Future Batteries 8 (2025) 100114

Contents lists available at ScienceDirect

Future Batteries

journal homepage: www.sciencedirect.com/journal/future-batteries

Battery-Insight-PSO: A machine learning model for accurate prediction of state of health and remaining useful life in lithium-ion batteries

Md Fazle Hasan Shiblee, Hannu Laaksonen *

School of Technology and Innovations, University of Vaasa, Vaasa, 65100, Finland

ARTICLE INFO

Keywords:
SOH
RUL
CBM
Li-ion battery
PSO
XGBoost
Hyperparameter tuning

ABSTRACT

Condition based monitoring (CBM) of the lithium-ion (Li-ion) battery has become very popular in recent years because of its wide usage as an energy storage for smart grids, power sources in various industrial equipment, electric vehicles (EVs), etc. As a result, predicting the state of health (SOH) and the remaining useful life (RUL) of Li-ion batteries with high accuracy ensures optimal performance and safe utilization, preventing non-scheduled failures and saving maintenance costs. This paper illustrates the significance of highly accurate SOH and RUL prediction for Li-ion batteries. This paper proposes a model called Battery-Insight-PSO, which employs the Extreme Gradient Boosting Regression (XGBoost) machine learning algorithm to forecast SOH and RUL. In this study, the Particle Swarm Optimization Algorithm (PSO) is used to optimize different parameters of XGBoost for ensuring precise and reliable predictions of SOH and RUL for Li-ion batteries. In this study, the National Aeronautics and Space Administration (NASA) Li-ion Battery Aging Datasets and the NMC LCO 18650 battery dataset from the Hawaii Natural Energy Institute (HNEI) were analyzed. Additionally, the performance of Battery-Insight-PSO was compared with other machine learning algorithms. Machine learning models were evaluated using various performance metrics. The estimation errors of Battery-Insight-PSO are very low, which means that this model can be highly accurate in predicting SOH and RUL. Moreover, the R^2 scores for the training and testing sets of this model also show high consistency with 0.9998 for each dataset, demonstrating high accuracy and reliable performance.

Introduction

Li-ion batteries are crucial for applications such as EVs and renewable energy storage systems, offering high energy density, long lifespan, and efficient power storage compared to other battery types [1]. The growing adoption of EVs, driven by environmental concerns and the need for energy efficiency, underscores the importance of optimizing battery performance. Li-ion batteries are key to this transition due to their safety, long-lasting nature, and low pollution [2]. The Battery Management System (BMS) in EVs plays a vital role in monitoring battery health, forecasting its usable life, and ensuring safety. Accurate SOH measurements are critical for predicting the RUL of batteries [3,4], but challenges remain in estimating battery degradation due to limited physical testing methods [5].

As the automotive industry increasingly adopts digital technologies such as the Internet of Things (IoT) and Artificial Intelligence (AI) for BMS, real-time monitoring and early failure prediction are becoming more accessible. Machine learning (ML) algorithms like XGBoost are extensively used for SOH and RUL prediction, with performance enhanced by hyperparameter optimization [6]. Despite these advancements, many existing ML models struggle to handle the nonlinear degradation patterns typical of fast-charging Li-ion batteries. Models such as [7]'s AT-CNN-BiLSTM and [3]'s IPSOBPNN offer improvements in prediction accuracy, yet still face limitations in capturing complex degradation behaviors under varying operational conditions. Similarly, hybrid models like [8]'s dilated CNN BiGRU achieve better performance on NASA and Oxford battery datasets, but do not fully address the challenges posed by real-world dynamics, where temperature variations and operational loads can significantly affect battery performance.

Recent studies, such as [9], developed hybrid frameworks like GRU-LSTM to predict SOH and cycle life with high accuracy using early-cycle data. While these models show promise, they often struggle with scalability and generalization across different battery types and real-world conditions. To address these challenges, hybrid models combining optimization techniques such as PSO with neural networks have been explored. For example, [10] combined PSO with neural networks to improve prediction performance and robustness, addressing limitations like slow convergence and local minima in traditional models. [11]

* Corresponding author.
E-mail address: hannu.laaksonen@uwasa.fi (H. Laaksonen).

https://doi.org/10.1016/j.fub.2025.100114
Received 23 May 2025; Received in revised form 28 September 2025; Accepted 8 October 2025
Available online 13 October 2025
2950-2640/© 2025 The Author(s). Published by Elsevier Ltd. This is an open access article under the CC BY license (http://creativecommons.org/licenses/by/4.0/).

proposed a novel hybrid model integrating Gated Recurrent Units (GRU), Multi-Head Attention (MHA), and ridge regression, which effectively captures both temporal and nonlinear degradation features, outperforming traditional and advanced methods on real-world datasets. Further attempts by [12] used a binary segmentation model with particle filtering to manage uncertainty, achieving a 27% average absolute error in laboratory data, but still struggled to capture real-world degradation behaviors. Furthermore, models like [13]'s hybrid deep learning approach, which uses GRU and MHA, have shown outstanding performance on datasets like NMC-LCO 18650 and NASA, achieving an MAE of 0.002 and an R^2 score of 99.99%. However, performance gains in methods like Random Forest and LightGBM are still limited by inherent model complexities and challenges with high-dimensional data [6]. Hybrid machine learning approaches, such as those combining SVR and Random Forest, have been developed to improve RUL predictions, yet they still struggle with noisy environments [14,15]. Additionally, [16] developed a multi-channel hybrid neural network integrating CNN, BiLSTM, GRU, and attention mechanisms, addressing the complexity of high-dimensional datasets. Their model, which incorporates advanced health feature selection and outlier removal, demonstrated excellent predictive accuracy across different battery types, outperforming simpler models like CNN and BiLSTM.

While significant progress has been made in developing machine learning models for SOH and RUL prediction, many models face challenges in capturing the complex and nonlinear degradation behaviors typical of real-world Li-ion batteries. Existing methods often lack scalability and fail to generalize across different battery types and operational conditions, leading to limited applicability in real-world environments. Additionally, many models rely on predefined degradation assumptions or limited sensor data, which hinders their robustness and predictive accuracy. Furthermore, the error metrics observed in existing literature remain high, with models showing substantial deviations in key metrics such as RMSE, MAE, and R^2, even in well-controlled experimental conditions. This highlights the need for more effective models. This paper proposes Battery-Insight-PSO to address these gaps by integrating PSO with XGBoost to optimize hyperparameters and leverage multi-sensor data, offering a more robust and scalable solution for battery health management.

In conclusion, Battery-Insight-PSO offers an innovative and comprehensive solution for estimating SOH and RUL, enhancing the accuracy, scalability, and reliability of battery health predictions. By integrating optimization and multi-sensor data, the model significantly improves predictive maintenance, ensuring greater safety and efficiency in battery-powered systems.

This research aims to advance the field through the following contributions:

- This study proposes a machine learning-based model, named Battery-Insight-PSO, for predicting the SOH and RUL of Li-ion batteries using the NASA and HNEI datasets. The model's performance is evaluated and compared with various other models, including Support Vector Regression (SVR), Random Forest Regression (RFR), Gradient Boosting Regression (GBR), K-Nearest Neighbors (KNN), and XGBoost. The results demonstrate that Battery-Insight-PSO achieves high accuracy, as measured by the R^2 value, and low values for evaluation metrics such as MAE, MSE, RMSE, MAPE, and Normalized Root Mean Squared Error (NRMSE), outperforming the other models in SOH and RUL prediction tasks.
- To enhance model performance, PSO has been employed for hyperparameter tuning, which significantly improves the accuracy and efficiency of battery RUL prediction. This approach showcases the potential of machine learning methodologies in revolutionizing CBM by enabling more accurate and efficient management of battery health and lifecycle.
- A comprehensive comparative evaluation is conducted on the SOH and RUL estimation performance of the Battery-Insight-PSO model, highlighting its superiority over conventional models.

The remainder of this paper is organized as follows: In Section "Data curation and analysis", there are descriptions of datasets and the preprocessing steps. Section "Methodology", presents the framework of Battery-Insight-PSO, which integrates the XGBoost algorithm with PSO for hyperparameter tuning. This section also includes a description of the baseline algorithms and a detailed explanation of the PSO algorithm. Section "Results and discussion" is related to the results and performance evaluation of Battery-Insight-PSO along with other models and algorithms. Finally, Section "Conclusion and future scope" concludes the paper by summarizing the contributions of this paper and discussing possible future research directions. The Supplementary Materials, which include additional figures and detailed experimental setups, are provided after the References section.

Data curation and analysis

There are three major areas in this category: data collection, the processing of data, and feature selection. The first subsection provides an overview of the dataset, including its origin and relevant characteristics. The second subsection outlines the various data processing techniques employed to prepare the data for analysis, ensuring its suitability for machine learning applications. Each of those areas is really important to understand and use the dataset effectively for predictive modeling.

Dataset description

The NASA Li-ion Battery Aging Dataset is from an experimental setup developed to investigate battery aging by accelerated charge–discharge cycles, published by the NASA Ames Research Center. 18650 Li-ion batteries with 2Ah capacity, 4.2 V charge cut-off voltage, and 2.7 V discharge cut-off voltage were used in the experiment. The batteries are considered to have reached their end of life (EOL) when the capacity of the batteries degrades to 70% of the rated capacity [3,5,7,8]. Discharge process features, including current, temperature, time, voltage, and capacity, from batteries B5, B7, B18, B47, and B48 were chosen. Many existing studies, such as [4], focus solely on batteries B5, B7, and B18 for method evaluation. To enhance model robustness, however, batteries B47 and B48 were additionally used as auxiliary datasets for training and testing. Notably, the maximum capacities of B47 and B48 are observed to be below 1.4 Ah, indicating significant performance degradation. The capacity of these datasets can be observed in Figs. 1 and 2. A dashed purple line indicates the failure threshold (for datasets B5, B7, and B18), the critical point where the SOH is insufficient for reliable battery operation. In the NASA dataset, RUL is determined using the cycle index, while SOH is calculated based on the battery's capacity. RUL indicates the remaining number of cycles until the battery reaches the end of its useful life and is computed by subtracting the current cycle number from the maximum cycle count. SOH reflects the battery's health by comparing its current capacity to its initial capacity.

$$\text{RUL} = \text{Cycles remaining until SOH reaches failure threshold} - \text{Current Cycle} \tag{1}$$

$$\text{SOH} = \frac{\text{Capacity at Current Cycle}}{\text{Initial Capacity}} \tag{2}$$

The heatmap in Fig. 3 illustrates the Pearson correlation between features in the NASA battery dataset, highlighting important relationships among them. Notably, cycle, SOH, and RUL exhibit nearly perfect correlations with each other $|\text{correlation}| \approx 1$, which aligns with their direct

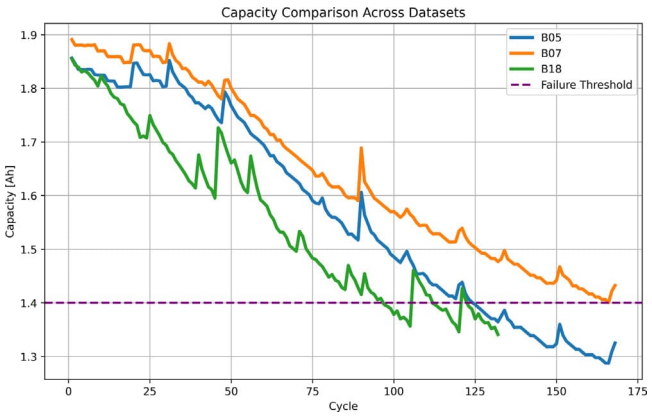

Fig. 1. Capacity observation of NASA battery datasets B5, B7 and B18.

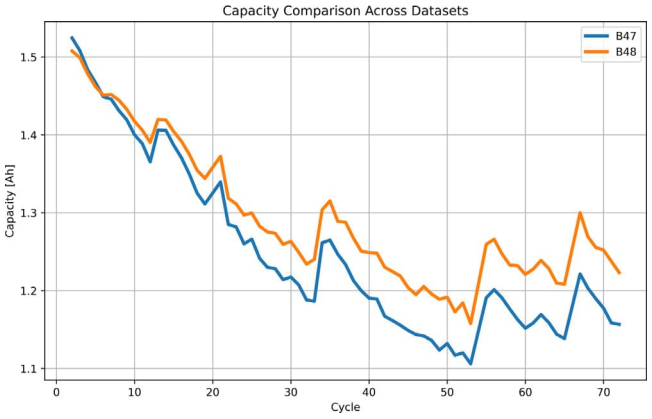

Fig. 2. Capacity observation of NASA battery datasets B47 and B48.

mathematical dependencies. Capacity also shows a very strong positive correlation with SOH and RUL, and a strong negative correlation with cycle, indicating that battery health and lifespan decline steadily over usage. Additionally, temperature has a strong positive correlation with time and a strong negative correlation with terminal_voltage, reflecting thermal patterns as the battery ages. Meanwhile, charge_current shows a moderate positive correlation with cycle, while other features like terminal_current and charge_voltage display relatively weak or mixed correlations. Overall, the heatmap provides useful insights into feature relationships, aiding in informed feature selection for modeling and analysis. The SHapley Additive exPlanations (SHAP) feature importance plots for the Battery-Insight-PSO model on the NASA battery dataset highlight the key features contributing to the prediction of SOH and RUL. As shown in Fig. 4, which corresponds to RUL prediction, cycle is the most influential feature, followed by capacity, while the contributions of other features are negligible. For SOH prediction, depicted in Fig. 5, capacity emerges as the dominant feature, with cycle having a smaller yet notable influence. All other features, including charge current, time, voltage, and temperature, have minimal impact in both cases. These observations are consistent with the Pearson correlation heatmap, where RUL is perfectly negatively correlated with cycle and strongly positively correlated with capacity, and SOH shows a near-perfect positive correlation with capacity and a strong negative correlation with cycle. This alignment between SHAP analysis and correlation results confirms the reliability and interpretability of the model's behavior.

In addition to the NASA dataset, this research also utilizes a dataset from HNEI. This dataset comprises 14 batteries with nickel manganese cobalt (NMC) and lithium cobalt oxide (LCO) mixed cathode

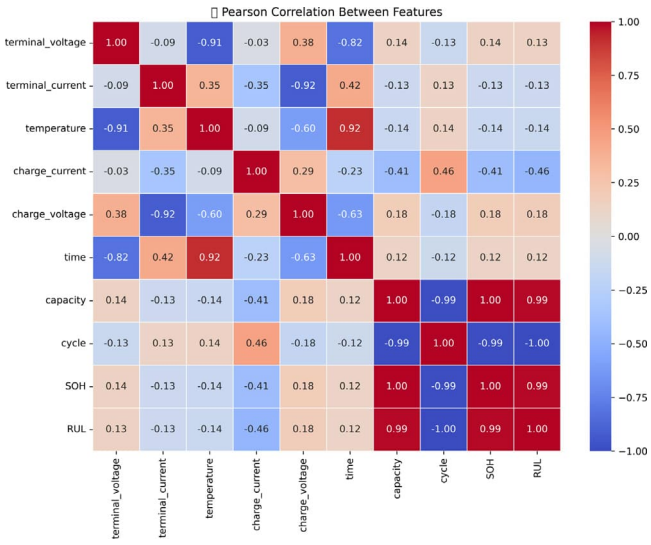

Fig. 3. The Pearson correlation between the features of the NASA dataset.

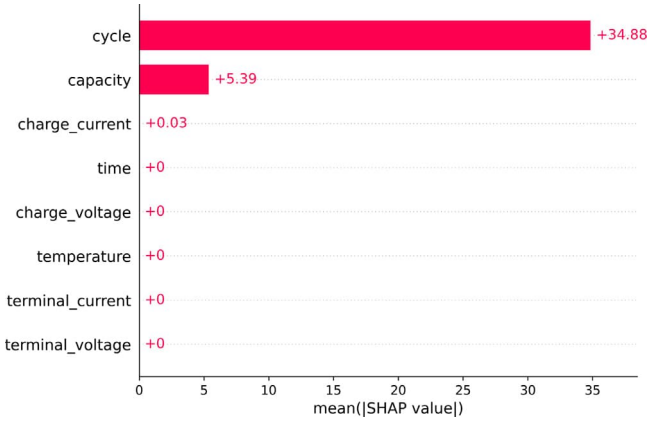

Fig. 4. SHAP feature importance for RUL prediction using the Battery-Insight-PSO model on the NASA battery dataset.

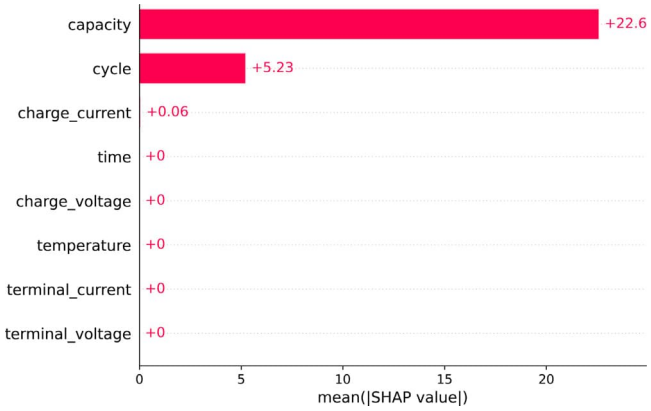

Fig. 5. SHAP feature importance for SOH prediction using the Battery-Insight-PSO model on the NASA battery dataset.

chemistries, each with a nominal capacity of 2.8 Ah per cell. It is employed to enhance the model's performance in predicting the RUL of batteries, offering complementary insights into battery degradation behavior. The batteries were cycled more than 1000 times at a constant

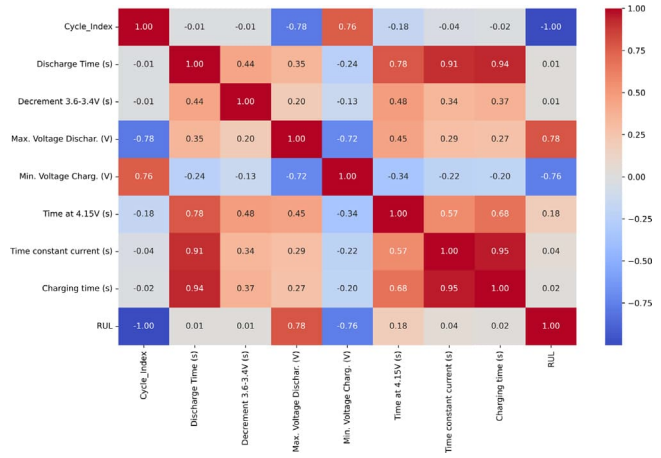

Fig. 6. The Pearson correlation between the features of the HNEI dataset.

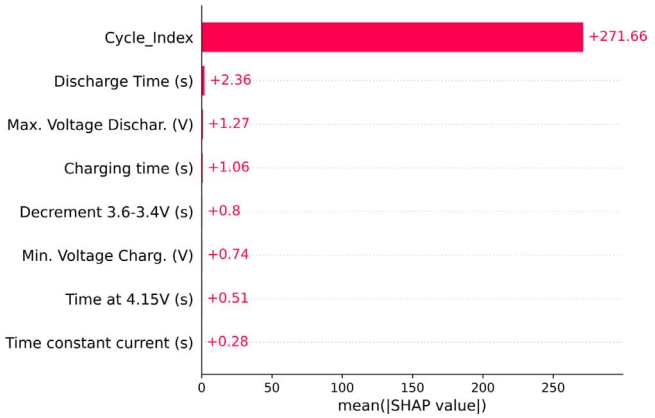

Fig. 7. SHAP feature importance for RUL prediction using the Battery-Insight-PSO model on the HNEI battery dataset.

temperature of 25 °C with currents between C/2 and 1.5C applied during the galvanostatic charge–discharge modes following the constant current–constant voltage (CC–CV) charging sequence, which extracted voltage characteristics as well as current behavior in order to predict RUL, an important indicator for battery health inspection metrics. With NMC-LCO batteries being used in electric vehicles (EVs) to a great extent, this dataset could provide valuable insights into the degradation patterns of these types of batteries along the automotive life cycle. Test was performed on separate training and testing sets of batteries to verify the reliability of the models [6,17,18]. In the HNEI dataset, RUL is determined by subtracting the current cycle index from the maximum cycle index. The dataset focuses on RUL prediction and investigates voltage relaxation behavior under different conditions; since capacity measurements are not provided, only RUL is performed and EOL is not defined.

$$\text{RUL} = \text{Maximum Cycle Index} - \text{Current Cycle Index} \tag{3}$$

Fig. 6 presents the Pearson correlation heatmap of all features in the HNEI lithium-ion battery dataset with respect to RUL. The plot reveals a strong negative correlation between Cycle Index and RUL (≈ -1.00), reflecting the expected decrease in remaining useful life as the number of cycles increases. Notably, Max. Voltage Discharge shows a strong positive correlation with RUL (≈ 0.78), while Min. Voltage Charge exhibits a strong negative correlation (≈ -0.76), indicating their significance in estimating RUL. Charging-related features such as Charging Time, Time Constant Current, and Time at 4.15 V are highly correlated with each other (up to 0.95), suggesting redundancy or interdependence in charging dynamics. Meanwhile, features like Decrement 3.6–3.4 V and Discharge Time show very weak correlations with RUL (≈ 0.01), implying limited predictive value. Overall, the heatmap highlights the most influential features and their interrelationships, which is crucial for developing accurate RUL prediction models. Fig. 7 illustrates the SHAP feature importance rankings for the RUL prediction model using the HNEI lithium-ion battery dataset. The Cycle Index overwhelmingly dominates feature importance with a mean SHAP value of approximately 271.66, far surpassing all other features. This indicates that the number of cycles alone explains the majority of the model's predictive power for estimating RUL. Other features such as Discharge Time, Max. Voltage Discharge, and Charging Time show marginal but non-negligible contributions. Overall, the results highlight that Cycle Index is the dominant feature for RUL prediction, underscoring its critical role in determining the battery's remaining useful life.

Data processing

After the initial data cleaning, multiple steps of configuring the dataset for analysis were performed. This process involved feature engineering, normalization, and data partitioning. Feature engineering derived new variables from the original data to capture useful information. Although this step is not mandatory for clustering, the features were normalized to have zero mean and unit variance, as this often improves the performance of many machine learning algorithms. Finally, the dataset was split into training, validation, and testing subsets using a 40:30:30 ratio. All machine learning algorithms were trained on the fixed training dataset. Due to stratification during partitioning, there was virtually equal representation of battery cycles across the sets. Both SOH and RUL predictions were performed on both the NASA and HNEI datasets. This consistent and balanced data preparation approach enhanced the model's robustness, enabling accurate predictions of SOH and RUL across diverse battery conditions.

Methodology

This section provides a comprehensive overview of the proposed Battery Insight-PSO model, including the methodologies and experimental setups employed for its development and evaluation. It presents an in-depth discussion on the model's architecture, functioning, and its predictive capabilities for estimating the SOH and RUL of lithium-ion batteries. Furthermore, this section highlights the application of PSO for hyperparameter tuning and the rationale behind the selection of the final proposed methodology. Together, these components form a cohesive framework aimed at enhancing prediction accuracy and offering practical insights into battery health assessment. A detailed description of the different baseline models used for comparison is provided in Section "Supplementary Materials", labeled as Supplementary Materials.

Extreme gradient boosting (XGBoost)

XGBoost is an ensemble machine learning method that has a gradient boosting framework and works specifically for constructing decision tree ensembles. It shines in making predictions about structured/tabular data and has a reputation for dealing well with interactions between features of your dataset. While artificial neural networks are the winning choice among all of the algorithms when dealing with unstructured data, XGBoost and other decision tree-based models continue to be top-ranked for small (when not too deep) to medium size datasets. As discussed above, one of the greatest advantages that XGBoost has is that it can learn and build upon the decision tree algorithm in an iterative

way, which makes it a very good model for prediction purposes. That is the reason for one of many examples and to mitigate more overfitting in high-dimensional feature spaces. XGBoost has a lot of regularization techniques. The equation for XGBoost in continuous form is:

$$F_m(x) = F_{m-1}(x) + \eta f_m(x;\theta_m) + \epsilon_m \tag{4}$$

Here η is the learning rate, which limits the contribution of each tree to prediction, and F_m(x) shows the predicted output at iteration m for input x. In some models, like XGBoost, you create individual base learners (weak predictors) through decision trees. This training method of reevaluating the weights against a target is done in sequence bias; it improves accuracy as all errors help correct subsequent predictions. Maximum depth and gamma are also some parameters that help define the specific structure of each decision tree, known as T_t(x). These parameters define the complexity of the tree and thus assist in managing overfitting by controlling how the feature space is split.

$$T_t(x) = \text{Tree}(r_t^i, x_i; \text{max\_depth}, \gamma) \tag{5}$$

Maximum Depth sets the maximum depth of each tree, and gamma is used to specify the minimum loss reduction required for partitioning a leaf node. Also, the number of trees in the model is shown by the number of estimators, which specifies how many different decision trees are built during training.

$$y_t^i = Y_{t-1}^i + \eta \cdot T_t(x_i) \tag{6}$$

Finally, the aggregate prediction for the ith instance across all trees can be summarized as:

$$\sum_{t=1}^{T} y_t^i = Y_0^i + \sum_{t=1}^{T} \eta \cdot T_t(x_i) \tag{7}$$

Here, T is the total number of ensemble estimators made up for making the final model output. The upper and lower bounds of optimizing hyperparameters with the PSO for the XGBoost algorithm were established in this study. The upper bounds (ub) are specified as follows: ub equals [500, 0.5, 15, 0.5], which means that there can be at most 300 estimators, a learning rate of 0.5, a maximum tree depth of 15, and a gamma value of 0.5. The lower bounds (lb) are set as lb equals [50, 0.01, 3, 0.0], indicating that there should be at least 50 estimators, a minimum learning rate of 0.01, a tree depth of at least 3, and a gamma value of 0. A learning rate of 0.01 is considered sufficient for smaller sample sizes, though it should not drop to zero, as this may result in model instability. The tree depth should not go below 3, as this would imply minimal splitting of the decision space, which limits the model's complexity. Lastly, a gamma value of 0 is crucial to ensure appropriate pruning in the algorithm. These bounds constrain the search space for hyperparameter optimization and enhance the model's predictive performance in forecasting the SOH and RUL of Li-ion batteries [14,19]. Fig. 8 demonstrates the framework of XGBoost algorithm.

PSO optimization algorithm

PSO is a nature-inspired optimization algorithm inspired by the social behaviors of birds flocking or fish schooling. The method starts by setting up a population of particles (which will constitute the population of possible solutions to the optimization problem) in the solution space with random positions and velocities for each particle. With an objective function that will evaluate the quality or fitness of each particle's solution, fitness is evaluated, and the flow chart can be observed in Fig. 9. In this study, PSO is employed to optimize the parameters of various machine learning algorithms, with the mean squared error (MSE) serving as the objective function. The algorithm then iteratively updates the position and velocity of each particle based on two key factors: the personal best position (personal best) its own previous best position (personal best), and the best position found globally by the entire swarm (global best). It is a process that has exploration: particles search areas of the solution space that they have not visited before, and it has exploitation: particles refine existing solutions. In the course of these iterative updates, the swarm seeks to find an optimal or acceptable settlement by grating particles to modify their positions or velocities. PSO is widely accepted as being simple to implement, easy to implement, and can solve complex, non-linear, multi-dimensional optimization problems [20].

Proposed model

This study introduces **Battery-Insight-PSO**, a machine learning-based model designed to accurately predict the SOH and RUL of Li-ion batteries. The model combines the XGBoost algorithm with PSO for hyperparameter tuning to enhance predictive performance. The methodology consists of four stages: data collection and preprocessing, feature extraction, model training and optimization, and performance evaluation.

Two well-known datasets are used: the NASA Li-ion Battery Aging Dataset (B5, B7, B18, B47, and B48) and the HNEI NMC-LCO 18650 Battery Dataset. Preprocessing steps include handling missing values, normalizing data, and selecting relevant features. Both datasets are divided into training, validation, and testing sets in a 40:30:30 ratio, with 5-fold cross-validation applied to reduce overfitting and ensure generalization.

PSO is employed to optimize key hyperparameters of XGBoost, namely the number of estimators, learning rate, maximum depth, and gamma. These parameters are tuned within defined ranges to balance model complexity and accuracy. The PSO process uses MSE as the objective function, with 10 particles and 10 iterations to maintain computational efficiency. While baseline models such as KNN, RF, SVR, GBR, and XGBoost complete training in about 4 s, the proposed method requires around 40 s due to optimization overhead. However, this additional cost yields significantly improved predictive performance. The need for hyperparameter tuning through PSO arises from the observed performance gap between XGBoost's accuracy in SOH prediction and its degraded performance in RUL prediction. As seen in Section "Results and discussion", XGBoost performed better than other baseline models in SOH and RUL prediction, but its performance was not satisfactory enough for RUL prediction. By introducing PSO to optimize the hyperparameters, we achieve superior results, improving RUL prediction accuracy and consistency. This tuning process is pivotal in ensuring robust performance across various datasets. To further ensure robustness, a callback mechanism stores the best performing parameter set, and early stopping prevents unnecessary computation when no improvement is observed. Models are evaluated using multiple performance metrics, including MSE, MAE, MAPE, RMSE, NRMSE, and R^2, to provide a comprehensive assessment. For fairness, baseline models were carefully configured rather than relying on default parameters. While XGBoost performed best among conventional methods, its accuracy was highly sensitive to hyperparameter choices, particularly for RUL prediction. By integrating PSO, the proposed Battery-Insight-PSO model achieves consistent and superior performance across all datasets. The optimized hyperparameters obtained for each dataset are summarized in Table 1, and the results are compared against baseline algorithms and prior studies to validate the model's effectiveness.

The flowchart showing the overall Battery-Insight-PSO methodology is provided in Fig. 10. The corresponding pseudocode is given below for a clearer understanding of the step-by-step process-

System specifications and configuration

All experiments were carried out on Kaggle. Table 2 represents technologies and tools used in the field of implementation, as well as configuration steps for them. The experimentation was done on a 64-bit Linux operating system with an Intel(R) Xeon(R) CPU running at

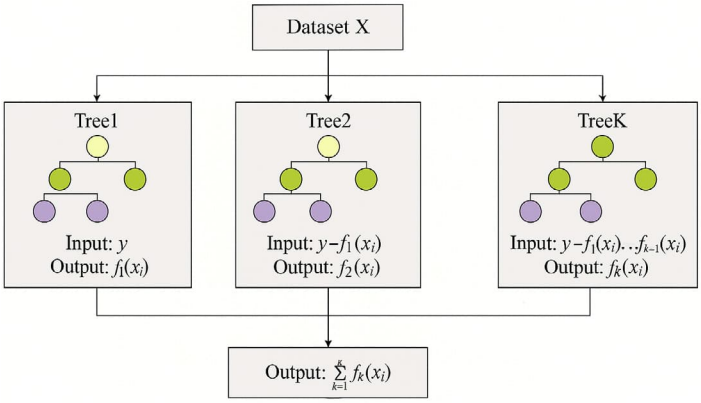

Fig. 8. Framework of XGBoost algorithm.

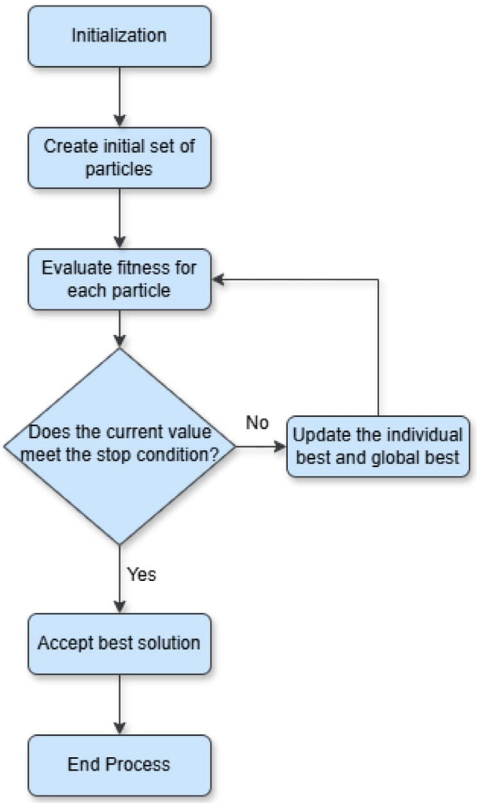

Fig. 9. Flow chart of PSO.

Table 1
Hyperparameter tuning results for battery-insight-PSO Across NASA and HNEI datasets.

Hyperparameter	B5	B7	B18	B47	B48	HNEI
Number of Estimators	320	410	280	195	65	250
Learning Rate	0.33	0.20	0.18	0.45	0.35	0.12
Maximum Depth	14.98	10.54	8.08	13.61	9.36	11.73
Gamma	0.015	0.008	0.025	0.005	0.01	0.03

2.00 GHz using the above strategy. This contained 2 T T4 GPUs along with 32 GB of RAM and was using Python 3.10.14 as the programming language.

Algorithm 1: Battery Health Prediction Workflow (Battery-Insight-PSO)

Input : NASA Li-ion Battery Aging Datasets (B5, B7, B18, B47, B48), Hawaii NMC-LCO Battery Dataset (HNEI)
Output: Trained XGBoost model with optimized hyperparameters, evaluation metrics

Function `BatteryHealthPrediction`:
  Step 1: Load and Preprocess Datasets;
  Load NASA Li-ion Battery Aging Datasets and Hawaii NMC-LCO Battery Dataset;
  Handle missing values and normalize the data;
  Select relevant features using domain knowledge and correlation analysis;
  Split each dataset into training (40%), validation (30%), and testing (30%);
  Step 2: Define Hyperparameter Bounds for XGBoost;
  Define parameter bounds for PSO optimization:;
    Upper Bound (UB) = [500, 0.5, 15, 0.5] ; `// num_estimators, learning_rate, max_depth, gamma`
    Lower Bound (LB) = [50, 0.01, 3, 0.0];
  Step 3: Optimize XGBoost Hyperparameters using PSO;
  Define PSO parameters and objective function (minimize validation MSE);
  Run PSO to obtain best hyperparameters for XGBoost;
  Step 4: Train Model with Optimized Hyperparameters;
  Train XGBoost model on the 40% training data using best parameters from PSO;
  Step 5: Evaluate Model Performance;
  Evaluate on 20% test data and compute metrics:
    - MAE, MSE, RMSE, NRMSE, MAPE, R^2

  Step 6: Compare with Other Algorithms;
  Train and evaluate the following models using the same train/val/test splits:; Random Forest (RF), Gradient Boosting Regressor (GBR), Support Vector Regressor (SVR), K-Nearest Neighbors (KNN), XGBoost;
  Compare Battery-Insight-PSO performance against the above models using evaluation metrics;

Table 2
System specification.

System component	Description
Operating System	64-bit Linux
CPU	Intel(R) Xeon(R) CPU
CPU GHz	2.00 GHz
GPU	Tesla T4 (x2)
RAM	32 GB
Programming Language	Python 3.10.14

Results and discussion

This chapter presents the test results of the Battery-Insight-PSO model in predicting battery status across multiple datasets. Various evaluation metrics are employed to assess the model's performance and to facilitate a comprehensive comparison with existing methods for battery health monitoring and life prediction. The results demonstrate that Battery-Insight-PSO consistently outperforms other approaches,

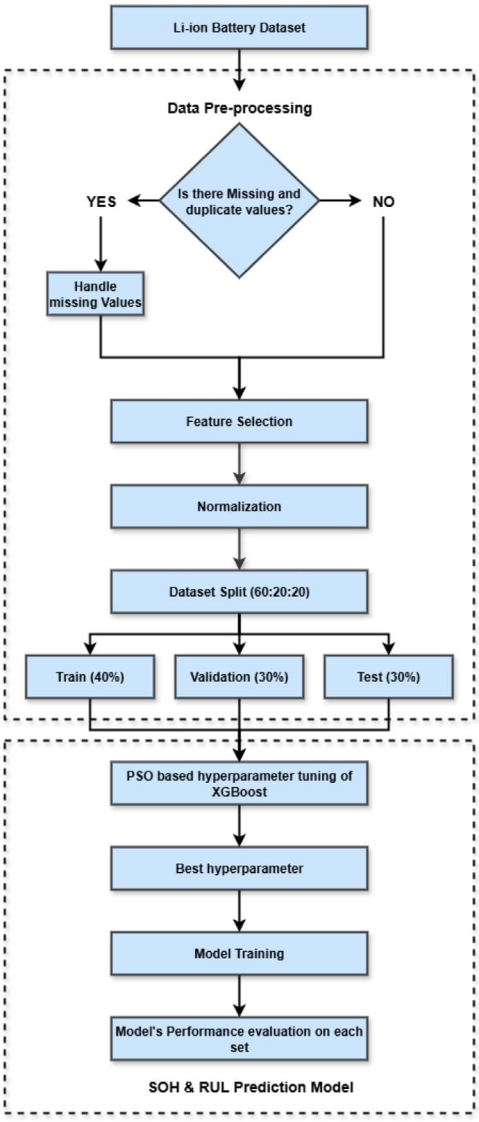

Fig. 10. Framework of Battery-Insight-PSO.

highlighting its effectiveness and robustness. Additionally, detailed descriptions of all the evaluation metrics used in the analysis are provided in Section "Supplementary Materials".

Table 3 compares the performance of machine learning models—GBR, KNN, SVR, RF, XGBoost, and the proposed Battery-Insight-PSO—across training, validation, and testing phases using key metrics: MSE, RMSE, MAE, MAPE, NRMSE, and R^2. Among the baseline models, XGBoost shows the best consistency and performance. In the training phase, XGBoost achieves the lowest MSE (0.879), RMSE (0.976), and NRMSE (0.0163), along with a high R^2 of 0.9985, outperforming models like SVR and RF. In the validation phase, XGBoost continues to lead with an MSE of 0.885, RMSE of 0.979, and MAE of 0.524, demonstrating its robustness. In the testing phase, XGBoost maintains strong performance with an MSE of 0.918, RMSE of 0.995, MAE of 0.530, MAPE of 3.695%, and R^2 of 0.9984. When compared to XGBoost, the Battery-Insight-PSO model shows substantial improvements. During training, Battery-Insight-PSO reduces MSE by 99.94%, RMSE by 97.55%, and MAE by 97.76%. In the validation phase, it achieves a 99.85% reduction in MSE, a 96.23% decrease in RMSE, and a 97.38% improvement in MAE. Most notably, during testing, Battery-Insight-PSO reduces MSE by 99.88%, RMSE by 96.53%, and MAE by 97.44%, while drastically lowering MAPE from 3.695% to 0.05502%, reflecting a 98.51% improvement. The R^2 score also improves marginally from 0.9984 to 0.9998, indicating near-perfect accuracy. These results highlight that while XGBoost performs well, the Battery-Insight-PSO framework, with PSO optimization, provides a significant enhancement in accuracy and robustness across all phases.

Figs. 11 and 17 present the RUL and SOH prediction performance for the B05 battery dataset on the test and validation sets, respectively. In Fig. 11, the real RUL is represented in bold red to emphasize the ground truth, with the predictions from the XGBoost model shown in lime green and those from the Battery-Insight-PSO method in blue. This color coding is consistent across all figures related to RUL and SOH to ensure clarity and ease of comparison between the models. It can also be observed from Figure 17 that both Battery-Insight-PSO and XGBoost perform well in SOH prediction, exhibiting similar accuracy. However, the performance of XGBoost degrades significantly when it comes to RUL prediction, as shown in Fig. 11. Therefore, for each battery set, the SOH prediction figures are included in the Supplementary Materials, labeled as Section "Supplementary Materials".

The figures show that while XGBoost captures the general trend of the real RUL curve, it deviates significantly in the later stages of the battery's life cycle, indicating its struggle to capture complex nonlinear degradation patterns. In contrast, the Battery-Insight-PSO model provides predictions that align more closely with the actual RUL, particularly during the early and mid-life stages, demonstrating better generalization and robustness. This comparison highlights XGBoost's limitations in accurately predicting RUL.

Table 4 compares the performance of various machine learning models—SVR, RF, GBR, KNN, XGBoost, and the proposed Battery-Insight-PSO—on the NASA battery dataset (B7) for SOH and RUL prediction. XGBoost performs well across all phases, achieving low error values and a high R^2 of 0.9990 during training. However, Battery-Insight-PSO significantly outperforms XGBoost in every metric. In the training phase, Battery-Insight-PSO reduces MSE by 99.97%, RMSE by 97.36%, and MAE by 98.01%. The model further improves these metrics during validation and testing, achieving reductions of 99.91% in MSE, 96.60% in RMSE, and 97.84% in MAE, with a drastic improvement in MAPE (99.58%). Fig. 12 demonstrates the RUL prediction, where Battery-Insight-PSO closely aligns with the ground truth, showing superior accuracy and robustness compared to XGBoost. Similarly, Figure 18 illustrates the SOH prediction, highlighting the model's strong generalization and enhanced performance. These results confirm the effectiveness of Battery-Insight-PSO over conventional models.

Table 5 provides a clear comparison of ML models applied to the NASA Battery Dataset (B18) for SOH and RUL prediction. XGBoost performs well, achieving a test MSE of 1.335, RMSE of 1.153, MAE of 0.673, MAPE of 0.775%, and R^2 of 0.9986. In contrast, models like RF and KNN show moderate performance, while SVR exhibits poor results with high error values and R^2 around 0.64. In comparison, the Battery-Insight-PSO model shows remarkable improvement. On the test set, it reduces MSE by 99.83%, RMSE by 95.74%, MAE by 98.81%, and MAPE by 93.80%, with R^2 increasing to 0.9998, indicating near-perfect predictive accuracy. Figs. 13 and 19 further validate these findings. Fig. 13 shows the RUL prediction, where Battery-Insight-PSO closely tracks the actual values. Similarly, Figure 19 demonstrates the SOH prediction, highlighting the model's strong generalization and precision across both validation and test sets.

Tables 6 and 7 summarize the performance of various machine learning models on the NASA battery datasets B47 and B48 for SOH and RUL prediction. XGBoost consistently outperforms traditional models like RF, GBR, and KNN, while SVR lags with high error rates and low R^2 scores. On dataset B47, XGBoost achieves excellent accuracy with a test MSE of 0.801, RMSE of 0.902, MAE of 0.512, MAPE of 1.21%, and R^2 of 0.9980. On dataset B48, XGBoost maintains similar precision with an MSE of 0.791, RMSE of 0.891, MAE of 0.511, MAPE of

Table 3
Performance metrics of ML algorithms for SOH and RUL prediction for NASA battery dataset (B5).

Model	Dataset	MSE	RMSE	NRMSE	MAE	MAPE (%)	R^2
GBR	Train	2.789	1.733	0.0294	0.902	6.012	0.9945
	Validation	2.857	1.760	0.0299	0.919	6.215	0.9942
	Test	2.961	1.788	0.0305	0.933	6.283	0.9940
KNN	Train	2.342	1.568	0.0271	0.519	4.497	0.9960
	Validation	3.962	2.021	0.0342	0.707	5.612	0.9944
	Test	4.298	2.098	0.0364	0.743	5.917	0.9940
SVR	Train	13.821	3.945	0.0664	1.775	11.215	0.5402
	Validation	13.485	3.777	0.0649	1.706	10.962	0.5451
	Test	14.323	3.977	0.0676	1.834	11.567	0.5381
RF	Train	1.257	1.165	0.0196	0.518	4.596	0.9975
	Validation	8.831	3.122	0.0521	1.378	7.718	0.9850
	Test	8.315	2.952	0.0511	1.372	7.935	0.9864
XGBoost	Train	0.879	0.976	0.0163	0.520	3.467	0.9985
	Validation	0.885	0.979	0.0163	0.524	3.563	0.9985
	Test	0.918	0.995	0.0168	0.530	3.695	0.9984
Battery-Insight-PSO	Train	0.000552	0.02401	0.000446	0.01163	0.04812	0.9998
	Validation	0.001301	0.03704	0.000553	0.01373	0.05608	0.9998
	Test	0.001102	0.03457	0.000556	0.01361	0.05502	0.9998

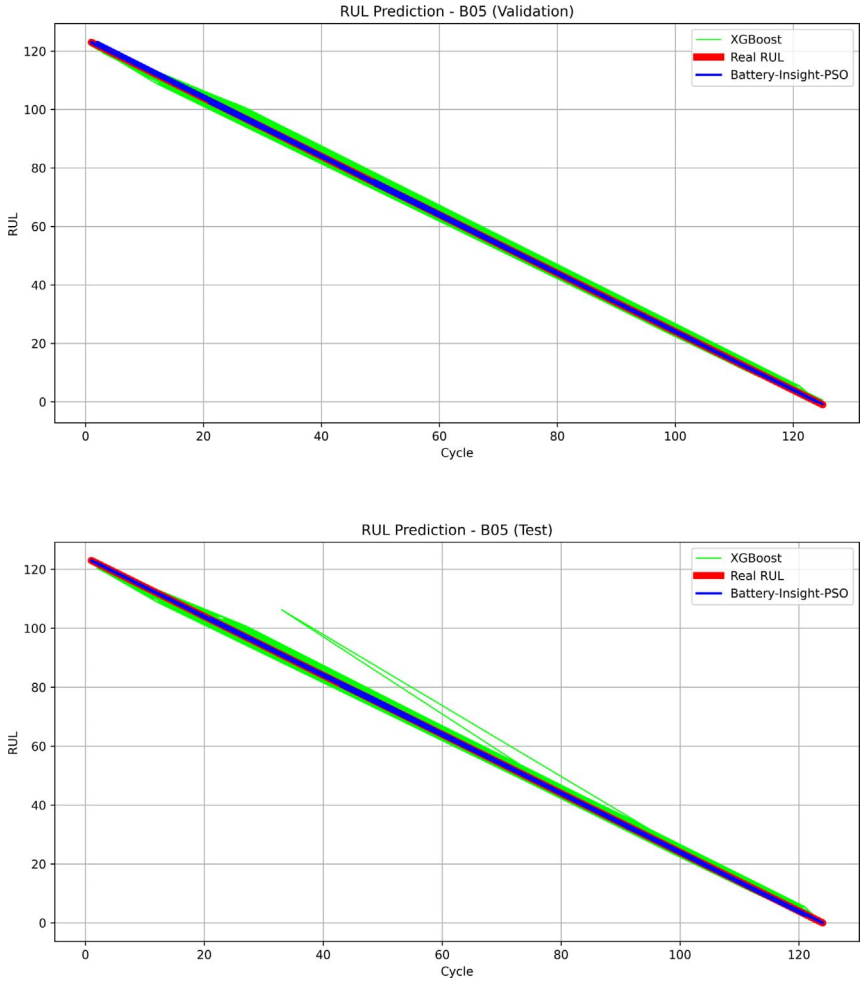

Fig. 11. RUL prediction of NASA dataset (B5) on validation and test set.

1.61%, and R^2 of 0.9979. Battery-Insight-PSO, however, demonstrates a significant leap in performance, achieving near-zero error in both datasets. For B47, it records MSE of 1.9e−9, RMSE of 4.53e−5, MAE of 3.32e−5, and MAPE of 0.00181%, with R^2 of 0.9998. For B48, it achieves an MSE of 1.81e−9, RMSE of 4.51e−5, MAE of 3.43e−5, MAPE of 0.00191%, and R^2 of 0.9998. Compared to XGBoost, Battery-Insight-PSO improves all metrics by over 99.99%. Figs. 14 and 15 show RUL predictions for B47 and B48, where Battery-Insight-PSO exhibits minimal error and excellent generalization. Figures 20 and 21 show SOH predictions, where the model closely tracks the ground truth, highlighting its superior accuracy.

Table 8 compares the performance of multiple machine learning models—KNN, GBR, SVR, RF, XGBoost, and the proposed Battery-Insight-PSO—on the HNEI battery dataset for RUL prediction. Unlike the NASA datasets, the HNEI dataset introduces a different data structure, making this evaluation crucial for assessing model generalizability

Table 4
Performance metrics of ML algorithms for SOH and RUL prediction for NASA battery dataset (B7).

Model	Dataset	MSE	RMSE	NRMSE	MAE	MAPE (%)	R^2
SVR	Train	13.687	3.985	0.0489	1.843	54.127	0.6103
	Validation	13.291	3.748	0.0476	1.792	46.215	0.6124
	Test	15.118	4.123	0.0510	1.924	58.104	0.6092
RF	Train	1.938	1.431	0.0185	0.643	20.124	0.9976
	Validation	12.397	3.637	0.0464	1.655	47.137	0.9886
	Test	14.527	3.807	0.0501	1.724	55.239	0.9861
GBR	Train	2.834	1.723	0.0221	0.934	18.045	0.9979
	Validation	2.773	1.703	0.0218	0.912	16.108	0.9979
	Test	2.963	1.762	0.0228	0.962	20.143	0.9976
KNN	Train	2.257	1.543	0.0196	0.573	9.823	0.9983
	Validation	3.698	1.936	0.0251	0.763	12.247	0.9970
	Test	4.092	2.024	0.0276	0.792	21.015	0.9965
XGBoost	Train	1.283	1.132	0.0151	0.633	11.523	0.9990
	Validation	1.274	1.133	0.0151	0.631	10.234	0.9990
	Test	1.353	1.162	0.0157	0.663	12.921	0.9989
Battery-Insight-PSO	Train	0.000445	0.02982	0.000331	0.01263	0.04912	0.9998
	Validation	0.001081	0.03803	0.000552	0.01405	0.05607	0.9998
	Test	0.001198	0.03953	0.000443	0.01432	0.05457	0.9998

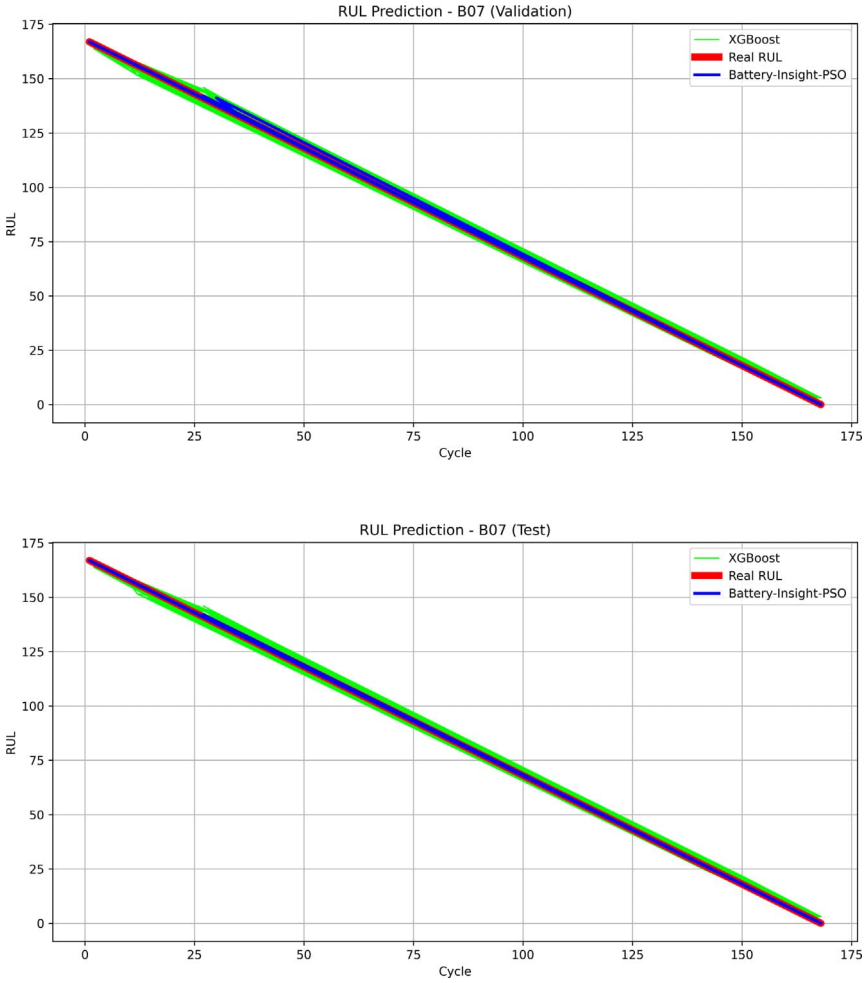

Fig. 12. RUL prediction of NASA dataset (B7) on validation and test set.

and robustness. XGBoost demonstrates excellent performance across all phases, achieving low error rates and high R^2 values. In training, it achieves an MAE of 0.743, MSE of 9.821, RMSE of 3.112, and R^2 of 0.9994, maintaining high accuracy during validation and testing with R^2 of 0.9993. In contrast, KNN and RF show uneven performance. KNN struggles on the test set, with MAE of 6.187 and RMSE of 26.433, while RF performs well during training but generalizes poorly, with test MAE of 8.591 and RMSE of 19.512. SVR continues to underperform, with the highest error metrics and the lowest R^2. Battery-Insight-PSO delivers a balanced performance, with slightly higher training error than XGBoost (MAE: 0.372, RMSE: 0.742), but better generalization on the test set. It reduces RMSE by 51.12% (2.541 vs. 5.206) and MAE by 40.97% (1.089 vs. 1.847), despite XGBoost having a slightly higher R^2. Fig. 16 compares the RUL predictions, where Battery-Insight-PSO aligns more closely with the true RUL than XGBoost, which shows noticeable fluctuations and underestimates in several regions. This

Table 5
Performance metrics of ML algorithms for SOH and RUL prediction for NASA battery dataset (B18).

Model	Dataset	MSE	RMSE	NRMSE	MAE	MAPE (%)	R^2
RF	Train	5.287	2.392	0.0412	1.182	1.856	0.9921
	Validation	36.128	6.094	0.1083	3.215	5.122	0.9592
	Test	34.298	5.921	0.1062	3.031	4.213	0.9611
GBR	Train	10.123	3.234	0.0562	1.784	4.013	0.9911
	Validation	10.912	3.285	0.0581	1.817	4.217	0.9906
	Test	9.839	3.124	0.0552	1.694	3.418	0.9916
KNN	Train	9.947	3.176	0.0561	1.352	2.215	0.9911
	Validation	16.231	4.107	0.0713	1.847	3.012	0.9861
	Test	15.415	3.973	0.0682	1.724	2.708	0.9871
SVR	Train	44.217	6.631	0.1186	3.312	10.623	0.6404
	Validation	46.089	6.789	0.1222	3.406	11.012	0.6381
	Test	42.129	6.487	0.1163	3.211	9.237	0.6423
XGBoost	Train	1.416	1.192	0.0212	0.693	0.948	0.9985
	Validation	1.463	1.213	0.0221	0.712	1.057	0.9984
	Test	1.335	1.153	0.0211	0.673	0.775	0.9986
Battery-Insight-PSO	Train	0.000113	0.01321	0.000223	0.00481	0.0201	0.9998
	Validation	0.00371	0.06312	0.00111	0.00862	0.0412	0.9998
	Test	0.00231	0.04908	0.000772	0.00803	0.0481	0.9998

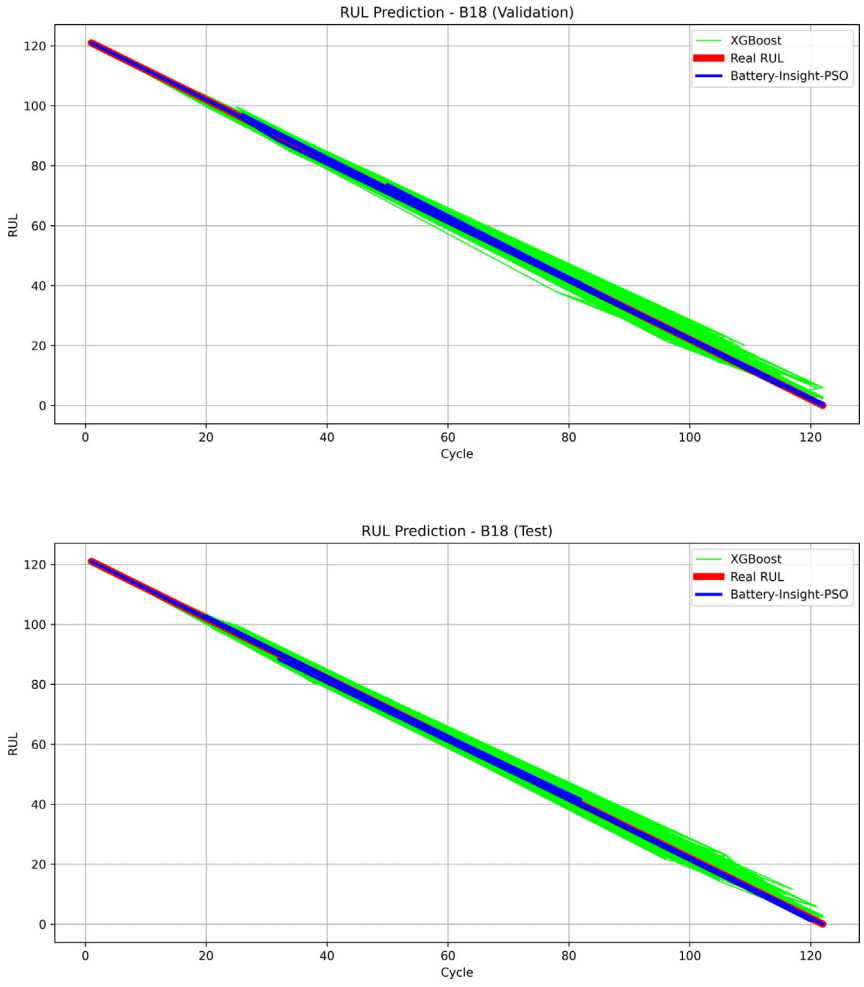

Fig. 13. RUL prediction of NASA dataset (B18) on validation and test set.

highlights the effectiveness of the PSO optimization in refining the model's performance.

Table 9 compares the performance of the Battery-Insight-PSO model with other existing models across various NASA battery datasets (B5, B7, B18, B47, and B48), using MSE, RMSE, NRMSE, MAE, and R^2 metrics. Battery-Insight-PSO consistently delivers excellent results, achieving very low MSE and RMSE values across all test datasets. The model excels in both SOH and RUL prediction, with R^2 values of 0.9999 in all experiments, indicating near-perfect accuracy. Compared to other models, Battery-Insight-PSO outperforms all in terms of accuracy, with near-zero error metrics and high R^2 scores across all datasets. This positions it as the top-performing model for battery health monitoring and prediction tasks.

Table 10 shows the comparison of Battery-Insight-PSO with existing models that have been trained on the Hawaii NMC-LCO 18650 battery dataset. Evaluation metrics are used like MAE, MSE, RMSE, MAPE,

Table 6
Performance metrics of ML algorithms for SOH and RUL prediction for NASA battery dataset B47.

Model	Dataset	MSE	RMSE	NRMSE	MAE	MAPE (%)	R^2
RF	Train	4.683	2.162	0.0653	1.082	3.53	0.9861
	Validation	33.127	5.748	0.1725	2.891	12.26	0.9102
	Test	30.912	5.562	0.1632	2.753	8.93	0.9163
GBR	Train	3.672	1.923	0.0592	1.133	2.91	0.9911
	Validation	3.812	1.952	0.0602	1.162	3.42	0.9908
	Test	3.592	1.891	0.0583	1.123	2.63	0.9913
KNN	Train	10.412	3.228	0.0985	1.173	2.62	0.9762
	Validation	17.123	4.135	0.1288	1.625	6.03	0.9602
	Test	16.342	4.048	0.1245	1.564	3.53	0.9612
SVR	Train	50.143	7.083	0.2155	3.608	16.95	0.3105
	Validation	51.276	7.152	0.2183	3.625	20.12	0.3052
	Test	48.897	6.987	0.2128	3.498	14.05	0.3092
XGBoost	Train	0.823	0.908	0.0273	0.523	1.31	0.9981
	Validation	0.862	0.934	0.0274	0.542	1.52	0.9979
	Test	0.801	0.902	0.0272	0.512	1.21	0.9980
Battery-Insight-PSO	Train	2.0e−9	4.52e−5	1.32e−6	3.21e−5	0.00162	0.9998
	Validation	1.9e−9	4.51e−5	1.31e−6	3.31e−5	0.00179	0.9998
	Test	1.9e−9	4.53e−5	1.32e−6	3.32e−5	0.00181	0.9998

Table 7
Performance metrics of ML algorithms for SOH and RUL prediction for NASA battery dataset B48.

Model	Dataset	MSE	RMSE	NRMSE	MAE	MAPE (%)	R^2
RF	Train	8.312	2.883	0.0882	1.513	5.53	0.9761
	Validation	59.137	7.694	0.2358	4.056	17.24	0.8463
	Test	57.278	7.562	0.2294	3.907	13.53	0.8481
GBR	Train	3.724	1.936	0.0603	1.122	3.02	0.9911
	Validation	3.791	1.953	0.0612	1.142	3.61	0.9906
	Test	3.843	1.962	0.0613	1.161	2.82	0.9905
KNN	Train	15.518	3.947	0.1212	1.463	2.72	0.9641
	Validation	26.047	5.113	0.1555	2.053	5.72	0.9401
	Test	24.514	4.956	0.1493	1.874	3.72	0.9421
SVR	Train	71.124	8.438	0.2635	4.089	19.72	0.2604
	Validation	70.893	8.421	0.2621	4.074	23.52	0.2552
	Test	69.524	8.337	0.2592	3.958	17.42	0.2583
XGBoost	Train	0.772	0.883	0.0281	0.503	1.81	0.9980
	Validation	0.801	0.897	0.0281	0.521	2.11	0.9979
	Test	0.791	0.891	0.0280	0.511	1.61	0.9979
Battery-Insight-PSO	Train	1.81e−9	4.51e−5	1.32e−6	3.42e−5	0.00191	0.9998
	Validation	1.82e−9	4.52e−5	1.32e−6	3.33e−5	0.00191	0.9998
	Test	1.81e−9	4.51e−5	1.32e−6	3.43e−5	0.00191	0.9998

Table 8
Performance metrics of ML algorithms for RUL prediction for HNEI dataset.

Model	Dataset	MAE	MSE	RMSE	NRMSE	MAPE (%)	R^2
KNN	Train	4.732	312.189	17.723	0.0331	10.82	0.9917
	Validation	6.218	372.451	19.298	0.0357	11.91	0.9898
	Test	6.187	699.862	26.433	0.0498	13.76	0.9849
GBR	Train	2.548	27.189	5.212	0.0095	1.902	0.9995
	Validation	2.751	32.487	5.702	0.0106	2.103	0.9994
	Test	2.703	31.526	5.648	0.0103	2.008	0.9994
SVR	Train	16.048	2001.489	44.512	0.0881	50.11	0.9647
	Validation	15.789	1849.632	42.889	0.0849	52.14	0.9628
	Test	16.823	2401.274	48.502	0.0901	60.22	0.9599
RF	Train	3.598	50.237	7.124	0.0126	4.698	0.9984
	Validation	8.721	519.672	22.803	0.0402	19.52	0.9899
	Test	8.591	421.238	19.512	0.0361	15.04	0.9910
XGBoost	Train	0.743	9.821	3.112	0.0055	0.653	0.9994
	Validation	1.899	27.481	5.298	0.0095	1.452	0.9993
	Test	1.847	26.967	5.206	0.0095	1.653	0.9993
Battery-Insight-PSO	Train	0.372	0.532	0.742	0.0013	6.503	0.9994
	Validation	1.098	6.097	2.549	0.0046	7.312	0.9994
	Test	1.089	5.987	2.541	0.0045	8.281	0.9994

R^2, and NRMSE. The table reveals that Battery-Insight-PSO achieves the best overall prediction performance with smaller error values and higher R^2 scores consistently compared to other models in estimating the RUL of Li-ion batteries. This evidence denotes the accuracy and efficiency of the proposed model for RUL estimation in battery health monitoring systems.

In summary, the proposed Battery-Insight-PSO model demonstrates consistent and superior performance across all evaluated datasets, including both the NASA and HNEI battery datasets. While XGBoost stands out as the strongest baseline among traditional machine learning models, delivering high R^2 values and stable performance, Battery-Insight-PSO significantly surpasses it in nearly every key metric. Across

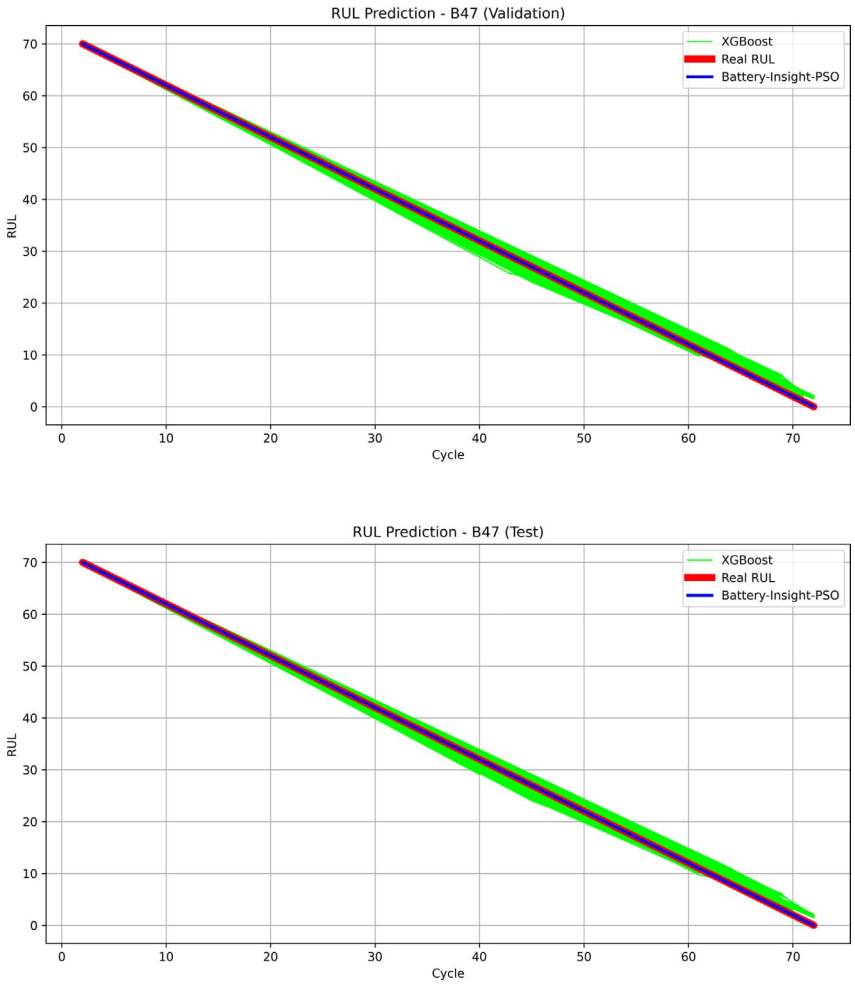

Fig. 14. RUL prediction of NASA dataset (B47) on validation and test set.

Table 9
Comparison of Battery-Insight-PSO with Existing Models for NASA Datasets.

Algorithm	Dataset	MAE	MSE	RMSE	NRMSE	MAPE	R^2
Battery-Insight-PSO	B5	0.01361	0.001102	0.03457	0.000556	0.05502	0.9998
	B7	0.01432	0.001198	0.03953	0.000443	0.05457	0.9998
	B18	0.00803	0.00231	0.04908	0.000772	0.0481	0.9998
	B47	3.32e−5	1.9e−9	4.53e−	5 1.32e−6	0.00181	0.9998
	B48	3.43e−5	1.81e−9	4.51e−5	1.32e−6	0.00191	0.9998
GRU + MHA + Ridge [13]	B5	0.005	0.0706	/	/	/	0.9964
	B7	0.012	0.105	/	/	/	0.9749
	B18	0.012	0.109	/	/	/	0.9597
AT-CNN-BiLSTM [7]	B5	/	/	0.006	/	/	0.9959
	B7	/	/	0.0053	/	/	0.9956
DilatedCNN-BiGRU [8]	B7	0.008	/	0.0143	/	/	/
LSTM [4]	B7	/	/	0.0085	/	/	/
	B18	/	/	0.0122	/	/	/
	B47	/	/	0.0368	/	/	/
	B48	/	/	0.0427	/	/	/

Table 10
Comparison of battery-insight-PSO with existing models over HNEI dataset.

Model	MAE	MSE	RMSE	NRMSE	MAPE	R^2
Battery-Insight-PSO	1.089	5.987	2.541	0.0045	0.08281	0.9994
RF with HHO [6]	49.306	3680	60.67	0.21	0.48	0.96
Light BGM with HHO [6]	47.95	3512	59.26	0.20	0.47	0.97
XGBoost [17]	8.191	245.993	15.684	–	–	0.99
Random Forest [21]	–	14.1186	3.7574	2.0930	–	0.99

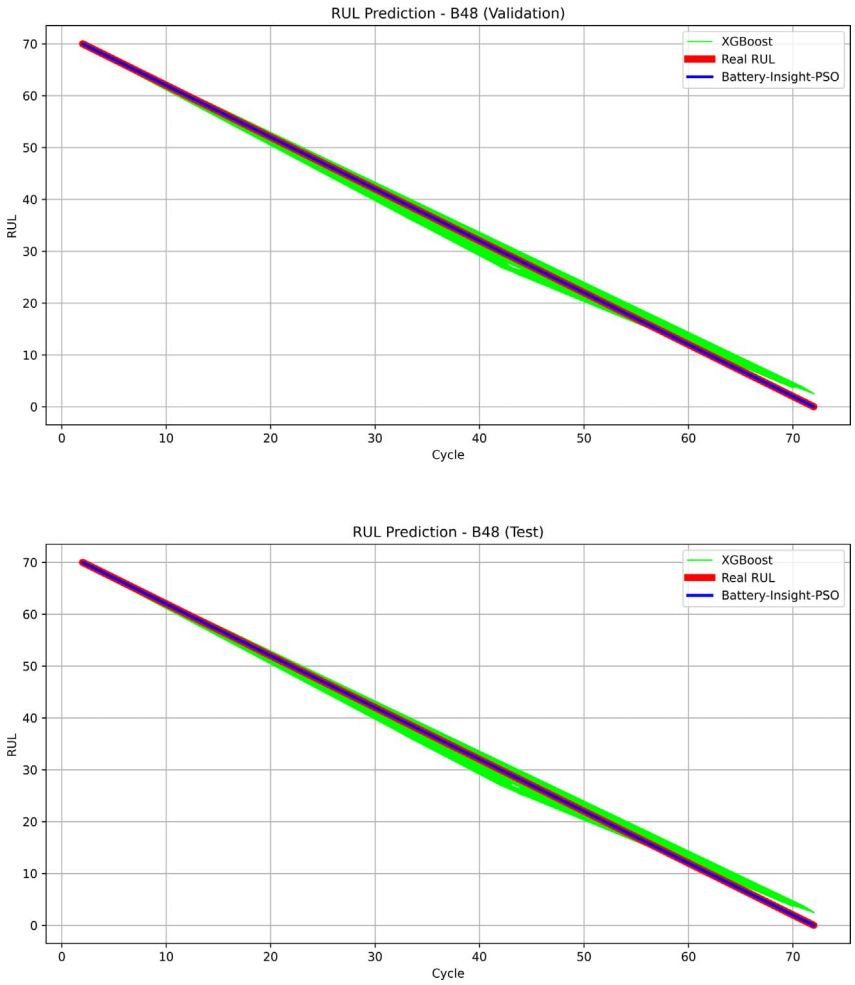

Fig. 15. RUL prediction of NASA dataset (B48) on validation and test set.

all NASA subsets (B5, B7, B18, B47, and B48), the model achieves drastic reductions in MSE, RMSE, MAE, and MAPE—often exceeding 99% improvement—while maintaining near-perfect R^2 scores of 0.9999, indicating exceptional predictive accuracy and robustness. On the structurally different HNEI dataset, Battery-Insight-PSO continues to outperform, achieving lower absolute errors than XGBoost while preserving high correlation with actual values. These results confirm the model's ability to generalize effectively across varying battery chemistries and data characteristics, positioning Battery-Insight-PSO as a powerful, reliable solution for SOH and RUL prediction in real-world battery management systems.

Conclusion and future scope

One of the main reasons for predicting SOH and RUL is to improve system reliability and safety, since this allows the development of effective maintenance strategies. The increasing tendency among different disciplines to rely on ML approaches for systemic asset SOH and RUL prediction has been explored in this work. Using historical data, organizations can improve maintenance planning and save significantly in costs due to the prevention of equipment failures. This work utilizes the XGBoost algorithm with the PSO algorithm to develop a model known as Battery-Insight-PSO for accurate RUL estimation of Li-ion batteries. This dataset comes from a real-life collection of NASA's and HNEI's Li-ion batteries and introduces research regarding the prediction of SOH and RUL by extensive experiments on ML algorithms like SVR, RFR, GBR, KNN, and the XGBoost regressor. Among the baseline models, XGBoost consistently performed best, particularly in SOH prediction, but its accuracy in RUL prediction was not satisfactory. This limitation motivated the use of PSO for hyperparameter tuning, enabling the model to overcome RUL prediction challenges and achieve improved efficiency in both SOH and RUL tasks. Keeping the hyperparameters of baseline or existing algorithms fixed does not yield accurate predictions; while parameter adjustment improves results, manual tuning is inefficient and suboptimal. Therefore, in this study, PSO was used to optimize the hyperparameters of the best-performing baseline algorithm—XGBoost. The high R^2 values very close to 0.9998 across training, validation, and test sets, as well as the low error metrics demonstrate that the Battery-Insight-PSO model is accurate enough for practical application of SOH and RUL prediction. Thus, this approach can serve as a valuable tool for CBM in applications ranging from electric vehicles to renewable energy systems.

This research could be further expanded by examining other types of batteries in the future and performance evaluation under different operating conditions. Such an extension would add further insights into the evaluation of this proposal regarding how well it works in a variety of settings. Furthermore, this study can be extended by investigating advanced optimization algorithms and feature selection techniques to improve the accuracy in SOH and RUL prediction for better preservation management of energy storage systems. This ultimately can help make the change over to new clean energy solutions and inspire innovation in what is one of the most important sectors for reducing emissions while aligning stakeholders across supply chains.

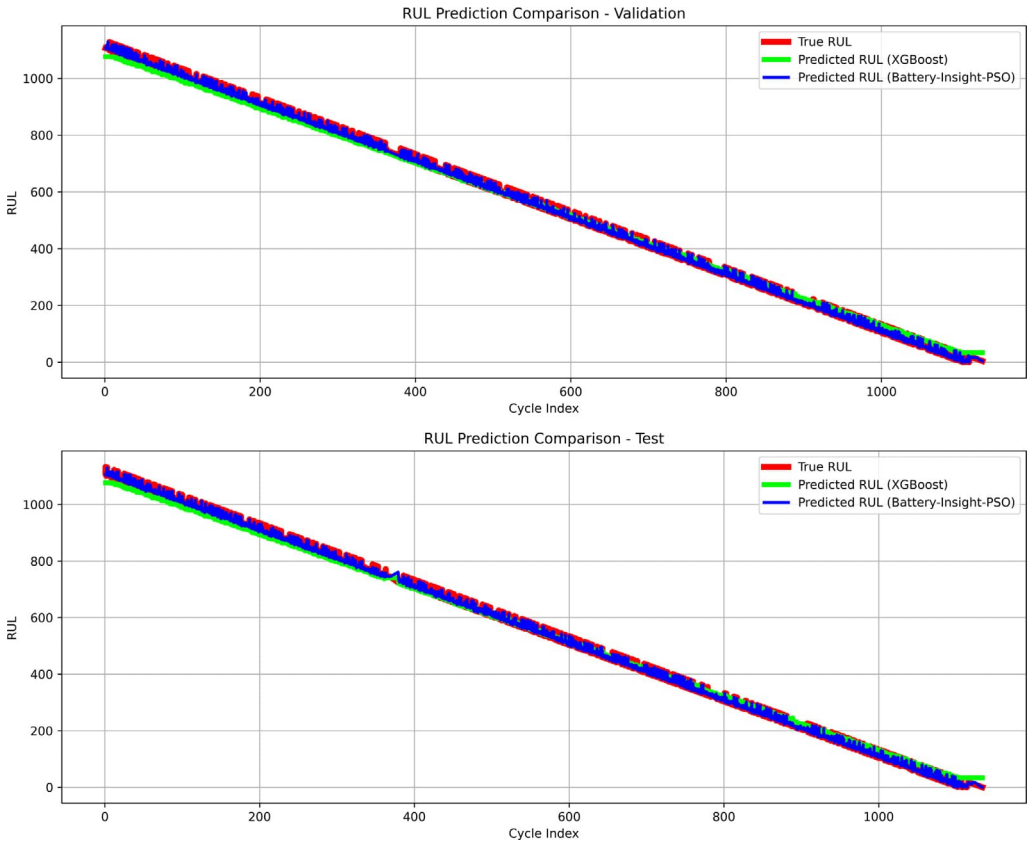

Fig. 16. RUL prediction of HENI dataset on validation and test set.

CRediT authorship contribution statement

Md Fazle Hasan Shiblee: Writing – review & editing, Writing – original draft, Visualization, Validation, Methodology, Investigation, Formal analysis, Data curation, Conceptualization. **Hannu Laaksonen:** Writing – review & editing, Supervision, Conceptualization.

Declaration of competing interest

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

Supplementary data

Additional figures, experimental details, and supporting content referenced in the main text. This section includes extensive figures, additional experimental details, or any other supplementary content referenced in the main text.

Supplementary material related to this article can be found online at https://doi.org/10.1016/j.fub.2025.100114.

Data availability

The used open data sets are described in the manuscript.

References

[1] Z.M. Ali, M. Calasan, F.H. Gandoman, F. Jurado, S.H.A. Aleem, Ain Shams Eng. J. 15 (2024) 102442.
[2] Y. Tavakol-Moghaddam, M. Boroushaki, M. and Astaneh, Results Eng. 23 (2024) 102532.
[3] Y. Ma, M. Yao, H. Liu, Z. Tang, J. Energy Storage 52 (2022) 104750.
[4] P. Li, Z. Zhang, Q. Xiong, B. Ding, J. Hou, D. Luo, Y. Rong, S. Li, J. Power Sources 459 (2020) 228069.
[5] M. Catelani, L. Ciani, R. Fantacci, G. Patrizi, B. Picano, IEEE Trans. Instrum. Meas. 70 (2021) 1–11.
[6] S. Jafari, Y.C. Byun, IEEE Access (2023).
[7] F.M. Zhao, D.X. Gao, Y.M. Cheng, Q. and Yang, Sci. Rep. 14 (29026) (2024).
[8] Z. Bao, J. Jiang, C. Zhu, M. Gao, Energies 15 (4399) (2022).
[9] Z. Yang, Y. Zhang, Y. Zhang, Future Batteries (2025) 100088.
[10] S. Ansari, M. Hannan, A. Ayob, M.G. Abdolrasol, M.A. Dar, J. Energy Storage 105 (2025) 114711.
[11] J. Wu, Z. Liu, Y. Zhang, D. Lei, Y. Zhang, Future Batteries (2025) 100074.
[12] R. Wang, M. Zhu, X. Zhang, H. and Pham, J. Energy Storage 59 (2023) 106457.
[13] A. Aljohani, S. and Aljohani, Energy Rep. 14 (2025) 294–309.
[14] S. Jafari, Y.C. Byun, S. Ko, IEEE Access 11 (2023) 131950–131963.
[15] R. Zheng, B. Yang, Y. Qian, H. Li, D. Gao, L. and Jiang, J. Energy Storage 114 (2025) 115891.
[16] B. Chen, Y. Liu, B. Xiao, J. Energy Storage 98 (2024) 113074.
[17] K. Karthick, S. Ravivarman, R. Priyanka, World Electric Vehicle J. 15 (60) (2024).
[18] G.Dos. Reis, C. Strange, M. Yadav, S. Li, Energy AI 5 (2021) 100081.
[19] S. Sifath, T. Islam, M. Erfan, S.K. Dey, M.M.U. Islam, M. Samsuddoha, T. and Rahman, Natural Lang. Processing J. 9 (2024) 100111.
[20] A.G. Gad, Arch. Comput. Methods Eng. 29 (2022) 2531–2561.
[21] J.C. Sekhar, B. Domathoti, E.D. Santibanez Gonzalez, Sustainability 15 (15283) (2023).

Further reading

[1] J.C. Huang, K.M. Ko, M.H. Shu, B.M. Hsu, Neural Comput. Appl. 32 (2020) 5461–5469.