batteries

MDPI

Article

A Methodological Framework for Incremental Capacity-Based Feature Engineering and Unsupervised Learning Across First-Life and Second-Life Battery Datasets

Matthew Beatty, Dani Strickland * and Pedro Ferreira

Wolfson School of Mechanical, Electrical, and Manufacturing Engineering, Loughborough University, Loughborough LE11 3TU, UK; m.s.c.beatty@lboro.ac.uk (M.B.); p.ferreira@lboro.ac.uk (P.F.)
* Correspondence: d.strickland@lboro.ac.uk

Abstract

Accurately assessing battery health across mixed datasets remains a challenge due to differences in chemistry, format, and usage history. This study presents a reproducible framework for preparing battery cycling data using incremental capacity analysis (ICA), with the aim of supporting machine learning (ML) workflows across both first-life and second-life battery datasets. The methodology includes IC curve generation, feature extraction, encoding and scaling, feature reduction, and unsupervised learning exploration. A two-tiered outlier detection system was introduced during preprocessing to flag edge-case samples. Two clustering algorithms, K-means and HDBSCAN, were applied to the engineered feature space to explore patterns in the IC feature space. K-means revealed broad health-related groupings with overlapping boundaries, while HDBSCAN identified finer clusters and flagged additional ambiguous samples as noise. To support interpretation, PCA and t-SNE were used to visualise the feature space in reduced dimensions. Rather than using clustering as a classification tool, the resulting cluster and noise labels are proposed as structure-aware meta-features for supervised learning. The framework accommodates heterogeneous battery datasets and addresses the challenges of integrating data from mixed sources with varying histories and characteristics. These outputs provide a structured foundation for future supervised classification of battery state of health.

Keywords: electric vehicles; second-life batteries; incremental capacity; sorting and grading; machine learning; methodology

Check for updates

Academic Editor: Rui Xiong

Received: 21 January 2026
Revised: 3 February 2026
Accepted: 4 February 2026
Published: 6 February 2026

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

1. Introduction

The widespread deployment of lithium-ion batteries (LIBs) in electric vehicles (EVs) and stationary energy storage systems has highlighted the need for reliable methods to monitor battery health and assess degradation. Accurate characterisation of battery ageing processes is needed for ensuring performance, safety, and cost-effectiveness throughout both first-life and second-life applications [1,2].

Second-life batteries (SLBs), repurposed from EVs for applications such as residential storage and grid services, present a growing opportunity to extend battery utilisation and reduce environmental impact [3,4]. However, assessing the state of health (SOH) and remaining useful life (RUL) of SLBs remains challenging. While initiatives such as a battery passport have been proposed to improve data sharing and traceability, they have not been widely adopted [5]. This means that SLBs are often designed by different

manufacturers with their own specifications, formats, and chemistries. Examples include chemistries such as lithium iron phosphate ($LiFePO_4$), nickel–manganese–cobalt (NMC), lithium cobalt oxide (LCO), and nickel–cobalt–aluminium (NCA), each with different voltage profiles, capacities, and ageing behaviours. SLBs are also manufactured in various formats, including cylindrical, prismatic, and pouch designs, adding further complexity to testing and analysis [6,7].

Alongside this, most SLBs enter the second-life market without records of how they were used during their first life. As a result, reliable information about usage history is often unavailable, making it difficult to apply condition-specific models or data-driven methods for health assessment. These factors make it challenging to apply consistent approaches to assessing battery health, particularly across mixed first-life and second-life datasets.

Incremental capacity analysis (ICA) provides a non-invasive technique for characterising battery degradation under these conditions. By transforming voltage and charge data into incremental capacity (IC) curves, ICA enables the extraction of features such as peak positions, voltage spans, and curve areas, which are known to evolve as batteries age [8,9]. While ICA has been applied to first-life batteries (FLBs) tested under controlled conditions, its potential for consistent application to undocumented, heterogeneous SLBs remains underexplored.

Building on previous work that established a standardised IC curve generation procedure for FLBs and SLBs [10], the present study advances the pipeline needed for machine learning (ML). Specifically, it moves beyond curve generation to multi-dataset IC feature extraction across heterogeneous sources and focuses on the next steps of feature extraction, systematic preprocessing, and unsupervised structure exploration using clustering and dimensionality reduction techniques. Even more, a recently published long-term SLB dataset generated under application-inspired cycling conditions is utilised to evaluate the applicability of ICA-derived features under second-life conditions [11].

The methodological framework developed in this study is designed to process both FLB and SLB datasets, including those with unknown operational histories and varied characteristics. As EV registrations increase and more SLBs enter the recycling and repurposing pathways, there is a need for a single, transparent process that can sort and grade batteries irrespective of chemistry or form factor [12]. Processing at scale requires methods that can handle large volumes of heterogeneous batteries efficiently. In practice, complete battery metadata such as capacity measurements, cycling history, and chemistry specifications can often be unavailable. Incremental capacity feature extraction addresses this limitation by deriving health indicators from basic charge and discharge data available from any battery test. Chemistries also differ in their operating voltage windows, nominal capacities, current limits, and protocol constraints. As summarised in prior work [10], studies have applied ICA to batteries with differing formats and chemistries, with capacities ranging from 0.4 Ah to 160 Ah and voltage ranges commonly near 2.5–4 V. This heterogeneity motivates analysing heterogeneous datasets in a shared IC feature space, where commonly available features and labels can be extracted, to progress towards a single process for SOH estimation across chemistries and formats.

To accommodate these differences, clustering excludes capacity and SOH variables from the inputs, and per-dataset scaling was applied to reduce dominance from larger capacity regimes. This scaling aims to normalise different operating ranges while seeking to preserve health-related patterns, supporting chemistry-agnostic analysis that can provide meaningful assessment across battery types. Together, these steps aim to yield a consistent representation in which shared structure can be examined and form the basis for future SOH classification using ML techniques. In contrast to previous studies that have focused on controlled, single-chemistry datasets, this work applies a reproducible feature extraction

and analysis methodology across batteries with differences in design, chemistry, and usage history.

The contributions of this work are as follows:

1. A unified, transparent methodology is presented for ICA feature extraction, data aggregation, and systematic preprocessing, applicable to both FLB and SLB datasets.
2. The methodology is applied across heterogeneous battery datasets to evaluate its applicability under varied operational histories, chemistries, and formats to test its applicability in a common IC feature space.
3. Unsupervised structure exploration techniques, including clustering and dimensionality reduction, are used to investigate trends in the resulting feature space.

The remainder of this paper is structured as follows: Section 2 describes the battery datasets used in this study and provides an overview of the ICA approach. Section 3 details the feature extraction, data aggregation, and preprocessing steps. Section 4 presents the unsupervised structure exploration methodology. Section 5 reports the results and visualisation of structural trends. Section 6 discusses the implications and limitations of the proposed framework, and Section 7 concludes the work.

2. Data Sources and Incremental Capacity Analysis

This study combines publicly available FLB datasets with a recently published SLB dataset to evaluate a unified methodological framework for ICA feature extraction and exploratory analysis. The dataset differs in chemistry, form factor, operational history, and testing conditions, providing an opportunity to apply the proposed methodology across batteries with varied characteristics. These datasets have been used in previous research focused on IC-DV curve generation methodologies [10]. In the present work, the same datasets are employed, with extended samples from the second-life dataset to incorporate additional use cases beyond those explored previously.

All analysis in this study was conducted in Python (version 3.12.4) using the Jupyter Notebook environment (version 7.0.8) for its flexibility in data manipulation and visualisation. The following libraries were used throughout:

- Pandas (2.2.3): Data handling and manipulation;
- NumPy (1.26.4): Numerical operations and array handling;
- Matplotlib (3.8.4): Data visualisation and plotting;
- Seaborn (0.13.2): Statistical visualisation and styling;
- SciPy (1.31.1): Peak detection, signal filtering, and numerical integration;
- Scikit-learn (1.4.2): Clustering, scaling, encoding, and dimensionality reduction;
- Hdbscan (0.8.40): Density-based clustering;
- Kneed (0.8.5): Knee point detection for clustering parameter tuning.

2.1. Dataset Description

The first life dataset is provided by the University of Oxford [13] and contains long-term cycling data for eight Kokam lithium nickel cobalt aluminium oxide (NCA) pouch cells. Each cell has a nominal capacity of 0.74 Ah and was cycled within a voltage range of 2.7 V to 4.2 V. The cells were tested at 40 °C using a repeated charge–discharge regime until end of life (EOL), defined as 70% SOH, with periodic characterisation tests conducted throughout the dataset.

The second first-life dataset originates from the Nature publication by Severson et al. [14] and comprises cycling data for 124 cylindrical 18650-format lithium iron phosphate ($LiFePO_4$) cells. These cells have a nominal capacity of 1.1 Ah and a voltage range of 2.0 V to 3.6 V. The cells were aged under varied fast-charging protocols at 30 °C to accelerate performance degradation and cycled to a defined EOL SOH of 80%.

The second-life dataset from Loughborough University comprises lithium-ion modules previously used in automotive applications, with the first-life operational history being unknown. The modules contain blended lithium manganese oxide (LMO) and lithium nickel oxide (LCO) chemistries and have a nominal capacity of 66 Ah. At the start of testing, the estimated SOH of the modules was approximately 70%. Each module has a voltage range of 5.0 V to 8.3 V. The modules have been cycled under six distinct, application-inspired use cases representative of stationary energy storage, grid services, and energy market participation [11]. In previous work [10], only one of these use cases was included in the ICA curve generation analysis, but in this study, the dataset has been extended to incorporate all six use cases.

2.2. Dataset Differences and Challenges

The combined datasets used in this study differ across multiple technical and operational characteristics. Three distinct battery formats are present: cylindrical cells, pouch cells, and large-format prismatic modules assembled from a 2S2P pouch cell configuration. The datasets also differ in nominal capacity, with values of 0.74 Ah, 1.1 Ah, and 66 Ah, and in voltage specifications, with ranges of 2.7–4.2 V, 2.0–3.6 V, and 5.0–8.3 V, respectively.

Chemistry variations are also present, including NCA, $LiFePO_4$, and blended LMO-LNO chemistries. In addition to these design differences, the first-life datasets provide complete cycling information from new to EOL, whilst the second-life dataset contains batteries with unknown first-life operational history. As a result, there is no available data capturing degradation between 100% and 70% SOH for the SLBs.

The combination of different formats, capacities, chemistries, voltage ranges, and incomplete operational histories introduces uncertainty regarding whether the SLBs can be treated in the same way as the FLBs for ICA feature extraction and analysis. More broadly, the heterogeneity across all three datasets presents challenges when applying a consistent, unified methodology. Nevertheless, this variability provides an opportunity to assess the applicability of a standardised ICA feature extraction and analysis process across batteries with different characteristics and histories. This design choice reflects real-world second-life scenarios, where detailed usage history and cycling protocol information can be unavailable, and motivates a data-driven, standardised framework that can operate under such uncertainty.

2.3. Incremental Capacity Analysis and Data Aggregation

To enable consistent health feature (HF) extraction across datasets, the IC curve generation methodology established in previous work [10] was applied to each usable charge cycle from the Loughborough Sweat Test, Oxford, and Nature datasets. This prior study systematically evaluated the influence of smoothing, differentiation, and filtering parameter choices on IC curve stability and signal integrity and established a processing procedure that balances noise suppression with retention of degradation-relevant structure. This approach ensured a unified set of IC curves for subsequent feature extraction and analysis.

For the Loughborough dataset, five of the six use cases were processed successfully. The EFR use case was excluded from standard processing due to its high-frequency pulsing test profile, which resulted in two distinct challenges. First, the brief 60 s charge and discharge pulses often produced incomplete IC curves with insufficient data points. Second, when IC curves were generated, they frequently displayed erratic patterns, characterised by repeating peaks and troughs rather than a clear degradation-related trend.

To address these challenges, a data-driven filtering criterion was applied. Based on iterative testing, a threshold of 200 valid dQ/dV data points was selected as the minimum requirement for curve retention. This ensured that each curve had sufficient resolution to

capture meaningful health-related features. Within the EFR use case, only capacity test cycles containing structured charge segments that met this threshold were treated. This filtering approach ensured that only consistent and interpretable IC curves were included for the Loughborough dataset.

Figure 1 illustrates an irregular IC curve with 175 valid dQ/dV points. It exhibits an erratic, multi-peak pattern and falls below the 200-point threshold and, therefore, was excluded.

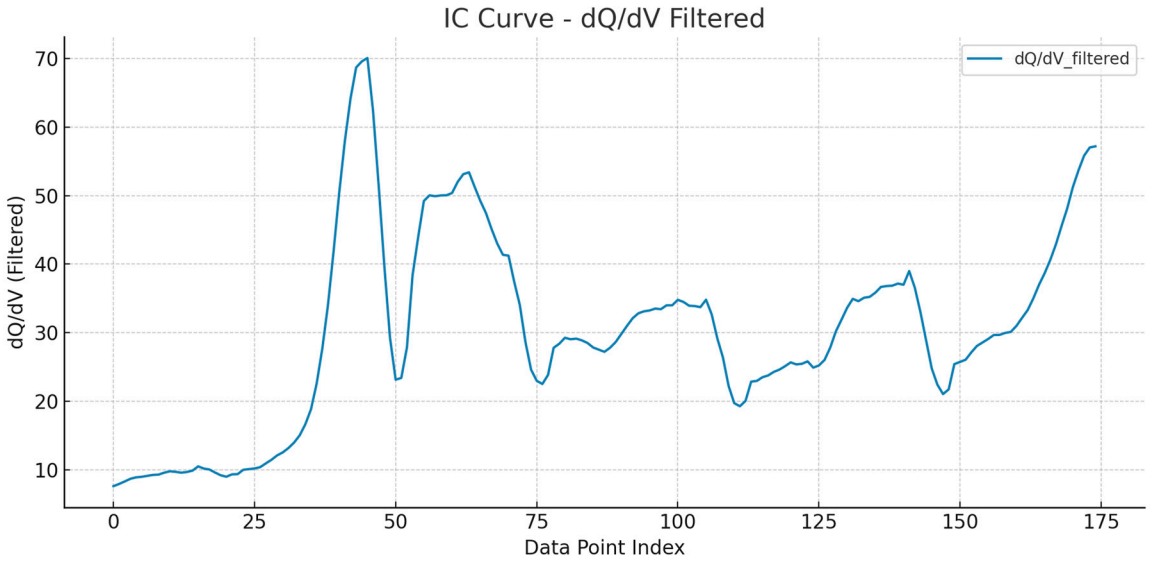


Figure 1. Loughborough sweat test data—irregular incremental capacity curve for EFR use case.

IC curves were generated independently for each dataset subset to account for differences in voltage ranges and battery chemistries. To remove low-voltage segments that contributed noise or non-diagnostic content, a lower voltage threshold was applied, retaining only data above this value. For the Loughborough dataset, a threshold of 7.0 V was used, reflecting the higher voltage range of the LMO-LNO modules. For the Nature dataset, containing lower-voltage LFP 18,650 cells, the voltage threshold was adjusted to 3.2 V to capture diagnostic features present in that chemistry. Accordingly, ICA-derived features are treated in this study as data-driven descriptors of voltage–capacity behaviour, without assuming electrochemical equivalence across chemistries.

Following successful IC curve generation, feature extraction was performed on each processed cycle using a custom Python script. General statistical properties of the dQ/dV signal, including the mean, standard deviation, and variance, were computed to reflect the spread and distribution of the IC response and have been commonly used to characterise curve shape and signal quality [15,16]. The area under the curve (AUC) was calculated using Simpson's rule from the Scipy library [17], providing a metric that the literature has associated with capacity loss and degradation [18].

The IC curves were further filtered to isolate regions above the defined voltage threshold, after which peak and trough detection was performed using the find_peaks() function from the SciPy library [19]. To ensure robust detection across curves of varying magnitude, the prominence threshold was dynamically set to 20% of the maximum dQ/dV value within the filtered region while suppressing noise-induced artefacts [20]. An example of the extracted peaks and troughs for an IC curve from the Loughborough dataset is shown in Figure 2.

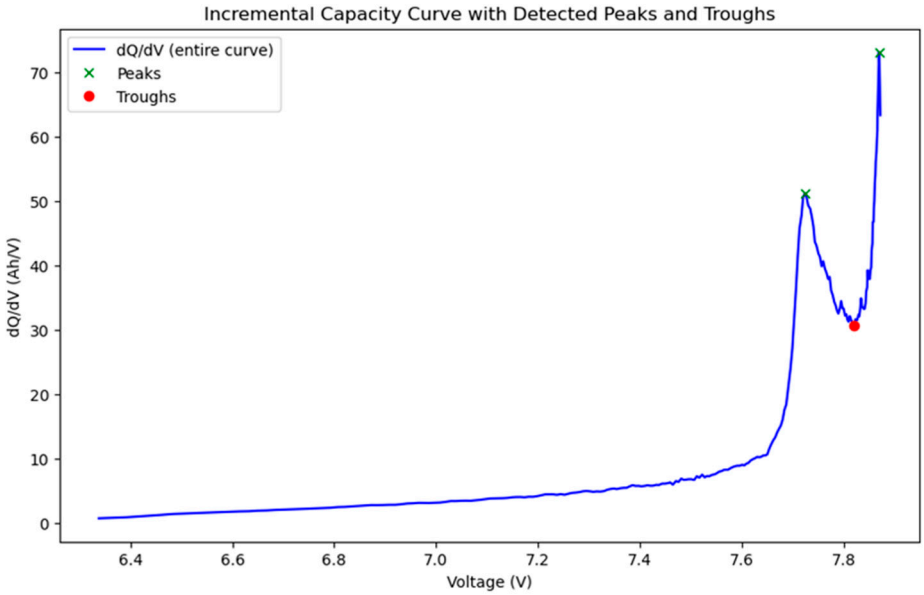


Figure 2. Loughborough sweat test data—extracted peak and troughs from the incremental capacity curve for channel 5, 2022, cycle 370.

To enhance diagnostic coverage, the global maximum peak and minimum trough were also included in the analysis in cases where they were omitted by the automatic detection process. From the identified peaks and troughs, the following HFs were extracted:

- The number of peaks and troughs, reflecting feature count variations linked to degradation phenomena such as electrode slippage and SEI growth [21].
- Peak height and trough depth, providing a measure of pronounced electrochemical transitions affected by ageing and loss of active material [22,23].
- Voltage positions corresponding to the most extreme features, which may shift due to phase transitions or electrode imbalance [24].
- A peak-to-trough ratio, capturing asymmetries in the IC curve structure, which can indicate uneven degradation or charging inefficiencies [25].

In addition to these HFs, each cycle was labelled with a corresponding SOH value. For the Oxford and Nature datasets, SOH was calculated from directly measured capacity values. For the Loughborough dataset, where per-cycle capacity data were unavailable, SOH was estimated by linearly interpolating between reliable capacity measurements obtained at the start and end of each use case. Time-stamped cycle information was used to perform this interpolation, providing an estimated SOH trajectory for all cycles.

A summary of the extracted features is provided in Table 1. These features were extracted consistently across all datasets and cycles and stored alongside metadata, including cycle number, test identifier, channel label, cell chemistry, and dataset origin.

Table 1. Description of the features extracted from the incremental capacity curves.

Feature Name	Description
mean_dQdV	Mean value of the filtered incremental capacity curve
std_dQdV	Standard deviation of the filtered incremental capacity curve
var_dQdV	Variance of the filtered incremental capacity curve
auc	Area under the incremental capacity curve (using Simpson's rule)
num_peaks	Number of peaks detected in the incremental capacity curve

Table 1. *Cont.*

Feature Name	Description
num_troughs	Number of troughs detected in the incremental capacity curve
max_peak_value	Maximum value of detected peaks
min_peak_value	Minimum value of detected troughs
peak_trough_ratio	Ratio of maximum peak to minimum trough
curve_width	Total number of data points in the incremental capacity curve
voltage_span	Voltage range between the start and end of the incremental capacity curve
cycle_number	Identifier for the cycle within the use case
SOH	State of health calculated as a percentage of nominal capacity
current_capacity	Interpolated or measured capacity for the cycle
channel_label	Battery tester channel identifier
cell_chemistry	Battery cell chemistry
cycle_peak_shift	Change in peak position relative to the reference cycle
auc_change_rate	Rate of change over the AUC over cycles
date_range	Time window associated with test cycle (used for capacity interpolation)
data_id	Unique ID for tracing data origin

Following feature extraction, the Oxford, Nature, and Loughborough datasets were combined into a single, aggregated dataset. Column structures were aligned across all datasets, and an identifier column was introduced to distinguish the source of each observation.

3. Preprocessing and Feature Engineering

The aggregated dataset described in Section 2 was used to prepare HFs for subsequent ML model development. A typical ML pipeline begins with data preparation, feature engineering, and exploratory analysis, followed by model training and evaluation. These steps ensure the dataset is suitable for downstream predictive modelling, including supervised learning for SOH or RUL classification.

The key stages of the ML pipeline are illustrated in Figure 3. In this study, only the data preparation, feature engineering, and exploratory analysis stages are covered. Supervised model training and evaluation will be presented in a separate follow-up paper. The following sub-sections describe the cleaning, feature engineering, and scaling processes applied to the aggregated dataset prior to model development.

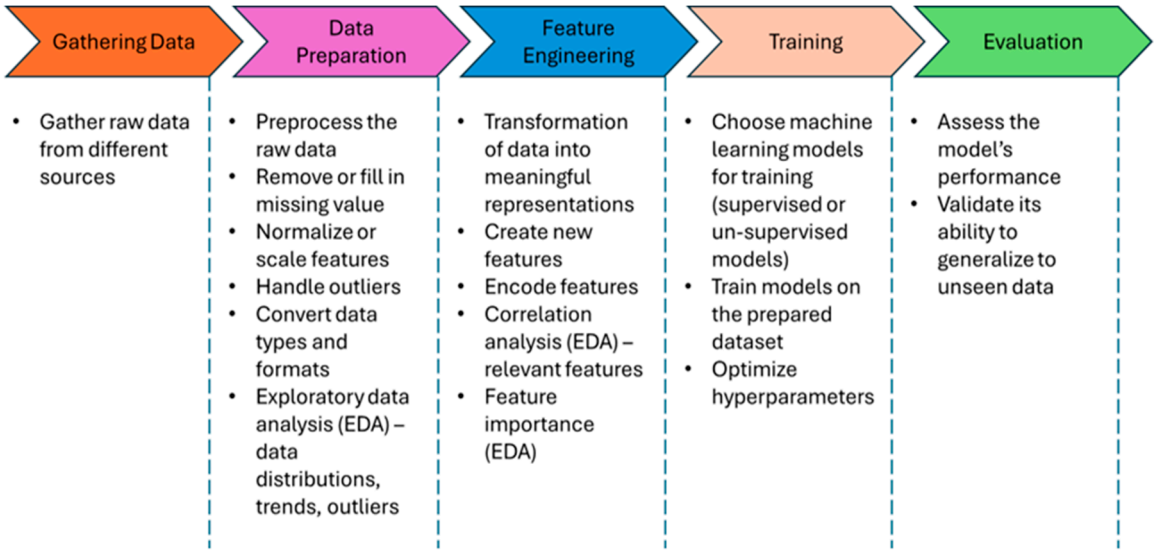


Figure 3. Overview of the machine learning pipeline.

3.1. Handling and Cleaning Data

A structured data cleaning and preparation process was applied to the aggregated dataset prior to feature engineering and model development. This step addressed missing values, managed outliers, and ensured consistency across the combined FLB and SLB datasets. The initial dataset comprised 40,689 rows of data across 20 features, where each row corresponds to a single IC cycle.

An assessment of dataset completeness identified missing values in five features, as summarised in Table 2. These missing values arise from the characteristics of the incremental capacity measurements and feature definitions rather than data processing errors.

Table 2. Missing data summary and imputation strategy.

Feature	Missing Values	% of Dataset	Reason for Missing Data	Imputation Method
min_peak_value	117	0.29%	No trough detected in the IC curve	Median
peak_trough_ratio	117	0.29%	No trough detected in the IC curve	Median
cycle_peak_shift	62	0.15%	First cycle in sequence (no previous cycle for comparison)	Median
auc_change_rate	62	0.15%	First cycle in sequence (no previous cycle for comparison)	Mean
date_range	39,273	96.5%	Dataset-specific identifier (Loughborough only)	Feature excluded

For trough-based features (min_peak_value and peak_trough_ratio), missing values occur when no troughs are detected within the analysed voltage window. As batteries degrade, the IC curve shape can change, and troughs can diminish or disappear entirely. This occurs naturally during the degradation process as the battery's internal structure changes and the characteristic features of the curve evolve. Delta features (cycle_peak_shift and auc_change_rate) are expected to produce some missing values due to their definition. These features calculate differences between consecutive cycles, so the first cycle in each test sequence inherently generates missing values since no previous cycle exists for comparison. This is an unavoidable consequence of computing temporal changes in battery behaviour.

The date_range feature represented a dataset-specific identifier used only in the Loughborough dataset for internal data tracking. This feature was not standardised across all datasets and was not relevant to the health assessment task. While it was mentioned here as part of the original Loughborough dataset, it was excluded from the analysis due to its limited applicability and high proportion of missing values.

Given the low overall proportion of missing data, affected features were retained with appropriate imputation strategies applied based on feature distributions, with median imputation for skewed features and mean imputation for centred distributions, as detailed in Table 2.

Following missing data handling, an interquartile range (IQR)-based analysis was performed to detect potential outliers, providing a robust approach suitable for non-symmetric distributions [26]. The IQR method defines outliers as values lying outside the following range:

$$\text{Outlier} < Q_1 - 1.5 \times \text{IQR} \text{ or } > Q_3 + 1.5 \times \text{IQR} \quad (1)$$

where:

- Q_1 is the first quartile (25th percentile);
- Q_3 is the third quartile (75th percentile);
- IQR is the interquartile range (Q_3–Q_1).

A two-tiered outlier handling strategy was implemented to balance the need for robust outlier removal whilst preserving sufficient data for model development. Features were categorised based on the proportion of outlier values they contained. For features with fewer than 10% outliers, all rows containing those outliers were removed. For features with 10% or more outliers, a maximum of 10% of rows were removed, and the remaining outliers were flagged using a binary outlier_flag column. This approach allowed for the influence of extreme values to be reduced while retaining potentially informative but highly variable data points. This approach is visualised in Figure 4.

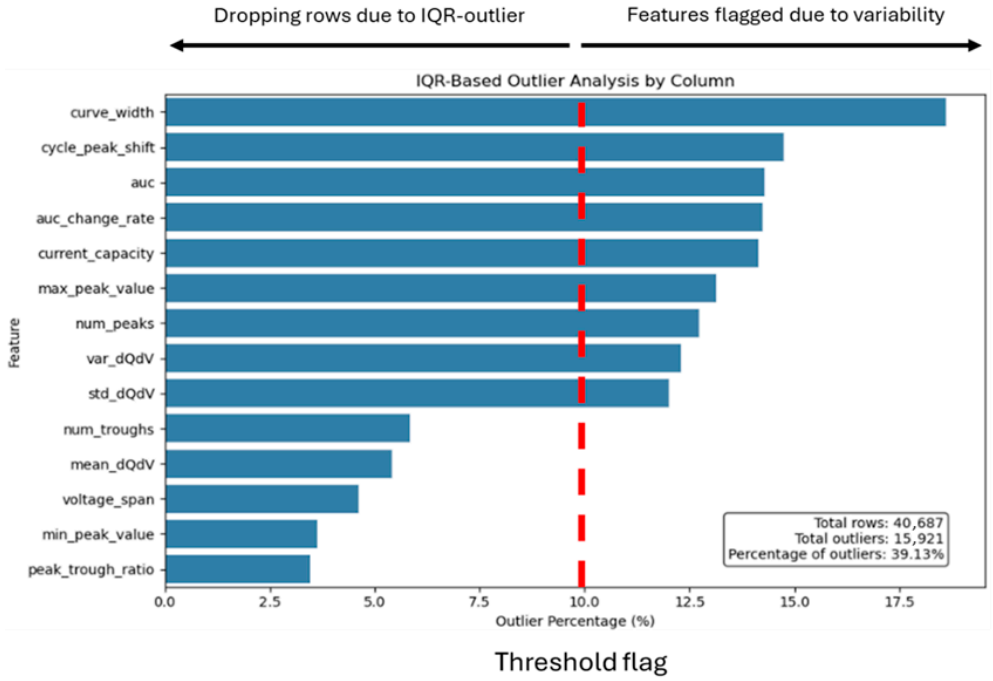


Figure 4. Visualisation of the percentage of outliers per column.

An initial global implementation of this two-tiered strategy was applied globally across the entire aggregated dataset using a 10% threshold for row removal. While effective for outlier removal, this approach disproportionately impacted smaller subsets, most notably the Oxford dataset, which was reduced to only three remaining rows.

To address this imbalance, the outlier detection strategy was refined by applying the IQR method separately to each dataset subset based on the data_id identifier. This per-dataset approach ensured that outlier detection was performed relative to the distribution of each dataset, rather than globally.

The two-tiered strategy was initially applied using a global 10% threshold under this per-dataset approach. However, this threshold remained too aggressive for certain subsets, with approximately 41% of the overall dataset removed. To achieve a better balance between data retention and outlier management, alternative thresholds were explored. A 3% threshold was selected as the optimal compromise, resulting in a 19.5% reduction in overall dataset size. The removed rows were primarily from the Nature subset, which contributed to the majority of the data, while the Oxford and Loughborough datasets experienced minimal reductions. The optimal IQR threshold is dependent on the characteristics of the datasets being analysed, and alternative datasets or operating conditions would require re-evaluation of this parameter. This data-driven approach facilitates generalisation across heterogeneous datasets by managing variability in feature behaviour without relying on assumptions about fault conditions or prior usage history.

The final distribution of rows by dataset, after applying the per-dataset IQR approach with a 3% threshold, is summarised in Table 3. Following this process resulted in 19.5% of rows of data being dropped and 31.6% of the rows being flagged using the outlier_flag column.

Table 3. Row counts per individual dataset before and after IQR outlier handling—global 10% threshold versus 3% per data_id threshold.

Individual Dataset	Before Outlier Analysis	After IQR (Global)	After IQR (Per Data_Id)
Nature	38,754	35,196 (9.18% dropped)	30,975 (22.07% dropped)
Oxford	519	3 (99.42% dropped)	473 (8.86% dropped)
Loughborough	1416	1416 (0.00% dropped)	1288 (9.04% dropped)

3.2. Feature Engineering

Feature engineering was carried out to introduce new, informative variables derived from the original features. This process aimed to help the ML models capture temporal trends, non-linear relationships, and interactions that may not be immediately evident from the raw values alone. Four main strategies were applied: SOH banding, delta features, interaction terms, and log transformations. Taken together, these descriptors provide complementary signals: deltas emphasise progression across cycles, interactions capture effects that arise only in combination, and logs stabilise scale and compress extremes while preserving signs. By combining these descriptor families, the representation becomes richer and more stable across datasets, supporting joint analysis in a shared IC feature space.

SOH Banding

Rather than estimating SOH as a continuous percentage value, a classification approach was adopted by discretising SOH into distinct health bands. This reflects practical considerations for second-life battery applications, where groups of cells or modules with similar SOH are often treated collectively to simplify system design and minimise performance limitations. Previous research by Yang et al. [27] demonstrated that the overall performance of repurposed battery systems is frequently constrained by the lowest-performing cell or module, with active balancing unable to fully mitigate this effect in some cases.

In this study, the bands are used to interpret the unsupervised results with no models being tuned with these labels. The number of bands is application-dependent and will influence the difficulty and error of any later prediction task, which will be reported in the follow-on supervised study using per-class metrics and confusion matrices. Grouping batteries into health bands provides a practical and interpretable framework for assessing and managing battery condition in second-life applications.

Prior to applying discretisation, a summary inspection of the SOH values was conducted. Using summary statistics, an unexpectedly high maximum SOH value was identified. Since SOH is expressed as a percentage of nominal capacity, all values were expected to fall within a realistic range. To allow for minor fluctuations due to measurement variation, a maximum threshold of 100.5% was defined. Two entries exceeded this limit, with values of 139.91% and 262.19%. These entries were then removed to maintain consistency within the dataset. The resulting SOH distribution is shown in Figure 5.

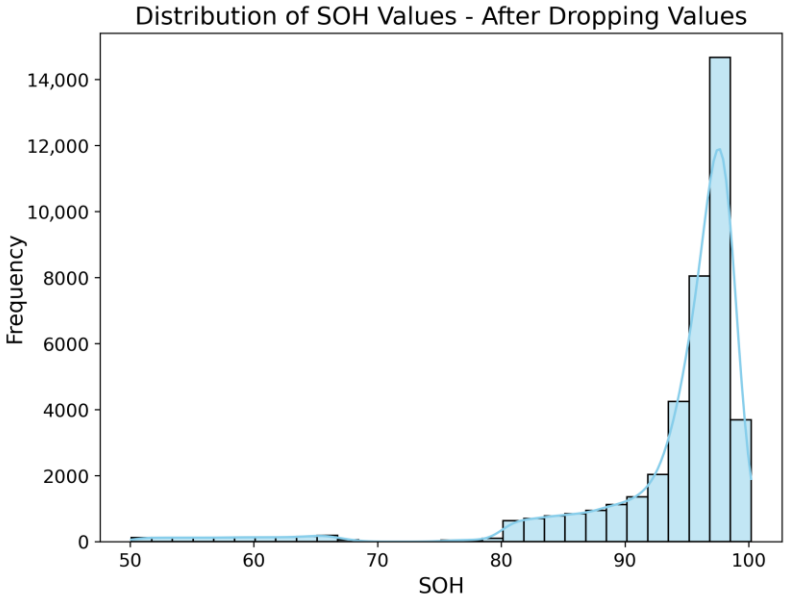


Figure 5. Visualisation of the distribution of the state of health values in the aggregated dataset.

An initial set of SOH band thresholds was defined to partition the continuous SOH values into five discrete categories:

- 0–65%: Very Bad;
- 65–75%: Bad;
- 75–85%: Okay;
- 85–95%: Good;
- 95–100.5%: Very Good.

While these bands aligned with general interpretations of battery health, the resulting class distribution was highly imbalanced, as shown in Table 4. As summarised in this table, over 24,000 samples were allocated to the Very Good band, while the Bad band contained only 264 samples.

Table 4. Summary of the state of health band distributions.

SOH Band (Number)	SOH Band (Label)	SOH Range (%)	Number of Samples
0	Very Bad	0–65	1155
1	Bad	65–75	264
2	Okay	75–85	1777
3	Good	85–95	8981
4	Very Good	95–100.5	24,438

To address this imbalance, alternative SOH band thresholds were explored. Several boundary configurations were trialled, and a more balanced set of thresholds was selected, as shown in Table 5 and visualised in Figure 6.

Table 5. Final state of health distributions following threshold adjustment.

SOH Band (Number)	SOH Band (Label)	SOH Range (%)	Number of Samples
0	Very Bad	0–60	700
1	Bad	60–75	590
2	Okay	75–90	2171
3	Good	90–95	5932
4	Very Good	95–100.5	23,343

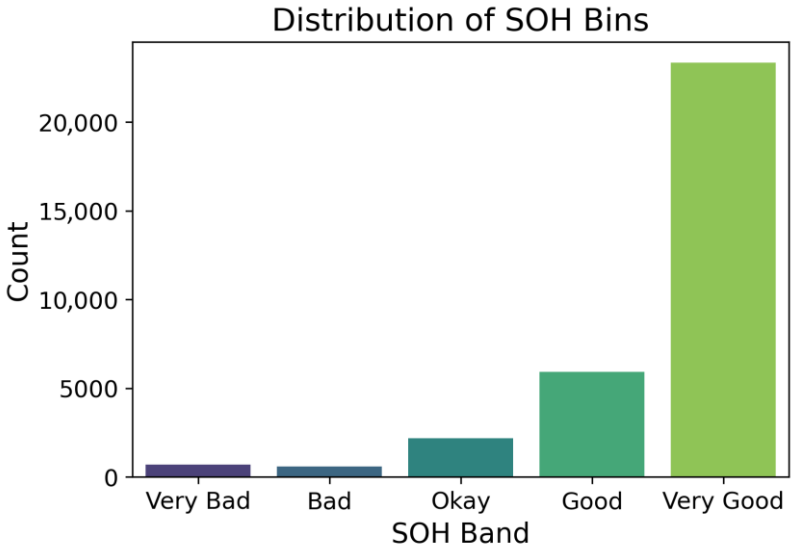


Figure 6. Visualisation of the distribution of the state of health bands following threshold adjustment.

For the second-life Loughborough dataset, per-cycle SOH values were estimated by linear interpolation due to the absence of continuous capacity measurements. To assess the potential uncertainty introduced by this assumption in the context of SOH band assignment, a sensitivity analysis was conducted in which the interpolated SOH values were perturbed by $\pm 3\%$ prior to discretisation. This perturbation was selected to represent a plausible uncertainty range rather than a precise error estimate. The resulting SOH band assignments were then compared with the original labels.

The analysis showed that approximately 83% of cycles retained the same SOH band under both positive and negative perturbations, while the remaining cycles transitioned exclusively between adjacent bands. No non-adjacent band transitions were observed. Since SOH banding is used to support comparative analysis of unsupervised structure in this study and to provide a consistent target representation for subsequent supervised modelling, these results indicate that uncertainty associated with SOH interpolation does not materially affect the interpretability or downstream use of the proposed framework.

Delta Features

To capture how key battery characteristics evolve throughout their lifecycle, delta features were computed using the absolute difference between successive cycles within each dataset subset. Grouping was conducted based on the data_id identifier to ensure battery-specific continuity and highlight trends such as linear or abrupt degradation.

The .diff() function was applied to each selected feature, and the first cycle in each group was filled with a value of zero [28]. Features selected for delta computation included:

- auc, auc_change_rate, and curve_width: Capture the shape and dynamics of the incremental capacity curve, which typically evolve with battery ageing.
- max_peak_value and min_peak_value: Can reflect the amplitude of the IC peak; changes in these values can indicate fading reactions or kinetic limitations.
- current_capacity: Direct measure of usable energy, and its rate of change can be a direct indicator of remaining useful life.
- cycle_peak_shift: Tracks movements of characteristic IC peaks, potentially indicating phase changes or resistance buildup.

Interaction Terms

Interaction terms were constructed to model relationships between features whose combined effects may differ from their individual contributions. These derived features represent combinations of existing variables, such as the product or ratio of two features, which

may reveal patterns not detectable when the features are considered independently [29–31]. Such interactions often improve the performance of machine learning models, particularly tree-based algorithms and neural networks, which benefit from richer, non-linear input representations.

The selected interaction terms were informed by their physical relevance to battery degradation and their interpretability in the context of IC analysis.

- *auc_voltage_interaction* (auc × voltage_span): Represents the electrochemical activity density across the voltage range. A declining value may indicate loss of overall electrochemical activity density due to loss of active material or lithium inventory.
- *dQdV_curve_interaction* (mean_dQdV × curve_width): Captures both the average height and spread of the IC curve. Lower values can indicate a loss in capacity or a narrowing of the voltage range over which reactions occur, which is indicative of ageing.
- *capacity_curve_ratio* (current_capacity ÷ curve_width): Relates deliverable capacity to the span of the voltage window. A reduction in this ratio may suggest rising internal resistance or declining electrode kinetic limiting capacity output per unit voltage.
- *peak_value_interaction* (max_peak_value × min_peak_value): Encodes the strength of the most prominent anodic and cathodic reactions. As both values tend to decrease degradation, their product is sensitive to fading reaction intensity and may help identify moderate or early-stage capacity loss.

These interactions were designed to expose complex patterns in the data and improve the model's ability to distinguish between batteries with different SOH bands.

Log Transformations

To mitigate the impact of skewed distributions and extreme values, natural logarithmic transformations were applied to selected features. This approach is particularly effective for right-skewed features, as it compresses the range of values and shifts extreme points closer to the mean, thereby improving the performance of ML models that are sensitive to scale and distribution [32–34]. Logarithmic transformation also enhances the learning capability of distance-based and gradient-based models by promoting more stable variance across inputs.

Transformations were implemented using the log1p() function, which computes log(1 + x). This function is used as it handles zero and near-zero values safely, avoiding the undefined behaviour of log(0) and reducing numerical instability. The features selected for transformation, and their justifications, are as follows:

- auc: The auc under the IC curve can span several orders of magnitude across cycles due to ageing. Log transformation helps normalise the feature's distribution, facilitating better learning of degradation-related trends.
- max_peak_value: Maximum values in the IC curve can vary significantly between cells due to differences in chemistry, capacity, and ageing. Applying a logarithmic scale reduces the impact of extreme peak values, stabilising the variance.
- curve_width: This feature captures the spread of the IC curve over the voltage axis. This feature can exhibit a wide range due to varying degradation patterns, and log transformations aid in compressing this variability for model stability.
- min_peak_value: Like its positive counterpart, minimum values can be widely distributed due to heterogeneous chemistries and degradation pathways. A log transformation adjusts the scale of negative peaks while maintaining proportionality.

In rare cases, log1p transformations resulted in NaN values when applied to zeroes or negative entries. These were addressed by replacing such values with the logarithm of

the smallest positive value available in that feature. This imputation strategy ensured data continuity without distorting the underlying distribution.

The engineered features derived throughout this process are summarised in Table 6 below, including the feature name and a brief description of its purpose.

Table 6. Summary of engineered features.

Feature Name	Description
SOH_bands	Categorical target variable based on SOH bins for SOH classification
auc_Delta	Per-cycle absolute change in auc, grouped by data_id
auc_change_rate_Delta	Per-cycle absolute change in auc_change_rate, grouped by data_id
curve_width_Delta	Per-cycle absolute change in curve_width, grouped by data_id
max_peak_value_Delta	Per-cycle absolute change in max_peak_value, grouped by data_id
min_peak_value_Delta	Per-cycle absolute change in min_peak_value, grouped by data_id
current_capacity_Delta	Per-cycle absolute change in current_capacity, grouped by data_id
cycle_peak_shift_Delta	Per-cycle absolute change in cycle_peak_shift, grouped by data_id
auc_voltage_interaction	Product of auc and voltage span
dQdV_curve_interaction	Product of mean_dQdV and curve_width
capacity_curve_ratio	Ratio of current_capacity to curve width, with a small constant added to prevent division by 0
peak_value_interaction	Product of max_peak_value and min_peak_value
log_auc	Natural logarithm of auc after clipping and NaN handling
log_max_peak_value	Natural logarithm of max_peak_value after clipping
log_curve_width	Natural logarithm of curve_width after clipping
log_min_peak_value	Natural logarithm of the absolute value of min_peak_value

The feature engineering process added 16 new features to the dataset, increasing the total number of features from 20 to 36 and representing an 80% increase in dimensionality. Broadening the feature set in this way aims to improve the model's capacity to capture complex patterns of degradation and electrochemical behaviour. This supports more effective SOH band classification by providing a richer feature space and enhancing the model's ability to generalise across diverse battery chemistries, formats, and usage histories.

3.3. Feature Encoding and Scaling

Before applying ML algorithms, it is necessary to ensure that all features are represented in compatible numerical formats and lie within consistent numerical ranges. Many models, particularly those relying on gradient descent or distance-based metrics, are sensitive to unscaled or improperly encoded data. This section describes the preprocessing steps taken to encode categorical variables and scale numerical features, ensuring model compatibility and improved training stability.

All features were first separated into numeric and non-numeric types. Numeric columns (including integer and float values) were retained for scaling, while non-numeric features were transformed using label encoding. A visualisation of the data types can be seen in Figure 7 below.

Label encoding was applied to the following object and categorical features:

- data_id;
- cell_chemistry;
- SOH_bands.

These features were converted to integer labels using the LabelEncoder from scikit-learn, ensuring all inputs were fully numerical and compatible with models that do not support object-type data.

Once encoding was completed, the next step was to standardise the numeric features. Logarithmic transformations were first applied to reduce skewness and limit the influence of extreme values. The following features were transformed:

- auc;
- peak_value_interaction;
- capacity_curve_ratio;
- auc_voltage_interaction.

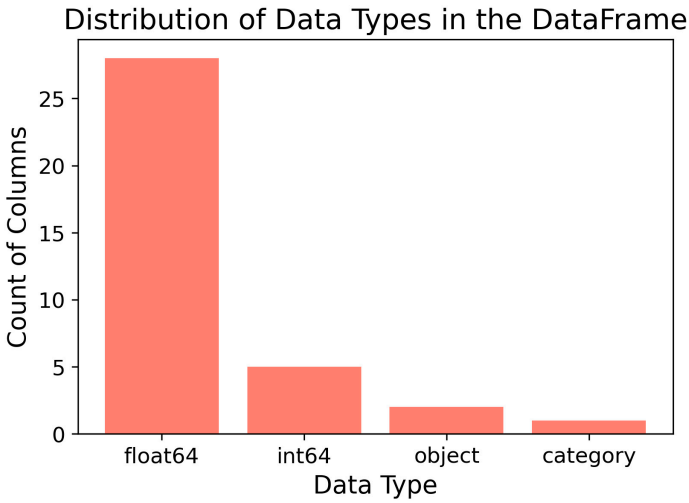


Figure 7. Visualisation of the distribution of data types in the aggregated dataset.

These features exhibited highly skewed distributions, often spanning several orders of magnitude due to the nature of battery degradation and the compounding effects of interaction terms. For example, the ratio of the absolute 99th percentile to the absolute 1st percentile fell from approximately 213 before transformation to approximately 45 after the log mapping. Without transformation, such skewness could negatively impact model convergence and performance, particularly for algorithms sensitive to feature distribution or scale.

The transformation used was log1p(abs(x)) $\times$ sign(x), which preserves the original sign of the data while compressing large absolute values. This approach ensures numerical stability and preserves directional trends, which is important for features such as capacity_curve_ratio, where small negative values may occur due to noise in curve width estimation.

Following the transformation, a per-group scaling approach was adopted. Each data_id subset was scaled independently using StandardScaler() [35]. This was performed after outlier handling and feature engineering to avoid distorting the statistical properties of the raw data. Given that the combined dataset included sources with different nominal capacities and operating ranges, global scaling could have caused the larger-magnitude Loughborough dataset to dominate, potentially masking patterns in the smaller-scale subsets.

Although scaling prior to outlier analysis might have reduced the number of detected outliers, particularly in the Oxford dataset, this was avoided. Scaling before filtering can suppress the magnitude of outliers and alter their physical meaning. Detecting outliers in the raw feature space ensures thresholds reflect the true underlying distribution. By scaling each group independently after filtering, features were normalised relative to their local context, preserving intra-dataset structure while maintaining consistency across the aggregated dataset.

Only relevant numeric features were scaled. Identifiers such as data_id, cell_chemistry, and the target variable SOH_bands were excluded from this process. The final scaled dataset was then reassembled and ordered for downstream processing.

3.4. Feature Reduction

Reducing the number of input features helps manage model complexity, mitigate overfitting, and improve computational efficiency. As part of the preprocessing pipeline, feature reduction was applied to remove redundant variables and minimise multicollinearity.

A Pearson correlation matrix was first computed to assess pairwise linear relationships between 33 numerical features; variables related to capacity, SOH, and metadata were excluded from this screening. The Pearson correlation matrix measures correlation on a scale from -1 (perfect negative correlation) to +1 (perfect positive correlation). Features with an absolute correlation coefficient above 0.90 were considered highly collinear, and one feature from each such pair was removed to reduce redundancy. This threshold was selected to identify strong linear dependence and remove near-duplicate information while retaining moderately correlated features that may still encode complementary physical characteristics. Similar high-correlation thresholds have been widely adopted in applied machine learning workflows as a conservative approach to mitigating multicollinearity without overly aggressive feature removal [36,37]. The correlation matrix used to guide this process is shown in Figure 8.

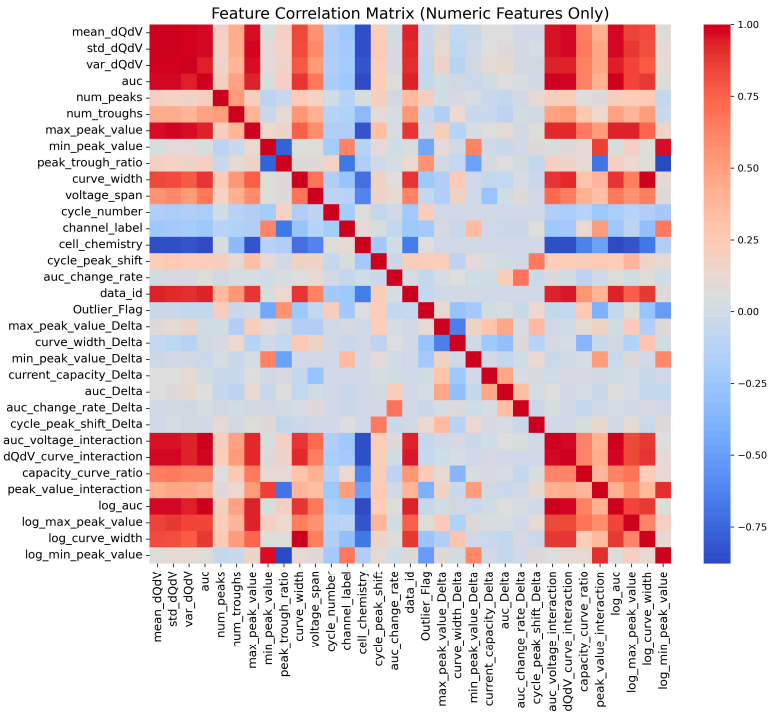


Figure 8. Pearson correlation matrix for feature reduction.

As expected, several features derived from similar base metrics exhibited strong mutual correlation. For example, std_dQdV, var_dQdV, and auc were closely related due to their dependence on the IC curve shape. Similarly, engineered features such as log_auc and auc_voltage_interaction showed high correlation with their untransformed or source components.

Out of the initial 33 features, eight were removed based on pairwise correlation analysis:

- std_dQdV;

- var_dQdV;
- log_auc;
- auc;
- log_curve_width;
- log_min_peak_value;
- auc_voltage_interaction;
- dQdV_curve_interaction.

Following this reduction step, the dataset retained 25 features, including selected engineered variables and the SOH_bands target for unsupervised analysis. The reduced dataset contained 32,736 samples across 25 features, representing a 19.5% reduction in rows and a 30% reduction in features compared to the engineered dataset shape of 40,689 samples and 36 features. This reduced dataset was used as the basis for clustering in Section 4.

4. Unsupervised Structure Exploration

Unsupervised learning refers to a class of ML techniques that operate on unlabelled data. These methods are particularly useful for uncovering hidden patterns or structural relationships in datasets where predefined output categories are unavailable or incomplete. Among these, clustering is one of the most widely used approaches, grouping data points based on similarity across multiple features. This is especially valuable in complex datasets, such as the one used in this study, where relationships between variables may be non-linear or difficult to interpret manually.

In this work, clustering was used to investigate whether any meaningful groupings could be identified within the engineered feature space, independent of the predefined SOH_bands. Two algorithms were selected for exploration:

- K-means;
- Hierarchical Density-Based Spatial Clustering of Applications with Noise (HDBSCAN).

While clustering does not yield direct predictions as supervised learning does, the resulting cluster labels can offer meaningful insights into how samples are organised in the IC feature space. These labels can also be used as meta-features, which serve as auxiliary inputs that augment supervised models by exposing group-level trends not captured by raw features alone. This hybrid strategy has the potential to improve downstream classification performance by leveraging both global structure and local relationships within the data.

4.1. Clustering Approaches

Clustering was used to explore whether the engineered feature space exhibited any inherent structure, independent of the predefined SOH_bands. Two clustering algorithms were tested: centroid-based clustering via K-means and density-based clustering via HDBSCAN. While DBSCAN was initially considered, it failed to identify well-separated clusters and was excluded from further analysis.

K-Means Clustering

K-means is a centroid-based clustering algorithm that partitions data by assigning each sample to the nearest cluster centre (centroid) based on feature similarity [38]. The algorithm iteratively adjusts the centroids to minimise the within-cluster variance.

A key limitation of K-means is the requirement to predefine the number of clusters (k). Choosing too few clusters can oversimplify the data, whilst too many clusters can fragment meaningful structure [39]. In addition, K-means assumes clusters are spherical in nature, roughly equal in size, and equally dense, which may not always reflect real-world data distributions. The algorithm is also sensitive to outliers, which can distort the centroid positions and reduce clustering quality [40,41].

To identify a suitable value of k, values from 2 to 10 were tested. Several features were excluded from clustering to preserve the unsupervised nature of the task and avoid bias from variables closely tied to battery health or capacity. These included:

- SOH_bands: This feature represents the classification target for downstream ML models.
- SOH, current_capacity, current_capacity_Delta, and capacity_curve_ratio: These features are directly correlated or derived from capacity metrics and thus would indirectly reflect SOH information.
- data_id, cycle_number, and channel_label: These are sequencing or metadata variables that, whilst not inherently indicative of battery performance, can indirectly encode information about degradation.

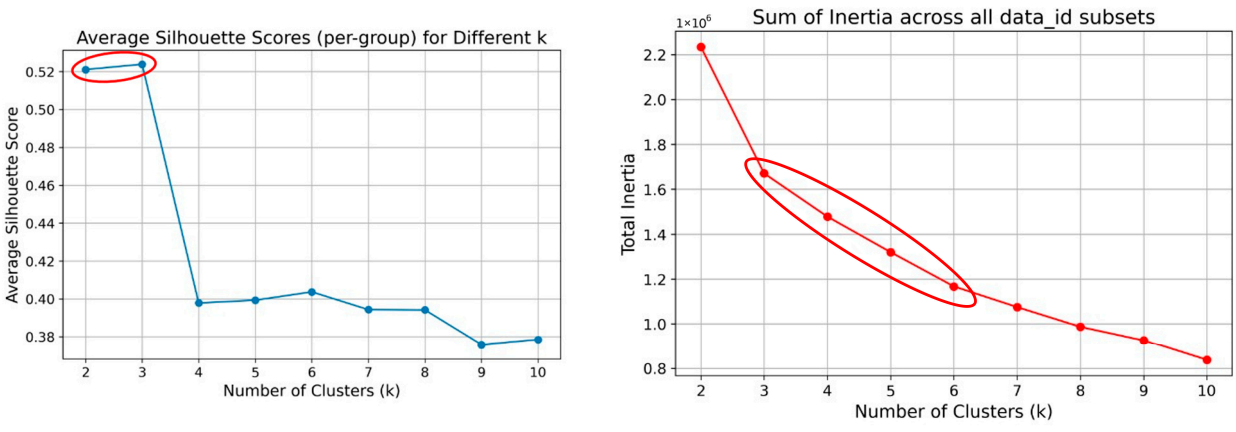
Following the exclusion of the above features, the final list of features used for clustering was:

- Base descriptors: mean_dQdV, num_peaks, num_troughs, max_peak_value, min_peak_value, peak_trough_ratio, curve_width, and voltage_span.
- Dynamics and deltas: cycle_peak_shift, auc_change_rate, curve_width_Delta, min_peak_value_Delta, auc_Delta, max_peak_value_Delta, cycle_peak_shift_Delta, and auc_change_rate_Delta.
- Interactions: peak_value_interaction.
- Log transforms: log_max_peak_value.
- Flag: Outlier_Flag.

Clustering was applied separately to each data_id subset to account for different scales and operating conditions. For each candidate k, the following metrics were computed and averaged across subsets:

- Silhouette score: To evaluate cluster cohesion and separation.
- Inertia: The within-cluster sum of squares, where lower values indicate more compact clusters [42,43].

Visual inspections of silhouette scores and inertia values are shown in Figures 9a and 9b, respectively. These values suggest that values of k between 3 and 6 provide a reasonable balance between interpretability and structural resolution. The optimal k is typically identified by selecting the value that gives the highest silhouette score, indicating well-separated clusters. For inertia, the "elbow" point marks where increasing k stops meaningfully improving the compactness of the clusters.



(**a**) (**b**)

Figure 9. (**a**) Visualisation of the average silhouette scores per k across all subsets. (**b**) Visualisation of the summed inertia (elbow method) per k across all subsets.

HDBSCAN Clustering

To evaluate which clustering algorithm provided more meaningful structure in the engineered feature space, both K-means and DBSCAN (Density-Based Spatial Clustering of Applications with Noise) were initially tested. However, DBCSAN consistently failed to separate data into distinct clusters, with most samples being assigned to a dominant cluster and only a small number labelled as noise or outliers. This lack of sensitivity to the underlying structure led to the exclusion of DBSCAN from further analysis.

As a more robust alternative, HDBSCAN was explored. HDBSCAN is an extension of DBSCAN that improves cluster detection by building a hierarchy of density-based clusters and selecting the most stable ones based on persistence. Like DBSCAN, it groups data points that are closely packed in space and identifies outliers as noise but does not require a global density threshold. Instead, it adapts to local density variations, making it more effective in high-dimensional or heterogeneous datasets, such as those derived from multiple battery chemistries and operational conditions [44].

HDBSCAN operates on similar principles to DBSCAN and inherits its key advantage, which is the ability to discover clusters of arbitrary shape and varying density, rather than relying on predefined spherical assumptions [45]. This flexibility makes it well-suited to datasets with complex, non-linear structure. In both algorithms, data points are classified into:

- Core points: Points with a sufficient nearby neighbour to form a dense region.
- Border points: Points in the neighbourhood of a core point that do not have enough neighbours themselves.
- Noise points (outliers): Points that do not belong to any cluster due to insufficient local density.

Unlike DBSCAN, HDBSCAN does not require the number of clusters to be specified in advance. Instead, it uses a single primary parameter, min_cluster_size, which determines the smallest grouping considered a cluster. This flexibility enables HDBSCAN to recover both large and small clusters, depending on the data's local density structure.

To tune HDBCSAN, a sweep of min_cluster_size values was conducted over the range {3, 5, 10, 20, 30, 40, 50, 75}. Clustering was performed using the default Euclidean distance metric applied to the reduced dataset. The distance metric was not varied further, and optimisation focused on the min_cluster_size parameter due to its direct influence on cluster density and noise identification. For each configuration, clustering was applied to the reduced dataset, and the number of clusters, proportion of noise points, and label assignments were recorded. Smaller min_cluster_size values produced many fine-grained clusters but also led to high noise (e.g., 60.11% for size = 3). Larger values yielded broader, more stable clusters with lower noise levels.

The best-performing configuration was selected based on a trade-off between cluster count and noise proportion. A setting of min_cluster_size = 40 produced seven clusters and 14.44% noise, offering a balance between resolution and robustness. Table 7 below shows all the results of HDBSCAN tuning across different min_cluster_size values.

Table 7. HDBSCAN tuning results across min_cluster_size values.

Min_Cluster_Size	Clusters Found	Noise Points	Noise Ratio (%)
3	937	19,677	60.11
5	152	13,254	34.38
10	49	11,442	34.95
20	20	11,674	35.66
30	9	4443	13.57

Table 7. *Cont.*

Min_Cluster_Size	Clusters Found	Noise Points	Noise Ratio (%)
40	7	4731	14.45
50	7	4980	15.21
75	6	5077	15.51

4.2. Dimensionality Reduction for Visualisation

To aid interpretation of the clustering results, dimensionality reduction techniques were applied to project the high-dimensional feature space into lower dimensions suitable for visual inspection. These projections helped assess how well the clusters identified by K-means and HDBSCAN aligned with the underlying structure in the dataset. Two techniques were employed for this purpose:

1. Principal component analysis (PCA) for linear dimensionality reduction.
2. t-distributed stochastic neighbour embedding (t-SNE) for non-linear projection.

PCA was used to create both 2D and 3D projections of the scaled feature space. This technique projects high-dimensional data into a lower-dimensional (2D or 3D) space while preserving local structure. It is particularly well-suited for visualising the clustering results with non-linear patterns. However, due to scale differences between datasets and the complex nature of the engineered features, PCA captured limited separability between clusters.

To better visualise local groupings and non-linear structure, t-SNE was applied to the same scaled dataset. t-SNE preserves local neighbourhoods by mapping similar points close together in a 2D space, making it effective for assessing structure in complex datasets [46]. Separate 2D t-SNE visualisations were generated for both K-means and HDBSCAN, with each clustering result applied independently to inspect how the data was grouped under each method.

t-SNE was computed on the scaled, correlation-reduced feature space using scikit learn's TSNE implementation with the following settings: n_components = 2 (2 for 2D plot, 3 for 3D plot), perplexity = 30, random_state = 42, PCA initialisation, Euclidean metric, early exaggeration = 12, n_iter = 1000, and the library default learning rate. The "Barnes Hut" approximation was used for 2D embeddings, and the "exact" method was used for 3D embeddings. Embeddings were used for visulisation only, and clustering was performed in the original feature space with no labels used during t-SNE fitting.

5. Results

This results section presents the clustering outcomes and visualisations derived from the engineered feature space. Both K-means and HDBSCAN clustering algorithms were applied to assess whether unsupervised structure exists within the dataset.

5.1. Clustering Outcomes and Structural Trends

K-Means Clustering

To evaluate structural patterns in the feature space, K-means clustering was applied with an optimal number of clusters selected as k = 3 based on the highest silhouette score and inertia analyses. Table 8 shows the distribution of samples across the three resulting clusters. Cluster 2 was the dominant grouping, containing 23,469 of the 32,736 total samples.

Table 8. K-means cluster distribution by dataset.

Dataset	Cluster 0	Cluster 1	Cluster 2	Total
Nature	873	7242	22,860	30,975
Oxford	92	183	198	473
Loughborough	590	287	411	1288

To better understand how cluster assignments varied across datasets, this table provided and showed a breakdown by data_id. This revealed consistent clustering behaviour across sources. For the largest dataset (Nature), cluster 2 accounted for 73.8% of samples. Cluster 2 was also dominant within the Oxford and Loughborough subsets, capturing 41.9% and 31.9% of their samples, respectively. This suggests that despite differences in format, chemistry, and operational history, K-means identified a shared structure across all three sources.

A further breakdown was conducted by SOH_bands to explore potential alignment between clustering outcomes and battery health classification. As shown in Table 9, cluster 2 was strongly associated with higher SOH bands. In particular, it contained over 72% of samples in Band 4 (Very Good) and a majority in Band 3 (Good). In contrast, the lower SOH bands (Very Bad to Okay) were distributed more evenly across all three clusters. This indicates that cluster 2 may reflect cells with stable or minimally degraded behaviour, while clusters 0 and 1 capture more deteriorated profiles.

Table 9. K-means cluster distribution by SOH band.

SOH Band	Cluster 0	Cluster 1	Cluster 2	Total
0 (Very Bad)	291	142	267	700
1 (Bad)	300	145	145	590
2 (Okay)	580	81	1510	2171
3 (Good)	258	1034	4640	5932
4 (Very Good)	126	6310	16,907	23,343

These results suggest that K-means clustering captured a structure that partially aligns with battery condition. Although clusters were not sharply separated by SOH, the grouping behaviour highlights trends that may be leveraged in downstream classification tasks.

HDBSCAN Clustering

Table 10 presents the HDBSCAN cluster distribution across the three dataset sources. A total of seven cluster labels were assigned (including label −1 for noise), with the majority of points in the Nature dataset assigned to cluster 5 (19,438 samples) and a smaller portion to cluster 1 (6594 samples). In contrast, the Loughborough dataset was predominantly assigned to cluster 0 (1171 samples), while the Oxford dataset produced a broader spread of assignments, primarily into cluster 3 (292 samples) and cluster 2 (81 samples).

Table 10. HDBSCAN cluster distribution by dataset.

Dataset	Cluster −1	Cluster 0	Cluster 1	Cluster 2	Cluster 3	Cluster 4	Cluster 5	Total
Nature	4860	0	6594	0	0	83	19,438	30,975
Oxford	100	0	0	81	292	0	0	473
Loughborough	117	1171	0	0	0	0	0	1288

The presence of noise (label −1) was evident across all three datasets, but it was particularly high in the Nature dataset (4860 samples) and the Loughborough dataset

(117 samples). These results reflect HDBSCAN's sensitivity to local density variation, with cluster membership determined by the density persistence rather than fixed thresholds. Unlike K-means, which produced uniform cluster counts across datasets, HDBSCAN formed clusters with varying membership and granularity, adapting to the heterogeneous feature space of each data subset.

Table 11 further breaks down the HDBSCAN cluster labels according to the SOH bands classification. Cluster 5 was the largest and most widely distributed across SOH bands, particularly dominating the higher health categories: Band 3 (3829 samples) and Band 4 (5687 samples), with smaller contributions in Band 3. In contrast, clusters 2, 3, and 4 captured small but distinct groupings of intermediate SOH states, such as Band 2. Noise points were dispersed across all SOH categories but were most concentrated in SOH Bands 2 to 4, with 2710 points flagged as noise in SOH Band 4 alone.

Table 11. HDBSCAN cluster distribution by SOH band.

SOH Band	Cluster −1	Cluster 0	Cluster 1	Cluster 2	Cluster 3	Cluster 4	Cluster 5	Total
0 (Very Bad)	73	627	0	0	0	0	0	700
1 (Bad)	44	544	0	1	1	0	0	590
2 (Okay)	1113	0	25	80	206	7	740	2171
3 (Good)	1137	0	882	0	84	0	3829	5932
4 (Very Good)	2710	0	5687	0	1	76	14,869	23,343

Notably, the cluster-to-SOH mapping in Table 11 shows a high proportion of zero-count entries, particularly in the lower SOH bands (0 and 1). For example, cluster 5 (which dominates SOH Band 4) contains no samples from SOH Bands 0 or 1. Likewise, clusters 1, 2, 3, and 4 are entirely absent from SOH Band 0. This pattern indicates that HDBSCAN effectively distinguishes low-health batteries from the rest, with minimal overlap. The clean separation observed across several clusters and SOH classes suggests that the clustering captures meaningful structural variation, reinforcing the potential value of the cluster labels as meta-features for supervised modelling.

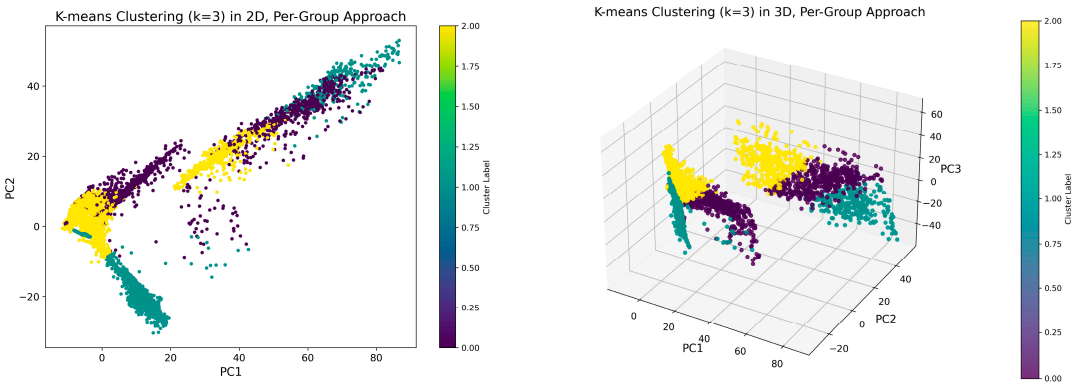
Compared to K-means, HDBSCAN clustering revealed greater nuance in structural variation, identifying both major and minor groupings while explicitly separating ambiguous points as noise. These properties reinforce its suitability as an unsupervised tool for structure discovery, offering complementary insights that can support downstream modelling.

5.2. Visualisation of Feature Space

K-Means Clustering

To assess the structure revealed by K-means clustering, the cluster labels (k = 3) were projected into reduced dimensions using PCA and t-SNE. These visualisations help evaluate the compactness, separation, and overlap between clusters in the engineered feature space.

Figure 10 presents the PCA projections in 2D and 3D. Cluster 2 dominates the central region in both views, with clusters 0 and 1 positioned at the fringes of the space. However, some boundaries overlap between clusters and are visible, especially near transitional regions in the PC axes. The PCA structure reflects a continuous gradient rather than strict cluster boundaries, which is consistent with the progressive nature of battery degradation. In this projection, two dominant lobes are apparent as PCA emphasises the strongest linear variance. With k = 3, K-means partitions this structure into two dominant groups plus a transitional group lying between them. This separation is less apparent in two dimensions and becomes clearer when additional components are considered.



(**a**) (**b**)

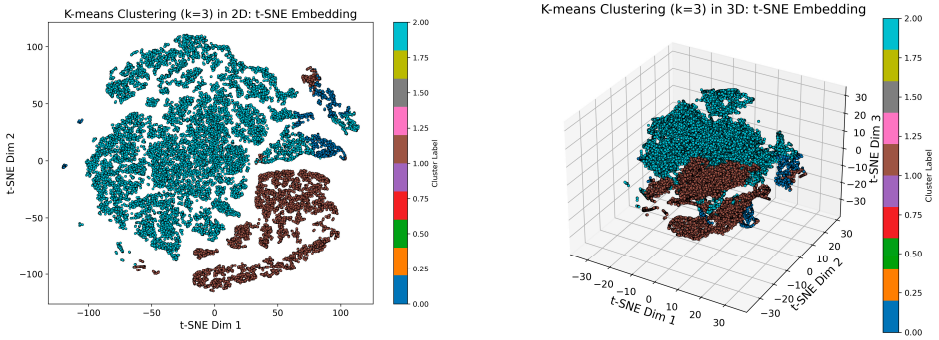
Figure 10. (**a**) K-means clustering—2D PCA visualisation. (**b**) K-means clustering—3D PCA visualisation.

The t-SNE projections in Figure 11 reveal a different structure. In the 2D t-SNE plot in Figure 11a, cluster 2 appears to form a dense and cohesive group in the central area of the embedding, while clusters 0 and 1 occupy more scattered and peripheral zones. The compact island at the top right contains a mixture of labels from all three K-means clusters. This arises because t-SNE preserves local neighbourhoods but not global geometry, so regions that are separated along directions now shown in two dimensions can overlay in the 2D map. The island is, therefore, interpreted as a dense transitional zone across adjacent degradation states, consistent with overlap between neighbouring SOH bands. However, in the 3D t-SNE visualisation in Figure 11b, the spatial distinctiveness between clusters 0 and 1 is less apparent, and significant overlap is visible across all three clusters. This suggests that the IC feature space contains structure, but boundaries between groups are soft rather than sharply defined.

These trends mirror earlier findings in Tables 8 and 9, where cluster 2 corresponds to higher SOH bands and contains most of the Nature dataset samples, while clusters 0 and 1 are more representative of degraded or transitional battery states. Increasing k beyond 3 resulted in further fragmentation of these groups without improving separability, supporting the use of three broad, overlapping clusters as a practical approximation.

HDBSCAN Clustering

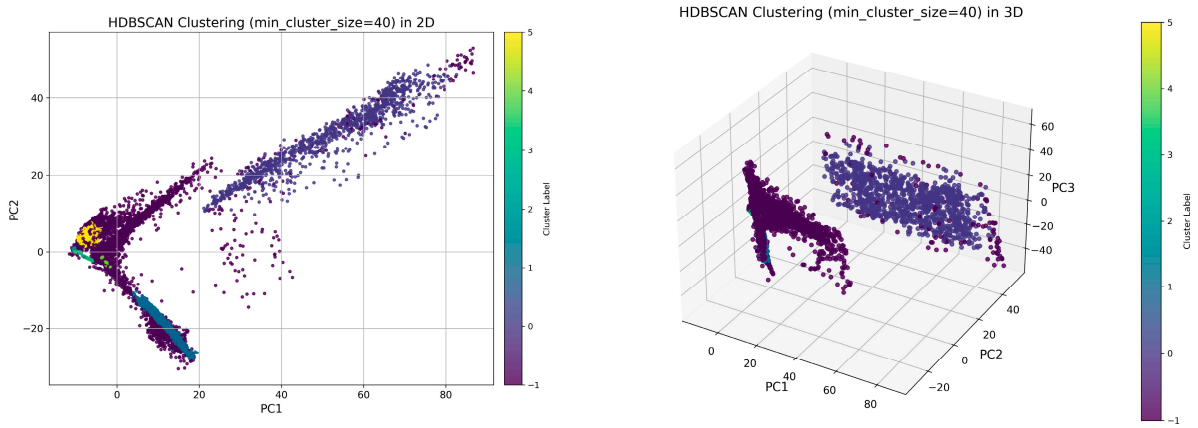
Visualisation of the HDBSCAN clustering results highlights several notable trends in the structure of the feature space. Compared to K-means, HDBSCAN generated a greater number of discrete clusters (six in total) while also identifying a substantial number of samples as noise (cluster −1). The visualisations below illustrate how these clusters are distributed in reduced dimensions.



(**a**) (**b**)

Figure 11. 2D (**a**) K-means clustering—2D t-SNE visualisation. (**b**) K-means clustering—3D t-SNE visualisation.

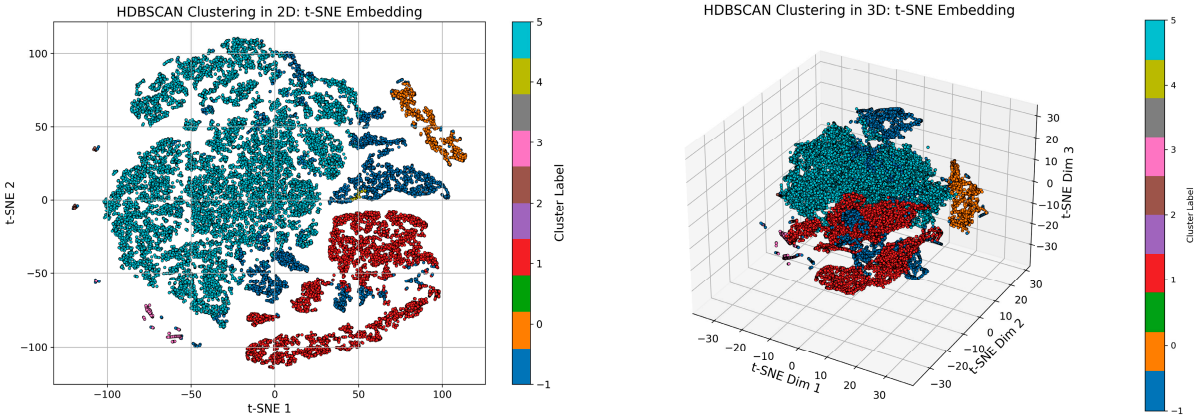
Figure 12 presents the PCA-based visualisations in two and three dimensions. The 2D PCA plot in Figure 12a shows cluster −1 distributed broadly across the space, forming a diffuse background that spans the primary PC1-PC2 axis. In contrast, several well-defined clusters, such as cluster 1 and cluster 5, appear in localised zones towards the extremities of the projection. The 3D PCA plot reinforces these observations, where most of the structure is compressed into two elongated volumes, with only modest separation between distinct clusters. This suggests that while PCA captures global trends, it does not adequately resolve the complexity present in smaller or transitional groupings.



(**a**) (**b**)

Figure 12. (**a**) HDBSCAN clustering—2D PCA visualisation. (**b**) HDBSCAN clustering—3D PCA visualisation.

Figure 13 provides the t-SNE visualisations of the same clustering assignments. The 2D t-SNE plot reveals strong visual separation between several clusters. In particular, cluster 1 forms a dense central group, while clusters 2, 3, and 4 are distributed in distinct, localised regions. Although there is some overlap between clusters, the separation is clearer than in the PCA plots and suggests that HDBSCAN has effectively captured the underlying structure in the feature space. However, the 3D t-SNE projection shows increased overlap between clusters and a more interwoven spatial distribution. This indicates that the true structure is more complex than what is visible in 2D, with some clusters potentially residing along curved or nested manifolds in higher dimensions.



(**a**) (**b**)

Figure 13. 2D (**a**) HDBSCAN clustering—2D t-SNE visualisation. (**b**) HDBSCAN clustering—3D t-SNE visualisation.

These spatial trends align with the cluster distributions shown in Tables 10 and 11. Cluster 1 includes the largest number of samples and is strongly associated with SOH Band 4 (Very Good), dominating the Nature dataset. Conversely, cluster 0 primarily consists of Loughborough samples and maps closely to lower SOH bands, suggesting that it represents degraded behaviour. Several other clusters contain only a small number of samples but appear in well-defined regions of the t-SNE space, which may correspond to particular usage conditions, calendar ageing, or inconsistent degradation mechanisms.

6. Discussion

The results from both clustering and visualisation reveal key strengths and limitations of the proposed methodological framework, particularly in relation to its ability to generalise across datasets with diverse use histories, chemistries, and health conditions.

The distribution of samples across clusters, as presented in Tables 8–11, illustrates several important trends. For both K-means and HDBSCAN, the clustering outcomes captured broad associations with SOH bands and dataset origin. In the case of K-means, one dominant cluster contained the majority of high SOH samples and most of the first-life data, while the remaining two clusters were more closely linked to degraded or transitional conditions. This segmentation reflects useful structure, but the boundaries between clusters were not sharply defined. Many samples in the intermediate SOH bands were split across multiple clusters, indicating a high degree of overlap and ambiguity. Attempts to increase the number of clusters only fragmented the groups further, reducing interpretability without resolving boundary clarity.

HDBSCAN, by contrast, produced a more granular segmentation, including smaller clusters and a large number of samples classified as noise. Some clusters were clearly localised to specific datasets or SOH bands, such as the small group in the Loughborough data linked to lower health categories. Others, such as cluster 1 in the Nature dataset, spanned a wide range of high-SOH cycles and formed a dense, well-defined structure. This more flexible assignment approach allowed HDBSCAN to better accommodate variability and heterogeneity, particularly in SLBs. However, the large noise group and the uneven distribution of cluster sizes also highlighted the difficulty of achieving stable segmentation when degradation behaviour is continuous and prior use is unknown.

The visualisations in reduced dimensions help contextualise these outcomes. PCA revealed broad gradients across the feature space but limited cluster separability, while t-SNE provided clear visual grouping in two dimensions. Notably, in the HDBSCAN results, 2D t-SNE embedding showed well-defined clusters with relatively low overlap. In contrast, the 3D t-SNE projection revealed much more intermixing, suggesting that the high-dimensional space contains local transitions that are not easily resolved in low-dimensional projections. These findings reinforce the importance of using multiple projections to avoid over-interpreting the structure seen in any single embedding. Manifold learning could sharpen apparent separation, but it was not examined in this study due to added transformations and tuning that can yield configuration-dependent groupings. The present work prioritises a simple, transparent baseline.

Taken together, these results suggest that IC feature space exhibits meaningful structure that reflects degradation-related behaviour. However, when examined through unsupervised clustering alone, the boundaries between different battery states are not sharply defined. The presence of overlapping clusters and ambiguous transitions, particularly in intermediate SOH bands, indicates that degradation may span a continuum rather than follow discrete category shifts. This effect is especially evident in SLBs, where varied and unknown prior usage leads to complex and sometimes inconsistent feature patterns. While

this presents challenges for unsupervised learning, the structured variation identified still provides a valuable foundation for supervised modelling in future work.

The results demonstrate that meaningful structure can be extracted from mixed chemistry datasets using the proposed normalisation. While this study does not provide detailed validation of the mechanisms underlying cross-chemistry compatibility, the clustering outcomes suggest that IC features capture degradation patterns that are sufficiently universal to enable joint analysis. The consistent alignment between cluster assignments and SOH bands across different chemistries supports the hypothesis that proper scaling can accommodate chemistry-specific operating conditions while preserving the health-related signal. Future work could explore the electrochemical basis for this compatibility and investigate whether chemistry-specific refinements could improve classification accuracy.

From a methodological perspective, the proposed framework separates generalisable analytical structure from dataset-sensitive parameter choices. Incremental capacity curve generation, ICA-based feature extraction, feature encoding and scaling, and the use of unsupervised cluster labels as meta-features are transferable components that rely only on standard cycling data and consistent signal processing. In contrast, parameters associated with data handling and cleaning (such as IQR thresholds), chemistry-dependent voltage windows, feature reduction thresholds, and density-based clustering hyperparameters are sensitive to dataset characteristics and require re-evaluation when applied to new or unseen battery datasets. This distinction enables controlled adaptation of the framework in transfer learning scenarios, where the overall pipeline structure is retained while dataset-dependent parameters are recalibrated based on the statistical properties of the target data.

Preprocessing parameters were tuned to balance comparability and data retention across sources. A global two-tier IQR rule with a 10% threshold reduced the Oxford subset to three rows, so the procedure was refined to a per-dataset basis, and thresholds were varied. The final 3% threshold was selected after experimenting with alternative thresholds and with global versus per-dataset IQR in supervised learning, where models were trained and assessed using classification reports and confusion matrices. This setting achieved the best trade-off between preserving samples and stabilising performance. Per-dataset scaling was applied to limit origin-driven scale effects prior to the unsupervised exploration.

Feature reduction in this study used a Pearson absolute correlation threshold of 0.90 to remove highly collinear variables, which addresses linear dependencies only. Non-linear redundancies may, therefore, persist and could, in principle, influence cluster formation, particularly for distance-based methods. Applying both K-means and HDBSCAN offers complementary perspectives, but a more decisive check will come in the supervised stage. In subsequent classification experiments, classification report metrics and confusion matrices will indicate whether any retained redundancies hinder predictive performance, and feature importance analysis will be used iteratively to identify and, where appropriate, remove less informative variables. Future extensions could consider mutual information-based screening to capture non-linear redundancy, but the present strategy prioritised transparency and reproducibility within the scope of this work.

For SLBs, SOH between capacity measures was interpolated linearly and used to create the SOH_bands target used in clustering. The IC-derived features and the clustering inputs do not include SOH, and variables directly related to capacity were excluded, so the analysis reflects IC-derived behaviour rather than capacity or SOH. In supervised settings, this construction contributes label noise rather than feature noise. Any effect is expected to be concentrated near band boundaries and to appear as increased confusion between adjacent classes in classification reports and confusion matrices. The reported alignments in this paper are, therefore, interpreted qualitatively.

Consequently, while unsupervised clustering alone does not yield discrete SOH groupings, the resulting cluster assignments may still be valuable as high-level meta-features. These labels can provide useful context in supervised learning pipelines by capturing structural similarity in the IC feature space, degradation regimes, or transitional behaviours not immediately evident from raw features.

In particular, the ability of HDBSCAN to explicitly identify noise points offers a practical advantage. These noise labels can be integrated into the supervised framework as an additional binary indicator, complementing the engineered flag column already introduced during preprocessing. This could allow learning algorithms to better distinguish uncertain or atypical datapoints, enhancing robustness when applied to real-world datasets that contain outliers or poorly defined patterns. Rather than serving as definite health classifications, the clusters and noise labels can augment downstream models with structure-aware signals that support classification or anomaly detection.

7. Conclusions

This study presents a reproducible framework for preparing and analysing battery cycling data using ICA, with a focus on supporting ML workflows across both first-life and second-life datasets. The methodology encompasses a complete data processing pipeline through initial data handling, IC-based feature engineering, encoding and scaling of variables, and feature space reduction using correlation analysis. This is followed by unsupervised structure exploration through clustering and dimensionality reduction techniques.

Two clustering algorithms, K-means and HDBSCAN, were applied to the engineered feature space. To assist interpretation, both linear and non-linear dimensionality reduction methods, PCA and t-SNE, were used to project the high-dimensional feature space into lower dimensions. These visualisations highlighted broad trends across datasets and degradation states while also exposing substantial overlap between clusters, particularly in regions representing transitional or intermediate SOH behaviour.

For K-means clustering, using three clusters yielded coarse groupings aligned with health trends but also revealed overlapping boundaries. HDBSCAN produced a finer segmentation and identified local groupings in the data while also assigning a large number of samples as noise. The presence of such uncertain or structurally atypical samples provides a key challenge when working with heterogeneous datasets, especially those involving SLBs with undocumented usage histories. To address this, a binary flag was introduced during preprocessing using a two-tiered IQR outlier detection system, designed to identify edge cases based on feature behaviour. HDBSCAN's noise detection further complemented this strategy by flagging additional ambiguous samples, enabling more flexible treatment of uncertainty in the dataset.

Rather than serving as target labels for classification, the cluster and noise assignments are proposed as structure-aware meta-features. These are incorporated into supervised learning pipelines as additional discrete input features (integer-encoded labels), along with the original ICA-derived features, allowing models to exploit cluster membership and noise identification as contextual information during training. The outcome of this study is, therefore, not a clustering-based classifier but a generalisable and interpretable methodology for ML preparation. All features and labels derived in this work will serve as the foundation for subsequent work focused on supervised SOH classification across the same mixed battery datasets. In that context, quantitative validation against the SOH bands is performed using classification performance metrics, confusion matrices, and feature importance analysis.

While the framework has been demonstrated across three heterogeneous sources, a full assessment of robustness on unseen datasets remains outstanding. Broader validation is currently limited by the scarcity of publicly available long-cycled second-life datasets. As the Loughborough dataset continues to accumulate cycles, additional data will become available to support further benchmarking and extended validation. Future studies will use the cluster and noise labels as meta features in supervised learning and will assess their influence on classification performance. This will enable the quantification of cross-dataset generalisation using classification report metrics and other complementary evaluation metrics. As feature reduction in this study relied on Pearson correlation and, therefore, addresses linear dependencies only, the supervised evaluation will indicate whether any non-linear redundancies affect performance. In parallel, expansion to other datasets and chemistries beyond lithium-ion will provide a broader test of generalisability.

Author Contributions: Conceptualisation, M.B. and P.F.; methodology, M.B., D.S., and P.F.; software, M.B.; validation, D.S. and P.F.; formal analysis, M.B.; investigation, M.B.; resources, D.S.; data curation, M.B.; writing—original draft preparation, M.B.; writing—review and editing, D.S. and P.F.; visualisation, M.B.; supervision, D.S. and P.F.; project administration, D.S. All authors have read and agreed to the published version of the manuscript.

Funding: This research was part-funded by the following project—Innovate UK, Faraday Challenge, 103951 "An ultra-low cost, resource efficient second life battery solution unlocking the mass-market for home energy storage" and Innovate UK project "Li.2".

Data Availability Statement: All data from the Loughborough dataset has been made publicly available through the Loughborough University Research Repository: https://doi.org/10.1038/s41597-025-05360-7. Any additional data or code is available upon request from Loughborough University by emailing d.strickland@lboro.ac.uk.

Acknowledgments: The author would like to acknowledge the help provided by Powervault for access to second-life batteries for the Loughborough dataset used in this study. Furthermore, the authors would like to acknowledge the help from Innovate UK.

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

Abbreviations

The following abbreviations are used in this manuscript:

EV	Electric vehicle
FLB	First-life battery
HDBSCAN	Hierarchical density-based spatial clustering of applications with noise
ICA	Incremental capacity analysis
IC	Incremental capacity
IQR	Interquartile range
LIB	Lithium-ion battery
ML	Machine learning
PCA	Principal component analysis
RUL	Remaining useful life
SLB	Second-life battery
SOH	State of health
t-SNE	t-distributed stochastic neighbour embedding

References

1. Berecibar, M.; Gandiaga, I.; Villarreal, I.; Omar, N.; Van Mierlo, J.; Van Den Bossche, P. Critical Review of State of Health Estimation Methods of Li-Ion Batteries for Real Applications. *Renew. Sustain. Energy Rev.* **2016**, *56*, 572–587. [CrossRef]
2. Guha, A.; Patra, A. State of Health Estimation of Lithium-Ion Batteries Using Capacity Fade and Internal Resistance Growth Models. *IEEE Trans. Transp. Electrif.* **2017**, *4*, 135–146. [CrossRef]
3. Salek, F.; Resalati, S.; Babaie, M.; Henshall, P.; Morrey, D.; Yao, L. A Review of the Technical Challenges and Solutions in Maximising the Potential Use of Second Life Batteries from Electric Vehicles. *Batteries* **2024**, *10*, 79. [CrossRef]
4. Azizighalehsari, S.; Venugopal, P.; Pratap Singh, D.; Batista Soeiro, T.; Rietveld, G. Empowering Electric Vehicles Batteries: A Comprehensive Look at the Application and Challenges of Second-Life Batteries. *Batteries* **2024**, *10*, 161. [CrossRef]
5. Hassini, M.; Redondo-Iglesias, E.; Venet, P. Battery Passports for Second-Life Batteries: An Experimental Assessment of Suitability for Mobile Applications. *Batteries* **2024**, *10*, 153. [CrossRef]
6. Beaudet, A.; Larouche, F.; Amouzegar, K.; Bouchard, P.; Zaghib, K. Key Challenges and Opportunities for Recycling Electric Vehicle Battery Materials. *Sustainability* **2020**, *12*, 5837. [CrossRef]
7. Patel, A.N.; Lander, L.; Ahuja, J.; Bulman, J.; Lum, J.K.H.; Pople, J.O.D.; Hales, A.; Patel, Y.; Edge, J.S. Lithium-Ion Battery Second Life: Pathways, Challenges and Outlook. *Front. Chem.* **2024**, *12*, 1358417. [CrossRef]
8. Schaltz, E.; Stroe, D.I.; Norregaard, K.; Ingvardsen, L.S.; Christensen, A. Incremental Capacity Analysis Applied on Electric Vehicles for Battery State-of-Health Estimation. *IEEE Trans. Ind. Appl.* **2021**, *57*, 1810–1817. [CrossRef]
9. Weng, C.; Cui, Y.; Sun, J.; Peng, H. On-Board State of Health Monitoring of Lithium-Ion Batteries Using Incremental Capacity Analysis with Support Vector Regression. *J. Power Sources* **2013**, *235*, 36–44. [CrossRef]
10. Beatty, M.; Strickland, D.; Ferreira, P. A Review of Methods of Generating Incremental Capacity–Differential Voltage Curves for Battery Health Determination. *Energies* **2024**, *17*, 4309. [CrossRef]
11. Beatty, M.; Strickland, D.; Warren, J.; Chan, J.; Ferreira, P. Long-Term Sweat Testing Dataset for Second-Life Batteries. *Sci. Data* **2025**, *12*, 1068. [CrossRef]
12. Yang, J.; Beatty, M.; Strickland, D.; Abedi-Varnosfaderani, M.; Warren, J. Second-Life Battery Capacity Estimation and Method Comparison. *Energies* **2023**, *16*, 3244. [CrossRef]
13. Birkl, C. *Oxford Battery Degradation Dataset 1*; University of Oxford: Oxford, UK, 2017. [CrossRef]
14. Severson, K.A.; Attia, P.M.; Jin, N.; Perkins, N.; Jiang, B.; Yang, Z.; Chen, M.H.; Aykol, M.; Herring, P.K.; Fraggedakis, D.; et al. Data-Driven Prediction of Battery Cycle Life before Capacity Degradation. *Nat. Energy* **2019**, *4*, 383–391. [CrossRef]
15. Spitthoff, L.; Vie, P.J.S.; Wahl, M.S.; Wind, J.; Burheim, O.S. Incremental Capacity Analysis (DQ/DV) as a Tool for Analysing the Effect of Ambient Temperature and Mechanical Clamping on Degradation. *J. Electroanal. Chem.* **2023**, *944*, 117627. [CrossRef]
16. Krupp, A.; Ferg, E.; Schuldt, F.; Derendorf, K.; Agert, C. Incremental Capacity Analysis as a State of Health Estimation Method for Lithium-Ion Battery Modules with Series-Connected Cells. *Batteries* **2020**, *7*, 2. [CrossRef]
17. SciPy Developers SciPy Simpson Integration Function. Available online: https://docs.scipy.org/doc/scipy/reference/generated/scipy.integrate.simpson.html (accessed on 19 May 2025).
18. Wang, J.; Zhang, C.; Meng, X.; Zhang, L.; Li, X.; Zhang, W. A Novel Feature Engineering-Based SOH Estimation Method for Lithium-Ion Battery with Downgraded Laboratory Data. *Batteries* **2024**, *10*, 139. [CrossRef]
19. SciPy Developers SciPy Find Peaks Function. Available online: https://docs.scipy.org/doc/scipy/reference/generated/scipy.signal.find_peaks.html (accessed on 19 May 2025).
20. Lin, H.; Kang, L.; Xie, D.; Linghu, J.; Li, J. Online State-of-Health Estimation of Lithium-Ion Battery Based on Incremental Capacity Curve and BP Neural Network. *Batteries* **2022**, *8*, 29. [CrossRef]
21. Dubarry, M.; Anseán, D. Best Practices for Incremental Capacity Analysis. *Front. Energy Res.* **2022**, *10*, 1023555. [CrossRef]
22. Su, L.; Xu, Y.; Dong, Z.; Zhaoyang Dong, C. State-of-Health Estimation of Lithium-Ion Batteries: A Comprehensive Literature Review from Cell to Pack Levels. *Energy Convers. Econ.* **2024**, *5*, 224–242. [CrossRef]
23. Lai, R.; Li, X.; Wang, J.; Viswanathan, V.; Lai, R.; Li, X.; Wang, J. Flexible Deep Learning-Based State of Health Estimation of Lithium-Ion Batteries with Features Extracted from Partial Charging Curves. *Batteries* **2024**, *10*, 164. [CrossRef]
24. Saini, U.; Renganathan, S. Incremental Capacity Analysis of Battery under Dynamic Load Conditions. *MethodsX* **2025**, *14*, 103331. [CrossRef] [PubMed]
25. Li, X.; Wang, Z.; Yan, J. Prognostic Health Condition for Lithium Battery Using the Partial Incremental Capacity and Gaussian Process Regression. *J. Power Sources* **2019**, *421*, 56–67. [CrossRef]
26. Xue, P.; Qiu, R.; Peng, C.; Peng, Z.; Ding, K.; Long, R.; Ma, L.; Zheng, Q. Solutions for Lithium Battery Materials Data Issues in Machine Learning: Overview and Future Outlook. *Adv. Sci.* **2024**, *11*, 2410065. [CrossRef]
27. Yang, J.; Strickland, D.; Abedi-Varnosfaderani, M. Comparison of Different Battery Balancing Methods for Use with Second Life Batteries. In Proceedings of the 9th International Conference on Renewable Energy Research and Applications, ICRERA 2020, Glasgow, UK, 27–30 September 2020; Institute of Electrical and Electronics Engineers Inc.: New York, NY, USA, 2020; pp. 112–117.

28. Pandas Developers Pandas DataFrame.Diff Function Documentation. Available online: https://pandas.pydata.org/docs/reference/api/pandas.DataFrame.diff.html (accessed on 28 April 2025).
29. Shanxuan, H.; Zuhong, L.; Bolun, Y.; Xu, G.; Biao, L.; Jingjing, Y. Learning to Fuse: Dynamic Integration of Multi-Source Data for Accurate Battery Lifespan Prediction. *arXiv* **2025**, arXiv:2504.18230. [CrossRef]
30. Hu, X.; Che, Y.; Lin, X.; Onori, S. Battery Health Prediction Using Fusion-Based Feature Selection and Machine Learning. *IEEE Trans. Transp. Electrif.* **2021**, *7*, 382–398. [CrossRef]
31. Jo, S.; Jung, S.; Roh, T. Battery State-of-Health Estimation Using Machine Learning and Preprocessing with Relative State-of-Charge. *Energies* **2021**, *14*, 7206. [CrossRef]
32. Fatima, N. Understanding Np.Log and Np.Log1p in NumPy. Available online: https://medium.com/@noorfatimaafzalbutt/understanding-np-log-and-np-log1p-in-numpy-99cefa89cd30 (accessed on 14 May 2025).
33. Thelen, A.; Huan, X.; Paulson, N.; Onori, S.; Hu, Z.; Hu, C. Probabilistic Machine Learning for Battery Health Diagnostics and Prognostics—Review and Perspectives. *npj Mater. Sustain.* **2024**, *2*, 14. [CrossRef]
34. Petkovski, E.; Marri, I.; Cristaldi, L.; Faifer, M. State of Health Estimation Procedure for Lithium-Ion Batteries Using Partial Discharge Data and Support Vector Regression. *Energies* **2023**, *17*, 206. [CrossRef]
35. Scikit Learn Developers Scikit StandardScaler Documentation. Available online: https://scikit-learn.org/stable/modules/generated/sklearn.preprocessing.StandardScaler.html (accessed on 28 April 2025).
36. Yang, Y.; Iqbal, M.Z. Cost-Optimised Machine Learning Model Comparison for Predictive Maintenance. *Electronics* **2025**, *14*, 2497. [CrossRef]
37. Aymerich, M.; García-Baizán, A.; Franco, P.N.; González, M.; San Miguel Fraile, P.; Ortiz-Rey, J.A.; Otero-García, M. Radiomics-Based Classification of Clear Cell Renal Cell Carcinoma ISUP Grade: A Machine Learning Approach with SHAP-Enhanced Explainability. *Diagnostics* **2025**, *15*, 1337. [CrossRef]
38. Sharma, P. K-Means Clustering Algorithm. Available online: https://www.analyticsvidhya.com/blog/2019/08/comprehensive-guide-k-means-clustering (accessed on 14 May 2025).
39. Sankalana, N. K-Means Clustering: Choosing Optimal K, Process, and Evaluation Methods. Available online: https://medium.com/@nirmalsankalana/k-means-clustering-choosing-optimal-k-process-and-evaluation-methods-2c69377a7ee4 (accessed on 14 May 2025).
40. Lee, K.C. Machine Learning 101—Classification vs. Clustering. Available online: https://kevin-c-lee26.medium.com/machine-learning-101-classification-vs-clustering-e11b12c71243 (accessed on 28 April 2025).
41. Ikotun, A.M.; Ezugwu, A.E.; Abualigah, L.; Abuhaija, B.; Heming, J. K-Means Clustering Algorithms: A Comprehensive Review, Variants Analysis, and Advances in the Era of Big Data. *Inf. Sci.* **2023**, *622*, 178–210. [CrossRef]
42. Scikit Learn Developers Scitkit Learn Silhoutte Score Documentation. Available online: https://scikit-learn.org/stable/modules/generated/sklearn.metrics.silhouette_score.html (accessed on 28 April 2025).
43. Scikit Learn Developers Scikit Learn K-Means Documentation. Available online: https://scikit-learn.org/stable/modules/generated/sklearn.cluster.KMeans.html (accessed on 28 April 2025).
44. McInnes, L.; Healy, J.; Astels, S. HDBSCAN: Hierarchical Density Based Clustering. *J. Open Source Softw.* **2017**, *2*, 205. [CrossRef]
45. Sharma, A. DBSCAN Clustering in Machine Learning. Available online: https://www.analyticsvidhya.com/blog/2020/09/how-dbscan-clustering-works/ (accessed on 28 April 2025).
46. Singh, A. Understanding T-SNE. t-SNE (t-Distributed Stochastic Neighbor Embedding). Available online: https://datainnovate.medium.com/understanding-t-sne-d869ece441a5 (accessed on 14 May 2025).

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.