Bayesian hierarchical modelling for battery lifetime early prediction

Zihao Zhou * **David A. Howey** *

* *University of Oxford, Parks Road, Oxford OX1 3PJ, UK (e-mail: david.howey@eng.ox.ac.uk).*

Abstract: Accurate prediction of battery health is essential for real-world system management and lab-based experiment design. However, building a life-prediction model from different cycling conditions is still a challenge. Large lifetime variability results from both cycling conditions and initial manufacturing variability, and this—along with the limited experimental resources usually available for each cycling condition—makes data-driven lifetime prediction challenging. Here, a hierarchical Bayesian linear model is proposed for battery life prediction, combining both individual cell features (reflecting manufacturing variability) with population-wide features (reflecting the impact of cycling conditions on the population average). The individual features were collected from the first 100 cycles of data, which is around 5-10% of lifetime. The model is able to predict end of life with a root mean square error of 3.2 days and mean absolute percentage error of 8.6%, measured through 5-fold cross-validation, overperforming the baseline (non-hierarchical) model by around 12-13%.

Keywords: Bayesian; lithium-ion battery; hierarchical model; lifetime variability

1. INTRODUCTION

Lithium-ion batteries are ubiquitous due to their relatively long lifetime and high energy density (Cano et al. (2018); Schmuch et al. (2018)), but their performance degrades with time and usage. Battery aging behaviours have been widely explored (Birkl et al. (2017)), but remain difficult to understand and quantify since batteries are complex electrochemical devices that operate in widely varying conditions.

Existing battery lifetime prediction models can be roughly divided into either physics-based (Reniers et al. (2019)) or data-driven approaches (Sulzer et al. (2021a)). Physics-based models are built from first principles, accounting for diverse aging mechanisms such as growth of the solid-electrolyte interphase (Liu et al. (2020)), lithium plating (Liu et al. (2016)), and particle cracking (Ai et al. (2019)). While these are good at explaining underlying aging mechanisms, developing models that predict behaviour under different cycling conditions is challenging, since the influential degradation mechanisms may be vary across cycling conditions (Su et al. (2016); Raj et al. (2020)).

Data-driven methods directly map from chosen health indicators (features) to battery lifetime (labels) without requiring underlying domain knowledge. For this purpose, (Severson et al. (2019)) proposed an input feature based on discharging curve differences in early cycles, called $\Delta Q(V)$, giving accurate predictions of lifetime using only early cycling data. Subsequently, other aging features have been proposed based on charging or discharging curve differences (Paulson et al. (2022)). However, the effectiveness of these may rely on the specific cycling conditions (Sulzer et al. (2021b)). Alternatively, instead of generating features from the raw signals (voltage/current/temperature)

and then using these as inputs for a model, other works implemented sophisticated machine learning methods to directly build models from raw measurements, such as ensemble learning (Li et al. (2019)) and long-short memory networks (Zhang et al. (2018)).

Although data-driven models achieve satisfactory results on their datasets, their ability to generalize is still unproven. Most of these models are built at a population level, i.e. all battery cells within a dataset follow the same feature-label mapping function, or to put it another way, the model that is learnt is an *average* model for the whole cell population. As a result, these models often lack the ability to give good performance on *individual* cells going through unseen cycling conditions. Recent work (Dechent et al. (2021); Strange et al. (2022)) also shows that cell-to-cell intrinsic manufacturing variability may not be properly addressed by a single population model. To address this, (Deng et al. (2022)) clustered cells under similar cycling conditions into subgroups, so that the usage variability could be reduced within each subgroup. Then, separate models were built for each subgroup. However, this separate modelling strategy needs to reach a balance between reducing within-group usage variability (more groups) versus group sample size (fewer groups). In reality, separate subgroup models often suffer from a lack of available data and tend to over-fit because of limited samples.

Different from either a single general population model or separate individual models, there is a reasonable middle ground in the form of hierarchical modelling (also called multi-level modelling). Hierarchical models are statistical models that have parameters at more than one level; they have proven to be a useful tool to deal with naturally structured data. For example, in a two-layer model there are pa-
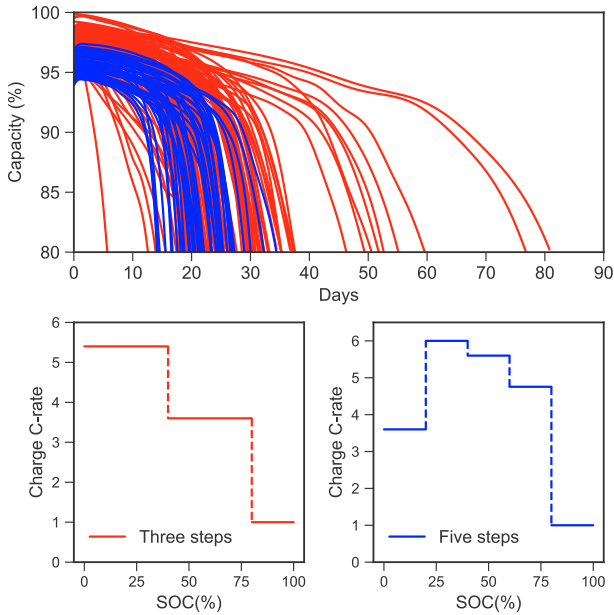
Fig. 1. Datasets of (Severson et al. (2019)) (red, three steps) and (Attia et al. (2020)) (blue, five steps) used here; top shows measured ageing trajectories, bottom shows exemplary individual charging protocols from each dataset respectively; in both cases the final step is a 1C constant current charge from 80–100% SOC.

rameters at the lower level which are assumed to be drawn from distributions, and these distributions are themselves parameterised by higher level parameters (Gelman and Hill (2006)). Hierarchical models have been widely used in different areas including ecology, psychology, sociology and computer vision (Lake et al. (2015); Pedersen et al. (2019)).

In this work, a hierarchical Bayesian model (HBM) is presented to achieve battery life prediction from early life measurements under varying cycling conditions. The individual cell feature-label relationships can vary across different cycling conditions according to cycling condition variables. Here we combine the MIT battery ageing dataset (Severson et al. (2019)) with extra samples from (Attia et al. (2020)) to generate a dataset consisting of 169 cells from 61 different cycling conditions. The lifetimes of these 169 cells range from 5 to 80 days. The individual battery health indicators (features) were calculated from the first 100 cycles of cycling data, which is only around 5-10% of a cell's average lifetime. We show that the HBM overperforms a baseline ridge regression model by 12% relative root-mean-square error (RMSE) and 13% mean-percentage-error (MPE).

2. DATA SOURCES AND FEATURE EXTRACTION

2.1 Data sources

In order to explore the impact of usage variability, different cycling conditions need to be included in the selected dataset. Here, the open source battery cycling datasets of (Severson et al. (2019)) and (Attia et al. (2020)) are combined together to give a dataset of 169 cells in total. The first dataset (Severson et al. (2019)) consists of 124

lithium iron phosphate/graphite 18650 Li-ion cells (A123) that were cycled at 30 °C. The second dataset (Attia et al. (2020)) is a follow up experiment that contains 45 cells of the same type as before. All these cells underwent identical discharge cycles at 4C but had different fast charging protocols. (In this paper C-rate has the conventional definition, i.e. charge or discharge current over nominal capacity.) The cells had a 1.1 Ah nominal capacity and the end of life (EoL) is defined as 80% of nominal capacity remaining. Cycling data (voltage/current/temperature) with time resolution of 1 second are included. More details can be found in (Severson et al. (2019)) As shown in Fig 1, the EoL time covers a wide range from 5 to 80 days (150–2300 cycles), with an average lifetime of 26 days (797 cycles). Cells from the first dataset experienced charging protocols with three steps, while cells from the second set experienced five-step charging protocols.

2.2 Feature extraction

Feature engineering is not the main focus of this work, hence we chose a small number of features at both the group and individual levels in the hierarchical model, guided by existing literature and our prior understanding of battery ageing. In the datasets considered, the discharging protocol is uniform across all cells, while the charging protocols are quite different from cell to cell. The main difference between them is the sequence of C-rates used during charging, so one way to compare them is to calculate some kind of average charging C-rate for each protocol. Here we used the 'SOC-average' charging C-rate as a metric to define differences between protocols—this is related to the total ohmic heat generation (I^2RT, where R is resistance) in the cell during each charging period, normalised by capacity, since

$$\frac{d(\text{SOC})}{dt} = \frac{I}{Q_0}, \quad d(\text{SOC}) = \frac{I}{Q_0}dt$$
$$\text{hence} \int_0^1 I d(\text{SOC}) = \int_0^T I^2 \frac{1}{Q_0} dt, \tag{1}$$

where the Q_0 is the nominal capacity, T the total charge time, and I indicates current. Taking the three steps protocol shown in Fig 1 as an example, the cell is first charged to 40% SOC at 5.4C, and then charged from 40% to 80% SOC at 3.6C. So, the average charging C-rate for this charging protocol is

$$q_1 = 40\%, \quad q_2 = 40\%, \quad q_3 = 20\%$$
$$5.4 \times q_1 + 3.6 \times q_2 + 1 \times q_3 = 3.8\text{C}. \tag{2}$$

Note that the final 80% to 100% SOC stage is always at 1C for all cells within the dataset.

As shown in Fig 2, the average charging C-rate has a strong predictive relationship with the end-of-life (EoL) time in this combined dataset. Multiple cells may have similar or the same average charging C-rate, and these cells will be clustered together into groups later. We denote this average charging C-rate metric as a group level feature g, as shown in Table 1.

At the individual cell level, three features which are widely used in existing literature (Severson et al. (2019); Paulson et al. (2022); Fei et al. (2021)) were calculated from the first 100 cycles of data for every cell. These are

Table 1. Input features used in his work

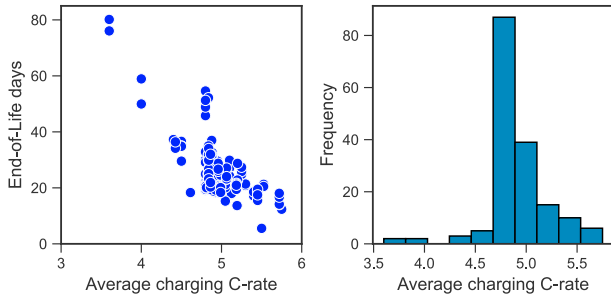| Symbol	Description	Level
g	Average charging C-rate	Group
F1	Variance of discharge $\Delta Q(V)$ curve between 10th and 100th cycles	Individual
F2	Minimum of discharge $\Delta Q(V)$ curve between 10th and 100th cycles	Individual
F3	Discharge capacity at cycle 2	Individual

Fig. 2. Left shows the relationship between the average charging C-rate and EoL time in days; the absolute linear correlation coefficient is 0.71. Right shows histogram of calculated average charging C-rates.
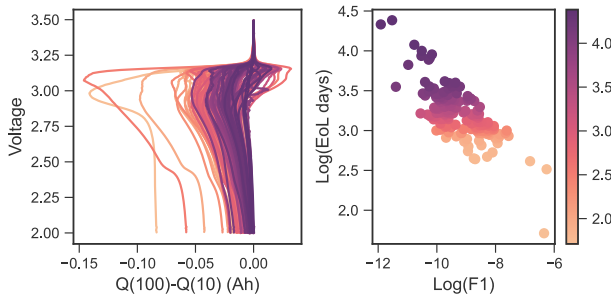

Fig. 3. Left shows the feature 'F1', the $\Delta Q(V)$ curve between the 10th and 100th discharge cycles; right shows the log-log relationship between the most predictive feature (F1) and the EoL in days. As already reported (Severson et al. (2019)), the correlation is strong, in this case the absolute value of Pearson's linear coefficient is 0.76.

denoted as individual-level features (F1-F3) in Table 1. The most predictive feature (the one having the largest linear coefficient with log of EoL time), and its relationship with log EoL time, is shown in Fig. 3.

3. METHODOLOGY

The overall lifetime variability consists of variability caused by usage differences, and variability from manufacturing differences. Ideally there would be many samples for every usage condition so that the cell-to-cell differences under usage could be estimated. However, in the dataset used, there are only 1-2 cells tested for each different protocol—a sample size too small to give reliable conclusions. To address this, we used constrained K-means clustering to gather cells into usage-related groups each having similar (although not necessarily identical) average charging C-rates.
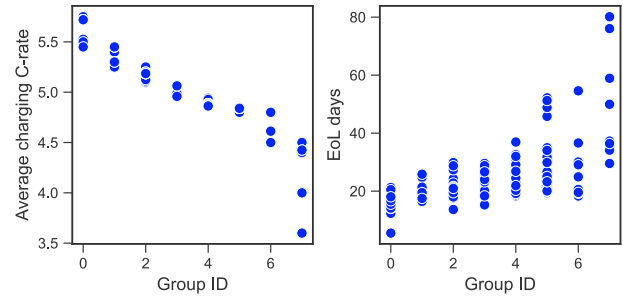

Fig. 4. Left shows average charging C-rate for each group with cells clustered into eight groups; right shows associated EoL days for each group. There is relatively large usage variability within the final group due to lack of cells at low (<3.5C) charging C-rates.

3.1 Cycling conditions clustering

A commonly used technique to address the balance between individual cell behaviour vs. population model accuracy is clustering, i.e. grouping of cells into 'similar' behavioural subgroups that each contain a relatively large number of cells. However, most previous works (Deng et al. (2022); Jiang et al. (2021)) cluster cells based on features from individual cell voltage signals (or cell EoL lifetime) rather than features representing usage. As a result, cells undergoing quite different cycling conditions may be grouped together. In this work, clustering uses the cycling condition level features (g), which enables cells with similar cycling conditions to be grouped together.

The tool used is constrained K-means clustering (Bhattacharya et al. (2018)). It is an improved version of K-means that avoids local solutions containing empty clusters, or clusters having very few samples. The goal is to have sufficient sample size (at least 10 cells) in each group, while also maintaining small usage variability within each group (at most 100 cells).

There are 169 cells with 61 different average charging C-rates in the dataset, so a reasonable number of clusters is $K \in [4, 16]$. Here, $K = 8$ is used as an example (further investigation of the influence of cluster number would be interesting future work). The corresponding clustering result is shown in Fig 4.

3.2 Bayesian hierarchical linear model

We now discuss the construction of the hierarchical model for life prediction. To motivate the need for a hierarchical model, one should first check whether the group level effect is significant in the target dataset. The variance partition coefficient (VPC), Equation 3, is a metric which represents

Hyper-priors

$\gamma \sim \mathcal{N}(0, 10)$

γ

Cycling Conditions Level-2:

$\boldsymbol{\theta_j} \sim \mathcal{N}(\boldsymbol{\gamma}^\top \boldsymbol{g_j}, \boldsymbol{\sigma_j^2})$

$\boldsymbol{\theta_1}, \boldsymbol{\sigma_1}$ ... $\boldsymbol{\theta_j}, \boldsymbol{\sigma_j}$ ... $\boldsymbol{\theta_J}, \boldsymbol{\sigma_J}$

Individual Cells Level-1:

$y_{ji} \sim \mathcal{N}(\boldsymbol{\theta_j^\top} \boldsymbol{x_{ji}}, \sigma_y^2)$

$y_{11}, \ldots, y_{1i}, \ldots, y_{1n_1}$ ... $y_{j1}, \ldots, y_{ji}, \ldots y_{jn_j}$ ... $y_{J1}, \ldots, y_{J,i}, \ldots y_{Jn_J}$
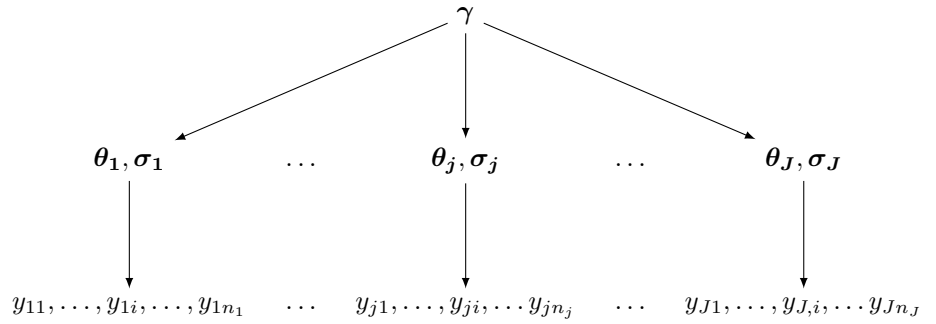
Fig. 5. Hierarchical model structure: parameters can be divided into individual cell level parameters $(\boldsymbol{\theta_j}, \boldsymbol{\sigma_j})$ and cycling condition level hyper-parameters $(\boldsymbol{\gamma})$. While j represents cycling condition group index, i represents individual cell index, y_{ji} represents lifetime of ith cell in jth cycling group. This two level structure allows the individual cell level feature-label $(x_{ji} - y_{ji})$ relationship to vary across different cycling conditions based on cycling condition level features $(\boldsymbol{g_j})$.

the percentage of label variance explained by including a grouping effect.

$$\text{VPC} = \frac{\sigma_{\text{group}}^2}{\sigma_{\text{individual}}^2 + \sigma_{\text{group}}^2}$$

$$\sigma_{\text{group}}^2 = \text{var}\left(\bar{u}_j - \bar{u}\right), \bar{u} = \frac{1}{N} \sum_{j=1}^{k} \sum_{i=1}^{n_j} y_{ji} \qquad (3)$$

$$\sigma_{\text{individual}}^2 = \text{var}\left(y_{ji} - \bar{u}_j\right), \bar{u}_j = \frac{1}{n_j} \sum_{i=1}^{n_j} y_{ji}$$

Here, σ_{group} defines the variance between groups, and $\sigma_{\text{individual}}$ defines the variance of the whole population where each cell is compared to the mean of its group, $\bar{u}$ is the average lifetime for all cells in the population, $\bar{\mu}_j$ is the average lifetime of the jth group, var has the usual definition of sample variance, y_{ji} is all individual lifetimes for the ith cell within the jth group, n_j is the total number of cells within the jth group and N is the total number of cells in the population. When the group variance $\sigma_{\text{group}}^2 = 0$, there is no difference between groups, that is, every group has same mean lifetime compared to the overall population battery mean lifetime. When the $\sigma_{\text{individual}}^2 = 0$, there is no differences within a group, that is, within each group, samples share the same lifetime value. For our dataset, VPC = 74.5%, which denotes that 74.5% of the lifetime variance is caused by the cycling condition group effect.

Given the large influence from cycling condition group, it is reasonable to apply a hierarchical model. The generative model structure is described by the schematic in Fig. 3.2. In this framework, within the jth group, the first (lower) level inference is done for individual cell lifetime regression parameters $(\boldsymbol{\theta_j})$ where θ_{j0} is a scalar representing the mean lifetime within the jth group and $\boldsymbol{\theta_j}$ is a 4×1 vector because the number of features is three. In this case because we only have three features, we could have explicitly written the lower level forward model for the jth group and ith cell within that group as

$$y_{ij} \sim \mathcal{N}(\theta_{j0} + \theta_{j1} x_{i1}^j + \theta_{j2} x_{i2}^j + \theta_{j3} x_{i3}^j, \sigma_y^2) \qquad (4)$$

Since the EoL time y_{ij} is accurately measured in the dataset, the sample noise variance σ_y was fixed to 1 here

(it can also be assigned as a distribution and learnt from the dataset). For the sake of flexibility and compactness we use vector notion. The second (upper) level inference is undertaken for cycling condition parameters $(\boldsymbol{\gamma_0}, \boldsymbol{\gamma_1}, \boldsymbol{\gamma_2}, \boldsymbol{\gamma_3})$, where the $\boldsymbol{\gamma_0}, \boldsymbol{\gamma_1}, \boldsymbol{\gamma_2}, \boldsymbol{\gamma_3}$ parameters are each 2×1 vectors (because each group has one group level feature and an intercept parameter), defined as

$$\boldsymbol{\theta_j} = \begin{bmatrix} \theta_{j0} \\ \theta_{j1} \\ \theta_{j2} \\ \theta_{j3} \end{bmatrix} \boldsymbol{g_j} = \begin{bmatrix} 1 \\ \frac{1}{n_j} \sum_{i=1}^{n_j} g_{ji} \end{bmatrix} \boldsymbol{x_{ji}} = \begin{bmatrix} 1 \\ x_{i1}^j \\ x_{i2}^j \\ x_{i3}^j \end{bmatrix}$$

$$\boldsymbol{\gamma_0} = \begin{bmatrix} \gamma_{0,0} \\ \gamma_{0,g} \end{bmatrix} \boldsymbol{\gamma_1} = \begin{bmatrix} \gamma_{1,0} \\ \gamma_{1,g} \end{bmatrix} \boldsymbol{\gamma_2} = \begin{bmatrix} \gamma_{2,0} \\ \gamma_{2,g} \end{bmatrix} \boldsymbol{\gamma_3} = \begin{bmatrix} \gamma_{3,0} \\ \gamma_{3,g} \end{bmatrix} \quad (5)$$

$$\boldsymbol{\gamma} = [\,\boldsymbol{\gamma_0}, \boldsymbol{\gamma_1}, \boldsymbol{\gamma_2}, \boldsymbol{\gamma_3}\,], Y_j = (y_{j1}, \ldots, y_{jn_j})$$

$$\{Y\} = \{Y_1, \ldots Y_j\},$$

where Y_j is an output vector of lifetimes for all cells within the jth group.

The individual slopes of each group model, $\boldsymbol{\theta_j}$, are assumed to be drawn from normal distributions centered at $\boldsymbol{\gamma}^\top \boldsymbol{g_j}$,

$$\boldsymbol{\theta_j} \sim \mathcal{N}\left(\boldsymbol{\gamma}^\top \boldsymbol{g_j}, \boldsymbol{\sigma_j^2}\right), \qquad (6)$$

where $\boldsymbol{g_j}$ is a 2×1 vector of known specific mean features for the jth group (i.e., the mean of the individual average charging C-rate for each group). In the top layer, weakly informative (i.e., wide) priors $\mathcal{N}(0, 10)$ are assigned for all hyper-parameters. And for setting the priors for θ_j, a non-centered reparameterization trick, proposed by (Betancourt and Girolami (2015)), is applied to improve sampler efficiency.

Our objective is to infer posterior distributions for both the group models and the population model, $P(\boldsymbol{\theta_j} \mid Y_j)$ and $P(\boldsymbol{\gamma} \mid \{Y\})$ respectively. The overall process is

(1) Calculate level-2 posterior distribution $P(\boldsymbol{\gamma} \mid \{Y\})$;
(2) Use level-2 posterior distribution as prior for level-1 parameters, calculate level-1 posterior distribution $P(\boldsymbol{\theta_j} \mid \boldsymbol{\gamma}, Y_j)$;
(3) Use level-1 posterior distribution to make prediction on individual labels.

Using Bayes' rule, the level-2 posterior can be estimated by multiplying the likelihood function with the hyper-prior:

$$P(\boldsymbol{\gamma} \mid \{Y\}) \propto \text{Likelihood} \times \text{Prior}$$

$$\text{Likelihood} = \prod_{j=1}^{J} P(Y_j \mid \boldsymbol{\gamma}) \tag{7}$$

$$\text{Prior} = \mathcal{N}(\mathbf{0}, S_0), S_0 = 10 \cdot I$$

Since the level-2 hyperparameters $\boldsymbol{\gamma}$ are not directly related to individual observations, the likelihood function can be calculated by integrating out the level-1 parameters, giving

$$P(Y_j \mid \boldsymbol{\gamma}) = \int_{\boldsymbol{\theta_j}} P(Y_j \mid \boldsymbol{\theta_j}) \cdot P(\boldsymbol{\theta_j} \mid \boldsymbol{\gamma}) \, d\boldsymbol{\theta_j}$$

$$P(Y_j \mid \boldsymbol{\theta_j}) = \prod_{i=1}^{n_j} \mathcal{N}\left(\text{err}_i(\boldsymbol{\theta_j}); 0, 1\right) \tag{8}$$

$$\text{err}_i(\boldsymbol{\theta_j}) = \left\{\boldsymbol{\theta_j}^\top x_{ij} - y_{ij}\right\}.$$

Notice that the level-2 likelihood component $P(Y_j \mid \boldsymbol{\gamma})$ is the model evidence (marginal likelihood) for the level-1 model of jth subgroup, which indicates there is an averaging effect for level-1 model selection of each subgroup in our proposed two level structure.

Given the level-2 likelihood expression (8), the posterior distribution for hyper-parameters $P(\boldsymbol{\gamma} \mid \{Y\})$ can be estimated by Markov chain Monte Carlo (MCMC). After that, the posterior distributions for level-1 parameters $P(\boldsymbol{\theta_j} \mid \boldsymbol{\gamma}, Y_j)$ can be estimated by applying Bayes' rule,

$$P(\boldsymbol{\theta_j} \mid \boldsymbol{\gamma}, Y_j) = \frac{P(Y_j \mid \boldsymbol{\theta_j}) \cdot P(\boldsymbol{\theta_j} \mid \boldsymbol{\gamma}) \cdot P(\boldsymbol{\gamma})}{P(Y_j \mid \boldsymbol{\gamma})}. \tag{9}$$

The whole inference process was performed in Python using the PYMC3 package Salvatier et al. (2016).

3.3 Baseline comparison and evaluation metrics

The performance of the approach was evaluated by comparing against a baseline ridge regression model under two different error metrics, defined below. To give a richer comparison, the ridge model was built on two different feature sets: one has 6 features (denoted as the "discharge" model in Severson et al. (2019)), and the other has 4 features (including g and F1-F3, the same features as the HBM). Notice that the 6 features used in the "discharge" model also include F1-F3 used in this paper.

The first error metric is the root mean square error (RMSE) of the EoL predictions. This is a commonly used metric for predictive performance, however, it is significantly influenced by outliers. Therefore, the second metric is mean absolute percentage error (MAPE), which measures the percentage difference in EoL time between the predicted y_{pred} and observed end of life y_{real}. The two are defined by

$$\text{RMSE} = \sqrt{\frac{\sum_{i=1}^{N}\left(y_{\text{true}}^{i} - y_{\text{pred}}^{i}\right)^2}{N}},$$

$$\text{MAPE} = \frac{100}{N}\sum_{i=1}^{N}\left|\frac{y_{\text{true}}^{i} - y_{\text{pred}}^{i}}{y_{\text{true}}^{i}}\right|. \tag{10}$$
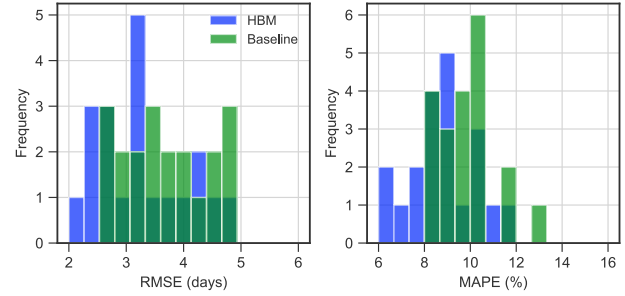
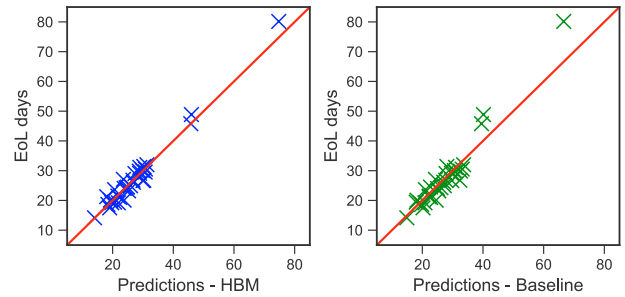

Fig. 6. Histograms of all results, as RMSE and MAPE

Fig. 7. Scatter plot of predicted versus actual EoL (days) for HBM vs. baseline model

K-fold cross validation is a commonly used paradigm to evaluate predictive ability (Arlot and Celisse (2010)). Here, the train and test sets were selected with 5-fold cross-validation, resulting in a 135 cells training set and a 34 cells test set. Then, 5-fold cross validation was performed 4 times (totalling 20 RMSE values) to observe the general predictive ability of the proposed model.

Table 2 summarises the results. The HBM gives 3.20 days median RMSE and 8.6% median MAPE, surpassing the baseline model on both feature sets. Notice that, by including usage feature g, the performance of the baseline model is also improved, which indicates the importance of cycling condition as a feature. More visualization of the results is shown in Fig. 6. For the HBM, the majority (70%) of the RMSE values on EoL time are below 3.5 days. For the baseline model, 50% of trials give RMSE values larger than 3.5 weeks. The case for MAPE is also similar. The HBM generally gives around 12-13% improvement in performance compared to the original baseline model in both RMSE and MAPE.

4. RESULTS AND DISCUSSION

Fig. 7 shows a scatter plot of predictions versus actual EoL time from one trial for the HBM and the baseline model. Both give relatively good predictions in the mid-life range (around 30 days). However, the baseline model fails to give good predictions for long-life (>40 days) cells. The HBM has a reasonably consistent predictive performance across the whole lifetime range.

The superior performance of the proposed HBM can be explained by its ability to account for relationships between different levels of features, or in other words, the inherent structure within the dataset. The cycling condition level feature influences the relationships between

Table 2. Summary of model predictive performance (RMSE is in days)

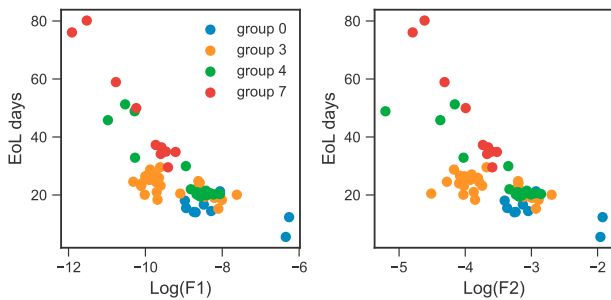|		Baseline (original)	Baseline (with feature g)	HBM	Improvement
RMSE	Median	3.67	3.38	3.20	12.8%
	Mean	3.72	3.49	3.26	12.4%
MPE	Median	9.9%	9.1%	8.6%	13.1%
	Mean	9.9%	9.2%	8.7%	12.1%

Fig. 8. Relationships between EoL vs. individual cell features and group labels based on different cycling conditions. Groups $0, 3, 4, 7$ are selected to show the influence of cycling conditions. See also Fig 4.

individual cell level features. Fig. 8 shows cells from four different cycling condition clusters—while all of them share negative correlations between features and labels, their slopes and input feature ranges are quite different. As the average charging C-rate decreases from group 0 to group 7, the slope become steeper and the range of log F1 values decreases.

In a conventional regression setting, the cycling condition features and individual cell features are treated in the same way, assuming they are independent. As a result, the inherent relationships between these two different categories of features are ignored. In contrast, in hierarchical approaches, this kind of inherent relationship is explicitly modelled, which enables better model performance when facing structured data.

5. CONCLUSION

A hierarchical Bayesian linear model was proposed to address the problem of battery early life prediction under varying usage conditions and using only the first 100 cycles data (only 5-10% of the entire life) as inputs. The proposed HBM was tested on a open data set consisting of 169 cells experiencing 61 different cycling protocols. In a 5-fold cross-validation experiment, the HBM gives a 3.20 days median RMSE and a 8.6% median MAPE, which overperforms the baseline model by 12%. These results show the effectiveness and further potential of the hierarchical models for battery early life prediction.

There are two directions for future work. First, the influence of cluster number and cluster size needs to be further investigated, as these two parameters are known to be important in many other fields when using hierarchical models (Gelman and Hill (2006)). In our dataset, eight clusters is a reasonable number to balance within-group usage variability versus sample size. However, this was an empirical choice based mainly on trial and error. Second, more different usage conditions need to be included. In the

used dataset, the usage differences only locate in charging period. The influence from discharge C-rate, depth of discharge and rest periods are not included. More useful group level features may be found by including more usage conditions. Third, nonlinear functions, or even non-parametric models such as Gaussian processes, could be implemented hierarchically, and may give better performance compared to linear regression models. The codes for this work is available online (Zhou (2023)). And a further collaboration work with Prof. Hu Chao's group aims to address above challenges is under preparation.

REFERENCES

Ai, W., Kraft, L., Sturm, J., Jossen, A., and Wu, B. (2019). Electrochemical thermal-mechanical modelling of stress inhomogeneity in lithium-ion pouch cells. *Journal of The Electrochemical Society*, 167(1), 013512.

Arlot, S. and Celisse, A. (2010). A survey of cross-validation procedures for model selection. *Statistics surveys*, 4, 40–79.

Attia, P.M., Grover, A., Jin, N., Severson, K.A., Markov, T.M., Liao, Y.H., Chen, M.H., Cheong, B., Perkins, N., Yang, Z., et al. (2020). Closed-loop optimization of fast-charging protocols for batteries with machine learning. *Nature*, 578(7795), 397–402.

Betancourt, M. and Girolami, M. (2015). Hamiltonian monte carlo for hierarchical models. *Current trends in Bayesian methodology with applications*, 79(30), 2–4.

Bhattacharya, A., Jaiswal, R., and Kumar, A. (2018). Faster algorithms for the constrained k-means problem. *Theory of computing systems*, 62(1), 93–115.

Birkl, C.R., Roberts, M.R., McTurk, E., Bruce, P.G., and Howey, D.A. (2017). Degradation diagnostics for lithium ion cells. *Journal of Power Sources*, 341, 373–386.

Cano, Z.P., Banham, D., Ye, S., Hintennach, A., Lu, J., Fowler, M., and Chen, Z. (2018). Batteries and fuel cells for emerging electric vehicle markets. *Nature Energy*, 3(4), 279–289.

Dechent, P., Greenbank, S., Hildenbrand, F., Jbabdi, S., Sauer, D.U., and Howey, D.A. (2021). Estimation of li-ion degradation test sample sizes required to understand cell-to-cell variability. *Batteries & Supercaps*, 4(12), 1821–1829.

Deng, Z., Lin, X., Cai, J., and Hu, X. (2022). Battery health estimation with degradation pattern recognition

and transfer learning. *Journal of Power Sources*, 525, 231027.

Fei, Z., Yang, F., Tsui, K.L., Li, L., and Zhang, Z. (2021). Early prediction of battery lifetime via a machine learning based framework. *Energy*, 225, 120205.

Gelman, A. and Hill, J. (2006). *Data analysis using regression and multilevel/hierarchical models*. Cambridge university press.

Jiang, B., Gent, W.E., Mohr, F., Das, S., Berliner, M.D., Forsuelo, M., Zhao, H., Attia, P.M., Grover, A., Herring, P.K., et al. (2021). Bayesian learning for rapid prediction of lithium-ion battery-cycling protocols. *Joule*, 5(12), 3187–3203.

Lake, B.M., Salakhutdinov, R., and Tenenbaum, J.B. (2015). Human-level concept learning through probabilistic program induction. *Science*, 350(6266), 1332–1338.

Li, Y., Zhong, S., Zhong, Q., and Shi, K. (2019). Lithium-ion battery state of health monitoring based on ensemble learning. *IEEE access*, 7, 8754–8762.

Liu, Q., Du, C., Shen, B., Zuo, P., Cheng, X., Ma, Y., Yin, G., and Gao, Y. (2016). Understanding undesirable anode lithium plating issues in lithium-ion batteries. *RSC advances*, 6(91), 88683–88700.

Liu, W., Liu, P., and Mitlin, D. (2020). Review of emerging concepts in sei analysis and artificial sei membranes for lithium, sodium, and potassium metal battery anodes. *Advanced Energy Materials*, 10(43), 2002297.

Paulson, N.H., Kubal, J., Ward, L., Saxena, S., Lu, W., and Babinec, S.J. (2022). Feature engineering for machine learning enabled early prediction of battery lifetime. *Journal of Power Sources*, 527, 231127.

Pedersen, E.J., Miller, D.L., Simpson, G.L., and Ross, N. (2019). Hierarchical generalized additive models in ecology: an introduction with mgcv. *PeerJ*, 7, e6876.

Raj, T., Wang, A.A., Monroe, C.W., and Howey, D.A. (2020). Investigation of path-dependent degradation in lithium-ion batteries. *Batteries & Supercaps*, 3(12), 1377–1385.

Reniers, J.M., Mulder, G., and Howey, D.A. (2019). Review and performance comparison of mechanical-chemical degradation models for lithium-ion batteries. *Journal of The Electrochemical Society*, 166(14), A3189.

Salvatier, J., Wiecki, T.V., and Fonnesbeck, C. (2016). Probabilistic programming in python using pymc3. *PeerJ Computer Science*, 2, e55.

Schmuch, R., Wagner, R., Hörpel, G., Placke, T., and Winter, M. (2018). Performance and cost of materials for lithium-based rechargeable automotive batteries. *Nature Energy*, 3(4), 267–278.

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. (2019). Data-driven prediction of battery cycle life before capacity degradation. *Nature Energy*, 4(5), 383–391.

Strange, C., Allerhand, M., Dechent, P., and dos Reis, G. (2022). Automatic method for the estimation of li-ion degradation test sample sizes required to understand cell-to-cell variability. *Energy and AI*, 100174.

Su, L., Zhang, J., Huang, J., Ge, H., Li, Z., Xie, F., and Liaw, B.Y. (2016). Path dependence of lithium ion cells aging under storage conditions. *Journal of Power Sources*, 315, 35–46.

Sulzer, V., Mohtat, P., Aitio, A., Lee, S., Yeh, Y.T., Steinbacher, F., Khan, M.U., Lee, J.W., Siegel, J.B., Stefanopoulou, A.G., et al. (2021a). The challenge and opportunity of battery lifetime prediction from field data. *Joule*, 5(8), 1934–1955.

Sulzer, V., Mohtat, P., Lee, S., Siegel, J.B., and Stefanopoulou, A.G. (2021b). Promise and challenges of a data-driven approach for battery lifetime prognostics. In *2021 American Control Conference (ACC)*, 4427–4433. IEEE.

Zhang, Y., Xiong, R., He, H., and Pecht, M.G. (2018). Long short-term memory recurrent neural network for remaining useful life prediction of lithium-ion batteries. *IEEE Transactions on Vehicular Technology*, 67(7), 5695–5705.

Zhou, Z. (2023). https://github.com/battery-intelligence-lab/hbm_ifac.git.