (61); nominal-level index in Equation (91).	The mean function is written with its argument and the unit counts with subscripts; the mass carries physical units and occurs only in Section 4.3; the remaining uses are indices or dimensions fixed by section.
N	Generic sample count and the counts N_{w}, N_{s}, N_{p}, N_{eff}, N_{d} and N_{c} in Equations (27), (31)–(34), (39)–(41), (44), (46), (47), (49), (59), (78), (80), (83)–(86), (88)–(90) and (94); the Gaussian distribution $\mathcal{N}$ in Equation (73); the differential operator $\mathcal{N}[\cdot]$ in Equation (94).	The counts are subscripted to identify the set counted, except the generic sample count; the Gaussian distribution and the differential operator are both set in script type and are distinguished by their arguments, $\mathcal{N}(0, \cdot)$ with two parameters in parentheses and $\mathcal{N}[\cdot]$ with a bracketed function.
n	Cycle index in Equations (2)–(5), (7), (9) and (12), with the failure index n_{f} in Equations (8) and (9); source and target sample counts n_{s} and n_{t} in Equation (93).	The bare symbol always denotes the cycle index; the failure index and the sample counts carry subscripts, and the sample counts are used only in Section 11.2.
P	Error covariance P_{k} in Equations (74)–(76); probability measure in the survival expression that follows Equation (16).	The covariance carries the time index; the probability is written with its argument.
p	Dimension of the measurement vector in Section 3; probability density or mass $p(\cdot)$ in Equations (42), (57), (60), (62), (78), (79) and (81).	The dimension appears only as the exponent of $\mathbb{R}$ in Section 3; the densities are written with their arguments, except the distributions p_{data} and p_z of Equation (62), which carry identifying subscripts.
Q	Delivered capacity in Equation (1) and the charge accumulated up to sample i in Equations (36) and (37); available charge Q_{avail} in Equation (4); capacity loss Q_{loss} in Equation (11); attention query matrix in Equation (72); process-noise covariance Q_{k} in Equations (73) and (74).	The delivered capacity is bare in Equation (1) and the accumulated charge carries a sample argument in Equations (36) and (37); the other capacities and the covariance carry a subscript or the time index; the query matrix occurs only in Equation (72).
q	Heat-generation rate, written with an overdot, in Equations (23) and (24), with its volumetric counterpart q''' in the text that follows; variational posterior in Equation (60); proposal density in Equation (79).	The heat-generation rates carry an overdot or a triple prime; the densities are written with their arguments.

Table A2. *Cont.*

Symbol	Distinct Meanings and the Equation That Fixes Each	How the Ambiguity is Resolved
R	Internal, ohmic, polarization and reference resistances in Equations (3), (6) and (17)–(19); gas constant R_{g} in Equations (10), (11) and (20); particle radius R_{p} in Equation (22); expected and empirical risk in Equation (44), together with the empirical Rademacher complexity $\Re_{\mathrm{N}}(\mathcal{F})$, set in fraktur type; pooling window R_{i} in Equation (68); measurement-noise covariance R_{k} in Equations (73) and (75); coefficient of determination R^2 in Equation (88).	All uses other than the risk carry a subscript, a superscript or the cycle argument; the risk is written with its argument, as $R(f)$, the Rademacher complexity is set in fraktur type, and the coefficient of determination occurs only in Section 10.1.
r	Radial coordinate in Equations (21) and (22); residual r_{K} of the empirical mode decomposition in Equation (30); sample-entropy tolerance in Equation (34); Spearman coefficient r_{s} in Equation (40); pseudo-residual r_{im} in Equation (52); reset gate r_{t} in Equation (71).	Distinguished by subscript and section; the bare symbol is the radial coordinate in Section 4.3 or the entropy tolerance in Section 5.3.
T	Absolute temperature in Equations (10), (11), (20), (23) and (24), with the ambient temperature T_{amb} in Equation (24); first-passage time T_{f} in Equations (14) and (15).	The bare symbol always denotes temperature; the first-passage time carries the subscript f, as does the failure index n_{f}, and the discharge duration of Equation (1) is written t_{d}.
t	Continuous time in Sections 3–5, with the particular instants t_0, t_{c}, t_{d}, t_1, t_2 and t_k; discrete step index in Equations (69)–(71).	The step index occurs only as the subscript of the recurrent quantities of Section 7.2, where it is stated; elsewhere the bare symbol is continuous time, and as a subscript or superscript t is a label, as in V_{t}, $\mathcal{D}_{\mathrm{t}}$, n_{t} and x_j^{t}.
V	Terminal, open-circuit, polarization and sampled voltages in Equations (6), (17)–(19), (23), (31) and (35)–(37); cell volume V_{cell} in the text following Equation (24); variance operator $\mathbb{V}$ in Equation (43); value function of the adversarial objective in Equation (62); attention value matrix in Equation (72).	The voltages are subscripted, written with a time or sample argument, or appear in a differential; the cell volume carries the subscript cell; the variance is set in blackboard type and the value function is written with its arguments; the value matrix occurs only in Equation (72).
v	Visible units of the restricted Boltzmann machine in Equation (61); pooled output in Equation (68); measurement noise v_{k} in Equation (73).	Distinguished by section and subscript.
W	Wavelet transform $W_{\mathrm{x}}(a, b)$ in Equation (29); weight matrices in Equations (63) and (69)–(71).	The transform carries the signal subscript and its two arguments; the weight matrices carry a layer superscript or a gate or connection subscript, except the candidate-state matrix of Equation (71), which occurs only in Section 7.2.
w	Failure threshold in Equations (14) and (15) and in the caption of Figure 14; weight vector of the support-vector formulations in Equations (46) and (49); coupling weights w_{ij} of the restricted Boltzmann machine in Equation (61); convolution kernel weights $w_{\mathrm{j}}^{\mathrm{l}}$ in Equation (67); process noise w_{k} in Equation (73).	The moving-average window length is written N_{w}. The threshold is used only in Section 4.2 and in the caption of Figure 14, and the weight vector only in Section 6.2; the coupling and convolution weights carry subscripts, the latter also a layer superscript, and the process noise occurs only in Section 8.
X	State space in Section 3; degradation path $X(t)$ in Equations (13) and (16); subscript of the correlation coefficient in Equation (39); random variable in Equation (42); matrix of training inputs, as the design matrix in Equation (45) and in Section 6.4.	The degradation path is written with its time argument and the state space occurs only in Section 3; in Section 5.4, X and Y denote the pair of variables compared in Equations (39) and (42), and the matrix of training inputs occurs only in Section 6.

Table A2. *Cont.*

Symbol	Distinct Meanings and the Equation That Fixes Each	How the Ambiguity is Resolved
x	Latent degradation state $x(n)$ in Section 3; initial value $x_0 = X(0)$ of the degradation path in Equations (13)–(15), with current value x_c in the text following Equation (15); argument of the increment density in Equation (16); signal or feature value in Equations (25)–(30), (32) and (33), including the moving average $\bar{x}(i)$ of Equation (27); paired observations x_i and their sample mean $\bar{x}$ in Equation (39); reference and compared sequences x_0 and x_i in Equation (41); model input in Sections 6 and 7; data sample in Equations (60) and (62); filter state x_k in Equations (73), (74) and (76)–(79); source, target and collocation samples in Equations (93) and (94).	The degradation state is written with the cycle argument and the filter state with the time index. The initial state x_0 is used only in Section 4.2 and in the caption of Figure 14, and the reference sequence x_0 only in Section 5.4; the moving average carries its sample argument, whereas the sample mean of Equation (39) does not. In Section 5.4, x_i is a paired value in Equation (39) and a candidate sequence in Equation (41), as stated there. Elsewhere x is the argument of the operation under discussion, fixed by the section.
Y	Subscript of the correlation coefficient in Equation (39); random variable in Equation (42); target matrix of the extreme learning machine in Equation (66); measured target $Y(i)$ and its mean $\bar{Y}$ in Equations (82)–(89).	The measured target carries the sample argument; in Section 5.4, X and Y denote the pair of variables compared in Equations (39) and (42), and the target matrix occurs only in Section 7.
z	Fade exponent in Equation (11); latent code of the autoencoders in Equations (58) and (60); noise input of the generator in Equation (62); argument of the activation functions in Equation (64); pre-activation in Equation (67); update gate z_t in Equation (71); measurement $z(n)$ and z_k in Section 3 and Equations (73), (76), (78) and (79).	Distinguished by subscript and section; the measurement appears as $z(n)$ or z_k throughout Sections 3 and 8.
α, β	Capacity-fade coefficients in Equation (12); Gamma shape coefficient and rate parameter α_Γ and β_Γ in Equation (16); anodic and cathodic charge-transfer coefficients α_a and α_c in Equation (20); elastic-net coefficient vector in Equation (45); dual variables α_i of the support-vector expansion in Equation (47); output weights of the extreme learning machine in Equation (66); nominal non-coverage level in Equation (92).	The Gamma parameters carry the subscript Γ and the transfer coefficients the subscripts a and c; the remaining uses are separated by section and by the objects they multiply.
γ	C-rate weighting in Equation (11); lower incomplete gamma function in the survival expression following Equation (16); grey relational grade in Equation (41); step size γ_m in Equation (51).	Distinguished by argument and subscript; the grade and the incomplete gamma function are written with their arguments, and the step size with its index.
Δ	Increment operator in Equation (16); grey relational deviation $\Delta_i(k)$ and its extremes Δ_{min} and Δ_{max} in Equation (41); worst-group coverage deviation Δ_{cov} in Equation (92).	The deviations carry subscripts; the bare symbol is the increment operator.
δ	Error of a reconstructed or inferred quantity, δV and δSOC, in Equation (6); confidence parameter of the generalization bound in Equation (44); Dirac delta in Equation (78).	The errors are written as prefixes to the quantity and the Dirac delta with its argument; the bare symbol is the confidence parameter.
ε	Error of the ampere-hour estimate ε_{SOC} in Equation (7); observation-noise variance $\sigma_\varepsilon{}^2$ in Equations (43), (54) and (55); insensitivity width of the support-vector formulation stated with Equation (46).	The estimate error is subscripted and the noise appears only as a subscript; the bare symbol is the insensitivity width.
η	Coulombic efficiency in Equations (5) and (7); surface overpotential η_{ov} in Equation (20); learning rate in Equation (65).	The overpotential is subscripted; the efficiency and the learning rate occur in Sections 3 and 7, respectively.
θ	Parameters of the time transformation in Equation (13); of the encoder in Equations (58) and (59); of the network in Equation (65); and of the Bayesian model in Equation (81).	A single symbol is retained deliberately, since each use is the parameter vector of the model under discussion; the model is named at each occurrence.
λ	Sparsity weight in Equation (38); lasso and ridge penalty weights λ_1 and λ_2 in Equation (45); regularization constant of the least-squares support-vector formulation, which weights the squared residuals, in Equation (49); physics-loss weight λ_p in Equation (94).	Distinguished by subscript, except in Equations (38) and (49), which occur in different sections: in Equation (38) λ weights the sparsity penalty, and in Equation (49) it weights the squared residuals against the regularizer $\frac{1}{2}\|w\|^2$.

Table A2. *Cont.*

Symbol	Distinct Meanings and the Equation That Fixes Each	How the Ambiguity is Resolved
μ	Drift of the Wiener process in Equations (13)–(15) and in the caption of Figure 14; sample mean μ_{x} in Equations (26) and (33); posterior mean μ_* of the Gaussian process in Equation (54).	The sample and posterior means carry a subscript; the bare symbol is the drift and occurs only in Section 4.2 and in the caption of Figure 14.
ρ	Target and observed mean activation in Equation (38); Pearson correlation coefficient in Equation (39).	The correlation carries the subscript of its two variables.
σ	Diffusion coefficient σ_{B} in Equations (13)–(15); sample standard deviation σ_{x} in Equations (26) and (33); Gaussian smoothing width, in samples, in the weights defined after Equation (28); observation-noise variance ${\sigma_\varepsilon}^2$ in Equations (43), (54) and (55); Gaussian-process signal variance ${\sigma_{\mathrm{f}}}^2$ and posterior variance ${\sigma_*}^2$ in Equations (55) and (56); logistic sigmoid $\sigma(\cdot)$ in Equations (63), (64) and (69)–(71).	Every use other than the smoothing width carries a subscript or an argument; the smoothing width is bare and occurs only in Section 5.2, and in the equations the sigmoid is always written with its argument.
τ	Integration variable in Equation (5); nominal coverage levels τ_{m} in Equation (91).	The levels carry the subscript m; the integration variable is a dummy variable confined to Equation (5).
ϕ, φ	Decoder parameters of the autoencoder in Equations (58) and (59); feature map of the support-vector kernel in Section 6.2 and of the kernel embedding in Equation (93).	The two glyphs are distinct and are used consistently: ϕ denotes the decoder parameters and φ the feature map.

References

1. Zhang, Y.; Li, Y.-F. Prognostics and health management of Lithium-ion battery using deep learning methods: A review. *Renew. Sustain. Energy Rev.* **2022**, *161*, 112282. [CrossRef]
2. Sebbagh, T.; Manfo, T.A.; Şahin, M.E. Advances in Battery Technologies for Next-Generation Energy Storage Systems. *Electronics* **2026**, *15*, 690. [CrossRef]
3. Chen, T.; Jin, Y.; Lv, H.; Yang, A.; Liu, M.; Chen, B.; Xie, Y.; Chen, Q. Applications of Lithium-Ion Batteries in Grid-Scale Energy Storage Systems. *Trans. Tianjin Univ.* **2020**, *26*, 208–217. [CrossRef]
4. Anik, M.A.H.C.; Bin Belal, T.M.A.I.; Islam, M.M.; Sharif, A. A Comprehensive Review of Next-Generation Grid-Scale Energy Storage Technologies and Their Role in a Sustainable Energy Future. *J. Control Sci. Eng.* **2025**, *2025*, 5577507. [CrossRef]
5. Redhwi, I.; Almalki, S.; Radhwi, H.; Jarwali, A.; Abaaltahin, A.; Fallatah, A. Advancements in battery technology: A comprehensive review. *Clean Energy Sci. Technol.* **2026**, *4*, 573. [CrossRef]
6. Guo, J.; Li, Y.; Pedersen, K.; Stroe, D.-I. Lithium-ion battery operation, degradation, and aging mechanism in electric vehicles: An overview. *Energies* **2021**, *14*, 5220. [CrossRef]
7. Feng, X.; He, X.; Ouyang, M.; Wang, L.; Lu, L.; Ren, D.; Santhanagopalan, S. A coupled electrochemical-thermal failure model for predicting the thermal runaway behavior of lithium-ion batteries. *J. Electrochem. Soc.* **2018**, *165*, A3748–A3765. [CrossRef]
8. Ghalkhani, M.; Bahiraei, F.; Nazri, G.-A.; Saif, M. Electrochemical–thermal model of pouch-type lithium-ion batteries. *Electrochim. Acta* **2017**, *247*, 569–587. [CrossRef]
9. Feng, X.; Ouyang, M.; Liu, X.; Lu, L.; Xia, Y.; He, X. Thermal runaway mechanism of lithium ion battery for electric vehicles: A review. *Energy Storage Mater.* **2018**, *10*, 246–267. [CrossRef]
10. Zio, E. Data-driven prognostics and health management (PHM) for predictive maintenance of industrial components and systems. In *Risk-Informed Methods and Applications in Nuclear and Energy Engineering: Modeling, Experimentation, and Validation*; Smith, C.L., Mandelli, D., Le Blanc, K., Eds.; Elsevier: Amsterdam, The Netherlands, 2024; pp. 113–137. [CrossRef]
11. Kong, J.; Liu, J.; Zhu, J.; Zhang, X.; Tsui, K.-L.; Peng, Z.; Wang, D. Review on Lithium-ion Battery PHM from the Perspective of Key PHM Steps. *Chin. J. Mech. Eng.* **2024**, *37*, 71. [CrossRef]
12. Dini, P.; Colicelli, A.; Saponara, S. Review on modeling and SOC/SOH estimation of batteries for automotive applications. *Batteries* **2024**, *10*, 34. [CrossRef]
13. Manoj, L.; Wang, P. Distributed Battery Prognostics via Quantized Edge Encoding and Cloud-Based Feature Decoding. In Proceedings of the 2026 Annual Reliability and Maintainability Symposium (RAMS), Miramar Beach, FL, USA, 19–22 January 2026; pp. 1–6. [CrossRef]
14. Saxena, S.; Xing, Y.; Pecht, M.G. PHM of Li-ion Batteries. In *Prognostics and Health Management of Electronics: Fundamentals, Machine Learning, and the Internet of Things*; Pecht, M.G., Kang, M., Eds.; Wiley: Hoboken, NJ, USA, 2018; pp. 349–375. [CrossRef]

15. Ruiz, P.L.; Damianakis, N.; Mouli, G.R.C. Physics-based and data-driven modeling of degradation mechanisms for lithium-ion batteries—A review. *IEEE Access* **2025**, *13*, 21164–21189. [CrossRef]
16. Seals, D.; Ramesh, P.; D'Arpino, M.; Canova, M. Physics-based equivalent circuit model for lithium-ion cells via reduction and approximation of electrochemical model. *SAE Int. J. Adv. Curr. Pract. Mobil.* **2022**, *4*, 1154–1165. [CrossRef]
17. Xiao, Y.; Wen, J.; Yao, L.; Zheng, J.; Fang, Z.; Shen, Y. A comprehensive review of the lithium-ion battery state of health prognosis methods combining aging mechanism analysis. *J. Energy Storage* **2023**, *65*, 107347. [CrossRef]
18. Zhu, J.; Dewi Darma, M.S.; Knapp, M.; Sørensen, D.R.; Heere, M.; Fang, Q.; Wang, X.; Dai, H.; Mereacre, L.; Senyshyn, A.; et al. Investigation of lithium-ion battery degradation mechanisms by combining differential voltage analysis and alternating current impedance. *J. Power Sources* **2020**, *448*, 227575. [CrossRef]
19. Yang, Q.; Xu, J.; Li, X.; Xu, D.; Cao, B. State-of-health estimation of lithium-ion battery based on fractional impedance model and interval capacity. *Int. J. Electr. Power Energy Syst.* **2020**, *119*, 105883. [CrossRef]
20. Melo, A.; Câmara, M.M.; Pinto, J.C. Data-driven process monitoring and fault diagnosis: A comprehensive survey. *Processes* **2024**, *12*, 251. [CrossRef]
21. Rauf, H.; Khalid, M.; Arshad, N. Machine learning in state of health and remaining useful life estimation: Theoretical and technological development in battery degradation modelling. *Renew. Sustain. Energy Rev.* **2022**, *156*, 111903. [CrossRef]
22. Shi, J.; Rivera, A.; Wu, D. Battery health management using physics-informed machine learning: Online degradation modeling and remaining useful life prediction. *Mech. Syst. Signal Process.* **2022**, *179*, 109347. [CrossRef]
23. Tang, X.; Liu, K.; Li, K.; Widanage, W.D.; Kendrick, E.; Gao, F. Recovering large-scale battery aging dataset with machine learning. *Patterns* **2021**, *2*, 100302. [CrossRef] [PubMed]
24. Khaleghi, S.; Hosen, M.S.; Van Mierlo, J.; Berecibar, M. Towards machine-learning driven prognostics and health management of Li-ion batteries. A comprehensive review. *Renew. Sustain. Energy Rev.* **2024**, *192*, 114224. [CrossRef]
25. Tong, Z.; Miao, J.; Tong, S.; Lu, Y. Early prediction of remaining useful life for lithium-ion batteries based on a hybrid machine learning method. *J. Clean. Prod.* **2021**, *317*, 128265. [CrossRef]
26. Zhao, J.; Feng, X.; Pang, Q.; Wang, J.; Lian, Y.; Ouyang, M.; Burke, A.F. Battery prognostics and health management from a machine learning perspective. *J. Power Sources* **2023**, *581*, 233474. [CrossRef]
27. Huo, Q.; Ma, Z.; Zhao, X.; Zhang, T.; Zhang, Y. Bayesian network based state-of-health estimation for battery on electric vehicle application and its validation through real-world data. *IEEE Access* **2021**, *9*, 11328–11341. [CrossRef]
28. Malekloo, A.; Ozer, E.; AlHamaydeh, M.; Girolami, M. Machine learning and structural health monitoring overview with emerging technology and high-dimensional data source highlights. *Struct. Health Monit.* **2022**, *21*, 1906–1955. [CrossRef]
29. Mawonou, K.S.R.; Eddahech, A.; Dumur, D.; Beauvois, D.; Godoy, E. State-of-health estimators coupled to a random forest approach for lithium-ion battery aging factor ranking. *J. Power Sources* **2021**, *484*, 229154. [CrossRef]
30. Sun, X.; Zhong, K.; Han, M. A hybrid prognostic strategy with unscented particle filter and optimized multiple kernel relevance vector machine for lithium-ion battery. *Measurement* **2021**, *170*, 108679. [CrossRef]
31. Sui, X.; He, S.; Vilsen, S.B.; Meng, J.; Teodorescu, R.; Stroe, D.I. A review of non-probabilistic machine learning-based state of health estimation techniques for Lithium-ion battery. *Appl. Energy* **2021**, *300*, 117346. [CrossRef]
32. Tagade, P.; Hariharan, K.S.; Ramachandran, S.; Khandelwal, A.; Naha, A.; Kolake, S.M.; Han, S.H. Deep Gaussian process regression for lithium-ion battery health prognosis and degradation mode diagnosis. *J. Power Sources* **2020**, *445*, 227281. [CrossRef]
33. Hussain, M.; O'Nils, M.; Lundgren, J.; Mousavirad, S.J. A comprehensive review on deep learning-based data fusion. *IEEE Access* **2024**, *12*, 180093–180124. [CrossRef]
34. Xie, J.; Xu, J.; Wei, Z.; Li, X. Fault isolating and grading for Li-ion battery packs based on pseudo images and convolutional neural network. *Energy* **2023**, *263*, 125867. [CrossRef]
35. Ma, Y.; Shan, C.; Gao, J.; Chen, H. Multiple health indicators fusion-based health prognostic for lithium-ion battery using transfer learning and hybrid deep learning method. *Reliab. Eng. Syst. Saf.* **2023**, *229*, 108818. [CrossRef]
36. Xu, F.; Yang, F.; Fei, Z.; Huang, Z.; Tsui, K.L. Life prediction of lithium-ion batteries based on stacked denoising autoencoders. *Reliab. Eng. Syst. Saf.* **2021**, *208*, 107396. [CrossRef]
37. Li, Y.; Li, K.; Liu, X.; Wang, Y.; Zhang, L. Lithium-ion battery capacity estimation-a pruned convolutional neural network approach assisted with transfer learning. *Appl. Energy* **2021**, *285*, 116410. [CrossRef]
38. Tian, J.; Xiong, R.; Shen, W.; Lu, J.; Yang, X.G. Deep neural network battery charging curve prediction using 30 points collected in 10 min. *Joule* **2021**, *5*, 1521–1534. [CrossRef]
39. Ding, P.; Liu, X.; Li, H.; Huang, Z.; Zhang, K.; Shao, L.; Abedinia, O. Useful life prediction based on wavelet packet decomposition and two-dimensional convolutional neural network for lithium-ion batteries. *Renew. Sustain. Energy Rev.* **2021**, *148*, 111287. [CrossRef]
40. Qian, C.; Xu, B.; Chang, L.; Sun, B.; Feng, Q.; Yang, D.; Ren, Y.; Wang, Z. Convolutional neural network based capacity estimation using random segments of the charging curves for lithium-ion batteries. *Energy* **2021**, *227*, 120333. [CrossRef]

41. Bhattacharjee, A.; Verma, A.; Mishra, S.; Saha, T.K. Estimating state of charge for xEV batteries using 1D convolutional neural networks and transfer learning. *IEEE Trans. Veh. Technol.* **2021**, *70*, 3123–3135. [CrossRef]
42. Ding, T.; von Jouanne, A.; Dong, L.; Fang, X.; Fang, T.; Rivas, P.; Yokochi, A. A Review of AI Applications in Lithium-Ion Batteries: From State-of-Health Estimations to Prognostics. *Energies* **2026**, *19*, 562. [CrossRef]
43. Sulzer, V.; Mohtat, P.; Aitio, A.; Lee, S.; Yeh, Y.T.; Steinbacher, F.; Khan, M.U.; Lee, J.W.; Siegel, J.B.; Stefanopoulou, A.G.; et al. The challenge and opportunity of battery lifetime prediction from field data. *Joule* **2021**, *5*, 1934–1955. [CrossRef]
44. Perez-Cerrolaza, J.; Abella, J.; Borg, M.; Donzella, C.; Cerquides, J.; Cazorla, F.J.; Englund, C.; Tauber, M.; Nikolakopoulos, G.; Flores, J.L. Artificial intelligence for safety-critical systems in industrial and transportation domains: A survey. *ACM Comput. Surv.* **2024**, *56*, 1–40. [CrossRef]
45. Huang, X.; Kim, H.S. Advancing battery prognostics and health management: Challenges and future perspectives. *J. Mech. Sci. Technol.* **2026**, *40*, 2405–2415. [CrossRef]
46. Sohail, M.; Tanveer, M.; Kim, H.S. Trustworthy Machine Learning and Mathematical Modelling for Lithium-Ion Battery State-of-Health Estimation. *Mathematics* **2026**, *14*, 1879. [CrossRef]
47. Martí-Florences, M.; Cecilia, A.; Costa-Castelló, R. Modelling and Estimation in Lithium-Ion Batteries: A Literature Review. *Energies* **2023**, *16*, 6846. [CrossRef]
48. Thelen, A.; Huan, X.; Paulson, N.; Onori, S.; Hu, Z.; Hu, C. Probabilistic machine learning for battery health diagnostics and prognostics—Review and perspectives. *npj Mater. Sustain.* **2024**, *2*, 14. [CrossRef]
49. Dong, G.; Chen, Z.; Wei, J.; Ling, Q. Battery health prognosis using Brownian motion modeling and particle filtering. *IEEE Trans. Ind. Electron.* **2018**, *65*, 8646–8655. [CrossRef]
50. Li, W.; Sengupta, N.; Dechent, P.; Howey, D.; Annaswamy, A.; Sauer, D.U. Online capacity estimation of lithium-ion batteries with deep long short-term memory networks. *J. Power Sources* **2021**, *482*, 228863. [CrossRef]
51. Hoque, M.A.; Nurmi, P.; Kumar, A.; Varjonen, S.; Song, J.; Pecht, M.G.; Tarkoma, S. Data driven analysis of lithium-ion battery internal resistance towards reliable state of health prediction. *J. Power Sources* **2021**, *513*, 230519. [CrossRef]
52. Bian, X.; Wei, Z.; He, J.; Yan, F.; Liu, L. A novel model-based voltage construction method for robust state-of-health estimation of lithium-ion batteries. *IEEE Trans. Ind. Electron.* **2021**, *68*, 12173–12184. [CrossRef]
53. Hong, J.; Wang, Z.; Chen, W.; Wang, L.; Lin, P.; Qu, C. Online accurate state of health estimation for battery systems on real-world electric vehicles with variable driving conditions considered. *J. Clean. Prod.* **2021**, *294*, 125814. [CrossRef]
54. Jia, J.; Wang, K.; Shi, Y.; Wen, J.; Pang, X.; Zeng, J. A multi-scale state of health prediction framework of lithium-ion batteries considering the temperature variation during battery discharge. *J. Energy Storage* **2021**, *42*, 103076. [CrossRef]
55. Zhang, M.; Yang, D.; Du, J.; Sun, H.; Li, L.; Wang, L.; Wang, K. A review of SOH prediction of Li-ion batteries based on data-driven algorithms. *Energies* **2023**, *16*, 3167. [CrossRef]
56. Zhang, S.; Zhang, X. A novel non-experiment-based reconstruction method for the relationship between open-circuit-voltage and state-of-charge/state-of-energy of lithium-ion battery. *Electrochim. Acta* **2022**, *403*, 139637. [CrossRef]
57. Knap, V.; Stroe, D.I. Effects of open-circuit voltage tests and models on state-of-charge estimation for batteries in highly variable temperature environments: Study case nano-satellites. *J. Power Sources* **2021**, *498*, 229913. [CrossRef]
58. Bian, C.; He, H.; Yang, S. Stacked bidirectional long short-term memory networks for state-of-charge estimation of lithium-ion batteries. *Energy* **2020**, *191*, 116538. [CrossRef]
59. Xu, C.; Cleary, T.; Wang, D.-W.; Li, G.; Rahn, C.; Wang, D.-H.; Rajamani, R.; Fathy, H.K. Online state estimation for a physics-based lithium-sulfur battery model. *J. Power Sources* **2021**, *489*, 229495. [CrossRef]
60. Shateri, N.; Shi, Z.; Auger, D.J.; Fotouhi, A. Lithium-sulfur cell state of charge estimation using a classification technique. *IEEE Trans. Veh. Technol.* **2021**, *70*, 212–224. [CrossRef]
61. Etxandi-Santolaya, M.; Canals Casals, L.; Corchero, C. Extending the electric vehicle battery first life: Performance beyond the current end of life threshold. *Heliyon* **2024**, *10*, e26066. [CrossRef] [PubMed]
62. Kong, J.-Z.; Wang, D.; Yan, T.; Zhu, J.; Zhang, X. Accelerated stress factors based nonlinear Wiener process model for lithium-ion battery prognostics. *IEEE Trans. Ind. Electron.* **2022**, *69*, 11665–11674. [CrossRef]
63. Miguel, E.; Plett, G.L.; Trimboli, M.S.; Lopetegi, I.; Oca, L.; Iraola, U.; Bekaert, E. Electrochemical model and sigma point Kalman filter based online oriented battery model. *IEEE Access* **2021**, *9*, 98072–98090. [CrossRef]
64. Zhang, H.; Mo, Z.; Wang, J.; Miao, Q. Nonlinear-drifted fractional Brownian motion with multiple hidden state variables for remaining useful life prediction of lithium-ion batteries. *IEEE Trans. Reliab.* **2020**, *69*, 768–780. [CrossRef]
65. Ahwiadi, M.; Wang, W. Battery Health Monitoring and Remaining Useful Life Prediction Techniques: A Review of Technologies. *Batteries* **2025**, *11*, 31. [CrossRef]
66. Omri, N.; Al Masry, Z.; Mairot, N.; Giampiccolo, S.; Zerhouni, N. Industrial data management strategy towards an SME-oriented PHM. *J. Manuf. Syst.* **2020**, *56*, 23–36. [CrossRef]

67. Madani, S.S.; Shabeer, Y.; Fowler, M.; Panchal, S.; Ziebert, C.; Chaoui, H.; Allard, F. Digital twin technologies for battery systems: Advancements, applications, and future directions. *Front. Batter. Electrochem.* **2026**, *5*, 1764210. [CrossRef]
68. Lyu, C.; Song, Y.; Zheng, J.; Luo, W.; Hinds, G.; Li, J.; Wang, L. In situ monitoring of lithium-ion battery degradation using an electrochemical model. *Appl. Energy* **2019**, *250*, 685–696. [CrossRef]
69. Downey, A.; Lui, Y.-H.; Hu, C.; Laflamme, S.; Hu, S. Physics-based prognostics of lithium-ion battery using non-linear least squares with dynamic bounds. *Reliab. Eng. Syst. Saf.* **2019**, *182*, 1–12. [CrossRef]
70. Peng, M.; Shen, W.J.; Luo, Z.D. Two-dimensional thermal modeling for laminated lithium ion battery. *Chin. J. Power Sources* **2018**, *42*, 1312–1315. (In Chinese) [CrossRef]
71. Tang, K.; Luo, B.; Chen, D.; Wang, C.; Chen, L.; Li, F.; Cao, Y.; Wang, C. The State of Health Estimation of Lithium-Ion Batteries: A Review of Health Indicators, Estimation Methods, Development Trends and Challenges. *World Electr. Veh. J.* **2025**, *16*, 429. [CrossRef]
72. Dong, G.; Yang, F.; Wei, Z.; Wei, J.; Tsui, K.-L. Data-driven battery health prognosis using adaptive Brownian motion model. *IEEE Trans. Ind. Inform.* **2020**, *16*, 4736–4746. [CrossRef]
73. Qiu, X.; Wu, W.; Wang, S. Remaining useful life prediction of lithium-ion battery based on improved cuckoo search particle filter and a novel state of charge estimation method. *J. Power Sources* **2020**, *450*, 227700. [CrossRef]
74. Gao, D.; Zhou, Y.; Wang, T.; Wang, Y. A method for predicting the remaining useful life of lithium-ion batteries based on particle filter using Kendall rank correlation coefficient. *Energies* **2020**, *13*, 4183. [CrossRef]
75. Zheng, L.; Zhu, J.; Wang, G.; Lu, D.D.-C.; He, T. Differential voltage analysis based state of charge estimation methods for lithium-ion batteries using extended Kalman filter and particle filter. *Energy* **2018**, *158*, 1028–1037. [CrossRef]
76. Doyle, M.; Fuller, T.F.; Newman, J. Modeling of galvanostatic charge and discharge of the lithium/polymer/insertion cell. *J. Electrochem. Soc.* **1993**, *140*, 1526–1533. [CrossRef]
77. Bernardi, D.; Pawlikowski, E.; Newman, J. A general energy balance for battery systems. *J. Electrochem. Soc.* **1985**, *132*, 5–12. [CrossRef]
78. Yu, H.; Zhang, H.; Zhang, Z.; Yang, S. State estimation of lithium-ion batteries via physics-machine learning combined methods: A methodological review and future perspectives. *eTransportation* **2025**, *24*, 100420. [CrossRef]
79. Li, Y.; Li, Y.; Zhu, J.; Dai, H.; Li, Z.; Jiang, B. Deep learning-based state estimation for sodium-ion batteries using long short-term memory network. *Batteries* **2026**, *12*, 6. [CrossRef]
80. Zuo, X.; Fu, X.; Han, X.; Sun, M.; Fan, Y. State of charge estimation for sodium-ion batteries based on LSTM network and unscented Kalman filter. *Batteries* **2025**, *11*, 274. [CrossRef]
81. Liu, Y.; Yang, L.; Liao, R.; Hu, C.; Xiao, Y.; He, C.; Wu, X.; Zhang, Y.; Li, S. Degradation mechanism of sodium-ion batteries and state of health estimation via electrochemical impedance spectroscopy under temperature disturbances. *Energy* **2025**, *332*, 137064. [CrossRef]
82. Tan, R.; Hong, W.; Tang, J.; Lu, X.; Ma, R.; Zheng, X.; Li, J.; Huang, J.; Zhang, T.-Y. BatteryLife: A comprehensive dataset and benchmark for battery life prediction. In Proceedings of the 31st ACM SIGKDD Conference on Knowledge Discovery and Data Mining V.2, Toronto, ON, Canada, 3–7 August 2025; pp. 5789–5800. [CrossRef]
83. Vanem, E.; Wang, S. Data-driven state of health and state of safety estimation for alternative battery chemistries—A comparative review focusing on sodium-ion and LFP lithium-ion batteries. *Future Batter.* **2025**, *5*, 100033. [CrossRef]
84. Propp, K.; Auger, D.J.; Fotouhi, A.; Marinescu, M.; Knap, V.; Longo, S. Improved state of charge estimation for lithium-sulfur batteries. *J. Energy Storage* **2019**, *26*, 100943. [CrossRef]
85. Huang, Z.; Couto, L.D.; Dangwal, C.; Xiao, S.; Lv, W.; Zhang, D.; Moura, S.J. On electrochemical model-based state estimation for lithium–sulfur batteries. *IEEE Trans. Control Syst. Technol.* **2024**, *32*, 849–861. [CrossRef]
86. Shateri, N.; Auger, D.J.; Fotouhi, A.; Brighton, J. An experimental study on prototype lithium-sulfur cells for aging analysis and state-of-health estimation. *IEEE Trans. Transp. Electrif.* **2021**, *7*, 1324–1338. [CrossRef]
87. Cheng, D.; Sha, W.; Wang, L.; Tang, S.; Ma, A.; Chen, Y.; Wang, H.; Lou, P.; Lu, S.; Cao, Y.-C. Solid-state lithium battery cycle life prediction using machine learning. *Appl. Sci.* **2021**, *11*, 4671. [CrossRef]
88. Sun, B.; Pang, J.; Zhu, L.; Zhao, X.; Zhang, W.; Ma, S.; Liu, X. A novel state of charge estimation method based on electrochemical impedance spectroscopy for solid-state batteries of next-generation space power sources under different states of health. *Space Sci. Technol.* **2025**, *5*, 0198. [CrossRef]
89. Xu, H.; Peng, Y.; Su, L. Health state estimation method of lithium ion battery based on NASA experimental data set. *IOP Conf. Ser. Mater. Sci. Eng.* **2018**, *452*, 032067. [CrossRef]
90. Tang, R.; Zhang, P.; Ning, S.; Zhang, Y. Prediction of battery SOH and RUL based on cooperative characteristics in voltage-temperature-time dimensions. *J. Electrochem. Soc.* **2023**, *170*, 060535. [CrossRef]
91. Center for Advanced Life Cycle Engineering (CALCE). Battery Data. University of Maryland, College Park. Available online: https://calce.umd.edu/battery-data (accessed on 4 September 2026).

92. Wu, J.; Liu, Z.; Zhang, Y.; Lei, D.; Zhang, B.; Cao, W. Data-driven state of health estimation for lithium-ion battery based on voltage variation curves. *J. Energy Storage* **2023**, *73*, 109191. [CrossRef]
93. Severson, K.A.; Attia, P.M.; Jin, N.; Perkins, N.; Jiang, B.; Yang, Z.; Chen, M.H.; Aykol, M.; Herring, P.K.; Fraggedakis, D.; et al. Data-driven prediction of battery cycle life before capacity degradation. *Nat. Energy* **2019**, *4*, 383–391. [CrossRef]
94. Attia, P.M.; Grover, A.; Jin, N.; Severson, K.A.; Markov, T.M.; Liao, Y.-H.; Chen, M.H.; Cheong, B.; Perkins, N.; Yang, Z.; et al. Closed-loop optimization of fast-charging protocols for batteries with machine learning. *Nature* **2020**, *578*, 397–402. [CrossRef] [PubMed]
95. Reniers, J.M.; Mulder, G.; Howey, D.A. *Oxford Energy Trading Battery Degradation Dataset*; Oxford University Research Archive: Oxford, UK, 2020. [CrossRef]
96. Steinbuss, G.; Rzepka, B.; Bischof, S.; Blank, T.; Böhm, K. FOBSS: Monitoring data from a modular battery system. In Proceedings of the Tenth ACM International Conference on Future Energy Systems, Phoenix, AZ, USA, 25–28 June 2019; pp. 456–459. [CrossRef]
97. Moy, K.; Khan, M.A.; Fasolato, S.; Pozzato, G.; Allam, A.; Onori, S. Second-life lithium-ion battery aging dataset based on grid storage cycling. *Data Brief.* **2024**, *57*, 111046. [CrossRef] [PubMed]
98. Zhang, H.; Gui, X.; Zheng, S.; Lu, Z.; Li, Y.; Bian, J. BatteryML: An open-source platform for machine learning on battery degradation. In Proceedings of the International Conference on Learning Representations (ICLR), Vienna, Austria, 7–11 May 2024. Available online: https://openreview.net/forum?id=sxGugrYhP9 (accessed on 11 September 2026).
99. Liu, H.; Li, C.; Hu, X.; Li, J.; Zhang, K.; Xie, Y.; Wu, R.; Song, Z. Multi-modal framework for battery state of health evaluation using open-source electric vehicle data. *Nat. Commun.* **2025**, *16*, 1137. [CrossRef] [PubMed]
100. Che, Y.; Deng, Z.; Lin, X.; Hu, L.; Hu, X. Predictive battery health management with transfer learning and online model correction. *IEEE Trans. Veh. Technol.* **2021**, *70*, 1269–1277. [CrossRef]
101. Madani, S.S.; Shabeer, Y.; Fowler, M.; Panchal, S.; Chaoui, H.; Mekhilef, S.; Dou, S.X.; See, K. Artificial Intelligence and Digital Twin Technologies for Intelligent Lithium-Ion Battery Management Systems: A Comprehensive Review of State Estimation, Lifecycle Optimization, and Cloud-Edge Integration. *Batteries* **2025**, *11*, 298. [CrossRef]
102. Huang, N.E.; Shen, Z.; Long, S.R.; Wu, M.C.; Shih, H.H.; Zheng, Q.; Yen, N.-C.; Tung, C.C.; Liu, H.H. The empirical mode decomposition and the Hilbert spectrum for nonlinear and non-stationary time series analysis. *Proc. R. Soc. Lond. A* **1998**, *454*, 903–995. [CrossRef]
103. Tao, T.; Ji, C.; Dai, J.; Rao, J.; Wang, J.; Sun, W.; Romagnoli, J. Data-based health indicator extraction for battery SOH estimation via deep learning. *J. Energy Storage* **2024**, *78*, 109982. [CrossRef]
104. Li, Z.; Yang, Y.; Li, L.; Wang, D. A weighted Pearson correlation coefficient based multi-fault comprehensive diagnosis for battery circuits. *J. Energy Storage* **2023**, *60*, 106584. [CrossRef]
105. Zhang, X.; Hong, J.; Xu, X. Fault diagnosis of real-scenario battery systems based on modified entropy algorithms in electric vehicles. *J. Energy Storage* **2023**, *63*, 107079. [CrossRef]
106. Zhang, M.; Yin, J.; Chen, W. SOH estimation and RUL prediction of lithium batteries based on multidomain feature fusion and CatBoost model. *Energy Sci. Eng.* **2023**, *11*, 3082–3101. [CrossRef]
107. Wu, J.; Su, H.; Meng, J.; Lin, M. State of health estimation for lithium-ion battery via recursive feature elimination on partial charging curves. *IEEE J. Emerg. Sel. Top. Power Electron.* **2023**, *11*, 131–142. [CrossRef]
108. Roman, D.; Saxena, S.; Robu, V.; Pecht, M.; Flynn, D. Machine learning pipeline for battery state-of-health estimation. *Nat. Mach. Intell.* **2021**, *3*, 447–456. [CrossRef]
109. Greenbank, S.; Howey, D. Automated feature extraction and selection for data-driven models of rapid battery capacity fade and end of life. *IEEE Trans. Ind. Inform.* **2022**, *18*, 2965–2973. [CrossRef]
110. Gong, D.; Gao, Y.; Kou, Y.; Wang, Y. State of health estimation for lithium-ion battery based on energy features. *Energy* **2022**, *257*, 124812. [CrossRef]
111. Kong, J.Z.; Yang, F.; Zhang, X.; Pan, E.; Peng, Z.; Wang, D. Voltage-temperature health feature extraction to improve prognostics and health management of lithium-ion batteries. *Energy* **2021**, *223*, 120114. [CrossRef]
112. Richman, J.S.; Moorman, J.R. Physiological time-series analysis using approximate entropy and sample entropy. *Am. J. Physiol. Heart Circ. Physiol.* **2000**, *278*, H2039–H2049. [CrossRef] [PubMed]
113. Tang, X.; Wang, Y.; Liu, Q.; Gao, F. Reconstruction of the incremental capacity trajectories from current-varying profiles for lithium-ion batteries. *iScience* **2021**, *24*, 103103. [CrossRef] [PubMed]
114. Feng, H.; Song, D. A health indicator extraction based on surface temperature for lithium-ion batteries remaining useful life prediction. *J. Energy Storage* **2021**, *34*, 102118. [CrossRef]
115. Ang, E.Y.M.; Paw, Y.C. Efficient linear predictive model with short term features for lithium-ion batteries state of health estimation. *J. Energy Storage* **2021**, *44*, 103409. [CrossRef]
116. Raj, T.; Wang, A.A.; Monroe, C.W.; Howey, D.A. Investigation of path-dependent degradation in lithium-ion batteries. *Batter. Supercaps* **2020**, *3*, 1377–1385. [CrossRef]

117. Song, L.; Zhang, K.; Liang, T.; Han, X.; Zhang, Y. Intelligent state of health estimation for lithium-ion battery pack based on big data analysis. *J. Energy Storage* **2020**, *32*, 101836. [CrossRef]
118. Tian, J.; Xiong, R.; Shen, W. State-of-Health estimation based on differential temperature for lithium ion batteries. *IEEE Trans. Power Electron.* **2020**, *35*, 10363–10373. [CrossRef]
119. Zhu, J.; Wang, Y.; Huang, Y.; Gopaluni, R.B.; Cao, Y.; Heere, M.; Mühlbauer, M.J.; Mereacre, L.; Dai, H.; Liu, X.; et al. Data-driven capacity estimation of commercial lithium-ion batteries from voltage relaxation. *Nat. Commun.* **2022**, *13*, 2261. [CrossRef] [PubMed]
120. Deng, J. Introduction to grey system theory. *J. Grey Syst.* **1989**, *1*, 1–24. [CrossRef]
121. Chen, M.; Ma, G.; Liu, W.; Zeng, N.; Luo, X. An overview of data-driven battery health estimation technology for battery management system. *Neurocomputing* **2023**, *532*, 152–169. [CrossRef]
122. Mohri, M.; Rostamizadeh, A.; Talwalkar, A. *Foundations of Machine Learning*, 2nd ed.; MIT Press: Cambridge, MA, USA, 2018.
123. Cortes, C.; Vapnik, V. Support-vector networks. *Mach. Learn.* **1995**, *20*, 273–297. [CrossRef]
124. Suykens, J.A.K.; Vandewalle, J. Least squares support vector machine classifiers. *Neural Process. Lett.* **1999**, *9*, 293–300. [CrossRef]
125. Li, X.; Yu, D.; Vilsen, S.B.; Stroe, D.-I. The development of machine learning-based remaining useful life prediction for lithium-ion batteries. *J. Energy Chem.* **2023**, *82*, 103–121. [CrossRef]
126. Wang, G.; Lyu, Z.; Li, X. An optimized random forest regression model for Li-ion battery prognostics and health management. *Batteries* **2023**, *9*, 332. [CrossRef]
127. Breiman, L. Random forests. *Mach. Learn.* **2001**, *45*, 5–32. [CrossRef]
128. Friedman, J.H. Greedy function approximation: A gradient boosting machine. *Ann. Stat.* **2001**, *29*, 1189–1232. [CrossRef]
129. Song, S.; Fei, C.; Xia, H. Lithium-ion battery SOH estimation based on XGBoost algorithm with accuracy correction. *Energies* **2020**, *13*, 812. [CrossRef]
130. Zhang, M.; Yin, J.; Feng, T. Lithium battery SOH estimation based on manifold learning and LightGBM. *Appl. Sci.* **2023**, *13*, 6540. [CrossRef]
131. Rasmussen, C.E.; Williams, C.K.I. *Gaussian Processes for Machine Learning*; MIT Press: Cambridge, MA, USA, 2006. [CrossRef]
132. Zhang, Y.; Tang, Q.; Zhang, Y.; Wang, J.; Stimming, U.; Lee, A.A. Identifying degradation patterns of lithium ion batteries from impedance spectroscopy using machine learning. *Nat. Commun.* **2020**, *11*, 1706. [CrossRef] [PubMed]
133. Li, X.; Yuan, C.; Li, X.; Wang, Z. State of health estimation for Li-ion battery using incremental capacity analysis and Gaussian process regression. *Energy* **2020**, *190*, 116467. [CrossRef]
134. Ng, M.-F.; Zhao, J.; Yan, Q.; Conduit, G.J.; Seh, Z.W. Predicting the state of charge and health of batteries using data-driven machine learning. *Nat. Mach. Intell.* **2020**, *2*, 161–170. [CrossRef]
135. Tipping, M.E. Sparse Bayesian learning and the relevance vector machine. *J. Mach. Learn. Res.* **2001**, *1*, 211–244.
136. Kingma, D.P.; Welling, M. Auto-encoding variational Bayes. In Proceedings of the International Conference on Learning Representations (ICLR), Banff, AB, Canada, 14–16 April 2014.
137. Hinton, G.E. Deep belief networks. *Scholarpedia* **2009**, *4*, 5947. [CrossRef]
138. Goodfellow, I.; Pouget-Abadie, J.; Mirza, M.; Xu, B.; Warde-Farley, D.; Ozair, S.; Courville, A.; Bengio, Y. Generative adversarial nets. In Proceedings of the Advances in Neural Information Processing Systems, Montreal, QC, Canada, 8–13 December 2014; pp. 2672–2680.
139. Srivastava, N.; Hinton, G.; Krizhevsky, A.; Sutskever, I.; Salakhutdinov, R. Dropout: A simple way to prevent neural networks from overfitting. *J. Mach. Learn. Res.* **2014**, *15*, 1929–1958.
140. Ioffe, S.; Szegedy, C. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In Proceedings of the 32nd International Conference on Machine Learning, Lille, France, 7–9 July 2015; pp. 448–456.
141. Huang, G.-B.; Zhu, Q.-Y.; Siew, C.-K. Extreme learning machine: Theory and applications. *Neurocomputing* **2006**, *70*, 489–501. [CrossRef]
142. Hochreiter, S.; Schmidhuber, J. Long short-term memory. *Neural Comput.* **1997**, *9*, 1735–1780. [CrossRef] [PubMed]
143. Ma, L.; Hu, C.; Cheng, F. State of charge and state of energy estimation for lithium-ion batteries based on a long short-term memory neural network. *J. Energy Storage* **2021**, *37*, 102440. [CrossRef]
144. Sun, Y.; Zhang, J.; Zhang, K.; Qi, H.; Zhang, C. Battery state of health estimation method based on sparse auto-encoder and backward propagation fading diversity among battery cells. *Int. J. Energy Res.* **2021**, *45*, 7651–7662. [CrossRef]
145. Chen, J.C.; Chen, T.L.; Liu, W.J.; Cheng, C.C.; Li, M.G. Combining empirical mode decomposition and deep recurrent neural networks for predictive maintenance of lithium-ion battery. *Adv. Eng. Inform.* **2021**, *50*, 101405. [CrossRef]
146. Cheng, G.; Wang, X.; He, Y. Remaining useful life and state of health prediction for lithium batteries based on empirical mode decomposition and a long and short memory neural network. *Energy* **2021**, *232*, 121022. [CrossRef]
147. Cho, K.; Van Merrienboer, B.; Gulcehre, C.; Bahdanau, D.; Bougares, F.; Schwenk, H.; Bengio, Y. Learning phrase representations using RNN encoder-decoder for statistical machine translation. In Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), Doha, Qatar, 25–19 October 2014; pp. 1724–1734. [CrossRef]

148. Wei, M.; Gu, H.; Ye, M.; Wang, Q.; Xu, X.; Wu, C. Remaining useful life prediction of lithium-ion batteries based on Monte Carlo Dropout and gated recurrent unit. *Energy Rep.* **2021**, *7*, 2862–2871. [CrossRef]
149. Jiang, J.; Zhao, S.; Zhang, C. State-of-Health estimate for the lithium-ion battery using chi-square and ELM-LSTM. *World Electr. Veh. J.* **2021**, *12*, 228. [CrossRef]
150. Kaur, K.; Garg, A.; Cui, X.; Singh, S.; Panigrahi, B.K. Deep learning networks for capacity estimation for monitoring SOH of Li-ion batteries for electric vehicles. *Int. J. Energy Res.* **2021**, *45*, 3113–3128. [CrossRef]
151. Zhang, H.; Tang, W.; Na, W.; Lee, P.Y.; Kim, J. Implementation of generative adversarial network-cls combined with bidirectional long short-term memory for lithium-ion battery state prediction. *J. Energy Storage* **2020**, *31*, 101489. [CrossRef]
152. Jiao, R.; Peng, K.; Dong, J. Remaining useful life prediction of lithium-ion batteries based on conditional variational autoencoders-particle filter. *IEEE Trans. Instrum. Meas.* **2020**, *69*, 8831–8843. [CrossRef]
153. Shen, S.; Sadoughi, M.; Li, M.; Wang, Z.; Hu, C. Deep convolutional neural networks with ensemble learning and transfer learning for capacity estimation of lithium-ion batteries. *Appl. Energy* **2020**, *260*, 114296. [CrossRef]
154. Zhang, W.; Li, X.; Li, X. Deep learning-based prognostic approach for lithium-ion batteries with adaptive time-series prediction and on-line validation. *Measurement* **2020**, *164*, 108052. [CrossRef]
155. Ungurean, L.; Micea, M.V.; Carstoiu, G. Online state of health prediction method for lithium-ion batteries, based on gated recurrent unit neural networks. *Int. J. Energy Res.* **2020**, *44*, 6767–6777. [CrossRef]
156. Yang, H.; Wang, P.; An, Y.; Shi, C.; Sun, X.; Wang, K.; Zhang, X.; Wei, T.; Ma, Y. Remaining useful life prediction based on denoising technique and deep neural network for lithium-ion capacitors. *eTransportation* **2020**, *5*, 100078. [CrossRef]
157. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Kaiser, L.; Polosukhin, I. Attention is all you need. In Proceedings of the Advances in Neural Information Processing Systems, Long Beach, CA, USA, 4–9 December 2017; pp. 5998–6008.
158. Chen, K.; Wang, D.; Guo, W. A method for estimating the SOH of lithium-ion batteries based on graph perceptual neural network. *Batteries* **2024**, *10*, 326. [CrossRef]
159. Wang, T.; Ma, Z.; Zou, S.; Chen, Z.; Wang, P. Lithium-ion battery state-of-health estimation: A self-supervised framework incorporating weak labels. *Appl. Energy* **2024**, *355*, 122332. [CrossRef]
160. Sylvestrin, G.R.; Maciel, J.N.; Amorim, M.L.M.; Carmo, J.P.; Afonso, J.A.; Lopes, S.F.; Ando Junior, O.H. State of the Art in Electric Batteries' State-of-Health (SoH) Estimation with Machine Learning: A Review. *Energies* **2025**, *18*, 746. [CrossRef]
161. Liu, C.; Li, H.; Li, K.; Wu, Y.; Lv, B. Deep Learning for State of Health Estimation of Lithium-Ion Batteries in Electric Vehicles: A Systematic Review. *Energies* **2025**, *18*, 1463. [CrossRef]
162. Mayemba, Q.; Ducret, G.; Li, A.; Mingant, R.; Venet, P. General machine learning approaches for lithium-ion battery capacity fade compared to empirical models. *Batteries* **2024**, *10*, 367. [CrossRef]
163. Ji, Y.; Chen, Z.; Shen, Y.; Yang, K.; Wang, Y.; Cui, J. An RUL prediction approach for lithium-ion battery based on SADE-MESN. *Appl. Soft Comput.* **2021**, *104*, 107195. [CrossRef]
164. Shen, D.; Wu, L.; Kang, G.; Guan, Y.; Peng, Z. A novel online method for predicting the remaining useful life of lithium-ion batteries considering random variable discharge current. *Energy* **2021**, *218*, 119490. [CrossRef]
165. Liu, S.; Wang, J.; Liu, H.; Liu, Q.; Tang, J.; Li, Z. Battery degradation model and multiple-indicators based lifetime estimator for energy storage system design and operation: Experimental analyses of cycling-induced aging. *Electrochim. Acta* **2021**, *384*, 138294. [CrossRef]
166. Lee, C.J.; Kim, B.K.; Kwon, M.K.; Nam, K.; Kang, S.W. Real-time prediction of capacity fade and remaining useful life of lithium-ion batteries based on charge/discharge characteristics. *Electronics* **2021**, *10*, 846. [CrossRef]
167. Zhang, Y.; Peng, Z.; Guan, Y.; Wu, L. Prognostics of battery cycle life in the early-cycle stage based on hybrid model. *Energy* **2021**, *221*, 119901. [CrossRef]
168. Gou, B.; Xu, Y.; Feng, X. State-of-Health estimation and remaining-useful-life prediction for lithium-ion battery using a hybrid data-driven method. *IEEE Trans. Veh. Technol.* **2020**, *69*, 10854–10867. [CrossRef]
169. Meng, H.; Li, Y.F. A review on prognostics and health management (PHM) methods of lithium-ion batteries. *Renew. Sustain. Energy Rev.* **2019**, *116*, 109405. [CrossRef]
170. Wang, D.; Kong, J.Z.; Zhao, Y.; Tsui, K.L. Piecewise model based intelligent prognostics for state of health prediction of rechargeable batteries with capacity regeneration phenomena. *Measurement* **2019**, *147*, 106836. [CrossRef]
171. Catelani, M.; Ciani, L.; Grasso, F.; Patrizi, G.; Reatti, A. Remaining Useful Life estimation for electric vehicle batteries using a similarity-based approach. In Proceedings of the 2022 IEEE International Workshop on Metrology for Automotive (MetroAutomotive), Modena, Italy, 4–6 July 2022; pp. 82–87. [CrossRef]
172. Wu, M.; Sun, M.; Zhang, F.; Wang, L.; Zhao, N.; Wang, J.; Huang, W. A fault detection method of electric vehicle battery through Hausdorff distance and modified Z-score for real-world data. *J. Energy Storage* **2023**, *60*, 106561. [CrossRef]

173. Sun, Z.; Han, Y.; Wang, Z.; Chen, Y.; Liu, P.; Qin, Z.; Zhang, Z.; Wu, Z.; Song, C. Detection of voltage fault in the battery system of electric vehicles using statistical analysis. *Appl. Energy* **2022**, *307*, 118172. [CrossRef]
174. Wang, Y.; Ren, D.; Feng, X.; Wang, L.; Ouyang, M. Thermal runaway modeling of large format high-nickel/silicon-graphite lithium-ion batteries based on reaction sequence and kinetics. *Appl. Energy* **2022**, *306*, 117943. [CrossRef]
175. Tran, M.K.; Fowler, M. A review of lithium-ion battery fault diagnostic algorithms: Current progress and future challenges. *Algorithms* **2020**, *13*, 62. [CrossRef]
176. Gao, W.; Li, X.; Ma, M.; Fu, Y.; Jiang, J.; Mi, C. Case study of an electric vehicle battery thermal runaway and online internal short-circuit detection. *IEEE Trans. Power Electron.* **2021**, *36*, 2452–2455. [CrossRef]
177. Zhang, L.; Zhao, P.; Xu, M.; Wang, X. Computational identification of the safety regime of Li-ion battery thermal runaway. *Appl. Energy* **2020**, *261*, 114440. [CrossRef]
178. Zhang, J.; Wang, Y.; Jiang, B.; He, H.; Huang, S.; Wang, C.; Zhang, Y.; Han, X.; Guo, D.; He, G.; et al. Realistic fault detection of Li-ion battery via dynamical deep learning. *Nat. Commun.* **2023**, *14*, 5940. [CrossRef] [PubMed]
179. Fioravanti, R.; Kumar, K.; Nakata, S.; Chalamala, B.; Preger, Y. Predictive-maintenance practices: For operational safety of battery energy storage systems. *IEEE Power Energy Mag.* **2020**, *18*, 86–97. [CrossRef]
180. Feng, X.; Zheng, S.; Ren, D.; He, X.; Wang, L.; Cui, H.; Liu, X.; Jin, C.; Zhang, F.; Xu, C.; et al. Investigating the thermal runaway mechanisms of lithium-ion batteries based on thermal analysis database. *Appl. Energy* **2019**, *246*, 53–64. [CrossRef]
181. Kang, Y.; Duan, B.; Zhou, Z.; Shang, Y.; Zhang, C. A multi-fault diagnostic method based on an interleaved voltage measurement topology for series connected battery packs. *J. Power Sources* **2019**, *417*, 132–144. [CrossRef]
182. Gneiting, T.; Balabdaoui, F.; Raftery, A.E. Probabilistic forecasts, calibration and sharpness. *J. R. Stat. Soc. Ser. B Stat. Methodol.* **2007**, *69*, 243–268. [CrossRef]
183. Lei, J.; G'Sell, M.; Rinaldo, A.; Tibshirani, R.J.; Wasserman, L. Distribution-free predictive inference for regression. *J. Am. Stat. Assoc.* **2018**, *113*, 1094–1111. [CrossRef]
184. Barua, N.; Collins, M.D.; Islam, M.S.; Barua, A.; Siddiquee, K.N.e.A. Predictor structure modulates validation inflation and cell-transfer reliability in battery state-of-health estimation. *Batteries* **2026**, *12*, 342. [CrossRef]
185. Li, Y.; Liu, K.; Foley, A.M.; Zülke, A.; Berecibar, M.; Nanini-Maury, E.; Van Mierlo, J.; Hoster, H.E. Data-driven health estimation and lifetime prediction of lithium-ion batteries: A review. *Renew. Sustain. Energy Rev.* **2019**, *113*, 109254. [CrossRef]
186. Eleftheroglou, N.; Mansouri, S.S.; Loutas, T.; Karvelis, P.; Georgoulas, G.; Nikolakopoulos, G.; Zarouchas, D. Intelligent data-driven prognostic methodologies for the real-time remaining useful life until the end-of-discharge estimation of the lithium-polymer batteries of unmanned aerial vehicles with uncertainty quantification. *Appl. Energy* **2019**, *254*, 113677. [CrossRef]
187. Hosen, M.S.; Kalogiannis, T.; Youssef, R.; Karimi, D.; Behi, H.; Jin, L.; Van Mierlo, J.; Berecibar, M. Twin-model framework development for a comprehensive battery lifetime prediction validated with a realistic driving profile. *Energy Sci. Eng.* **2021**, *9*, 2191–2201. [CrossRef]
188. Karimi, D.; Khaleghi, S.; Behi, H.; Beheshti, H.; Hosen, M.S.; Akbarzadeh, M.; Van Mierlo, J.; Berecibar, M. Lithium-ion capacitor lifetime extension through an optimal thermal management system for Smart grid applications. *Energies* **2021**, *14*, 2907. [CrossRef]
189. Gretton, A.; Borgwardt, K.M.; Rasch, M.J.; Schölkopf, B.; Smola, A. A kernel two-sample test. *J. Mach. Learn. Res.* **2012**, *13*, 723–773.
190. Raissi, M.; Perdikaris, P.; Karniadakis, G.E. Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations. *J. Comput. Phys.* **2019**, *378*, 686–707. [CrossRef]
191. Wang, F.; Zhai, Z.; Zhao, Z.; Di, Y.; Chen, X. Physics-informed neural network for lithium-ion battery degradation stable modeling and prognosis. *Nat. Commun.* **2024**, *15*, 4332. [CrossRef] [PubMed]
192. Kohtz, S.; Xu, Y.; Zheng, Z.; Wang, P. Physics-informed machine learning model for battery state of health prognostics using partial charging segments. *Mech. Syst. Signal Process.* **2022**, *172*, 109002. [CrossRef]
193. Lundberg, S.M.; Lee, S.-I. A unified approach to interpreting model predictions. In Proceedings of the Advances in Neural Information Processing Systems, Long Beach, CA, USA, 4–9 December 2017; pp. 4765–4774.
194. Ribeiro, M.T.; Singh, S.; Guestrin, C. Why should I trust you? Explaining the predictions of any classifier. In Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, San Francisco, CA, USA, 13–17 August 2016; pp. 1135–1144. [CrossRef]
195. Fan, Z.; Guo, Y.; Song, L.; Du, J.; Wang, H.; Li, X. BatteryTSFM: Generalizable long-horizon degradation prediction across conditions and chemistries via time series foundation models. *Energy AI* **2025**, *22*, 100646. [CrossRef]
196. Wong, K.L.; Im, S.K.; Fang, X.; Tang, S.-K.; Lam, C.T.; Pau, G. Prognostics by generalists: Large language models for lithium-ion batteries health forecasting. *Sci. Rep.* **2026**, *16*, 16723. [CrossRef] [PubMed]
197. Kröger, T.; Belnarsch, A.; Bilfinger, P.; Ratzke, W.; Lienkamp, M. Collaborative training of deep neural networks for the lithium-ion battery aging prediction with federated learning. *eTransportation* **2023**, *18*, 100294. [CrossRef]

198. Lai, R.; Wang, J.; Tian, Y.; Tian, J. FedCBE: A federated-learning-based collaborative battery estimation system with non-IID data. *Appl. Energy* **2024**, *368*, 123534. [CrossRef]
199. Li, W.; Rentemeister, M.; Badeda, J.; Jöst, D.; Schulte, D.; Sauer, D.U. Digital twin for battery systems: Cloud battery management system with online state-of-charge and state-of-health estimation. *J. Energy Storage* **2020**, *30*, 101557. [CrossRef]
200. Bendaouia, A.; Wang, F.; Ouifak, H.; Li, J. Digital twin for battery energy storage systems. *Renew. Sustain. Energy Rev.* **2026**, *226*, 116347. [CrossRef]

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