BIG DATA MINING AND ANALYTICS
ISSN 2096-0654 04/15 pp42–54
DOI: 10.26599/BDMA.2023.9010003
Volume 7, Number 1, March 2024

Cell Consistency Evaluation Method Based on Multiple Unsupervised Learning Algorithms

Jiang Chang*, Xianglong Gu, Jieyun Wu, and Debu Zhang

Abstract: Unsupervised learning algorithms can effectively solve sample imbalance. To address battery consistency anomalies in new energy vehicles, we adopt a variety of unsupervised learning algorithms to evaluate and predict the battery consistency of three vehicles using charging fragment data from actual operating conditions. We extract battery-related features, such as the mean of maximum difference, standard deviation, and entropy of batteries and then apply principal component analysis to reduce the dimensionality and record the amount of preserved information. We then build models through a collection of unsupervised learning algorithms for the anomaly detection of cell consistency faults. We also determine whether unsupervised and supervised learning algorithms can address the battery consistency problem and document the parameter tuning process. In addition, we compare the prediction effectiveness of charging and discharging features modeled individually and in combination, determine the choice of charging and discharging features to be modeled in combination, and visualize the multidimensional data for fault detection. Experimental results show that the unsupervised learning algorithm is effective in visualizing and predicting vehicle core conformance faults, and can accurately predict faults in real time. The “distance+boxplot” algorithm shows the best performance with a prediction accuracy of 80%, a recall rate of 100%, and an F1 of 0.89. The proposed approach can be applied to monitor battery consistency faults in real time and reduce the possibility of disasters arising from consistency faults.

Key words: battery consistency; charging segment data; unsupervised learning

1 Introduction

With the rapid development of new energy vehicles and machine learning technologies, the integration of new energy batteries and machine learning technology has become challenging. The inconsistency of a cell or cells in the battery pack occurs due to the existence of such a “short board” single cell, and affects the charging and discharging of the battery pack with good performance and full capacity from being fully charged and discharged. This inconsistency leads to a decrease in the overall battery system and a remarkably degraded performance[1]. For the “short board” of the single cell, the potential failures of the battery pack and system can only be effectively avoided by timely detecting the inconsistency fault[2] and timely replacing and renewing the single faulty cell to minimize the production losses brought by the failure.

The existing supervised learning algorithm has a high overfitting risk in inconsistent battery prediction and poor performance in unbalanced sample prediction. Compared with the supervised learning algorithm, the unsupervised learning algorithm has achieved good

• Jiang Chang, Xianglong Gu, Jieyun Wu, and Debu Zhang are with Stellantis China Technology Center, Shanghai 200233, China. E-mail: tqchangjiang@163.com; xianglong.gu@sts.com; jieyun.wu@sts.com; 392441820@qq.com.
* To whom correspondence should be addressed.
Manuscript received: 2022-12-28; revised: 2023-02-14; accepted: 2023-03-13

results in reducing the overfitting risk and effectively solving sample imbalance to overcome these shortcomings.

Detecting the inconsistency fault of a single battery in a battery pack in real time can help effectively improve the performance of the battery pack during the effective working time[3] and the service life of the battery. Some scholars and researchers proposed the equalization control strategy to effectively improve the battery pack inconsistency fault; however, only a few methods are available for detecting single-cell consistency[4] and "short boards". Given that the charging and discharging cycle of a single cell under actual operating conditions is not continuous, the battery pack of a new energy vehicle can rarely reach a full charging cycle (i.e., a complete charging cycle from 0% to 100%). As a result, the data acquired and detected from actual vehicles are mostly charging fragment data, including voltage, rather than completely equispaced time-series data so that the initial dataset is spaced.

Researchers often adopt a variety of methods when screening inconsistent batteries, such as the battery charging and discharging energy efficiency and voltage difference, the distance of the charging voltage curve, and the impedance variation trend during discharge[4]. In actual battery usage and inconsistency fault detection, we need to judge the fault according to the previous occurrence of consistency faults. Different battery systems have different fault performances between battery packs, and the inconsistency of individual battery within them varies according to the actual battery conditions. Therefore, the consistency of the battery system is not an absolute concept but a relative concept to be compared. The evaluation of the actual inconsistency fault of different units is often based on the empirical value[5], but changes in actual working conditions or parameters often lead to subjective errors in the results. Therefore, finding a method that can accurately and effectively predict single-cell inconsistency fault is important[6].

Among the existing methods for lithium battery cell or pack consistency evaluation, the technique in Ref. [7] considers charging data from the cloud in analyzing the consistency of voltage, temperature, internal resistance, capacity, and power, and adopts fuzzy hierarchical analysis to develop a resume weighted scoring system, which helps in evaluating and detecting consistency problems. Sun[8] proposed a segmented SOC consistency evaluation model considering the influence of voltage. Ji et al.[9] studied the dynamic consistency of power batteries in use and adopted an estimated SOC combined with the battery voltage to describe battery consistency.

In addition to the combination of data-driven and statistical correlation algorithms, we employ single-battery charge-discharge characteristic data to construct a fault prediction model. Using the unsupervised clustering algorithm as a basis[10], we explore outliers for anomaly detection. We also apply the statistical method of distance+boxplot to rank the urgency of problems according to the optimization coefficient. The above methods can effectively[11] and accurately detect outliers in the data in real time, and assess the consistency of the battery cells.

Our research adopt fragment data on lithium battery charging from actual running new energy vehicles, and apply various unsupervised algorithms to explore the core consistency evaluation approach. The specific steps include data preparation, i.e., exploration, feature engineering, and model construction.

2 Data Preparation and Exploration

2.1 Data preparation

A total of 1210 cell data are recorded for three battery types: A: PHEV, B: PHEV, and C: BEV. The analysis of cell consistency focuses on these three types of batteries. Each battery has different cell parameters, temperature measurement points, and nominal capacity and energy at the numerical level. The specific parameter values are shown in Table 1.

2.2 Data exploration

Data exploration involves considering the single-cell inconsistency fault, exploring and analyzing the data distribution of the maximum pressure difference in the single cell, and practicing the subsequent data-driven algorithm according to the actual data distribution of the three types of batteries.

Table 1 Information of battery parameter.

Battery type	Cell voltage quantity	Cell temperature quantity	Nominal Power (kW·h)
A: PHEV	96	50	13.20
B: PHEV	88	50	11.80
C: BEV	108	54	45.24

3 Feature Engineering

Feature engineering mainly includes three parts: data preprocessing, feature extraction, and Principal Component Analysis (PCA) for dimensionality reduction.

3.1 Data preprocessing

Data preprocessing includes screening and rejecting outliers and extracting and deleting duplicate values for the charge and discharge data of a single cell. Feature extraction requires predetermining the data range for feature extraction, and the selected data range is the last 1000 km of data for each battery. Thus, feature extraction is performed on this part of the data range, and the nearest data are selected. Abnormal value screening and rejection are performed specifically for the feature data related to the subsequent feature extraction, such as voltage, SOC, range, and other data. The normal range of SOC is 0–100, and that of voltage is 0–4.4 V. The range must not show a sudden and substantial increase. With the above normal value range used as a basis, the abnormal values of the features are screened and eliminated.

The nearest data are selected in feature extraction because, in the continuous charging and discharging of a battery, the consistency problem of its single cell may occur gradually but may also change abruptly in a short period of time. In general, fault rebound is impossible. When a single cell already has a consistency problem, it generally does not return to normal on its own but only gets worse. Therefore, the selection of the most recent data can ensure that the battery data in the near future will not have consistency problems. The reason for choosing data within 1000 km in the previous section is that the battery undergoes 3–5 charging and discharging cycles within 1000 km, and the produced charging fragment data are also the charging over discharging fragment data within nearly 1000 km. The recent 3–5 charges are enough to show whether the battery cell consistency problem exists.

3.2 Feature extraction

Feature extraction requires feature identification and extraction of all experimental core data. The specific features include the mean values of

(1) the maximum difference between the cores during the charging process;

(2) the standard deviation of the cores during the charging process;

(3) the entropy of the cores in the charging process;

(4) the maximum difference between the cores during the discharging process;

(5) the standard deviation of the cores during the discharging process; and

(6) the entropy of the cores during the discharging process.

The entropy value represents the chaos of a series of data.

3.3 PCA for dimensionality reduction

PCA downscaling in feature extraction aims to filter the extracted features. Given that the extracted features are multidimensional, processing them will reduce the training speed in the subsequent model building, and the results cannot be conveniently visualized. In addition, not all features are valid features, so PCA dimensionality reduction must be performed for multidimensional features to reduce the overall dimensionality of the feature attribute set. As a result, the model training speed can be substantially improved, and the modeling results can be effectively visualized each time.

After the PCA algorithm reduces the dimensionality, the new dimension represents the main direction of change in the dataset and is called the principal component. PCA_feature1, PCA_feature2, and PCA_feature3 are the first, second, and third principal components, respectively, after dimensionality reduction . Each principal component contains partial information from the dataset and is orthogonal to other principal components. The first few principal components contain the main information of the dataset. The coordinates of the sample on these new dimensions represent the information content of the sample in the main change patterns. Dimension reduction represents the main trend and pattern information of the original data by retaining the main principal components.

A current effective PCA method is chosen for data dimension reduction, and its core logic is as follows:

Suppose the algorithm input dataset is $x_{m\times n}$,

(1) Calculate the average value x_{mean} of dataset x by column, decentralize the data, and set $x_{\text{new}} = x - x_{\text{mean}}$;

(2) Solve for the covariance matrix x_{new} and call it $\text{Cov} = \frac{1}{m} x_{\text{new}} x_{\text{new}}^{\text{T}}$;

(3) Compute the eigenvalues and the corresponding eigenvectors of the covariance matrix Cov;

(4) Sort the eigenvalues from largest to smallest, select the largest k among them, and form the eigenvector matrix $w_{n\times k}$ with the corresponding k eigenvectors as column vectors;

(5) Calculate x^{w}_{new}, that is, project dataset x_{new} onto the selected feature vector to obtain the desired k-dimensional dataset x^{w}_{new} that has been reduced in dimension.

For the dataset used in this paper, the largest advantages of PCA dimension reduction are as follows:

(1) We can use PCA to reduce the dimension of the data and classify the importance of the newly obtained "principal element" vector. The most important part in the previous selection is determined by the need, and the following dimension is omitted to achieve dimensionality reduction and simplify the model or data compression. The original data are retained.

(2) The orthogonality among the principal components can eliminate the mutual influence among the original data components.

Finally, the features collected for the three battery types of new energy vehicles are processed by PCA dimensionality reduction, and the corresponding information for each type of vehicle can be found in Table 2. The information loss after dimensionality reduction is extremely small, and the effect is also small compared with the model speed and visualization perspective.

Data preprocessing and feature selection involve the data from charging and discharging. The charging segment data with SOC > 70 are selected for charging, and the discharging segment data with SOC < 30 are selected for discharging. The reason for choosing the range of SOC segments above is that battery inconsistency occurs within these ranges. The cell consistency problem is most evident at the end of charging and discharging; that is, it becomes evident as the SOC increases during charging.

4 Model Construction

The following are the two main reasons for choosing unsupervised learning to build the model:

Table 2 Information retention of different models after dimensionality reduction.

Battery type	Amount of information saved after dimensionality reduction (%)
A: PHEV	99.33
B: PHEV	98.42
C: BEV	99.91

(1) We take into account that the number of normal and abnormal samples is extremely unbalanced in the study of the consistency problem of batteries. The number of abnormal batteries is small. Applying a supervised learning algorithm for model construction will lead to a large machine-learning error and a large risk of overfittings. Unsupervised learning can overcome this problems. In particular, unsupervised learning algorithms include anomaly detection and novelty detection algorithms, both of which are suitable for finding a few problematic samples in a large dataset.

(2) The sample dataset trained in this work is small. If we use supervised learning, problems will arise in the proportion division of training samples and test samples. When the data training is complete, the test samples will not be enough. Unsupervised learning can overcome this problem by treating all samples as test samples.

The model construction comprises four parts: Density-Based Spatial Clustering of Applications with Noise (DBSCAN) algorithm based on density clustering, Isolation Forest and Local Outlier Factor (LOF) algorithm based on anomaly detection, One-Class SVM algorithm based on classification, and K-Nearest Neighbors (KNN) algorithm based on distance and statistics.

4.1 DBSCAN algorithm based on density clustering

The model-building step of the DBSCAN identifies isolated points that cannot form clusters as anomalies or noise points. EPS represents the radius threshold of the density clustering in the DBSCAN algorithm, which defines the maximum distance to determine whether two samples belong to the same cluster. EPS defines the neighborhood radius of a sample (if the distance between a sample and another sample in the dataset is less than or equal to EPS, then these two samples are considered to be samples within the neighborhood). For any sample, if the number of samples in its neighborhood (i.e., within a circular area with the sample as the center and EPS as the radius) reaches a certain minimum sample number MinPts, then the sample is a core sample. The core sample and samples within its neighborhood will be clustered into the same class, forming a cluster. If a sample is neither a core sample nor a neighborhood of any core sample, it will be labeled as a noise point. In summary, EPS

defines the maximum sample distance to determine whether it belongs to the same cluster. It, together with MinPts, determines the rules for clustering formation. Reasonably setting EPS and MinPts is the most critical step in the DBSCAN algorithm. Generally, EPS has a good effect in obtaining the moderate value of the distance distribution between samples in the dataset. The flow of the algorithm is as follows.

Step 1: Select an arbitrary data object point p from the dataset.

Step 2: If the selected data object point p is the core point for parameters Eps and MinPts, then find all data object points that are reachable from p density to form a cluster.

Step 3: If the selected data object point p is an edge point, then select another data object point.

Step 4: Repeat Steps 2 and 3 until all points are processed.

In the model construction, the training features and corresponding samples of the model must be established in advance. Although the extraction features must be extracted from the data during charging and discharging, past experiments showed that individual modeling based on charging features and discharging features is not effective. Hence, charging and discharging features are combined for modeling, as shown in Fig. 1, in the distance-based clustering algorithm using DBSCAN.

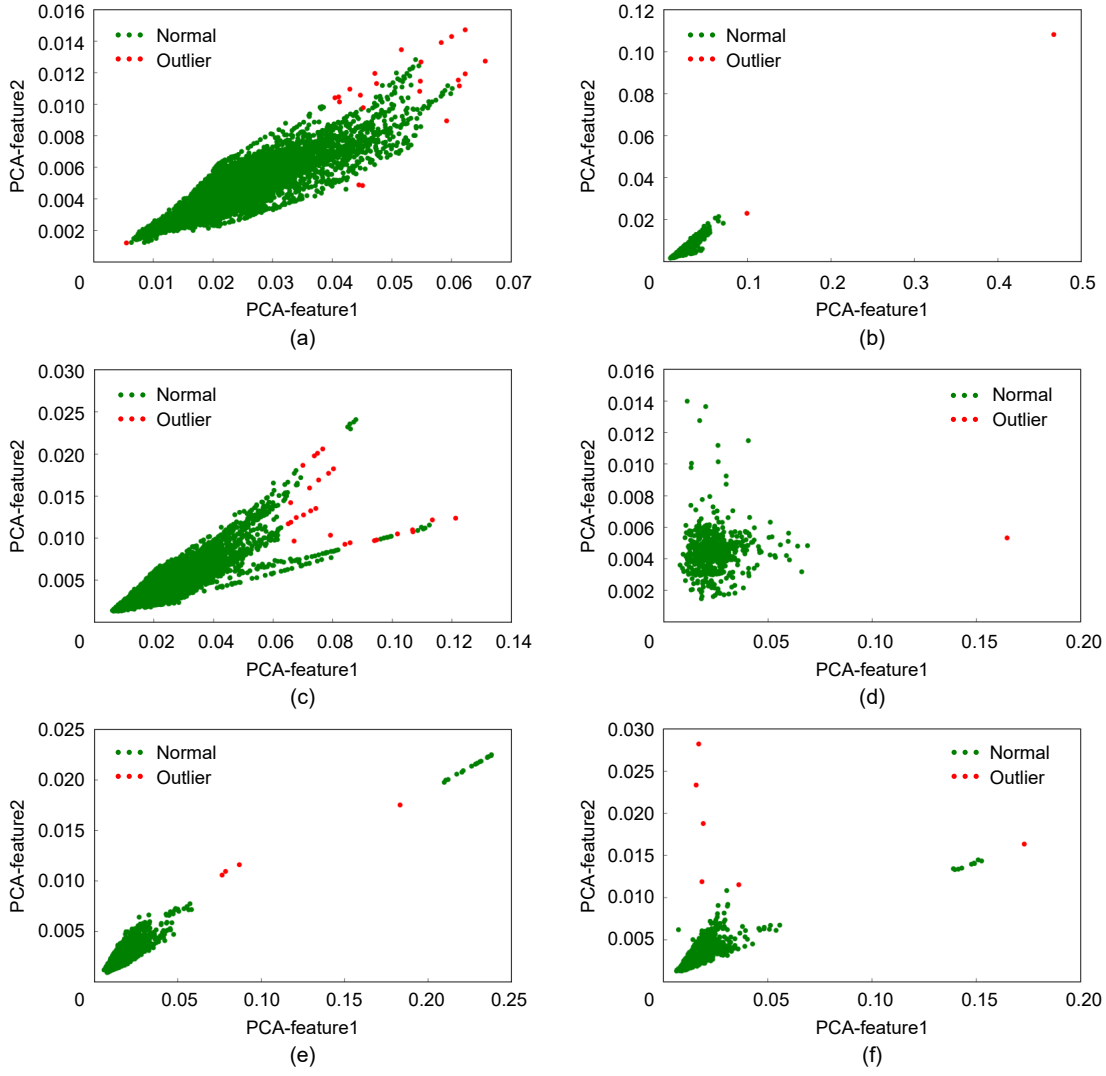


Fig. 1 Test results of charging and discharging characteristics. (a) Test result of charging characteristic for Type A battery, (b) test result of discharge characteristic for Type A battery; (c) test result of charging characteristic for Type B battery, (d) test result of discharge characteristic for Type B battery, (e) test result of charging characteristic for Type C battery, and (f) test result of discharge characteristic for Type C battery.

4.2 Isolation Forest and LOF algorithms based on anomaly detection

The Isolation Forest algorithm is an algorithm for anomaly detection. Outliers are detected to separate those which are easy to be isolated, and the points with sparse distribution and far away from the population with high density are considered abnormal data. In the known feature space, the sparsely distributed region represents the probability of events occurring in this region, so it can be considered as the outlier of abnormal data. The Isolation Forest algorithm adopts an efficient random segmentation strategy to recursively randomly segment the dataset until all the sample points become isolated. As a population of isolated trees, Isolation Forest algorithm identifies points with short path lengths as outliers. Different numbers act as experts for different outliers, among which abnormal outliers often have short paths.

The details within the Isolation Forest algorithm are as follows:

(1) Two subspaces can be generated at a time by using a random hyperplane to cut a data space. Next, we continue to randomly select hyperplanes to cut the two subspaces obtained in the first step and continue the cycle until each subspace contains only one data point. During the above process, those clusters with high density need to be cut many times before they stop being cut; that is, each point exists in a subspace alone. However, sparsely distributed points often enter early subspaces. Therefore, the whole idea of the Isolation Forest algorithm is that the abnormal samples are likely to fall into the leaves quickly, or the abnormal samples become close to the root node on the decision tree.

(2) M features are randomly selected, and the data points are split by randomly selecting a value between the maximum and minimum values of the selected features. The partitioning of observations is repeated recursively until all observations are isolated.

The schematic of the Isolated Forest algorithm is shown in Fig. 2.

The LOF algorithm in the model building step, namely, the local anomaly factor algorithm, is a classical algorithm based on density. LOF algorithm is an unsupervised anomaly detection algorithm that realizes anomaly detection by calculating the local density deviation of a given data point which is relative to its neighborhood. An outlier factor that depends on neighborhood density is assigned to each data point to determine whether the data point is an outlier. Its advantage is that it can quantify the abnormal degree of each data point.

The overall algorithm is as follows:

(1) For each data point, calculate its distance from all other points and sort it from near to far;

(2) For each data point, find its KNN and calculate the LOF score.

4.3 One-Class SVM algorithm based on classification

The One-Class SVM algorithm in the model-building

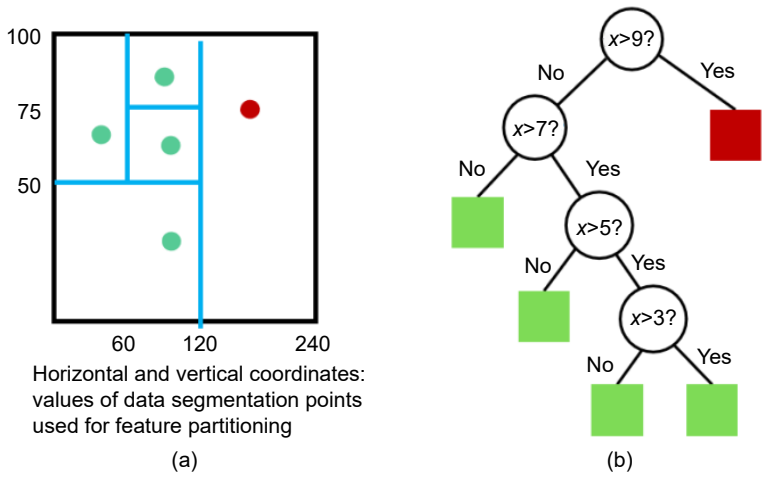


Fig. 2 Illustration of the Isolated Forest algorithm. (a) Partitioning graph illustrates how the feature space is recursively divided by the Isolation Forest algorithm to isolate anomalies and (b) Isolated Forest recursively splits the data space from the root node to leaf nodes, so that anomalies are further isolated into smaller partitions (green boxes represent all possible judgment results, and the red box represents the actual judgment result. *x* represents the feature value used for judgment).

step is a single classification algorithm. By finding a hyperplane, the positive example circle in the sample is drawn. The modified hyperplane is used for prediction, and the samples in the circle are positive samples. When the data dimension is not high, and no assumption on the distribution of relevant data has been established, we can regard the single classification algorithm as a simple and fast algorithm for unsupervised classification and anomaly detection.

This alogrithm is to find a hyperplane to circle the positive examples in the sample, and prediction is to use this hyperplane to make decisions. Samples inside the circle are considered positive samples, while samples outside the circle are considered negative samples. The division of the single classification algorithm to predict the sample signal is shown in Fig. 3.

4.4 KNN algorithm based on distance and statistics

The KNN algorithm in the model establishment step is the nearest neighbor algorithm. Its basic idea is as follows: a sample is most similar to K samples in the dataset. If most of the K samples belong to the same category, then the sample also belongs to this category. The algorithm flow is as follows: calculate the average distance between each sample point and its nearest K samples, and compare the calculated distance with the threshold. If the distance is greater than the threshold, then the sample is considered an outlier. The advantage of this algorithm is that it does not need to assume the distribution of data. The disadvantage is that only global outliers can be found, not local outliers. The abnormal prediction result is shown in Fig. 4, where each point represents a battery, the red point represents the normal point, and the green point represents the abnormal detection point.

In the distance+boxplot algorithm, the distance algorithm uses Euclidean distance to calculate the

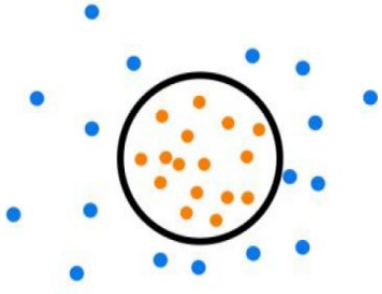


Fig. 3 Schematic of the One-Class SVM algorithm.

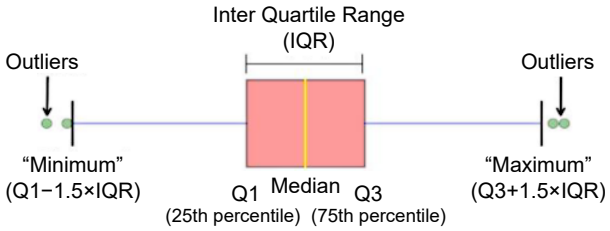


Fig. 4 Representation of the distance+box line graph algorithm.

distance between each point and other points. The boxplot algorithm is based on the interquartile range, which is much less for the abnormal point of retrieval, and the anomalous point probe signal is shown in Fig. 4.

4.5 Model parameter settings

The specific parameter adjustment of the model is carried out by grid cross-validation, as shown in Table 3.

In Fig. 1, PCA-feature1 and PCA-feature2 are the remaining two dimensional features after PCA dimensionality reduction on the basis of the original features, and the features of these two dimensions can represent all the feature information before PCA dimensionality reduction, and the specific values have no practical meaning, but only as numerical outlines. The same is true later in the article.

5 Experimental Result and Analysis

The abnormal prediction result of the DBSCAN algorithm is shown in Fig. 5, where each point represents a battery, the red point represents the normal point, and the green point represents the abnormal detection point.

In this work, we combine the charging and discharging characteristics of the three types of batteries to train the Isolated Forest algorithm. The training results are shown in Fig. 6.

In the distribution of the outliers of three types of batteries, each point represents a battery, the red points represent normal points, and the green points represent abnormal outliers. The Isolated Forest algorithm can effectively separate the cells corresponding to abnormal charge and discharge data and identify the single cells with consistent faults.

We combine the charging and discharging characteristics of the three types of batteries to train the LOF algorithm, and the training results are shown in Fig. 7. In the distribution of outliers of three types of

Table 3 Model parameter setting.

Algorithm	Parameter	Meaning	Setting	Best choose
Isolation Forest	n_estimators	How many iTrees to build	[50, 100, 150, 200]	150
	max_feature	Maximum characteristic number	[1, 2]	2
	bootstrap	Whether to replace sampling when building iTrees	[True, False]	True
LOF	m_neighbors	Number of neighbors	[20, 40, 60, 80, 100]	40
One-Class SVM	gamma	—	Default value	Default value
KNN	k_neighbors	Number of neighbors	[20, 40, 60, 80, 100]	20
DBSCAN	eps	Distance threshold of the neighborhood threshold of sample number in the neighborhood required for the sample point to become the core object	Get it by the elbow method	7
	min_samples		[2, 3, 4, 5, 6]	4
Logistics	—	No parameter adjustment is made; default parameters are used	—	—
Decision Tree	max_depth	Maximum depth of the tree maximum characteristic index used to divide features	[6, 7, 8, 9]	7
	max_features		[1, 2]	2

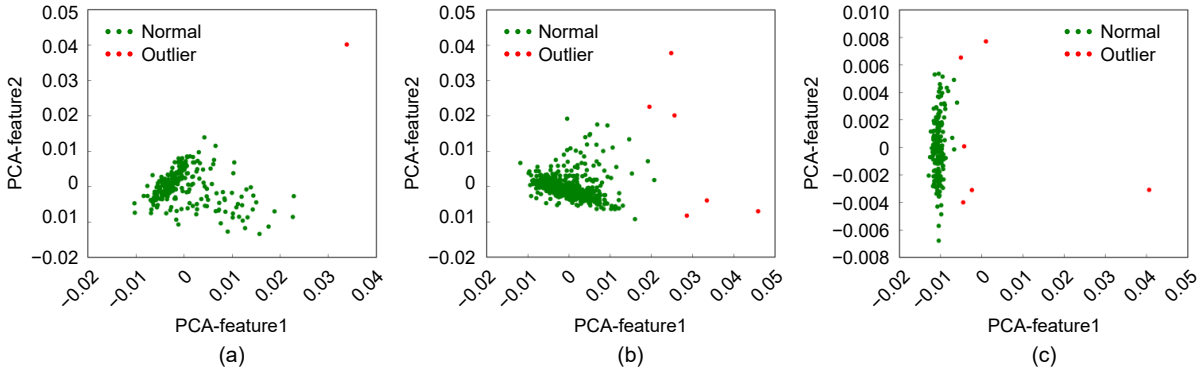


Fig. 5 DBSCAN results for three types of batteries. (a) Test results of charge and discharge characteristics for Type A battery, (b) test results of charge and discharge characteristics for Type B battery, and (c) test results of charge and discharge characteristic for Type C battery.

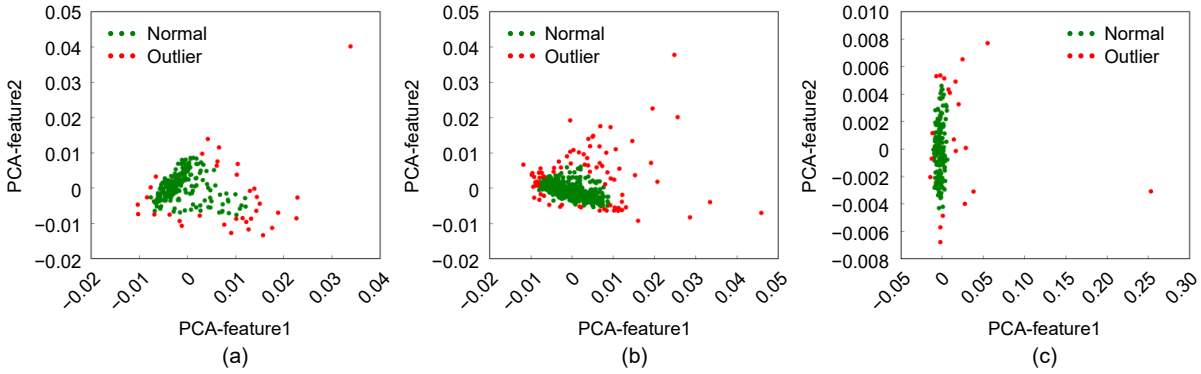


Fig. 6 Isolation Forest results for three types of batteries. (a) Test results of charge and discharge characteristics for Type A battery, (b) test results of charge and discharge characteristics for Type B battery, and (c) test results of charge and discharge characteristic for Type C battery.

batteries, each point represents a battery, the red point represents normal point, and the green point represents abnormal outlier. The abnormal prediction result of the One-Class SVM algorithm is shown in Fig. 8, where each point represents a battery, the red point represents the normal point, and the green point represents the abnormal detection point. The abnormal prediction result of the KNN algorithm is shown in Fig. 9, where each point represents a battery, the red point represents the normal point, and the green point represents the abnormal detection point. We use the distance+boxplot algorithm to predict abnormal outliers, as shown in

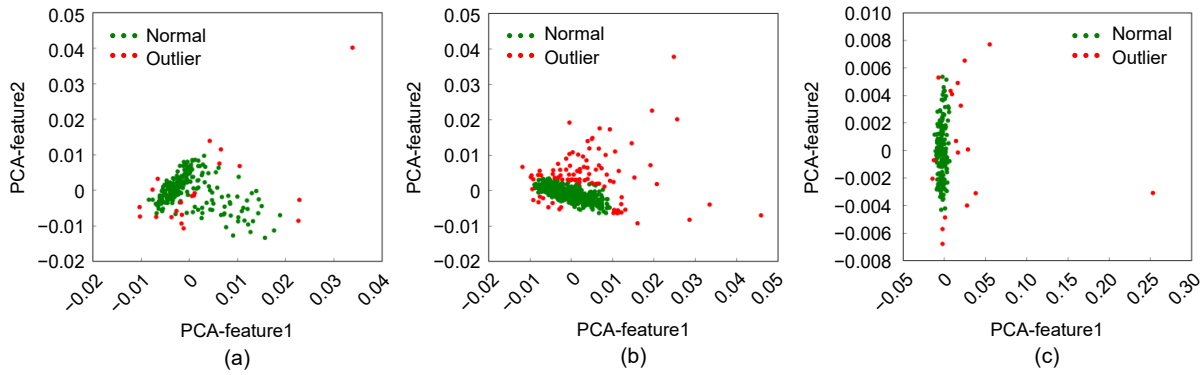


Fig. 7 LOF results for three types of batteries. (a) Test results of charge and discharge characteristics for Type A battery, (b) test results of charge and discharge characteristics for Type B battery, and (c) test results of charge and discharge characteristic for Type C battery .

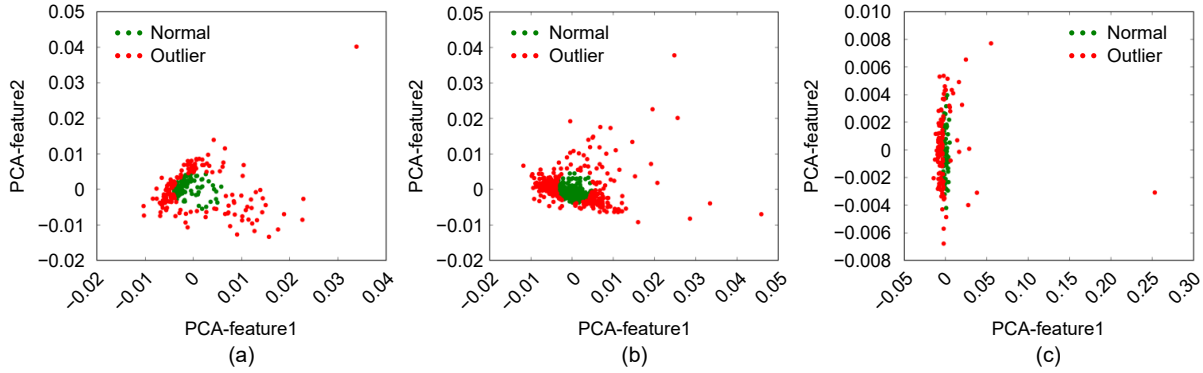


Fig. 8 One-Class SVM results for three types of batteries. (a) Test results of charge and discharge characteristics for Type A battery, (b) test results of charge and discharge characteristics for Type B battery, and (c) test results of charge and discharge characteristic for Type C battery.

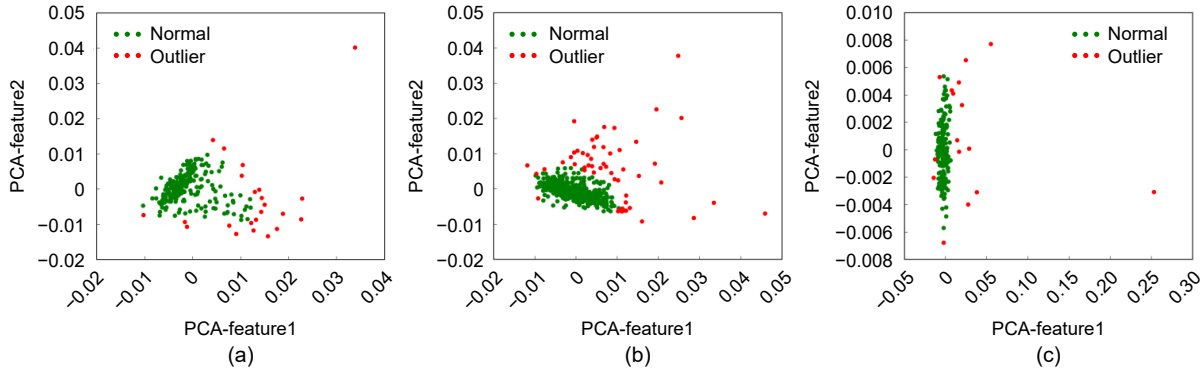


Fig. 9 KNN results for three types of batteries. (a) Test results of charge and discharge characteristics for Type A battery, (b) test results of charge and discharge characteristics for Type B battery, and (c) test results of charge and discharge characteristic for Type C battery.

Fig. 10, where each point represents a single cell, the red point represents the abnormal point, the yellow point represents the potential abnormal point, and the green point represents the normal point.

Analysis of experimental and modeling results shows that the DBSCAN algorithm and the distance+boxplot algorithm have the best anomaly detection and classification effect in terms of the detection and classification effect of outliers. In the actual training process, the modeling and DBSCAN algorithm trains the artificial parameter whose value is slightly influenced by the disturbance to the classification

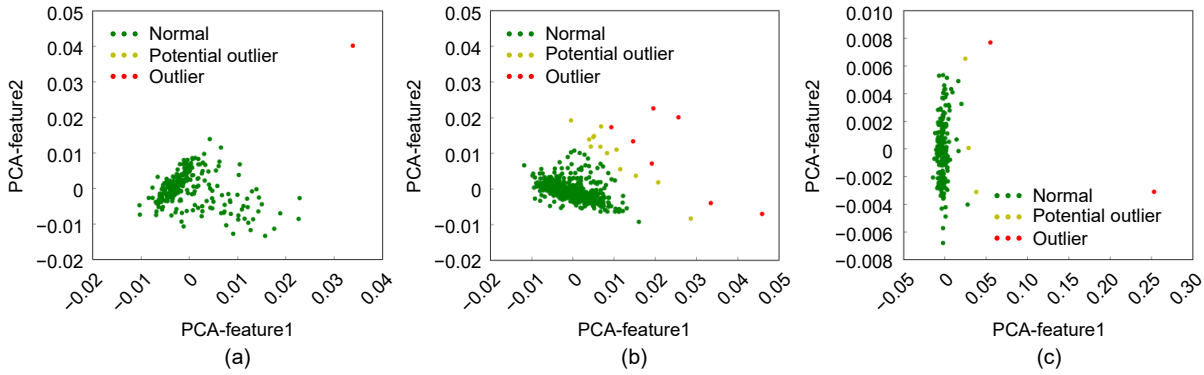


Fig. 10 Distance+boxplot results for three types of batteries. (a) Test results of charge and discharge characteristics for Type A battery, (b) test results of charge and discharge characteristics for Type B battery, and (c) test results of charge and discharge characteristic for Type C battery .

result. Meanwhile, the distance+boxplot algorithm has a simple and efficient implementation, can realize fully automated and output multilevel anomalies (i.e., potential and actual anomaly), and can classify various points.

Given their similar classification effect, this work choose the distance+boxplot algorithm over the DBSCAN algorithm to detect the consistency fault of a single cell.

The distance+boxplot modeling algorithm reveals abnormal data distribution shown in the form of 3D figures; they are shown in Fig. 11 for Type A battery, Fig. 12 for Type B battery, and Fig. 13 for Type C battery.

For the detection of abnormal cell consistency, this work selects one sample for each battery type to display the data. Figure 14 shows the data for Type A battery, Fig. 15 shows the data for Type B battery, and Fig. 16 shows the data for Type C battery. The different colored lines in Figs. 14–16 represent

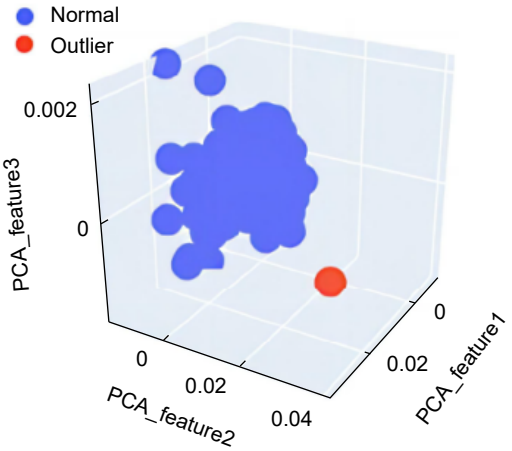


Fig. 11 3D characteristics of Type A battery.

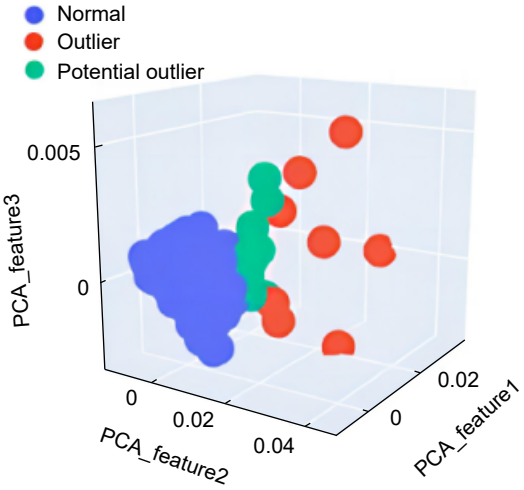


Fig. 12 3D characteristics of Type B battery.

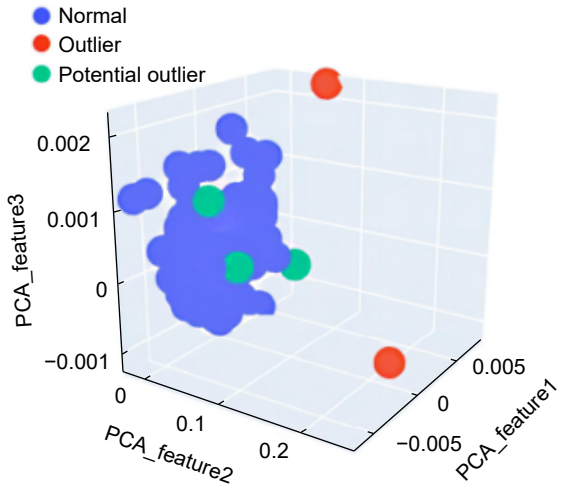


Fig. 13 3D characteristics of Type C battery.

different cell voltages. In particular, Type A has 88 cells, Type B has 96 cells, and Type C has 108 cells. Given that many cell voltages are close to each other

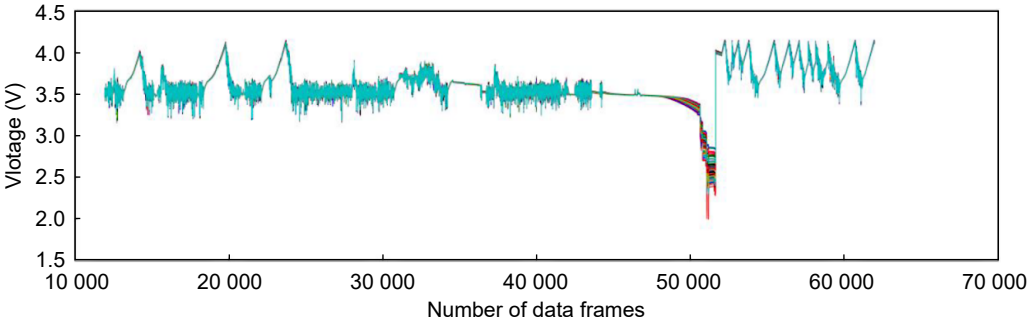


Fig. 14 Visualization of the single-cell consistency anomaly data detected in Type A battery.

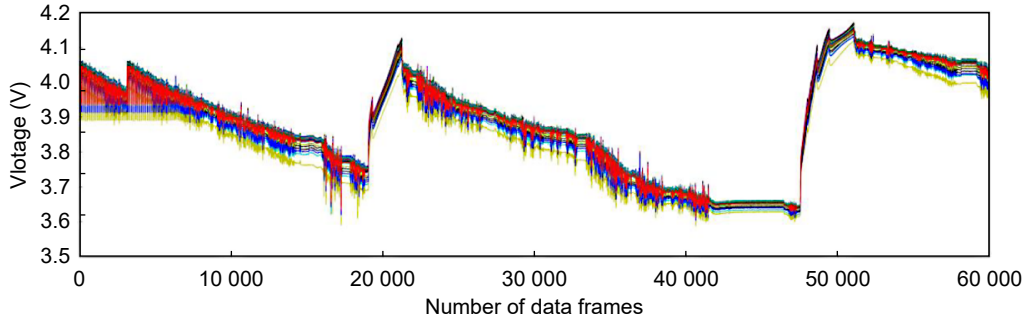


Fig. 15 Visualization of the single cell consistency anomaly data detected in Type B battery.

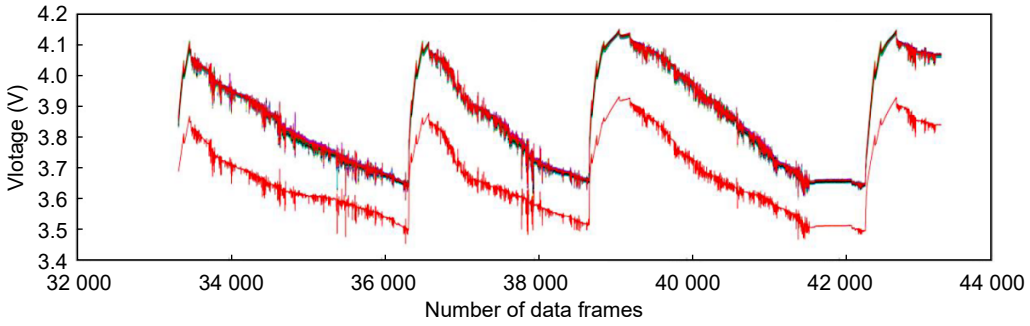


Fig. 16 Visualization of the single cell consistency anomaly data detected in Type C battery.

and the curve is repeated, not all the cells can be seen from Figs. 14–16. Instead, only the cells with consistency problems are visualized.

As shown in Figs. 14–16, the cells with abnormal consistency all have battery charging and discharging problems, and the cell consistency problems of each type are different. The algorithm adopted in this work can effectively detect these abnormalities.

To increase the reliability of the experimental results, we select some algorithms that are not unsupervised learning, and compare them with the unsupervised learning algorithms. We select eight abnormal samples and 80 normal samples as the test set. Recall rate, accuracy rate, and F1 index are measured for evaluation, and the test results are shown in Table 4.

Table 4 Index for evaluation of different algorithms.

Algorithm	Recall rate	Accuracy rate	F1 index
Logistic	0.75	0.60	0.67
Decision tree	0.88	0.64	0.74
Isolation Forest	1.00	0.29	0.45
LOF	1.00	0.31	0.47
One-Class SVM	1.00	0.17	0.29
KNN	1.00	0.21	0.35
DBSCAN	0.88	0.73	0.80
Distance+boxplot	1.00	0.80	0.89

We visualize the algorithm evaluation index, as shown in Fig. 17.

Figure 17 shows that the recall rate of supervised learning algorithms is generally not as high as that of

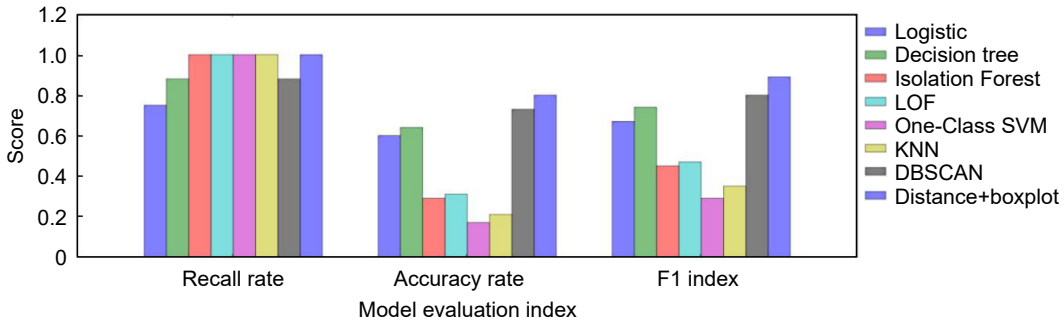


Fig. 17 Effect of consensus algorithms.

unsupervised learning algorithms. In terms of accuracy performance, most unsupervised learning methods have a low accuracy rate. Meanwhile, supervised learning algorithms have a relatively high accuracy rate. In summary, only two algorithms exhibit high recall and accuracy rates: distance+boxplot and DBSCAN. Compared with DBSCAN, the distance+boxplot algorithm has a better recall rate and accuracy rate. Therefore, the most effective algorithm is the distance+boxplot.

6 Conclusion

This work investigates the battery consistency problem in new energy vehicles with lithium batteries by adopting an unsupervised learning approach. Targeting the charging fragment data with uneven distribution of positive and negative samples in the actual working conditions, we select the charging and discharging features for PCA dimensionality reduction and compare the results with those from the supervised learning algorithm. We also compare the prediction effects of the charging and discharging features modeled individually and in combination. Experimental results show that the combination of charging and discharging features modeled by the unsupervised learning method can effectively overcome the overfitting problem and exhibit superior performance in terms of accuracy and recall rate.

References

[1] Y. J. Han, H. Y. Yuan, J. Li, J. Du, Y. M. Hu, and X. J. Huang, Study on influencing factors of consistency in manufacturing process of vehicle Lithium-Ion battery based on correlation coefficient and multivariate linear regression model, *Advanced. Theory. Simul.*, vol. 4, pp. 1–8, 2021.

[2] F. Wang, Z. Zhao, J. Ren, Z. Zhai, S. Wang, and X. Chen, A transferable lithium-ion battery remaining useful life prediction method from cycle-consistency of degradation trend, *J. Power Sources*, vol. 521, p. 230975, 2022.

[3] W. Han, K. Yu, L. Mao, Q. He, Q. Wu, and Z. Li, Evaluation of lithium-ion battery pack capacity consistency using one-dimensional magnetic field scanning, *IEEE. Trans. Instrum. Meas.*, vol. 71, p. 3507610, 2022.

[4] F. Wang, Z. Zhao, Z. Zhai, S. Wang, B. Ding, and X. Chen, Remaining useful life prediction of lithium-ion battery based on cycle-consistency learning, in *Proc. Int. Conf. Sensing, Measurement & Data Analytics in the Era of Artificial Intelligence*, Nanjing, China, 2021, pp. 1–6.

[5] Q. Wang and W. Qi, Study on influence of sorting parameters to lithium-ion battery pack life-cycles based on cell consistency, *Int. J. Electr. Hyb. Veh.*, vol. 10, no. 3, pp. 223–235, 2018.

[6] H. Wang, Z. Tao, Q. Ma, Y. Fu, H. Bai, Y. Zhu, H. Xiao, and H. Bai, Impact of initial open-circuited potential on the consistency of lithium ion battery, *IOP Conf. Ser.: Earth Environ. Sci.*, vol. 153, no. 2, p. 022023, 2018.

[7] Y. Lu, K. Li, X. Han, X. Feng, Z. Chu, L. Lu, P. Huang, Z. Zhang, Y. Zhang, F. Yin, et, al., A method of cell-to-cell variation evaluation for battery packs in electric vehicles with charging cloud data, *eTransportation*, vol. 6, pp. 2590−1168, 2020.

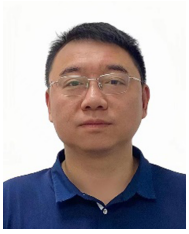
[8] S. Sun, Consistency and capacity estimation of lithium ion batteries for vehicles, (in Chinese), Master dissertation, Qingdao University of Science & Technology, Qingdao, China, 2019, pp. 1–32.

[9] Y. R. Ji, J. Pang, L. Tang, and Z. Ding, Research progress in evaluation methods of consistency of Li-ion power battery, *Battery*, vol. 44, no. 1, pp. 53–56, 2014.

[10] J. Q. Tian, Y. J. Wang, C. Liu, and Z. H. Chen, Consistency evaluation and cluster analysis for lithium-ion battery pack in electric vehicles, *Energy*, vol. 194, p. 116944, 2020.

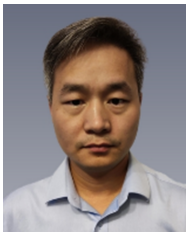
[11] X. Bai, J. Tan, X. Wang, L. Wang, C. Liu, L. Shi, and W. Sun, Study on distributed lithium-ion power battery grouping scheme for efficiency and consistency improvement, *J. Clean. Prod.*, vol. 233, pp. 429–445, 2019.

Jiang Chang received the MEng degree in energy and electric automatic science from Polytech Orleans University, French in 2017, and the Engineer degree in energy and electric automatic science from Paris-Saclay University, French in 2016. He is currently a project manager at Stellantis China Technology Center, Shanghai, China. His main research interests include optimizing control for energy flow, advanced electric/hybrid powertrain architecture design, and big data science and engineering.



Jieyun Wu received the MEng degree in powertrain engineering from Tsinghua University, China in 2005. He is currently the head of Powertrain Department, Stellantis China Technology Center, Shanghai, China. His main research direction is in the field of powertrain and high voltage battery, including machine learning, data science and engineering, and data mining.

Xianglong Gu received the MEng degree in internal combustion engine from Tianjin University, China in 2005. He is currently a senior manager at Stellantis China Technology Center, Shanghai, China. His main research interests include propulsion system performance simulation, software control strategy, and system integration and calibration.



Debu Zhang received the MEng degree in vehicle engineering from Shanghai Jiaotong University, China in 2010. He is currently a section manager at Stellantis China Propulsion System, Stellantis China Technology Center, Shanghai, China. His main research interests include calibration tuning for new energy vehicle and battery big data analysis.