

Article

Lithium Battery SOH Estimation Based on Manifold Learning and LightGBM

Mei Zhang, Jun Yin * and Tao Feng

College of Electrical and Information Engineering, Anhui University of Science and Technology (AUST), Huainan 232002, China; xwz098@163.com (M.Z.); ft985211@163.com (T.F.)

* Correspondence: yj19955433079@163.com; Tel.: +86-199-5543-3079

Abstract: In order to accurately identify the state of health (SOH) and remaining useful life (RUL) of lithium-ion batteries, this paper proposes an SOH estimation algorithm for lithium-ion batteries based on stream learning and LightGBM. To address the problem of inconsistent data length, which makes it difficult to establish the state mapping relationship between degraded data and health state, the health factors in this paper are extracted from capacity degradation features, entropy features, and correlation coefficient features. Then, the landmark isometric mapping (L-ISOMAP) manifold learning algorithm is used to dimensionally reduce the input feature set and map the high-dimensional features to the low-dimensional space to solve the dimensional explosion problem. Finally, a LightGBM prediction model is developed to perform SOH prediction on different datasets, and the superiority of the multidimensional model is evaluated. The experimental results show that the goodness-of-fit is 0.98 and above, and the MSE values are below 4×10^{-4} . Comparing several prediction models, the LightGBM model has the best performance and better results in several indexes, such as MSE and RMSE. Under different working conditions, the proposed model in this paper has a goodness-of-fit of more than 0.98 in dataset B, which proves that the proposed model has a strong generalization ability.

Keywords: capacity degradation feature; SOH prediction; manifold learning; LightGBM model



Citation: Zhang, M.; Yin, J.; Feng, T. Lithium Battery SOH Estimation Based on Manifold Learning and LightGBM. *Appl. Sci.* **2023**, *13*, 6540. <https://doi.org/10.3390/app13116540>

Academic Editor: Xianpeng Wang

Received: 17 April 2023

Revised: 22 May 2023

Accepted: 25 May 2023

Published: 27 May 2023



Copyright: © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

As one of the most commonly used energy storage devices, lithium-ion batteries are widely used in aerospace, new energy vehicles, and portable electronic devices [1–5]. Due to the chemical reactions inside the battery and the influence of the external environment, the service life of the battery gradually ages or even fails with the use time. Therefore, timely prediction of the health condition and failure threshold of Li-ion batteries is beneficial for planning and managing batteries to ensure safe and reliable operation of devices. Therefore, accurate prediction of the state of health (SOH) and remaining useful life (RUL) of Li-ion batteries is of great significance in the field of Li-ion battery use [6–8].

In recent years, theories and technologies related to the fields of artificial intelligence, machine learning, and data mining have become more and more mature, and data-driven battery SOH prediction has received more and more attention. Researchers have used advanced machine learning and neural networks and other methods to establish the mapping relationships between battery SOH and RUL and features to achieve the prediction task [9–12]. Shi Yongsheng et al. used early battery cycling data, observed the discharge voltage–capacity degradation curve, and established a hybrid prediction model of a WOA–XGboost algorithm with input time series data to achieve SOH prediction on a time series [13]. In the research of Ji Wu, the voltage during the charging process of Li-ion batteries served as an indirect health factor, and a model for predicting the RUL of Li-ion battery was built based on an artificial neural network (ANN). Nevertheless, fewer features were extracted, and the model generalization ability of the model with

fewer features in the actual prediction should be verified [14]. Ping Wang et al. proposed a Gaussian regression-based, joint principal component analysis method to process the charging curve to obtain indirect health features, which was combined with a battery aging model and achieved high accuracy. Pan, H. et al. proposed constructing health indicators (HI) characterizing battery decline under dynamic operating conditions and introduced an ELM decline model for offline training of the whole battery life cycle by an extreme learning machine (ELM) to achieve SOH online estimation, which is more robust than the traditional estimation methods [15]. Wen proposed a model based on incremental capacity analysis and a BP neural network to predict the SOH of batteries at different ambient temperatures. By analyzing the correlation between IC curve characteristics and SOH, the mapping relationship between temperature and IC curve characteristics was established by the least squares method to obtain the SOH prediction model at different temperatures [16]. Jixuan Zhang combined the filtering algorithm with the autoregressive sliding average algorithm and used the fusion algorithm to build the remaining life prediction model for lithium-ion batteries, which effectively improved the training speed [17].

Due to the complexity of chemical reactions in actual lithium-ion batteries, various types of partial differential equations need to be established, and the parameters and matrix calculations involving the equivalent circuit model are relatively large, which makes it difficult to guarantee the accuracy of the model in practical applications. Furthermore, SOH cannot be directly measured, often relying on certain technical means for the correct extraction of the internal parameters and characteristic relationships of the battery to estimate SOH [18–20]. Data-driven and predictive model building-based approaches usually extract typical features from degraded data, and, using currently popular machine learning, models can construct state mapping relationships between degraded data and health states to overcome the problems of sample size and nonlinearity of data, thus, achieving SOH and RUL prediction tasks. Song Zhe et al. extracted eight degradation parameters, such as equal time interval voltage difference, equal voltage drop discharge time, and root mean square of discharge voltage value from voltage, current, and temperature profiles to jointly predict the RUL of Li-ion batteries but ignored the redundancy and deficiency among the degradation parameters [21]. Yang Zanshe et al. proposed a gray wolf optimization support vector regression method to build a degradation model based on the life cycle data of sample batteries to achieve the evaluation and prediction of the degradation state of Li-ion batteries, but the battery capacity is difficult to measure directly and does not have real-time functionality [22].

In addition, although PCA (principal component analysis), LDA (linear discriminant analysis) and NMF (non-negative matrix factorization) are more effective in feature extraction applications, they are all linear. However, they are linear models that can only discover the global features of the data, and LDA is supervised and requires a given label information during training. Recently, a large amount of research has started to focus on the unsupervised domain. Researchers in different fields have found that data points in high-dimensional space lie approximately on a sub-flowform in the embedded low-dimensional space. To discover the nonlinear structure of the high-dimensional data space, some researchers have proposed kernel methods, but they are computationally intensive and do not take into account the intrinsic manifold structure of the original high-dimensional data. However, the manifold-based methods can effectively solve the above problems.

To this end, this paper proposes an SOH estimation and RUL prediction algorithm for lithium-ion batteries based on manifold learning and the LightGBM model, which integrates each cycle curve of the battery to establish feature engineering. In this paper, we choose to extract a total of three types of features from different perspectives, which are capacity degradation features, entropy value features, and correlation coefficient features. Among them, the capacity degradation features extract different feature parameters for the discharge voltage curves in charge/discharge cycles, including the average voltage decay and IC curve feature parameters; the entropy value features are the generalized multiscale sample entropy of each attribute using the algorithm described in the previous

paper to measure the information uncertainty; finally, the autocorrelation coefficient and partial autocorrelation coefficient are extracted in the time domain. Then, the L-ISOMAP manifold learning algorithm is used to dimensionally reduce the input feature set to solve the dimensional explosion problem. Finally, LightGBM prediction models are established to perform SOH prediction on different datasets, and multidimensional model superiority assessment is performed, including the selection of different prediction models, the comparison of the selection of different manifold learning methods, and model generalization validation experiments.

The approximate structure of this paper is as follows. Section 2 shows the algorithms used for SOH prediction. It includes the algorithms used in data preprocessing, as well as the principle of the generalized multiscale sample entropy algorithm, L-ISOMAP algorithm, and the LightGBM regression prediction model. Section 3 shows the flow-based learning and LightGBM prediction process constructed in this paper. Section 4 is an example validation of the datasets from different sources using specific battery data examples, combined with the prediction model built in Section 3, and the analysis of the obtained prediction results. Section 5 sets up multiple sets of experiments to validate the multidimensional superiority of the model proposed in this paper. Section 6 concludes by summarizing the whole paper.

2. Algorithm Principle

2.1. The Generalized Multiscale Sample Entropy

2.1.1. Multiscale Sample Entropy (MSE)

The MSE algorithm process is as follows:

For the time series $\{x(i), i = 1, 2, \dots, N\}$, define the coarse-grained sequence $y^{(s)}$ using the following equation:

$$y^{(s)} = \frac{1}{s} \sum_{i=(j-1)s+1}^{js} x_i \tag{1}$$

where s is the scale factor, $1 \leq j \leq \frac{N}{s}$.

Calculate the sample entropy values of the coarse-grained sequence $y^{(s)}$ under different scale factors s , i.e., the multiscale sample entropy, as follows:

$$E_{MSE}(x, s, m, r) = E_{SE}(y^{(s)}, m, r) = -\ln\left(\frac{n_s^{(m+1)}}{n_s^{(m)}}\right) \tag{2}$$

where m is the embedding dimension, r is the similarity tolerance, $E_{SE}()$ is the sample entropy value, and $n_s^{(m+1)}$ and $n_s^{(m)}$ are the number of $m + 1$ and m dimensional vectors of the coarse-grained sequence, respectively.

2.1.2. Generalized Multiscale Sample Entropy

In this paper, we extend the mean calculation of the MSE coarse-grain process to second-order moments to overcome the deficiency of “neutralizing” the mutation behavior of the original signal brought by the coarse-grain method of homogenized data, and then propose a generalized multiscale sample entropy (GMSE) algorithm. The specific procedure is as follows:

For the time series described in the previous section $x(i)$, the following equation is used to calculate the generalized coarse-grained series $y_G^{(s)}$:

$$y_G^{(s)}(j) = \frac{1}{s} \sum_{i=(j-1)s+1}^{js} (x_i - \bar{x}_i)^2 \tag{3}$$

$$\bar{x}_i = \frac{1}{s} \sum_{h=0}^{s-1} x_{i+h} \tag{4}$$

where $1 \leq j \leq \frac{N}{s}$.

Calculate the sample entropy values for the generalized coarse-grained sequence $y_G^{(s)}$ at different scale factors s , as follows:

$$E_{(GMSE)}(x, s, m, r) = E_{SE}(y_G^{(s)}, m, r) = -\ln\left(\frac{n_{G,s}^{(m+1)}}{n_{G,s}^{(m)}}\right) \tag{5}$$

2.2. Landmark Isometric Mapping Algorithm (L-ISOMAP)

The basic idea of landmark-isometric mapping (L-ISOMAP) is similar to MDS, that is, to find the “geodesic distance” between any two sample points from high-dimensional data, and to achieve the “geodesic distance” in the low-dimensional space by mapping. The “geodesic distance” remains approximately the same in the low-dimensional space. Compared with the ISOMAP algorithm, L-ISOMAP has a faster operation rate and wider application range and can represent the low-dimensional features of high-dimensional data well.

The specific calculation steps of the L-ISOMAP algorithm are as follows:

Select the appropriate landmark points. Select n landmark points from N samples, and then find the Euclidean distance between N all sample points and the selected n points to obtain the matrix d , where d_{ij} represents the Euclidean distance between the sample points x_i and landmark points x_j .

Construct the adjacency graph. Each edge weight in the adjacency graph G is $d_E(i, j)$, and if the sample points x_i and x_j are connected in G , then the initial value of the shortest path between them is $d_G(i, j) = d_E(i, j)$; otherwise, $d_G(i, j) = \infty$. Let $q = 1, 2, \dots, N$ and calculate the following equation:

$$d_G(i, j) = \min\{d_G(i, j), d_G(i, q), d_G(q, j)\} \tag{6}$$

From this, the geodesic distances between all sample point pairs can be formed into a geodesic distance matrix at $D_G = \{d_G(i, j)\}$ and the shortest paths can be calculated.

Compute the low-dimensional embedding coordinates. Applying the MDS algorithm to the geodesic distance matrix D_G , the following objective function is minimized to obtain the low-dimensional embedding coordinates of the sample Y .

$$E = \|\tau(D_G) - \tau(D_Y)\|_{L^2} \tag{7}$$

where $\tau(D_G) = -HSH/2$, $S_{ij} = D_{ij}^2$, $H = I - \frac{1}{m}ee^T$, and I denote the unit matrix and e denotes the unit column vector. The optimal embedding coordinates of L-ISOMAP are obtained by minimizing the objective function and finding the maximum eigenvector Y .

2.3. LightGBM Algorithm

2.3.1. Gradient Boosting Algorithm (GBDT)

Gradient boosting is a machine learning algorithm of the boosting class. GBDT is based on the principle of addition, which accumulates the results of multiple regression trees to obtain the final result, and effectively avoids the problem of easy overfitting of a single tree by controlling the weights of the regression tree results in each round of iteration [23].

The steps of the GBDT algorithm are as follows:

Step 1: Input the training dataset, as follows:

$$T = \{(x_1, y_1), (x_2, y_2), \dots, (x_n, y_n)\} \tag{8}$$

where $x_i \in X \subseteq R^n$, $y_i \in Y \subseteq R^n$.

Step 2: Initialize the weak learner $f_0(x)$, and define the prediction function of the model $f(x)$, as follows:

$$f_0(x) = \text{Arg min}_c \sum_{i=1}^m L(y_i, c) \tag{9}$$

$$f(x) = \sum_{i=0}^N f_i \tag{10}$$

Step 3: Construct a weak learner for each round in the iteration number $t = 1, 2, \dots, T$ with the training sample $D = \{x_i, y_i\}_{i=1}^N$ and the corresponding loss function $L(y_i, f(x_i)) = (y_i - f(x_i))^2$ to solve for the negative gradient, as follows:

$$r_{ti} = -\left[\frac{\partial L(y_i, f(x_i))}{\partial f(x_i)}\right]_{f(x)=f_{t-1}(x)} \tag{11}$$

Step 4: Using the solved negative gradients r_{ti} and $x_i (i = 1, 2, \dots, N)$, a CART regression tree is fitted according to the squared error minimization training decision tree to obtain the t th regression tree, whose corresponding leaf region is $R_{tj} (j = 1, 2, \dots, J)$, where j is the number of leaf nodes of the regression tree t , as in the following Equation (12):

$$C_{tj} = \text{Arg min}_c \sum_{x_i \in R_{tj}} L(y_i, f_{t-1}(x_i) + c) \tag{12}$$

Step 5: Update the strong learner, as follows:

$$f_t(x) = f_{t-1}(x) + \sum_{j=1}^J C_{tj} I(x \in R_{tj}) \tag{13}$$

If the corresponding decision function satisfies the convergence condition, the iteration stops and the expression of the final strong learner $f(x)$ is as follows:

$$f(x) = f_0(x) + \sum_{t=1}^T \sum_{j=1}^J C_{tj} I(x \in R_{tj}) \tag{14}$$

2.3.2. LightGBM Algorithm

The traditional GBDT algorithm has some limitations in implementation, such as poor performance on high-dimensional sparse datasets, serial training process, and relatively poor parallelism capability. Therefore, the LightGBM (light gradient boosting machine) algorithm is further proposed on the basis of GBDT. In model building, the basic idea is the same as GBDT, but the computational cost and complexity of the model are reduced compared with GBDT, and the performance and performance is better than the commonly used machine learning algorithms.

LightGBM uses a histogram to optimize the sample processing speed by pre-constructing a training function to transform the continuous eigenvalues into discrete K box eigenvalues before training, thereby building a histogram with K entries that traverses the entire sample. In this process, LightGBM accumulates statistics in the histogram with K discrete values and finally finds the best splitting point from the discrete values [24]. LightGBM starts with a constant tree model and minimizes the loss function by training a new tree model, as follows [25–27]:

$$Y_i^{(t)} = Y_i^{(t-1)} + f_t(x_i) \tag{15}$$

where $Y_i^{(t)}$ is the new model, $Y_i^{(t-1)}$ is the tree model of the previous iteration, and $f_t(x_i)$ is the tree model to be added in the next step, denoted as follows:

$$f_t = \min L = \min \left(\sum_{i=1}^n l(y_i, Y_i) \right) \tag{16}$$

where L is the loss function of the algorithm, denoted by the predicted value Y_i and the true value y_i of the i th sample. In addition, the leaves of the GBDT algorithm use level-wise growing by layer, which treats the leaves of the same layer indiscriminately, while in fact many leaves have low splitting gain and there is no need for splitting. To address this shortcoming, the LightGBM algorithm uses a more efficient leaf-wise algorithm to find the leaf with the largest splitting gain from a layer and repeat it over and over again to obtain a higher accuracy with the same number of splits.

3. Li-Ion Battery SOH Prediction Process

For Li-ion battery SOH prediction, this paper designs a Li-ion battery SOH prediction flow based on manifold learning and LightGBM. This prediction flow is shown in Figure 1.

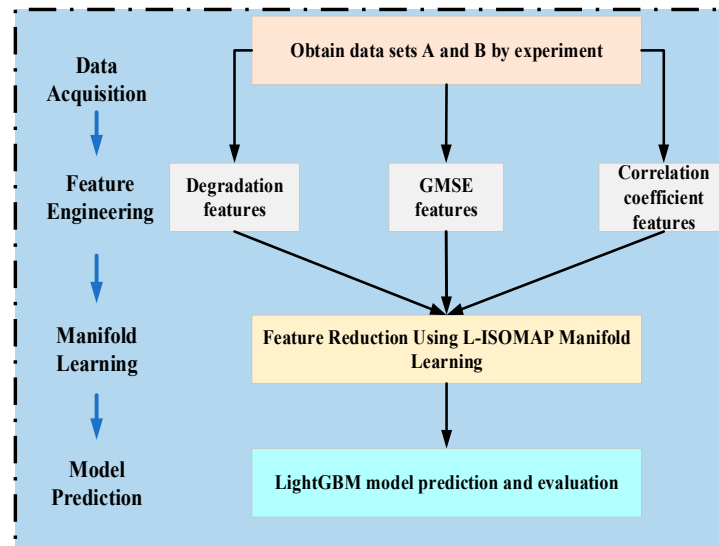


Figure 1. Manifold learning and LightGBM-based SOH estimation process for a Li-ion battery.

As can be seen from Figure 1, the battery SOH prediction model proposed in this paper includes four major parts: data acquisition, feature engineering establishment, manifold learning, and model prediction. Firstly, the battery charging and discharging cycle experiment is designed to obtain the dataset B. Secondly, the feature engineering is established, and this paper chooses to extract a total of three types of features from different angles, which are the capacity degradation features, entropy value features, and correlation coefficient features. Among them, the capacity degradation features mainly focus on the discharge voltage curve in the charge/discharge cycle and extract different feature parameters, including the average voltage decay and IC curve feature parameters; the entropy value features are the generalized multiscale sample entropy of each attribute using the algorithm described in the previous paper, which measures the information uncertainty; finally, the autocorrelation coefficients and partial autocorrelation coefficients are extracted in the time domain; then, the L-ISOMAP manifold learning algorithm is used. Dimensional simplification is performed on the input feature set to solve the dimensional explosion problem; finally, a LightGBM prediction model is established to perform SOH prediction on different datasets, and model evaluation is performed.

4. Data Acquisition

4.1. Experimental Protocol

4.1.1. Instrumentation

The following apparatus were used: battery tester, thermostat, upper computer, and alligator clips.

In this paper, a battery tester is used to perform cyclic charge/discharge experiments on the battery and to detect parameters, such as voltage, current, power, and capacity during the process to construct the dataset B. The battery is an NCM battery, which is also the most widely used battery in method vehicles, and a thermostat is used to create an experimental environment of 15 °C [16]. The experimental equipment is shown in Figure 2.

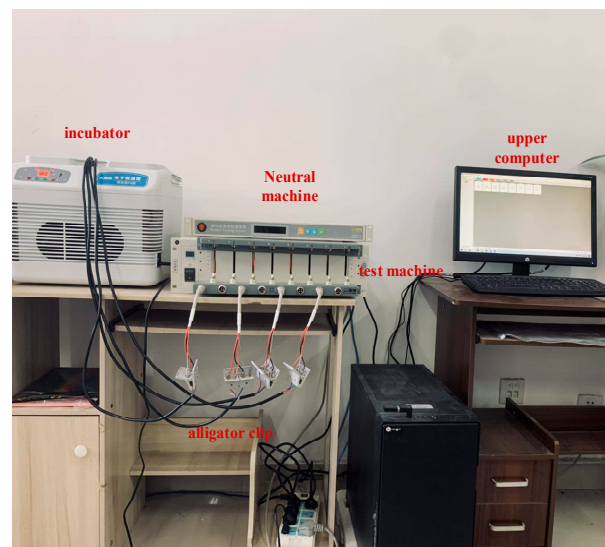


Figure 2. Experimental equipment.

The battery to be tested is an 18650 Li-ion battery, and its details are shown in Table 1.

Table 1. Battery detailed parameter table.

Projects	Specification	Projects	Specification
Housing material	Nickel-plated steel	Charging strategy (CC/CV)	Standard Fast
Nominal capacity	1300 mAh	Operating temperature	Charging
Rated capacity	3.7 V		Discharge
Charging voltage (Max)	4.2 V	Storage	Storage
Discharge cut-off voltage	2.7 V		
Charging current (Max)	1C A ₅		
Discharge current (Max)	3C ₅ A		
Internal resistance (Max at 1000 Hz)	≤25 mΩ		

4.1.2. Experimental Steps

The experimental procedure is shown in Figure 3.

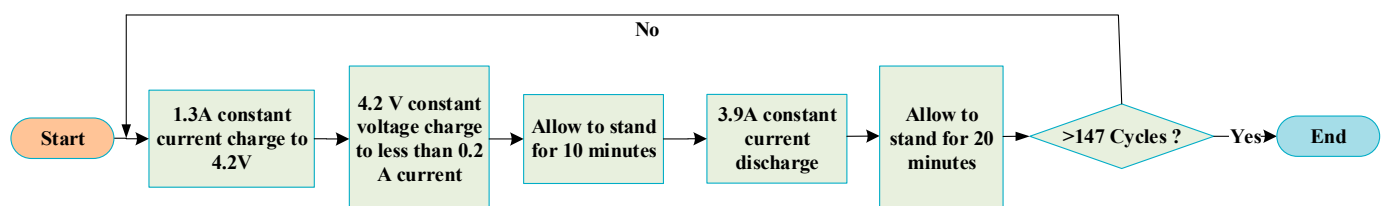


Figure 3. Experimental procedure.

4.2. Introduction to the Dataset

To verify the prediction effect of the proposed algorithm for Li-ion batteries under different charging and discharging strategies, two datasets are selected for the experiments, which are noted as dataset A and dataset B.

Dataset A: Lithium-ion battery test data from the NASA prediction center (prognostics center of excellence, PCoE). The battery used is the 18650 battery, with a rated capacity of 2 Ah. Take the B0005, B0006, and B0007 battery data as experimental data, including the battery cycle charge and discharge process voltage. The experimental process is as follows. During the charging process, the battery is charged with 1.5 A constant current, and when the voltage reaches 4.2 V, it is switched to constant voltage charging until the charging current is less than 20 mA to stop charging; during the discharging process, the battery is discharged with 2.0 A constant current until the voltage is less than 2.5 V. According to the above process, the battery is charged and discharged for 168 cycles [28–30].

Dataset B: Dataset B is the data obtained from the measurements according to the experimental procedure in Section 4.1, including the battery data of voltage, current, and power during charging, resting, and discharging. The parameters of 148 cycles of the battery are obtained after processing.

5. Instance Validation

In this paper, we take dataset A as the research object, perform feature extraction, and then build a prediction model to predict SOH and RUL of each group of batteries. Meanwhile, to verify that the feature engineering proposed in this paper has strong generalization ability, we take dataset B as the validation object, build feature engineering again for dataset B, and build a prediction model to observe the prediction effect.

5.1. Feature Extraction

The dataset selected in this paper includes charging and discharging cycles, including voltage measured, current measured, temperature measured, current load, and voltage load for charging experiments, and voltage measured, current measured, temperature measured, and current load, voltage load for discharging experiments, as well as temperature measured, current load, and voltage load, a total of 10 attributes. Some of the discharge voltage curves are shown in Figure 4.

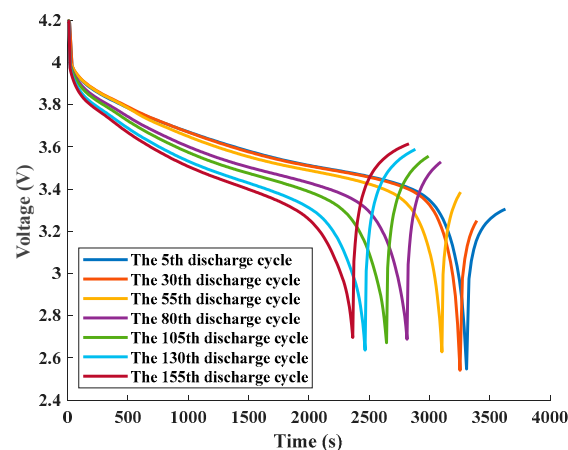


Figure 4. Discharge voltage curves under different cycles.

As can be seen from Figure 4, each curve represents the change in the discharge voltage property in the discharge experiment under different cycles. Since the capacity of each charge and discharge is different, the sampling length of each curve is inconsistent, resulting in inconsistent lengths of the data sequence measured by the instrument, so it is difficult to directly input the property as a feature quantity directly into the prediction model for battery life prediction, and feature extraction must be performed on it.

5.1.1. Capacity Degradation Characteristics

The capacity of a Li-ion battery is related to its complex internal physicochemical and thermal effects. It is necessary to find the health factors that replace the degradation state of the battery, so it is necessary to extract the characteristic sequence of the degradation state of the lithium battery for the parameters that can be easily monitored and to construct the set of health factors.

First is the voltage decay characteristic. The time used by a brand new fully charged battery to discharge the power will gradually decrease with the increase in the number of discharge cycles. As can be seen from the discharge voltage curves in Figure 4, within 500~1500 s, the value of the voltage drop in different cycle curves increases with the number of cycles, i.e., the distance between the nominal voltage and the real-time voltage increases, so the voltage decay can be used to quantify the capacity decay. Several discharge voltage curves in Figure 4 were randomly selected, and the discharge voltage decay schematic is shown in Figure 5.

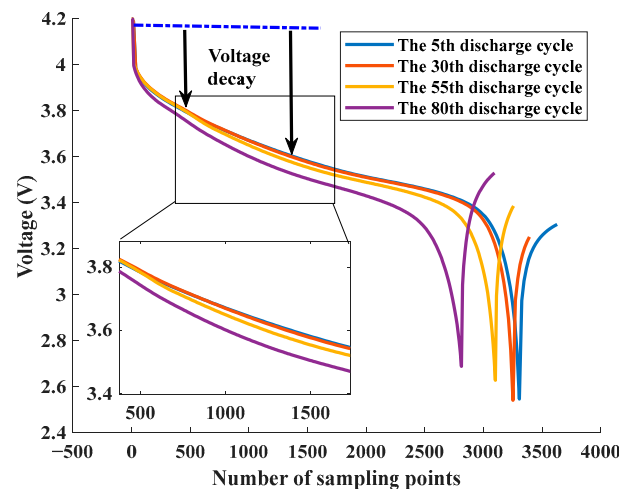


Figure 5. Schematic diagram of discharge voltage decay.

As shown in Figure 5, in order to describe the voltage decay more comprehensively, this paper chooses to use the mean voltage decay (MVF) as the health factor. Specify 500~1500 s as the time range for extracting the health factor, and define MVF by averaging 50 voltage points within the defined time range, with each sampling time point defined as j , where the MVF during the i th discharge cycle can be expressed as follows:

$$MVF_i = \frac{\sum_{j=1}^{50} |V_n - V_j|}{50} \tag{17}$$

where MVF_i is the discharge voltage drop of the i th charge/discharge cycle; V_j is the j th discharge voltage point in the defined time range; V_n is the nominal voltage of 4.2 V.

In addition to the average voltage drop sequence, there are some physical quantities that can also reflect the battery capacity degradation state, such as the capacity increment curve [31]. The essence of the capacity increment curve is the differentiation of the capacity Q charged or discharged by the battery during charging and discharging against the voltage V , which is the magnitude of the capacity change corresponding to the unit voltage change in the battery when it occurs. The voltage plateau appearing in the charging and discharging voltage change curve can be transformed into the capacity increment with significant characteristics (dQ/dV). The specific equation for the IC in the discharging mode is as follows:

$$\frac{dQ}{dV} \approx \frac{\Delta Q}{\Delta V} = \frac{Q_{v2} - Q_{v1}}{V_2 - V_1} \tag{18}$$

where Q_{v1} represents the battery capacity at V_1 and Q_{v2} represents the battery capacity at V_2 . The IC curves in the discharge experiment are shown in Figure 6a.

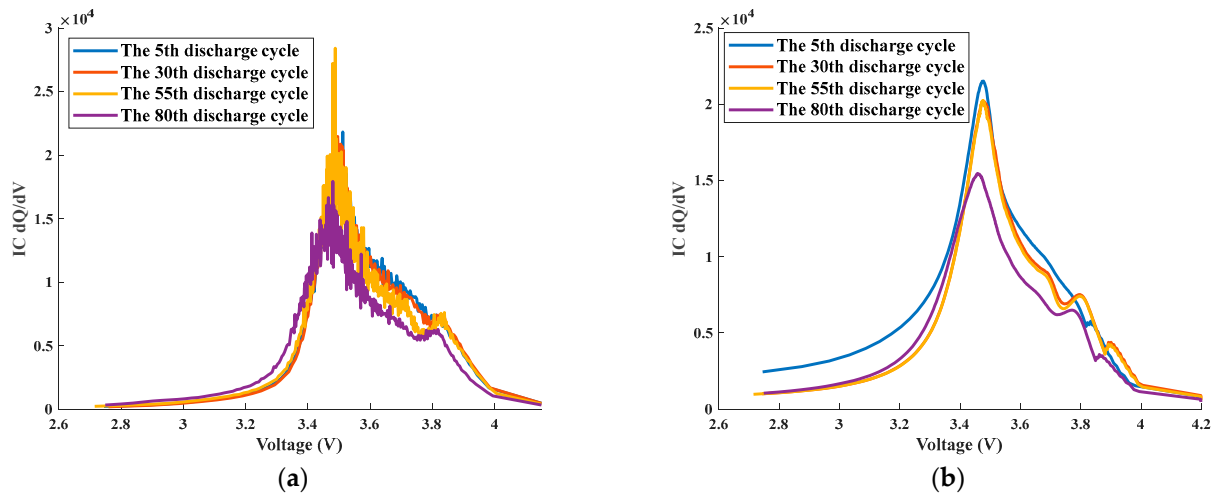


Figure 6. Discharge IC curves. (a) Original discharge IC curve; (b) discharge IC curve after filtering.

As can be seen from Figure 6a, with the decline in the battery, there is a more obvious change in the IC curve, and the change is more significant at the peak of the IC curve and its corresponding position.

To quantitatively analyze the battery aging state by using the capacity increment curve, it is also necessary to filter and smooth the curve, so that it is easier to find the characteristics and change patterns of the curve. In this paper, the generalized moving average method is chosen to smooth and filter the curve. The filtered discharge IC curve is shown in Figure 6b.

As can be seen from Figure 6b, the IC curve after filtering is obviously smooth and the peak height is clearly visible, and the peak height shows a better correlation with capacity decline as the battery aging degree deepens. Meanwhile, analyzing the IC curve characteristics of different aging degrees during the battery cycle aging process, such as the left and right positions, the area contained in the peak, and other parameters, the mode and mechanism of battery aging decline can be analyzed and judged, and then the health status of the battery can be diagnosed [31,32]. Taking the fifth discharge cycle as an example, the schematic representation of each parameter in its capacity increment curve is shown in Figure 7.

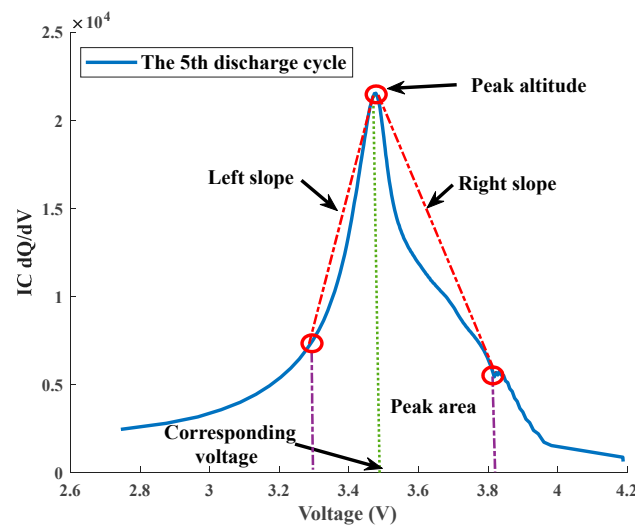


Figure 7. Schematic diagram of the characteristic parameters in the IC curve.

As shown in Figure 7, the feature parameters selected for extraction in this paper include the peak height, the position of the crest (the voltage value corresponding to the horizontal axis at the peak), the area contained in the crest, and the slope of the left and right sides of each peak. Among them, the peak height and the position of each peak are the peak at the highest point in the IC curve and its corresponding voltage value. For the left and right slope of the peak, we need to determine the voltage point on the left and right side. From the IC curves under different cycles in Figure 7, we can see that the curve rises sharply at about 3.3 V, indicating that it is greatly influenced by the discharge voltage at this time, so the voltage on the left side of the wave peak is determined to take the value of 3.3 V; the voltage point on the right side is calculated by calculating the position where the first trough appears and taking it as the voltage on the right side of the wave peak. The area contained in the wave crest is calculated as the area enclosed by the IC curve and the voltage points on the left and right side.

In summary, the characteristic parameters in the MVF and discharge IC curves are selected as the first type of capacity degradation characteristics in this paper.

5.1.2. Generalized Multiscale Sample Entropy

Entropy is a measure of information uncertainty, so entropy features are often used as a class of effective features for feature extraction in regression models. To measure the uncertainty and complexity of the signal distribution, the entropy of different frequency bands can be used to quantify the information contained in the signal. The mean value feature of entropy can reflect the complexity of the information within the system as well as the error orientation.

The most commonly used entropy functions include scatter entropy, fuzzy entropy, and sample entropy, as well as the derived multiscale entropy. Among them, the generalized multiscale sample entropy, which is improved on the basis of multiscale sample entropy, has the advantages of robustness, simple calculation, and fast operation, which can overcome the shortage brought by the coarse-grained method of using homogenized data in the calculation process on the sample.

In this paper, the generalized multiscale sample entropy is selected as the entropy feature, and the optimal scale factor is selected by analyzing the change trend of the GMSE mean value under different scales s . Figure 8 shows the GMSE mean values of the discharge voltage attributes under different cycles when different scale factors are selected, and the most suitable scale is selected based on the scale–mean decay relationship, i.e., the dimensionality of the entropy feature vector is determined.

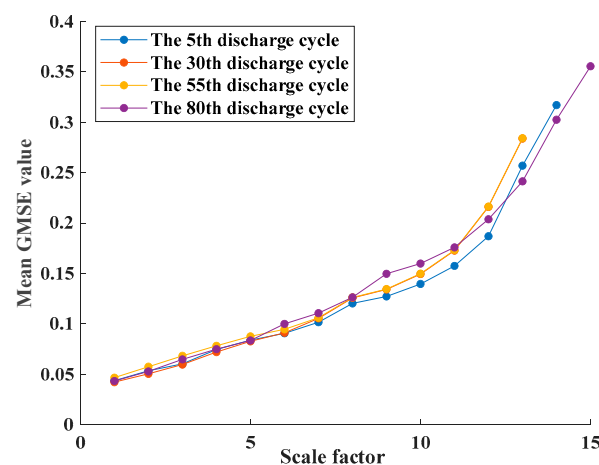


Figure 8. Mean values of GMSE at different scale factors.

As can be seen from Figure 8, with the increase in the scale factor, the mean GMSE values of the discharge voltage signals of each cycle show a continuous increasing trend, and at the scale factor of 5, the voltage curves of different cycles appear to overlap and

cross. Considering that the dimensionality of the feature set is closely related to the aging information, a smaller scale factor cannot fully extract the feature information and is prone to information overlap; a larger scale factor will cause redundancy of feature information and affect the model prediction effect. Therefore, the GMSE values of the first six scale factors constitute the entropy feature sample set, which is used as the second type of entropy feature input.

5.1.3. Autocorrelation Coefficient Characteristics

In time series forecasting, the autocorrelation and partial autocorrelation coefficients can be used to measure the correlation between current and past series values and to indicate the most useful past series values for predicting future values.

Autocorrelation is the correlation of a signal with itself at different points in time, and it can identify repetitive signals (such as periodic signals masked by noise), as well as fundamental frequencies that disappear implicitly in the harmonic frequencies of the signal. The partial autocorrelation is a summary of the relationship between the time series and the previous time series after the interference is removed.

The autocorrelation coefficients and partial autocorrelation coefficients of each attribute are extracted to form the timing characteristics. The autocorrelation coefficients and partial autocorrelation coefficients of the discharge voltage attributes in the first cycle are shown in Figure 9.

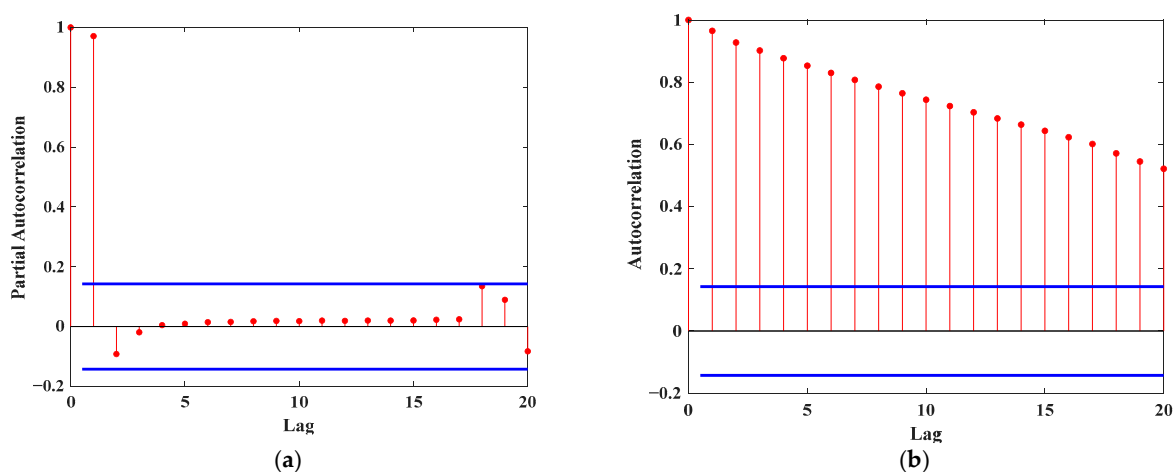


Figure 9. Autocorrelation coefficient and partial autocorrelation coefficient. (a) Partial autocorrelation coefficient; (b) autocorrelation coefficient.

In Figure 9, The upper and lower blue horizontal lines indicate the upper and lower limits of the autocorrelation coefficient and the partial autocorrelation coefficient, respectively, and the part beyond the bounds indicates the existence of a correlation. As can be seen from Figure 9, The red line represents the magnitude of the value at different lags the absolute value of the autocorrelation coefficient maintains a large value for a long time, and there is a gradually decreasing trend, which indicates the phenomenon of “trailing”. The partial autocorrelation plot, on the other hand, fluctuates around the value of zero after the second order, i.e., the “truncated tail” phenomenon, which indicates that the time series is a smooth series. In this paper, the autocorrelation coefficient and partial autocorrelation coefficient of each attribute are selected as the third type of characteristics.

5.2. L-ISOMAP Manifold Learning

Manifold learning is used to recover the structure of low-dimensional stream shapes from high-dimensional sampled data, i.e., to find low-dimensional stream shapes in high-dimensional space and find the corresponding embedding mappings for dimensional simplification or data visualization. It is used to find the essence of things from observed phenomena and to find the inner laws that generate data.

ISOMAP is an unsupervised streamlining method that can maintain global characteristics and is a generalization of the MDS (multi-dimensional scaling) algorithm for nonlinear feature extraction. The ISOMAP algorithm uses nonlinear geodesic distances instead of Euclidean distances as the similarity degree between sample points. The L-ISOMAP (landmark ISOMAP) algorithm is an improved algorithm based on ISOMAP, which only calculates the geodesic distances from each sample point to landmark points to generate the dimensional matrix and obtain the Euclidean embedding of the observed data. In this paper, we use L-ISOMAP to dimensionally approximate the results of feature extraction described in the previous paper, and, as such, achieve the purpose of solving the dimensional disaster and overfitting problems.

As can be seen from Figure 10, firstly, after the landmark points are selected, the neighbor-joining graph is constructed based on the high-dimensional health factor sequences obtained in the previous section to obtain its representation in the low-dimensional space, and the six-dimensional feature subset is finally obtained after L-ISOMAP manifold learning. The Spearman correlation coefficient is then used to characterize the intrinsic correlation between the feature set and the capacity after feature screening, and for the variables x_i, y_i , its Spearman correlation coefficient is calculated as follows:

$$\rho = 1 - \frac{6 \sum_{i=1}^n (R_i - Q_i)^2}{n(n^2 - 1)} \tag{19}$$

where R_i represents the rank of x_i , Q_i represents the rank of y_i , $R_i - Q_i$ is the difference in the rank of the variables, x_i, y_i , and n denotes the number of samples. Figure 11 shows the Spearman correlation coefficient plot between the six-dimensional health factor and the capacity obtained after L-ISOMAP manifold learning.

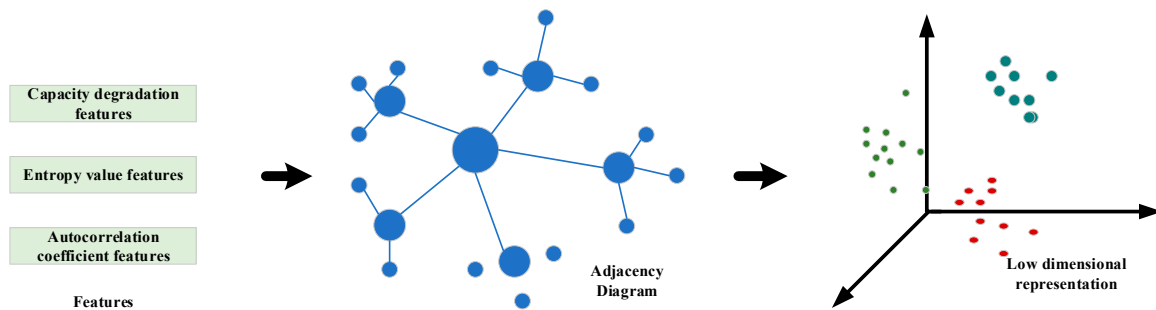


Figure 10. L-ISOMAP algorithm flow.

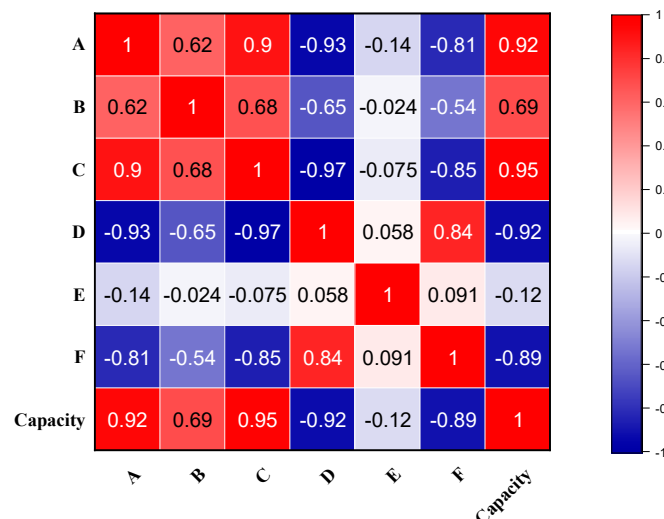


Figure 11. Spearman correlation coefficient between health factor and capacity.

A through F in the horizontal and vertical axes in Figure 11 characterize the six-dimensional health factors above, in that order. In addition, blue indicates that the correlation coefficient is less than 0 and red indicates that the correlation coefficient is greater than 0. The darker the color, the larger the value. From the definition of Spearman correlation coefficient, it is clear that the absolute value of the correlation coefficient is approximately close to 1, which represents a higher correlation, and 0 indicates no correlation. From Figure 11, it can be seen that the Spearman correlation coefficient values between the six-dimensional health factor sequences obtained by manifold learning and the capacity are larger, indicating that the correlation between them is higher, which is conducive to the subsequent input model and achieving accurate prediction of SOH.

5.3. SOH Prediction Based on LightGBM Model

In this paper, the cell capacity is used to describe the SOH of the cell, defined as follows:

$$SOH = 1 - \frac{C_{init} - C_{batt}}{0.2C_{init}} \quad (20)$$

where C_{init} represents the rated capacity, C_{batt} represents the actual capacity, and $0.2 < C_{batt} < C_{init}$.

Battery RUL indicates the number of cycles between the capacity decay of a lithium battery and its 80% capacity. According to the EOL standard, the failure thresholds of B0005, B0006, and B0007 batteries are set to 1.38 Ah, 1.38, Ah and 1.5 Ah, respectively, which can be calculated as 128 cycle count points for B0005 to reach the failure threshold point for the first time, 112 cycle count points for B0006 to reach the failure threshold point for the first time, and 125 cycle count points for B0007 to reach the failure threshold point for the first time.

The six-dimensional health factor sequence obtained above is used as the input of the LightGBM model for battery SOH prediction. For the 168 cycles of battery B0005, B0006, and B0007 data obtained after feature processing, the first 84 cycles were taken as the training set and the 85th to 168th cycles were taken as the test set. Here, the grid search method is used to find the optimization of the hyperparameters of the LightGBM model. The `num_leaves`, `learning_rate`, and `n_estimators` of the LightGBM model are selected as the grid search parameters, and five-times cross-validation is used in the adjustment process of each parameter to obtain the optimal LightGBM model with negative mean square error (NMSE) as the objective function. The optimal values of the above parameters for the LightGBM model were obtained after the grid search method to find the optimal values, as shown in Table 2.

Table 2. Optimization search results.

Battery Number	Num_Leaves	Learning_Rate	n_Estimators
5	17	0.1837	60
6	26	0.0163	60
7	10	0.1563	71
Parameter range	[2, 50]	[0.001, 0.5]	[1, 100]

The prediction results for their test set SOH and RUL are shown in Figure 12. To demonstrate that statistical significance is reflected, a 95% confidence interval is added to the prediction results as an indication that there is only a 5% or smaller margin of probability that the true results will occur outside the confidence interval.

As can be seen from Figure 12, on the test set results of dataset A, the fit of each prediction curve is good, the overlap between the prediction curves and the true capacity curves is high, the overall prediction curves fit the true values well, and the true values of the capacities all fall within the 95% confidence interval predicted by the LightGBM model,

indicating that the model can achieve the prediction purpose well on the SOH estimation and RUL prediction in dataset A.

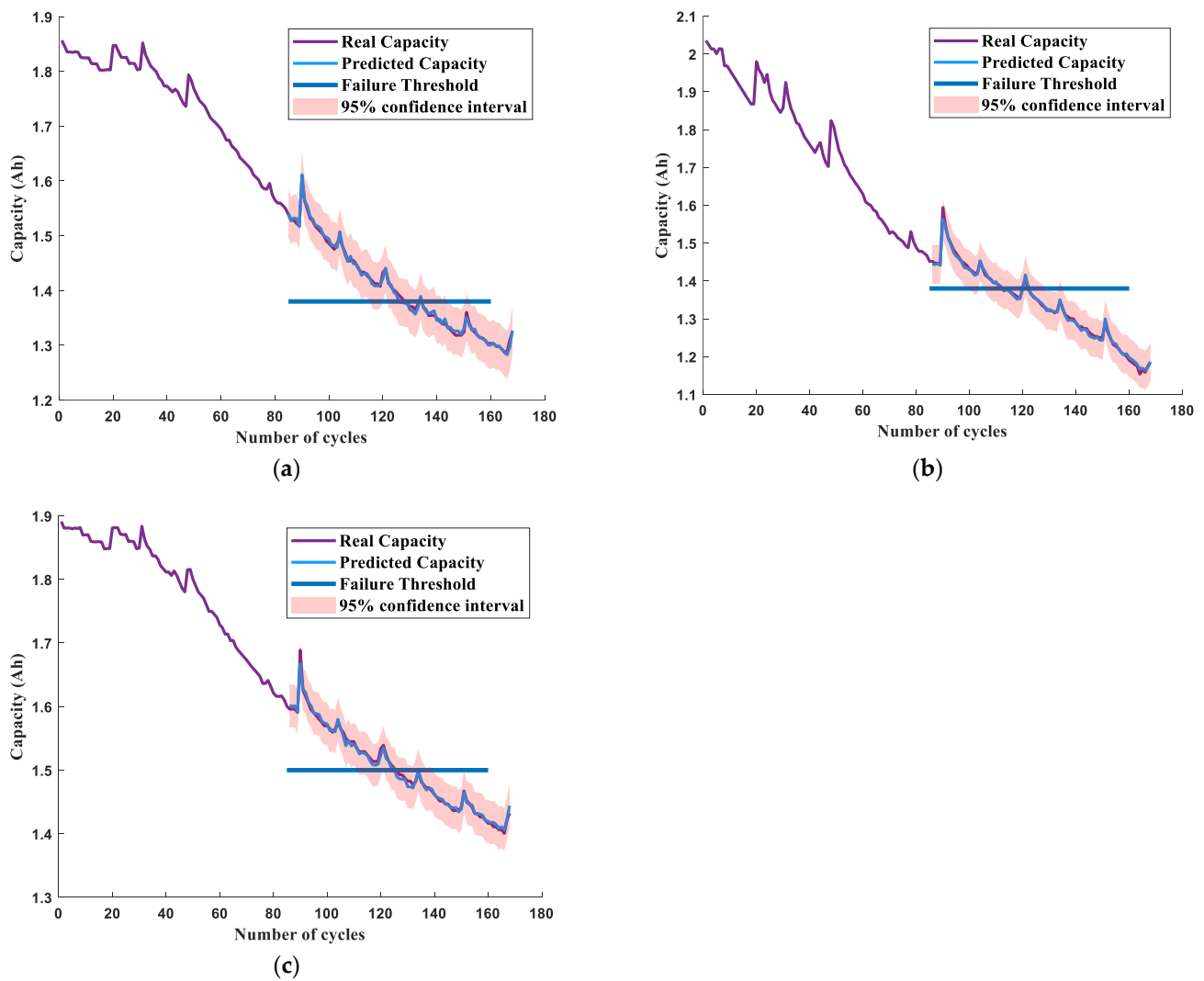


Figure 12. Prediction results of the dataset A test set. (a) Battery 5 test set prediction results. (b) Battery 6 test set prediction results. (c) Battery 7 test set prediction results.

In order to show the prediction accuracy of each cell in more detail, mean square error (MSE), mean absolute error (MAE), and goodness-of-fit (R-Squared, R^2) are chosen as the prediction models. The prediction results are evaluated by the evaluation indexes. The specific information and error distribution of the prediction evaluation indexes are shown in Table 3 and Figure 13, respectively.

Table 3. Specific information on predictive evaluation indicators.

Battery Number	MSE	MAE	AE	R^2
5	1.2740×10^{-4}	0.0076	0	0.9889
6	5.4299×10^{-4}	0.0175	1	0.9865
7	9.0437×10^{-5}	0.0069	0	0.9921

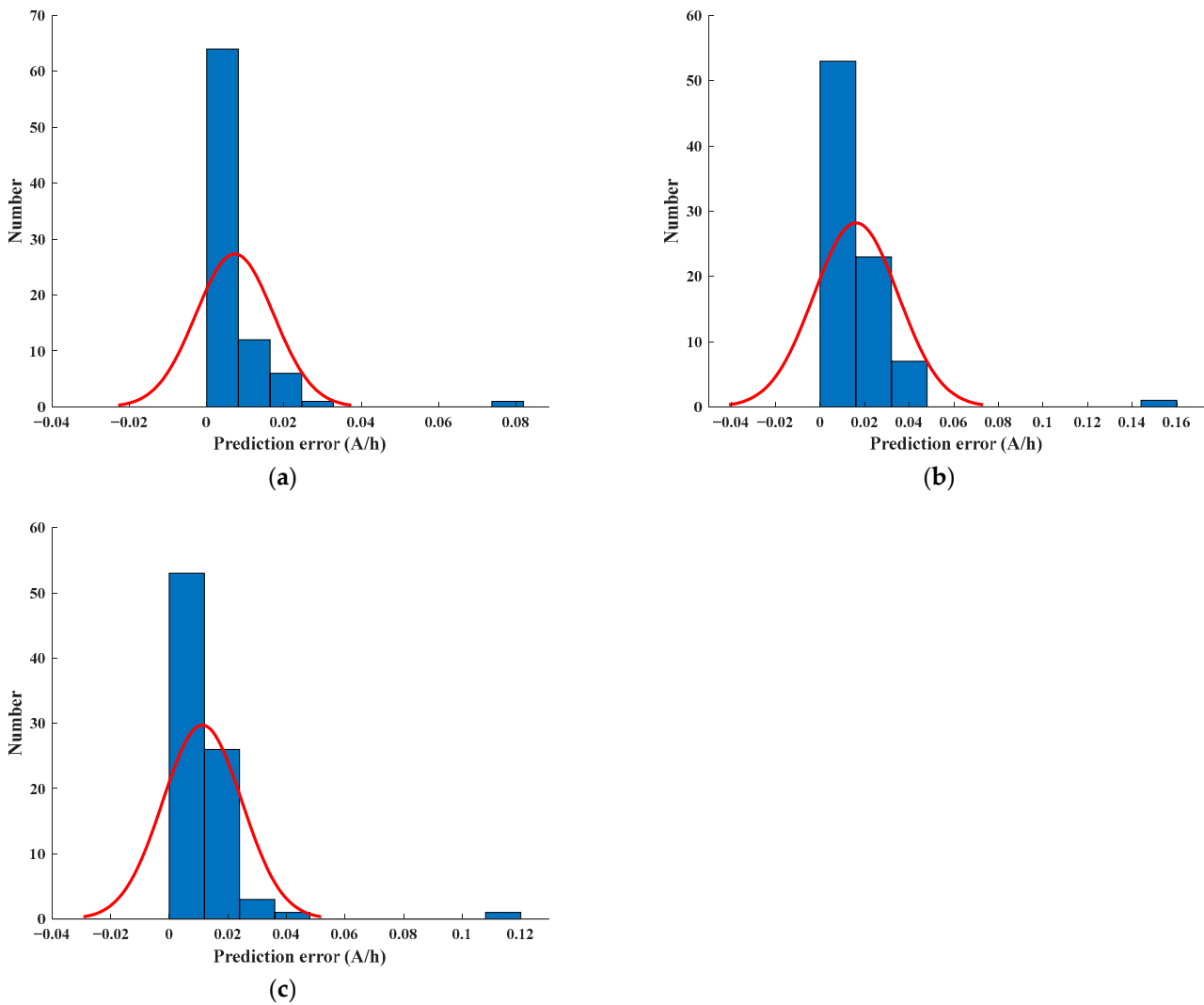


Figure 13. SOH prediction error distribution of dataset A. (a) Battery 5 test set prediction error. (b) Battery 6 test set prediction error. (c) Battery 7 test set prediction error.

Here, MSE, MAE, and R^2 are characterized by the following formulas, respectively:

$$MSE = \sum_{i=1}^N \frac{1}{N} (f(x_i) - y_i)^2 \tag{21}$$

$$MAE = \frac{1}{N} \sum_{i=1}^m |f(x_i) - y_i| \tag{22}$$

$$R^2 = 1 - \frac{SS_{residual}}{SS_{total}} \tag{23}$$

In the above equation, $f(x_i)$ is the true value; y_i is the predicted value; $i = 1, 2, \dots, N$, N is the sample size; R^2 is a measure of the overall fit of the regression equation, which takes values between $[0, 1]$; $SS_{residual}$ represents the regression sum of squares; SS_{total} represents the total sum of squares.

AE in Table 2 indicates the absolute value of the difference between the actual RUL and the predicted RUL of the cell.

Figure 13 characterizes the prediction error distribution of batteries 5, 6, and 7, the horizontal axis represents the error between the predicted value and the true value, and the vertical axis is the frequency of the occurrence of this error interval. Combined with

Figure 12 and Table 3, for R^2 , the effect of all three groups of batteries was above 0.98, with a mean value of 0.9892; for AE, an index for evaluating the predictive effect of RUL, the error of both battery 5 and battery 7 was 0, while battery 6 showed an error of one cycle from the true RUL. The error intervals of the three batteries are basically concentrated between 0 and 0.05, especially for battery No. 5, whose error is basically below 0.02 Ah. It can be proved that the feature engineering and LightGBM model established in this paper still have greater advantages in the field of SOH estimation.

5.4. Multidimensional Superiority Assessment

To verify the superiority and effectiveness of the algorithms used in different stages of this paper in the field of SOH estimation, multidimensional model superiority assessment experiments are conducted in this section, which include the comparison of the selection of different prediction models, the comparison of the selection of different stream shape learning methods, and the verification of model generalization.

5.4.1. Comparison of the Effects of Different Models

XGBoost provides an integrated decision tree-based learning method that excels in accuracy and automatically handles missing values and outliers, and which also supports parallel computation [33]. XGBoost's ability to process high-dimensional sparse data quickly and accurately with high operational efficiency and parallel processing makes it one of the very popular algorithms in the field of machine learning, performing well on multiple datasets. Random forest is an integrated learning method based on decision trees, which consists of multiple decision trees and is "randomized" to improve model performance and stability [34]. First, each base tree is sampled by "bootstrap aggregating" with put-back sampling, and second, each base tree uses "feature randomization" to randomly select only a portion of features for splitting when selecting nodes for splitting. This will reduce the correlation between each base decision tree and increase the stability of the algorithm. Random forests are widely used in modern machine learning in various fields, such as finance, healthcare, and natural language processing.

As well as ensemble learning models, regression models commonly used in machine learning, such as ELM and LSTM, are also widely applied in the field of SOH prediction. An ELM (extreme learning machine) [35] is a fast neural network algorithm that differs from other neural network algorithms in that ELM no longer requires initialization and iterative optimization of weight parameters in the network. It directly transmits randomly generated input weights from the input layer to the hidden layer through matrix operations, and then forms output weights by linearly combining the output vectors generated by the input layer in the hidden layer, thereby achieving the classification, regression, or clustering of data samples. The LSTM (long short-term memory) is a recurrent neural network (RNN) commonly used to process sequential data. The LSTM algorithm uses a series of gates to filter the input and control the flow of information, effectively maintaining and updating the content and length of memory units, thus, avoiding the "gradient disappearance" and "gradient explosion" problems encountered in the training process of traditional RNN algorithms [36]. It improves the accuracy and generalization ability of the network when processing long sequence data.

To verify the superiority of the LightGBM model selected in this paper, the ELM, LSTM, random forest, and XGboost models were selected, and the capacity features obtained in the previous paper were input to compare the prediction results with the LightGBM model selected in this paper [37]. The next step is to compare the computational cost of each model and characterize the operational efficiency by calculating the model run time. The prediction results are shown in Table 4.

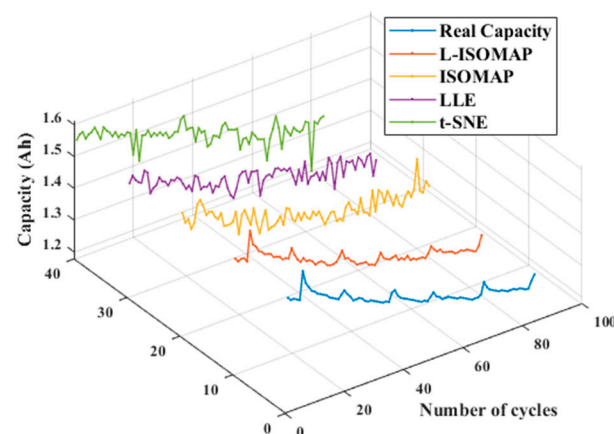
Table 4. Comparison of multi-model prediction effects.

Predictive Models	Battery Number	MSE	MAE	AE	R^2	Running Time (s)
ELM	5	0.0025	0.0421	4	0.9489	28.6184
	6	0.1827	0.4158	5	0.9254	35.0837
	7	0.0023	0.0374	4	0.9521	29.7031
LSTM	5	0.0033	0.0519	3	0.9675	42.9855
	6	0.0012	0.0258	5	0.9405	49.3947
	7	0.0016	0.0366	4	0.9677	36.0618
Random forest	5	4.5492×10^{-4}	0.0169	2	0.9802	42.9855
	6	8.0754×10^{-4}	0.0209	3	0.9769	49.3947
	7	3.0842×10^{-4}	0.0112	2	0.9713	36.0618
XGBoost	5	1.3182×10^{-4}	0.0081	2	0.9812	49.3246
	6	6.8945×10^{-4}	0.0225	1	0.9875	54.5812
	7	1.4913×10^{-4}	0.0072	1	0.9722	50.5719
LightGBM	5	1.2740×10^{-4}	0.0076	0	0.9889	46.9862
	6	5.4299×10^{-4}	0.0175	1	0.9865	51.6523
	7	9.0437×10^{-5}	0.0069	0	0.9921	44.2364

From Table 4, it can be seen that the LightGBM, XGBoost, and random forest algorithms all have good regression prediction ability, and the goodness-of-fit and root mean square error of all three are relatively close, reflecting the good adaptability of the feature engineering in this paper. However, the average MSE value of LightGBM for the three batteries is 2.5361×10^{-4} , which is 2.700×10^{-4} and 0.9685×10^{-4} lower than that of the random forest and XGBoost, respectively. In addition, the prediction goodness-of-fit of the LightGBM model is improved. Therefore, the LightGBM method can effectively improve the SOH estimation accuracy and operation rate and has good SOH estimation performance.

5.4.2. Comparison of the Effect of Different Manifold Learning Methods

To verify the superiority of the L-ISOMAP manifold learning algorithm selected in this paper, the data are now processed sequentially using the ISOMAP, LLE, and t-SNE methods, and then input into the LightGBM model for prediction. Take the prediction of battery No. 5 as an example; the prediction results of its test set are shown in Figure 14.

**Figure 14.** Comparison of the prediction effects of different stream shape learning algorithms.

As can be seen from Figure 14, the prediction effect using the t-SNE method is the worst, while the prediction curve using the L-ISOMAP manifold learning algorithm has the highest overlap with the true capacity curve and the best fitting effect, and its prediction effect is significantly better than the prediction effect processed by the rest of the stream

shape algorithms, indicating that the L-ISOMAP manifold learning algorithm used in this paper has high superiority.

5.4.3. Model Generalizability Validation

To verify the generalization ability of the feature engineering and prediction model proposed in this paper, feature extraction is performed again for dataset B and SOH prediction based on the LightGBM model is performed in this paper. There are 148 cycles of data in dataset B. The first 60 cycles are used as the training set and the last 88 cycles are used as the test set, the model built in the previous paper is used to train and predict, and the prediction results are shown in Figure 15.

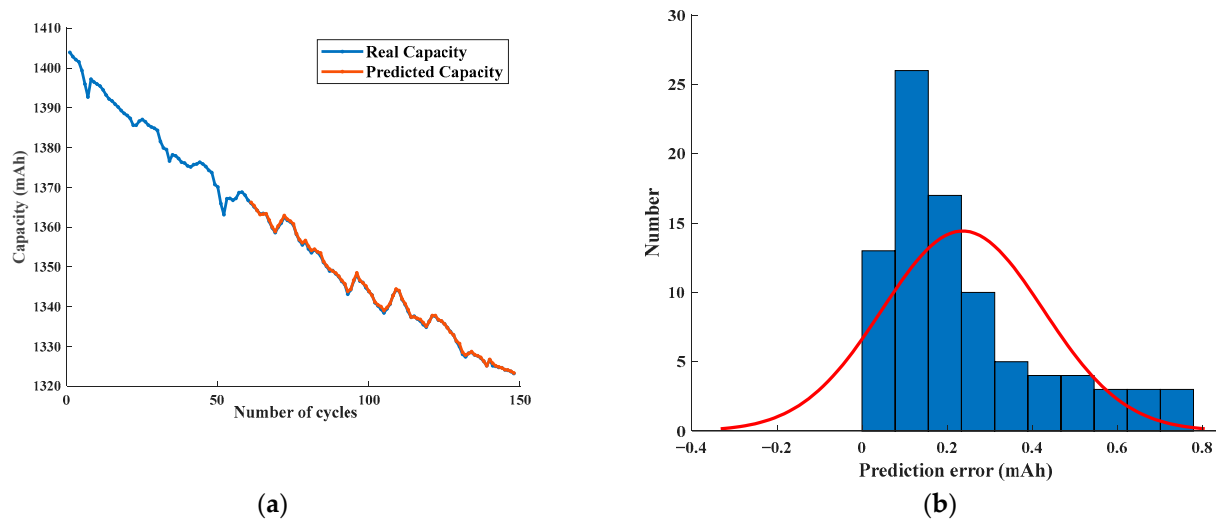


Figure 15. Prediction results analysis for the test set of dataset B. (a) Prediction results for the test set of dataset B; (b) dataset B test set prediction error.

Combined with Figure 15, it can be seen that in dataset B, the LightGBM model also achieves better prediction results in the feature engineering established in this paper, with an MSE value of 0.1031, MAE value of 0.2655, and goodness of fit of 0.9898, all of which achieve excellent prediction results; secondly, from the error distribution graph of dataset B, it can be seen that the capacity errors are mostly gathered in the range of 0–0.2 mAh, indicating that the model can achieve accurate SOH prediction under different datasets.

In summary, the prediction accuracy of the proposed manifold learning and LightGBM prediction models in different datasets is good, indicating that the models established in this paper have good generalization ability.

6. Conclusions

In order to solve the problem of low accuracy of SOH and RUL prediction caused by the difficulty of establishing the feature engineering of Li-ion batteries, this paper proposes an SOH and RUL prediction model based on manifold learning and LightGBM, and conducts relevant experiments to obtain the following conclusions:

- (1) In this paper, various features including average voltage decay, IC curve parameters, generalized multiscale entropy values, and autocorrelation coefficients are extracted from different angles, and the high-dimensional features are mapped to the low-dimensional space based on the L-ISOMAP manifold learning method, which effectively achieves dimensional simplification. The final obtained health factor sequences have high Spearman correlation coefficients between them and the true capacity.
- (2) In the field of battery SOH estimation, the feature engineering and prediction model established in this paper can effectively achieve its accurate prediction. In dataset A, the battery R^2 is higher than 0.98. In dataset B, the feature engineering and model

established in this paper also achieve good prediction results, which proves that the method in this paper has good generalizability.

- (3) Comparing with the random model and XGboost model commonly used for battery SOH prediction, the LightGBM model used in this paper has better performance and better results in several indicators, such as MSE and RMSE, and the MSE is reduced by 2.700×10^{-4} and 0.9685×10^{-4} compared with the random forest algorithm and XGBoost, respectively, where the AE indicators are all 0, indicating that the model has a lower RUL prediction error.
- (4) In the comparison experiments of different stream shape learning algorithms, the L-ISOMAP stream shape learning method used in this paper significantly outperforms the rest of the methods in terms of SOH prediction, and its prediction results have the highest overlap with the true capacity curve and perform the best.

Author Contributions: Conceptualization, J.Y.; data curation, M.Z.; methodology, J.Y.; project administration, M.Z.; resources, M.Z. and J.Y.; validation, J.Y.; visualization, T.F.; writing—original draft, J.Y.; writing—review and editing, M.Z. and J.Y. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the Natural Science Foundation of the Higher Education Institute of Anhui Province (KJ2020A0309), National Natural Science Foundation of China (51874010).

Data Availability Statement: The access URL for dataset A in the manuscript is: <https://ti.arc.nasa.gov/tech/dash/groups/pcoe/prognostic-data-repository/> (accessed on 10 February 2023).

Conflicts of Interest: The authors declare no conflict 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.

References

1. Liu, Y.; Zhang, D.; Li, X.; Han, Z. Research on SOH estimation method based on SOC-OCV curve characteristics. *Automot. Eng.* **2019**, *41*, 1158–1163.
2. Lin, M.; Wu, D.; Zheng, G.; Wu, J. Health state estimation of lithium battery based on surface temperature and incremental capacity. *Automot. Eng.* **2021**, *43*, 1284–1290.
3. Wang, Y.; Huang, H.; Yang, J.; Xu, S.; Yin, M. Research on power battery health condition prediction based on SSA-ELM. *Transm. Technol.* **2022**, *36*, 3–6+13.
4. Chu, Y.; Chen, Y.F.; Mi, Y. A CNN-LSTM Li-ion battery health state estimation based on attention mechanism. *Power Technol.* **2022**, *46*, 634–637+651.
5. Khumprom, P.; Yodo, N. A Data-Driven Predictive Prognostic Model for Lithium-ion Batteries based on a Deep Learning Algorithm. *Energies* **2019**, *12*, 660. [CrossRef]
6. Haifeng, D.; Xuezhe, W.; Zechang, S. A new SOH prediction concept for the power lithium-ion battery used on HEVs. In Proceedings of the 2009 IEEE Vehicle Power and Propulsion Conference, Dearborn, MI, USA, 7–10 September 2009; pp. 1649–1653.
7. Goh, H.H.; Lan, Z.; Zhang, D.; Dai, W.; Kurniawan, T.A.; Goh, K.C. Estimation of the state of health (SOH) of batteries using discrete curvature feature extraction. *J. Energy Storage* **2022**, *50*, 104646. [CrossRef]
8. Messing, M.; Shoa, T.; Habibi, S. Estimating battery state of health using electrochemical impedance spectroscopy and the relaxation effect. *J. Energy Storage* **2021**, *43*, 103210. [CrossRef]
9. Deng, Z.; Hu, X.; Li, P.; Lin, X.; Bian, X. Data-driven battery state of health estimation based on random partial charging data. *IEEE Trans. Power Electron.* **2021**, *37*, 5021–5031. [CrossRef]
10. Vichard, L.; Ravey, A.; Venet, P.; Harel, F.; Pelissier, S.; Hissel, D. A method to estimate battery SOH indicators based on vehicle operating data only. *Energy* **2021**, *225*, 120235. [CrossRef]
11. Qu, J.; Liu, F.; Ma, Y.; Fan, J. A neural-network-based method for RUL prediction and SOH monitoring of lithium-ion batteries. *IEEE Access* **2019**, *7*, 87178–87191. [CrossRef]
12. 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]
13. Shi, Y.S.; Li, J.; Ren, J.R.; Zhang, K. WOA-XGBoost-based prediction of remaining lifetime of lithium-ion batteries. *Energy Storage Sci. Technol.* **2022**, *11*, 3354–3363. [CrossRef]
14. Wu, J.; Zhang, C.; Chen, Z. An online method for lithium-ion battery remaining useful life estimation using importance sampling and neural networks. *Appl. Energy* **2016**, *173*, 134–140. [CrossRef]
15. Wang, P.; Fan, L.F.; Cheng, Z. Joint estimation method of SOH and RUL for lithium-ion batteries based on health characteristic parameters. *Chin. J. Electr. Eng.* **2022**, *42*, 1523–1534.

16. Wen, J.; Chen, X.; Li, X.; Li, Y. SOH prediction of lithium battery based on IC curve feature and BP neural network. *Energy* **2022**, *261 Pt A*, 125234. [CrossRef]
17. Zhang, J.; Jia, J.; Zeng, J. Residual life prediction of lithium batteries for electric vehicle power supply systems. *J. Electron. Meas. Instrum.* **2018**, *32*, 60–66. [CrossRef]
18. Zhang, M.; Chen, W.; Yin, J.; Feng, T. Health factor extraction of lithium-ion batteries based on discrete wavelet transform and soh prediction based on LightGBM. *Energies* **2022**, *15*, 5331. [CrossRef]
19. Wang, J.; Deng, Z.; Yu, T.; Yoshida, A.; Xu, L.; Guan, G.; Abudula, A. State of health estimation based on modified Gaussian process regression for lithium-ion batteries. *J. Energy Storage* **2022**, *51*, 104512. [CrossRef]
20. Kim, T.; Qiao, W.; Qu, L. Online SOC and SOH estimation for multicell lithium-ion batteries based on an adaptive hybrid battery model and sliding-mode observer. In Proceedings of the 2013 IEEE Energy Conversion Congress and Exposition, Denver, CO, USA, 15–19 September 2013; pp. 292–298.
21. Song, Z.; Gao, J.-P.; Pan, L.-S.; Xi, J.-G. Health state prediction of lithium-ion batteries based on principal component analysis and improved support vector machine. *Automot. Technol.* **2020**, *11*, 21–27. [CrossRef]
22. Yang, Z.S.; Wang, Y.H.; Kong, C.Z. Residual lifetime prediction of lithium batteries based on GWO-SVR. *J. Power Sources* **2021**, *4*, 1445–1457. Available online: <http://kns.cnki.net/kcms/detail/12.1420.TM.20210520.1300.002.html> (accessed on 29 March 2023).
23. Liu, Q.; Zhang, P. Remaining lifetime prediction of lithium batteries based on GBDT algorithm. *J. Electron. Meas. Instrum.* **2022**, *36*, 166–172.
24. Long, S.P.; Yuan, J.; Li, Z.W.; Chen, J.P.; Ning, K.; Sun, A. Application of integrated algorithms for remaining life prediction of engineering core components. *Equip. Manag. Maint.* **2021**, 34–37. [CrossRef]
25. Wang, J.; Wang, Z.; Li, J.; Peng, Y. An Interpretable Depression Prediction Model for the Elderly Based on ISSA Optimized LightGBM. *J. Beijing Inst. Technol.* **2023**, *32*, 168–180. [CrossRef]
26. Li, G.; Huang, M.; Wang, R.; Zheng, W. SOH Estimation of Power Battery Based on the Real Vehicle Multi-feature and LightGBM Algorithm. In Proceedings of the 2022 4th International Conference on Frontiers Technology of Information and Computer (ICFTIC), Qingdao, China, 2–4 December 2022; pp. 511–514. [CrossRef]
27. Nasiri, H.; Kheyroddin, G.; Dorrigiv, M.; Esmaeili, M.; Nafchi, A.R.; Ghorbani, M.H.; Zarkesh-Ha, P. Classification of COVID-19 in Chest X-ray Images Using Fusion of Deep Features and LightGBM. In Proceedings of the 2022 IEEE World AI IoT Congress (AIIoT), Seattle, WA, USA, 6–9 June 2022; pp. 201–206. [CrossRef]
28. Aggarwal, A.; Chakradar, M.; Bhatia, M.S.; Kumar, M.; Stephan, T.; Gupta, S.K.; Alsamhi, S.H.; Al-Dois, H. COVID-19 Risk Prediction for Diabetic Patients Using Fuzzy Inference System and Machine Learning Approaches. *J. Healthc. Eng.* **2022**, *2022*, 4096950. [CrossRef] [PubMed]
29. Ge, D.D.; Zhang, Z.D.; Kong, X.D.; Wan, Z.P. Extreme Learning Machine Using Bat Optimization Algorithm for Estimating State of Health of Lithium-Ion Batteries. *Appl. Sci.* **2022**, *12*, 1398. [CrossRef]
30. Li, Q.L.; Li, D.Z.; Zhao, K.; Wang, L.C.; Wang, K. State of health estimation of lithium-ion battery based on improved ant lion optimization and support vector regression. *J. Energy Storage* **2022**, *50*, 104215. [CrossRef]
31. Zhang, C.; Zhao, S.; Yang, Z.; Chen, Y. A reliable data-driven state-of-health estimation model for lithium-ion batteries in electric vehicles. *Energy Storage* **2022**, *10*, 1013800. [CrossRef]
32. Guo, Q.; Zhang, C.-P.; Gao, Y.; Jiang, J.; Jiang, Y. A health state estimation method for ternary lithium-ion batteries based on capacity incremental curve. *Glob. Energy Internet* **2018**, *1*, 180–187. [CrossRef]
33. Chelgani, S.C.; Nasiri, H.; Tohry, A.; Heidari, H. Modeling industrial hydrocyclone operational variables by SHAP-CatBoost—A ‘conscious lab’ approach. *Powder Technol.* **2023**, *420*, 118416. [CrossRef]
34. Dabiri, H.; Farhangi, V.; Moradi, M.J.; Zadehmohamad, M.; Karakouzian, M. Applications of Decision Tree and Random Forest as Tree-Based Machine Learning Techniques for Analyzing the Ultimate Strain of Spliced and Non-Spliced Reinforcement Bars. *Appl. Sci.* **2022**, *12*, 4851. [CrossRef]
35. Yang, R.; Xiong, R.; Shen, W.; Lin, X. ELM-Based Model in Extreme Learning Machine-Based Thermal Model for Lithium-Ion Batteries of Electric Vehicles under External Short Circuit. *Engineering* **2021**, *7*, 395–405. [CrossRef]
36. Zhao, S.; Zhang, C.; Wang, Y. Lithium-ion Battery Capacity and Remaining Useful Life Prediction Using Board Learning System and Long Short-Term Memory Neural Network. *J. Energy Storage* **2022**, *52 Pt B*, 104901. [CrossRef]
37. Liu, X.T.; Liu, X.J.; Wu, G.; He, Y.; Liu, X.T. SOH estimation for lithium-ion batteries based on curve compression with XGBoost algorithm. *J. Jilin Univ.* **2022**, *52*, 1273–1280. [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.