A naive Bayes model for robust remaining useful life prediction of lithium-ion battery

Selina S.Y. Ng *, Yinjiao Xing, Kwok L. Tsui

Department of Systems Engineering and Engineering Management, City University of Hong Kong, Tat Chee Avenue, Kowloon, Hong Kong

HIGHLIGHTS

- Robustness of RUL predictions for lithium-ion batteries is analyzed quantitatively.
- RUL predictions of the same battery over cycle life are evaluated.
- RUL predictions of batteries over different operating conditions are evaluated.
- Naive Bayes (NB) is proposed for predictions under constant discharge environments.
- Its robustness and accuracy are compared with that of support vector machine (SVM).

ABSTRACT

Online state-of-health (SoH) estimation and remaining useful life (RUL) prediction is a critical problem in battery health management. This paper studies the modeling of battery degradation under different usage conditions and ambient temperatures, which is seldom considered in the literature. Li-ion battery RUL prediction under constant operating conditions at different values of ambient temperature and discharge current are considered. A naive Bayes (NB) model is proposed for RUL prediction of batteries under different operating conditions. It is shown in this analysis that under constant discharge environments, the RUL of Li-ion batteries can be predicted with the NB method, irrespective of the exact values of the operating conditions. The case study shows that the NB generates stable and competitive prediction performance over that of the support vector machine (SVM). This also suggests that, while it is well known that the environmental conditions have big impact on the degradation trend, it is the changes in operating conditions of a Li-ion battery over cycle life that makes the Li-ion battery degradation and RUL prediction even more difficult.

© 2013 Elsevier Ltd. All rights reserved.

1. Introduction

Battery-powered systems constitute an essential part of our everyday life. The consequences of battery failure can range from inconvenience, production downtime, to catastrophic failure. The health of a battery may refer to a number of parameters that indicate the battery condition, such as the cycle number, the full charge capacity (FCC), and the state-of-charge (SoC). Three critical parameters for battery health management are the real-time estimation of the remaining charge within one discharge cycle, or SoC estimation; the real-time estimation of the battery's capability to deliver its specified output, or state-of-health (SoH) estimation; and the prediction of remaining useful life (RUL). The ultimate goal of SoC estimation to ensure optimum control of the charging/discharging process, thereby maximize the run-time per discharge cycle [1]. SoH estimation and RUL prediction, on the other hand, aim at determining when the battery can no longer hold a useful amount of energy and needs to be replaced, thereby maximize the number of cycles attainable for the life of the battery. These two problems are modeled with battery data on different time-scales. Some recent papers on SoC and capacity estimation include [2–6]. This paper focuses on the RUL to end-of-life (EOL) prediction in SoH estimation. A more comprehensive introduction to battery health management and Li-ion batteries is in [7].

There are fewer papers on battery SoH estimation and RUL prediction than in SoC estimation. One challenge in battery RUL prediction is the controversy over the definition of SoH. For example, capacity indicates how much electricity can be stored, and power indicates how fast electricity can be given out. They represent different aspects of battery health. Both capacity and power fade have been used by Pattipati et al. [1] as SoH indicators. Recently, Liu et al. [8] proposed a novel health indicator with observable parameters instead of parameters from battery properties,
although more evidence of its effectiveness is required. The end-of-life (EOL) is usually defined as the point at which the full charge capacity (FCC) drops by a certain percentage from its initial rated capacity, typically 20–30% [9,10]. Predicting the RUL from the SoH usually involves finding the k-step-ahead projection of SoH, then transforming the capacity predictions to time-to-EOL [9]. In this paper, the most popular health indicator, the capacity, is used and the EOL is defined at 70% of rated capacity as in [10].

Techniques for prognostics and health management (PHM) [11] of systems and components can be roughly classified into the Physics-of-failure-based approach (PoF-based approach), the data-driven approach, and the fusion approach. A short review on various methods for Li-ion battery prognostics and health monitoring can be found in [12]. A discussion of SoH estimation methods categorized under these approaches can be found in [13]. PoF-based approach takes into account the physics-of-failure of a component or system. Ramadass et al., Ning et al., and Zhang and White [14–16] proposed different PoF-based techniques for modeling capacity fade and cycle life of Li-ion batteries. However, physical models are usually presented in the form of partial differential equations with many unknown parameters. They are accurate but not desirable in online applications due to a high requirement for memory and computation.

In contrast, data-driven approach does not require product-specific knowledge. Statistical techniques and/or machine learning algorithms are used to detect parameter changes, isolate faults and estimate RUL for failure diagnosis and prognosis [17]. He et al. [4] proposed a SoH and RUL estimation method using Dempster–Shafer theory and Bayesian update with Monte Carlo approach. Pattipati et al. [1] proposed a data-driven battery management system (BMS) that used support vector machine (SVM) for both SoH and SoC estimation; and support vector regression (SVR) for RUL prediction. Xing et al. [18] predicted RUL with a combination of exponential and polynomial regression followed by particle filtering (PF). Data-driven techniques are flexible and can be applied to problems with a similar format, even when the underlying physics are different. In recent years, techniques that utilize PoF-based models with data-driven methods for parameter estimation are sometimes referred to as the fusion approach. The relevance vector machine with particle filtering (RVM–PF) framework for battery RUL prediction proposed by Saha et al. [19] falls into this category.

A common problem in the existing data-driven models is that modeling battery degradation was based on the capacity data, which was collected at a constant discharge rate and a constant ambient temperature. In practice, the battery is operated in a varying operational condition. Uncertainties will make these models underperform. Although the PF-based models have paid more attention to address the uncertainties, the degradation data have still been employed under a constant environmental condition, such as Saha et al. [19] and Xing et al. [18] model. The fact that more uncertainties are introduced can reduce both the performance and the computational efficiency of the PF-based methods. Therefore, it is important to model the battery degradation taking into account different operational conditions including different usage conditions and ambient temperatures. Goebel et al. [10] compared the capacity estimation and RUL prediction over time of three methods qualitatively. Xing et al. [20] studied the influences of ambient temperature on SoC estimation. The authors are not aware of any studies on the robustness of RUL prediction over different operations conditions of different batteries. This paper is a quantitative study on the robustness of RUL prediction over time for the same battery and for different batteries under different constant operations conditions. The purpose procedure is to develop an effective battery degradation model considering both various operational conditions and the feasibility for real application. It will help a better planning of battery replacement schedule, a proper maintenance strategy, and an early precaution to protect against battery failure.

In this paper, the problem of Li-ion battery SoH estimation is formulated such that a data-driven classification method, the naive Bayes (NB), can be applied directly for RUL prediction. Their RUL prediction accuracy and robustness against constant discharging at different operating conditions are evaluated. Section 2 is a brief background on naive Bayes. Section 3 introduces the proposed prognostics procedure and Section 4 shows the empirical analysis with publicly available Li-ion battery data. This is followed by a discussion of the results in Section 4 and the conclusion in Section 5.

2. Modeling the capacity depletion

The NB used in the proposed algorithm belongs to the data-driven approach. NB is a popular data mining algorithm [21] known for its empirical performance.

NB model is the simplest form of a Bayesian network [22,23]. It calculates the probabilities of an observation belonging to a particular class according to the Bayes’ theorem, by assuming that each predictor is conditionally independent of every other predictor [24]. An NB normally assigns an observation to the class with the largest probability, but it has also been applied to regression problems [25] in which the probabilities of each class can be combined differently. It is well-known for its competitive performance in real-world applications, even though the conditional independence assumption is rarely valid. Its strengths include its simplicity, efficiency, and robustness to noise and missing data. It also permits the use of more than two classes.

2.1. A short review on the naive Bayes literature

The naive Bayes has a long history in the literature, sometimes under various names in earlier articles [26]. A recent summary of the technique can be found in [21].

It is well documented in the literature that the NB often performs surprisingly well and outperforms more sophisticated algorithms in classification, even at times where the features are clearly not independent [27]. This has motivated three kinds of research [28]: (1) improve the algorithm by relaxing the independence assumption; (2) modify the feature set to comply with the independence assumption as much as possible; and (3) explain why the independence assumption is not necessary. Quantifying the degree of dependence which the NB can tolerate was started by Ref. [29]. Identifying the NB’s true region of optimal performance was started by Ref. [27]. A flexible tree-augmented naive Bayes using kernel density estimation was proposed by Ref. [23]. More recently, a “non-parametric” naive Bayes was proposed by Ref. [30], motivated by surprisingly well performance of the NB on a medical data set.

There has been continued interest in explaining and quantifying the conditions for good NB performance for practical application [22,26,27,31]. One obvious reason for good classification performance is the insensitivity of zero-one loss. This refers to a model’s ability to assign the correct class to a test observation as long as the probability of belonging to this class is greater than that of other class(es), even though the probability estimates may not be accurate. However, this property applies to all classification algorithms in general and does not explain why the NB performs better than more complicated classification algorithms such as C4.5 decision tree, SVM, and neural networks on real data sets. Another reason for good performance unique to the NB is that the conditional independence is only a sufficient but not a
necessary condition for its well classification performance. This has led to the important recognition that the NB has a much greater range of applicability than the conditional independent case. In particular, [22] pointed out that the conditional independence assumption can be replaced by the weaker linked independence assumption suggested by Ref. [32]. Zhang further hypothesized that the dependency distribution is more important than the magnitude of the dependencies itself, and illustrated that for the two-class case, the NB probability estimates are correct in three cases: (1) when the attributes are independent, (2) when the dependence of each attribute distributes evenly in both classes, and (3) when the dependence of an attribute supporting a class is canceled out by that of the other class.

Although the capacity readings of Li-ion batteries over cycles are not statistically independent, the dependence is likely to fall in case 2 or 3 and the NB is likely to perform reasonably well.

2.2. NB Capacity depletion model

Let $\mathbf{C}$ be the $n \times d$ matrix of capacity over time (features) used in training, and $\mathbf{Y}$ be the $n$ vector of corresponding battery life (classes) assigned to each battery (observation), where $n$ is the number of training observations and $d$ the number of features. A classification algorithm, or classifier, can be viewed as a set of discriminant functions $g_i(c)$, one for each possible class $y$ in the finite set $\mathcal{Y}$, which are to be computed for each observation so that the observation $\mathbf{b}$ is assigned to the one class whose function value is maximum [23], i.e., $b_y = m$ if $g_m(c) > g_y(c) \ \forall y \neq m$. This can be alternatively expressed as

$$\hat{b}_y = \underset{y \in \mathcal{Y}}{\arg\max} g_i(c), \ \forall y \in \mathcal{Y}. \quad (1)$$

A Bayes model makes use of the Bayes' theorem as the discriminant function, together with the probabilities of different values of each feature and each class in the training data set, i.e.,

$$g_i(c) = P(y|c) = \frac{P(y)P(c|y)}{P(c)}. \quad (2)$$

Instead of the probability calculation $P(c|y) = P(c_1|y)P(c_2|y, c_1)P(c_3|y, c_1, c_2) \ldots P(c_d|y, c_1, \ldots, c_{d-1})$ for the general case, the naive Bayes assumes conditional independence on the features and employs the simplified probability calculation

$$P(c|y) = \prod_{j=1}^{d} P(c_j|y), \quad (3)$$

such that

$$g_i(c) = P(y|c) = \frac{P(y)\prod_{j=1}^{d} P(c_j|y)}{\sum_{y'} P(y')\prod_{j=1}^{d} P(c_j|y')}]. \quad (4)$$

Thus

$$\hat{b}_y = \underset{y \in \mathcal{Y}}{\arg\max} \left[ \frac{P(y)\prod_{j=1}^{d} P(c_j|y)}{\sum_{y'} P(y')\prod_{j=1}^{d} P(c_j|y')} \right], \quad (5)$$

where the common denominator can be omitted without affecting the classification results [33], i.e.,

$$\hat{b}_y = \underset{y \in \mathcal{Y}}{\arg\max} P(y|c) = \underset{y \in \mathcal{Y}}{\arg\max} \left[ \frac{P(y)\prod_{j=1}^{d} P(c_j|y)}{\sum_{y'} P(y')\prod_{j=1}^{d} P(c_j|y')} \right] \quad (6)$$

In contrast to the common practice of choosing the most probable class in classification problems, the probability distribution can be utilized when the NB is applied to a regression problem. The classes $y$ are the numeric battery life in this context, and the probabilities of each class from each trained model are combined to form a probability estimate

$$\hat{y}_y = \sum_{y \in \mathcal{Y}} y P(y|c) \quad (7)$$

The word ‘Bayes’ in the naive Bayes refers to the use of the Bayes’ theorem to indirectly estimate $P(c|y)$ from $P(c|y)$. The NB is not a ‘Bayesian’ method and a frequentist interpretation is adopted.

3. RUL prediction algorithm

In contrast to the usual practice of using capacity as the response variable in Li-ion battery prognostics, in our formulation the actual battery life in number of cycles is used directly in a non-parametric regression. We utilize the strength of the NB for the RUL to EOL prediction of Li-ion batteries at different times. The resulting NB predictions are not limited to the set of response values in the training set. A similar algorithm has been used for bearing-degradation prognosis with vibration data [34]. This algorithm can also be applied to the RUL to EOD problem using the data points from one cycle as one observation. The proposed algorithm is summarized as a flowchart in Fig. 1, where notations are explained in the below paragraphs.

3.1. Initialization

Let $\mathbf{C}$ be a set of $n$ input vectors of Li-ion battery capacity from the first cycle to a pre-specified EOL threshold, and let $t_{\text{EOL}}$ be the $n$-vector of actual battery life, i.e., $t_{\text{EOL}}(i)$ is the number of cycles to EOL for battery $i$. The $n \times d$ matrix $\mathbf{C}$ of predictors used in training has the elements

$$c_{ij} = \hat{c}_{ij}, \quad i = 1 \ldots n; \quad j = 1 \ldots t_{\text{EOL}} \quad (8)$$

where $d = \max (t_{\text{EOL}})$, i.e. the maximum number of cycles to EOL in the training set, and NA denotes missing data. The corresponding $n \times 1$ vector $Y$ of responses is

$$y_i = t_{\text{EOL}}(i), \quad i = 1 \ldots n. \quad (9)$$

3.2. NB model training

Let $\mathcal{C}$ denote the finite set of ranges obtained from each $n$-vector $C_i$ using Fayyad and Irani’s MDL method of supervised discretization [35] and $\mathcal{Y} = \{t_{\text{EOL}}\}$ denote the finite set of classes (responses) in the training set. If the number of classes is large, they can also be discretized into ranges [23].

The training data is divided into $F$ subsets, for $F$-fold cross-validation (CV). In each fold, an NB model calculates, with $F - 1$ folds of data, the probabilities $P(y|c)$, $P(c)$, and $P(y)$ for all $c \in \mathcal{C}$ and $y \in \mathcal{Y}$ as in Section 2.1. Any missing data are ignored in this training process. Typically, a 10-fold CV is used, i.e. $F = 10$. If the number of training observations is less than 10, a leave-one-out cross validation is used, i.e. $F = n$. As NB requires no parameter tuning, the validation sets in the CV are not used. This results in $F$ approximations of each of the probabilities and these probabilities approximations are used in the online prediction for each new observation.

3.3. RUL prediction

For a new observation $c_0$, $F$ approximations of the probabilities of belonging to each class $y \in \mathcal{Y}$ are calculated from each of the $F$ NB models obtained by cross validation

$$P(y|c_k) = \frac{P(y)\prod_{j=1}^{d} P(c_j|y)}{\sum_{m} P(m)\prod_{j=1}^{d} P(c_j|m)}. \quad (10)$$
The classes $y$ are the numeric battery life in this context, and the probabilities of each class from each trained model are combined to form a probability estimate

$$b_f = \frac{1}{F} \sum_{m=1}^{F} P(y | c_m) \quad f = 1 \ldots F,$$

The $F$ estimates of $\hat{Y}$ from all $F$ folds are averaged to form the NB’s final prediction for observation $k$

$$\hat{\mu}_k = \frac{1}{F} \sum_{f=1}^{F} y_f \quad f = 1 \ldots F.$$

The 5th and 95th percentiles are taken from the sorted list of $y_f$ estimates in the $F$ folds by interpolation. At prediction time $t$, $c_k$ contains the capacity of cycles up to $t$ and missing from element $t + 1$ onwards, i.e., $c_{k \leq t}, \ldots, c_{k F} = \text{NA}$. The NB handles any missing values in a predictor by assigning a probability of one to all classes, i.e., $P(\text{NA} | y) = 1$. This has no effect on the multiplication of equation (10), hence the resulting probabilities belong to each class. This implies that historical capacity readings at later times are not used for the prediction at earlier times. This is statistically valid according to the conditional independence assumption.

To better identify the predicted life at different time-points, we denote the predicted life of observation $k$ at time $t$ (obtained with $c_k$, $c_{k+1}$, ..., $c_{k F}$) by $\hat{\mu}_k(t)$. Then, the RUL to EOL of battery $k$ predicted at time $t$ is simply

$$RUL_k(t) = \max[\hat{\mu}_k(t) - t, 0],$$

and $t$ can similarly be subtracted from the 5th and 95th percentiles of $\hat{\mu}_k(t)$ to obtain the percentiles of $RUL_k(t)$.

This method is applied to experimental battery data available online and the prediction results of the NB are compared with those of the SVM. More details of the SVM formulation used can be found in Section 4.2.

4. Experimental results

4.1. The data set

In this case study we use the NASA Ames Li-ion battery cycle-life test data [36], as in papers including [10,19,37,38]. In a battery prognostics testbed, 18,650 sized Li-ion batteries were run through three operational profiles: charge, discharge, and impedance. Repeated charge and discharge cycles accelerated aging while impedance measurements were used to keep track of changes in the battery’s internal parameters. Charging was performed using a 1.5 A constant current (CC) until the battery voltage of 4.2 V was reached, then in a constant voltage (CV) mode until the current dropped to 20 mA. Discharging was carried out at two constant levels of current (1 A, 2 A) and ambient temperature (4 °C, 24 °C) until four levels of discharge voltage (2 V; 2.2 V; 2.5 V; 2.7 V) [36]. Data collection for the batteries terminated once the full charge capacity (FCC) reached the EOL threshold at 30% below the rated capacity, according to the description. The experimental parameters are summarized in Table 1.

The capacity depletion data are plotted in Fig. 2. Although capacity is affected by the depth of load profiles and the ambient temperature, the effects of these parameters on capacity loss are not modeled in this case study. These parameter values are kept hidden from the NB to validate its robustness in this context. The
capacity values of zero are treated as missing data, which are ignored by the NB and replaced with mean value by the SVM. In both cases, the missing data should have no effect on the model training.

Nine observations are used to train the NB and three separate observations for testing according to the methodology described in Section 3, with stratification by experimental parameters ambient temperature and discharge current. In other words, the four battery observations in each pair of ambient temperature and discharge current setting are randomly assigned such that each time three are used for training and one is used for RUL prediction testing. To utilize the dataset, the jackknifing resampling technique (Fig. 3) is used to obtain RUL predictions for each of the battery observations from a separate training set. Thus each battery observation is used exactly once in testing and its RUL predictions are obtained from a NB model trained from nine other batteries in the data set (Table 2). For each test observation, $k = 1, 2$, a $\hat{c}_k(t)$ and hence $\hat{RUL}(t)$ are calculated for each time $t = 1 \ldots t^E_{\text{EOL}}$, using the capacity up to prediction time $t$ (and $C_d t^E_{\text{EOL}} = \text{NA}$). As time progressed, more predictors are used as input for the NB prediction. The predicted battery life $\hat{c}_k(t)$ is compared to the actual battery life $t^E_{\text{EOL}}$, i.e., time to EOL, for battery $k$. The root mean squared error (RMSE) for NB is calculated.

4.2. Performance comparison

The process is repeated by replacing in Section 3.2 with the SVM [24,25], another popular data-driven method [21] known for its empirical performance. SVM is used for comparison with our proposed method due to its popularity in PHM related research, including but not limited to Li-ion batteries. Recent examples of applying SVM in RUL prediction in the literature include [1,39,40].

SVM is a kernel method first introduced in [41], which builds a model by constructing a hyperplane that best separates two classes. Although a SVM model depends on only a subset of the training data, it is well-known for its good performance in the classification of high-dimensional data. The SVM is built using sequential minimal optimization (SMO) [42], with a polynomial kernel. A common choice of kernel parameters and the penalty margin $C$ are selected by cross-validation. However, it does not naturally handle multiple classes or generate proper probability

### Table 1
Discharging parameters of the battery data.

<table>
<thead>
<tr>
<th>Battery ID</th>
<th>Temperature (°C)</th>
<th>Discharge current (A)</th>
<th>End-of-discharge voltage (V)</th>
</tr>
</thead>
<tbody>
<tr>
<td>B0005</td>
<td>24</td>
<td>2</td>
<td>2.7</td>
</tr>
<tr>
<td>B0006;</td>
<td>24</td>
<td>2</td>
<td>2.5</td>
</tr>
<tr>
<td>B0018</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>B0007</td>
<td></td>
<td></td>
<td>2.2</td>
</tr>
<tr>
<td>B0045</td>
<td>4</td>
<td>1</td>
<td>2.2</td>
</tr>
<tr>
<td>B0046</td>
<td>4</td>
<td>1</td>
<td>2.2</td>
</tr>
<tr>
<td>B0047</td>
<td></td>
<td></td>
<td>2.5</td>
</tr>
<tr>
<td>B0048</td>
<td></td>
<td></td>
<td>2.7</td>
</tr>
<tr>
<td>B0053</td>
<td></td>
<td>2</td>
<td>2.2</td>
</tr>
<tr>
<td>B0054</td>
<td>4</td>
<td>2</td>
<td>2.2</td>
</tr>
<tr>
<td>B0055</td>
<td></td>
<td>2</td>
<td>2.5</td>
</tr>
<tr>
<td>B0056</td>
<td></td>
<td></td>
<td>2.7</td>
</tr>
</tbody>
</table>

### Table 2
Jackknifing of the battery data.

<table>
<thead>
<tr>
<th>Battery ID</th>
<th>Jackknifing</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1</td>
</tr>
<tr>
<td>B0005</td>
<td>Test</td>
</tr>
<tr>
<td>B0006</td>
<td>Train</td>
</tr>
<tr>
<td>B0007</td>
<td>Train</td>
</tr>
<tr>
<td>B0018</td>
<td>Train</td>
</tr>
<tr>
<td>B0045</td>
<td>Test</td>
</tr>
<tr>
<td>B0046</td>
<td>Train</td>
</tr>
<tr>
<td>B0047</td>
<td>Train</td>
</tr>
<tr>
<td>B0048</td>
<td>Train</td>
</tr>
<tr>
<td>B0053</td>
<td>Train</td>
</tr>
<tr>
<td>B0054</td>
<td>Train</td>
</tr>
<tr>
<td>B0055</td>
<td>Train</td>
</tr>
<tr>
<td>B0056</td>
<td>Test</td>
</tr>
</tbody>
</table>

Fig. 2. Plots of the data – capacity over time.

Fig. 3. Illustration of a statistical resampling technique – jackknifing.
estimates. In this paper, multiple classes are handled by pairwise classification (1-vs-1) and proper probability estimates are obtained by fitting logistic models to the SVM outputs and Hastie and Tibshirani’s pairwise coupling [43], which estimates a Gaussian probability density function for each class in the pairwise model.

In the formal statistical sense, both the NB and the SVM do not belong to techniques of the ‘Bayesian’ approach. While historical data are used to train the models, no notion of subjective probability is introduced thus these two methods do not require updating of priors. An example of a ‘Bayesian’ technique is the relevance vector machine (RVM) [12].

4.3. Results – prediction accuracy

Ideally, the predicted battery life should be the same as the actual battery life at all times, i.e. \( \hat{\mu}_k(t) = t^\text{EOL}_k, \quad \forall t = 1 \ldots t^\text{EOL}_k \). The better a method, the closer to zero is the error and squared error. The root mean squared error (RMSE) is a measure of accuracy that penalizes predictions that deviate from the actual value and it is zero for a perfect method. The NB and the SVM each produced \( t^\text{EOL}_k \) predictions for battery \( k \). To ensure that each testing observation contributed the same weight to RMSE irrespective of the number of capacity readings, the RMSE was first calculated for each \( k \) over all \( t^\text{EOL}_k \) predictions then averaged over all \( k \) testing observations

\[
\frac{1}{k} \sum_{k} \frac{1}{t^\text{EOL}_k} \sum_{t=1}^{t^\text{EOL}_k} (RUL_k(t) - \hat{\mu}_k(t))^2 ,
\]

which is equivalent to

\[
\frac{1}{k} \sum_{k} \frac{1}{t^\text{EOL}_k} \sum_{t=1}^{t^\text{EOL}_k} (\hat{\mu}_k(t) - t^\text{EOL}_k)^2 ,
\]

where \( k = 1 \ldots 2 \) and \( t = 1 \ldots t^\text{EOL}_k \).

Table 3 shows the RMSEs for the NB and the SVM for each test battery and the average. The smaller RMSE generated from better prediction performance are highlighted in bold for each testing battery. Of the 12 test batteries, the NB predictions give smaller RMSE hence better performance for 10 batteries, same RMSE as the SVM predictions for one battery and larger RMSE for one battery. The average RMSE of the NB is 16.1 cycles (0.17%), with standard deviation of 10.7 cycles (0.15%), much better than the average 26.5 cycles (0.27%) and standard deviation 13.2 cycles (0.20%) of the SVM.

Consider the battery observations with the best NB performance, namely B0007; B0046; B0047. Although the number of cycles for B0007 (168 cycles) is more than twice that of B0046 (72 cycles), the two NB RMSEs, 4.6 and 3.1 cycles (0.03% and 0.04%), are close to each other. The RMSEs for the SVM, 31.1 and 6.6 cycles (0.19% and 0.09%), are much further apart and the RMSE of the longer B0007 is double that of B0046. The only test battery with smaller RMSE from the SVM than the NB is B0045. This corresponds to the battery data with small capacity values throughout (in Fig. 2) and may be considered as an outlier. Also note that all testing batteries where SVM performed well, namely B0045–B0048, come from the same group of experimental setting.

Table 3

<table>
<thead>
<tr>
<th>No. of cycles</th>
<th>%</th>
<th>No. of cycles</th>
<th>%</th>
</tr>
</thead>
<tbody>
<tr>
<td>B0005</td>
<td>14.9</td>
<td>0.09</td>
<td>35.5</td>
</tr>
<tr>
<td>B0006</td>
<td>16.8</td>
<td>0.10</td>
<td>32.0</td>
</tr>
<tr>
<td>B0007</td>
<td>4.6</td>
<td>0.03</td>
<td>31.1</td>
</tr>
<tr>
<td>B0018</td>
<td>35.6</td>
<td>0.27</td>
<td>35.6</td>
</tr>
<tr>
<td>B0045</td>
<td>21.1</td>
<td>0.29</td>
<td>11.7</td>
</tr>
<tr>
<td>B0046</td>
<td>3.1</td>
<td>0.04</td>
<td>6.6</td>
</tr>
<tr>
<td>B0047</td>
<td>3.3</td>
<td>0.05</td>
<td>7.4</td>
</tr>
<tr>
<td>B0048</td>
<td>6.7</td>
<td>0.09</td>
<td>13.9</td>
</tr>
<tr>
<td>B0053</td>
<td>30.7</td>
<td>0.56</td>
<td>46.9</td>
</tr>
<tr>
<td>B0054</td>
<td>21.6</td>
<td>0.21</td>
<td>32.9</td>
</tr>
<tr>
<td>B0055</td>
<td>12.1</td>
<td>0.12</td>
<td>27.7</td>
</tr>
<tr>
<td>B0056</td>
<td>22.2</td>
<td>0.22</td>
<td>36.7</td>
</tr>
<tr>
<td><strong>RMSE</strong></td>
<td>16.1</td>
<td>0.17</td>
<td>26.5</td>
</tr>
<tr>
<td><strong>s</strong></td>
<td>10.7</td>
<td>0.15</td>
<td>13.2</td>
</tr>
</tbody>
</table>

Fig. 4. Plot of the NB (top) and SVM (bottom) predictions of B0046 in evaluation data set.
(ambient temperature 4 °C; discharge current 1 A), indicating that the SVM is more susceptible to changes in operational parameters and a different SVM model should be trained for each ambient temperature and discharge current pair for good prediction performance in this case study.

4.4. Results – predicted battery life over time

The predicted battery life over time for B0046, B0047 and B0007, the three best performing test batteries (least RMSE) for the NB, are plotted in Figs. 4–6. The first two batteries are also the best performing test batteries for the SVM. Ideally, the predicted battery life should be the same as the actual battery life at all times, i.e. \( \mu_k(t) = e_{\text{EOL}}^k \), \( \forall t = 1 \ldots t_{\text{EOL}}^k \). The better a method, the closer the \( \mu_k(t) \) and the percentiles are to the dotted horizontal line \( e_{\text{EOL}}^k \). Note that the predictions are plotted against the actual battery life, i.e. a constant, not the capacity.

The upper plots in Figs. 4–6 show the NB predicted battery life \( \hat{\mu}_k(t) \) over time and the same information for the SVM predictions are shown at the bottom plots. The predictions from both methods
move towards the actual battery life as more historical capacity data became available then stabilized around $t_{50\%}$. Whereas the NB predictions stabilize in less than 10% of the battery life for the three batteries, the SVM predictions deviated a lot more and only stabilize after 20% of the battery life for B0046 and B0047. For the B0007 with a more than double life-span, the SVM predictions deviated a lot more and only stabilize after 20% of the battery life.

### 4.5. Results – RUL over time

The RUL is calculated from $t_{50\%}$ by Eq. (13). Table 4 summarizes the RUL of the battery life for B0046 and B0047. For the B0007 with a more than double life-span, the SVM predictions deviated a lot more and only stabilize at around 50% of the battery life.

#### 4.5.1. Effect of training set size on prediction performance

In this paper, nine observations are used for training and three separate observations are used for testing. It is well known that the size of training set used affects the test performance of a method. In general, when the training set size increases, the prediction size of training set used affects the test performance of a method. It is well known that the curse of dimensionality.

**Table 4** Predicted RUL of evaluation data set at four different times using NB and SVM.

<table>
<thead>
<tr>
<th>Battery ID</th>
<th>5th %</th>
<th>25th %</th>
<th>SVM 5th %</th>
<th>75th %</th>
<th>SVM 75th %</th>
<th>95th %</th>
<th>SVM 95th %</th>
</tr>
</thead>
<tbody>
<tr>
<td>B0005</td>
<td>25%</td>
<td>75%</td>
<td>72.0%</td>
<td>74.7%</td>
<td>75.0%</td>
<td>68.2%</td>
<td>68.9%</td>
</tr>
<tr>
<td>B0006</td>
<td>25%</td>
<td>75%</td>
<td>75.0%</td>
<td>75.0%</td>
<td>75.0%</td>
<td>68.2%</td>
<td>69.1%</td>
</tr>
<tr>
<td>B0007</td>
<td>25%</td>
<td>75%</td>
<td>75.0%</td>
<td>75.0%</td>
<td>75.0%</td>
<td>68.2%</td>
<td>69.1%</td>
</tr>
<tr>
<td>B0018</td>
<td>25%</td>
<td>75%</td>
<td>80.4%</td>
<td>80.4%</td>
<td>80.4%</td>
<td>66.3%</td>
<td>72.2%</td>
</tr>
<tr>
<td>B0045</td>
<td>25%</td>
<td>75%</td>
<td>67.3%</td>
<td>60.7%</td>
<td>60.7%</td>
<td>60.5%</td>
<td>60.7%</td>
</tr>
<tr>
<td>B0046</td>
<td>25%</td>
<td>75%</td>
<td>75.0%</td>
<td>75.0%</td>
<td>75.0%</td>
<td>67.9%</td>
<td>78.6%</td>
</tr>
<tr>
<td>B0047</td>
<td>25%</td>
<td>75%</td>
<td>75.0%</td>
<td>75.0%</td>
<td>75.0%</td>
<td>67.9%</td>
<td>78.6%</td>
</tr>
<tr>
<td>B0048</td>
<td>25%</td>
<td>75%</td>
<td>75.0%</td>
<td>77.2%</td>
<td>77.2%</td>
<td>72.2%</td>
<td>72.2%</td>
</tr>
<tr>
<td>B0053</td>
<td>25%</td>
<td>75%</td>
<td>81.1%</td>
<td>82.9%</td>
<td>82.9%</td>
<td>82.9%</td>
<td>82.9%</td>
</tr>
<tr>
<td>B0054</td>
<td>25%</td>
<td>75%</td>
<td>50.0%</td>
<td>50.0%</td>
<td>50.0%</td>
<td>50.0%</td>
<td>50.0%</td>
</tr>
<tr>
<td>B0055</td>
<td>25%</td>
<td>75%</td>
<td>63.9%</td>
<td>71.1%</td>
<td>74.0%</td>
<td>72.5%</td>
<td>72.5%</td>
</tr>
<tr>
<td>B0056</td>
<td>25%</td>
<td>75%</td>
<td>63.4%</td>
<td>73.7%</td>
<td>74.5%</td>
<td>72.7%</td>
<td>72.7%</td>
</tr>
</tbody>
</table>

In this paper, nine observations are used for training and three separate observations are used for testing. It is well known that the size of training set used affects the test performance of a method. In general, when the training set size increases, the prediction
accuracy on the training set increases. The testing prediction accuracy also increases, until the noises in the training set are also modelled (overfitting). After this point, the testing prediction accuracy decreases with increase in training set size, although the training prediction accuracy continues to increase. Fig. 7 shows the testing performance of the two methods when different sizes of training set are used. The jackknifing resampling technique is again used with the proposed method to obtain RUL predictions of each of the battery data from a separate training set. The average RMSE of the NB is lower than that of the SVM for training set sizes of three or more. For this data set, the average RMSE of the NB is only worse than that of SVM at training size of two, which may be due to the use of an outlier for training combined with a small training size. Intuitively, when a particular loading condition is not included in the training data (such as in training sizes one and two), the prediction performance depends on how well the training data represents the population as a whole. The average RMSE at training size two shows the SVM’s better generalization property. However, when the training data is a proper representation of the battery life cycle (training size one, three and more), the NB outperforms in terms of average RMSE. In particular, the NB can still work even the exact loading condition may not exist in the training data (the case of training size one).

To summarize, the NB predictions result in smaller RMSEs than the SVM for most of the test batteries and also on average (Table 3). The prediction performance over time for the NB is closer to the target than that of the SVM. The NB-predicted battery life, and hence the estimated RUL, is on average 10 cycles closer to the actual value than the SVM-predicted battery life. These results show the robustness of the NB method at different times and with different batteries in this data set. Although the NB produces close-to-actual battery life predictions in less than 10% of total battery life (Figs. 4–6), the target value for some batteries has never been reached (Table 4). This can be due to the bias introduced in using probability estimates instead of zero-one loss function of classification, deviations from ideal dependency distribution in variables, or the adverse effect of the curse of dimensionality. One the other hand, while the SVM method eventually produces the correct prediction for each of the test batteries, the predictions fluctuate in the early stages and stabilize at different cycles for different batteries. Overall, the prediction performance of the NB is more stable and practical than that of the SVM in this case study.

In this proposed methodology, the features used are the full-charge capacity of the battery over different cycles and the response is the life-span (No. of cycles) of the battery. Although the capacity readings are not statistically independent, the dependence is likely to fall in case 2 or 3 in the appendix and the NB is likely to perform reasonably well. However, the probability estimates of the NB are utilized in the RUL prediction so bias introduced by any deviations from the linked independence assumption is reflected in the accuracy of the final RUL predictions.

The results of this case study also suggested that, while it is well known that the environmental conditions have big impact on the degradation trend, it is the changes in operating conditions of a Li-ion battery over cycle life that makes the Li-ion battery degradation and RUL prediction even more difficult. This paper considers relatively constant operating conditions over the cycle life of a particular battery. The situation of varying operating conditions over cycle life is out of the scope of this paper. Further work includes robustness study with consideration of varying operating conditions over the cycle life of a particular battery. The constant discharge can be replaced with dynamic tests, and constant ambient temperature can be replaced with seasonal change.

5. Conclusion

This paper studies the modeling of battery degradation under different usage conditions and ambient temperatures, which is seldom considered in the literature. We have proposed a method using naive Bayes for robust prediction of Li-ion battery RUL over discharging cycles in different operating conditions. We have utilized available classification techniques in our algorithm and the proposed RUL prediction method is simple and efficient. Stable and accurate RUL predictions over time are obtained for different batteries under different operating conditions. Publicly available Li-ion battery data are used, RUL has been calculated at four time intervals and RMSEs are calculated for performance evaluation. We have shown that the NB generates competitive prediction performance over the SVM, even though the batteries under different operating conditions are considered together in the modeling.

Acknowledgement

This research was supported in part by RGC CRF #CityU8/CRF/09 and RGC CRF #CityU121410.

References


[37] Zhang J. Health monitoring and prognostics of Li-ion battery. University of Cincinnati, 2010.


