

Prediction of Lithium-Ion Battery's Remaining Useful Life Based on Relevance Vector Machine

Zhiyun Zhang, Miaohua Huang, Yupu Chen, and Shuanglong Zhu
Wuhan University of Technology

ABSTRACT

In the field of Electric Vehicle (EV), what the driver is most concerned with is that whether the value of the battery's capacity is less than the failure threshold because of the degradation. And the failure threshold means instability of the battery, which is of great danger for drives and passengers. So the capacity is an important indicator to monitor the state of health (SOH) of the battery. In laboratory environment, standard performance tests can be carried out to collect a number of related data, which are available for regression prediction in practical application, such as the on-board battery pack.

Firstly, we make use of the NASA battery data set to form the observed data sequence for regression prediction. And a practical method is proposed to determine the minimum embedding dimension and get the recurrence formula, with which a capacity model is built. Afterwards, an optimized Relevance Vector Machine (RVM) algorithm is utilized to improve the prediction performance as well as the operating efficiency and get a tradeoff between training time and computational complexity. Finally, given the conditional probability distributions of hyper-parameters, we can capture the uncertainty in our prediction of the capacity. The calculation results make it clear that the algorithm we develop have better accuracy and performance than others.

CITATION: Zhang, Z., Huang, M., Chen, Y., and Zhu, S., "Prediction of Lithium-Ion Battery's Remaining Useful Life Based on Relevance Vector Machine," *SAE Int. J. Alt. Power.* 5(1):2016, doi:10.4271/2015-01-9147.

INTRODUCTION

As EV users, we are always worried about the driving range as soon as we start to drive the vehicle, no matter how much energy of the on-board battery pack is available, which is an obvious disadvantage. However, what impress us most are the low cost and zero emission of the EV. A battery unit is series or parallel connected in several sets, so ideally, the on-board battery pack may be regarded as an ordinary battery with larger capacity. And it is of great importance to have an accurate prediction of the capacity of an individual cell. As times goes on, the battery experiences aging and degradation, which leads to the limited remaining useful life (RUL).

As the most important indicator of battery's SOH, a reliable prediction of capacity can provide a better reference for analysis, which not only ensure the battery's safe operating condition, but also make it helpful for the optimization of battery management system (BMS) to maximize the RUL. Different kinds of algorithms have been developed to predict the accurate capacity, but the predicted value seems to be unbelievable without taking the uncertainty of the prediction into consideration. Some scientific literatures have mentioned how to predict the capacity with various algorithms. And these algorithms may be grouped into two categories: model-based prognostics and data-driven prognostics.

The key to carry out the model-based prognostics is that we should figure out the model, which can describe the complex electrochemical process in mathematical expressions. And the model is always non-linear, where analytical solutions are difficult to be explicit. Lumped-element equivalent-circuit components such as resistors and capacitors are used to represent the behavior of a battery cell [1]. With the development of the measurement technology and reduced cost, Electrochemical Impedance Spectroscopy (EIS), a non-invasive method, has been used to characterize the battery capacity degradation through variations of the internal parameters. In other words, battery life can be determined and predicted from the impedance changes [2]. In this work, an empirical model based on the physical degradation behavior of lithium-ion batteries is developed. Model parameters are initialized by combining sets of training data based on DST (Dempster-Shafer Theory). BMC (Bayesian Monte Carlo) is then used to update the model parameters and predict the RUL based on available data through battery capacity monitoring [3].

For data-driven prognostics the battery is taken as a black box without giving too much insight into complex electrochemical process. We just need to find out the statistical rules with the measured data to predict the capacity or the RUL. And the prediction accuracy is mainly dependent on the algorithms or the related parameters we choose. Furthermore, the result can help us have a better understanding of the degradation and verify the statistical rules

we propose. A novel fusion prognostic framework is proposed, in which the data-driven time series prediction model is adopted as observation equation, and combined to PF (Particle Filter) algorithm for lithium-ion battery cycle life prediction [4]. The Gaussian process model is presented in [5] to realize the prognostics for battery health. Because of the advantages of flexible, probabilistic, nonparametric model with uncertainty predictions, the Gaussian process model can provide variance around its mean predictions to describe associated uncertainty in the evaluation and prediction. As stated by [6], models for battery SOH monitoring based on SVR-PF (Support Vector Regression- Particle Filter) are developed with novel capacity degradation parameters introduced to determine battery health in real time. Moreover, the RUL prediction model is proposed, which is able to provide the RUL value and update the RUL probability distribution to the End-of-Life cycle. In [7], an ensemble model is developed to characterize the capacity degradation and predict the remaining useful performance (RUP) of lithium-ion batteries. The model fuses an empirical exponential and a polynomial regression model to track the battery's degradation trend over its cycle life based on experimental data analysis. Model parameters are adjusted online using a particle filtering (PF) approach. Auto-regressive moving average, fuzzy logic, neural network, support vector machine, and nonlinear filtering methods are combined together to predict the remaining useful life of batteries [8, 9, 10].

However, the papers mentioned above have an obvious disadvantage that we have to spend much time in training collected data to achieve high prediction accuracy. What's more, we choose parameters of the recurrence equation in a subjective way, which may not offer good generalization performance. In this paper, we utilize the data-driven methods to make prediction with the optimized RVM algorithm, which not only improve performance of calculating the mean predicted value with the variance, but also decrease the training time and computational complexity. And the uncertainty of the predicted value is captured and described by the probability distribution function. We set the confidence to 95% to determine the confidence interval, providing a more convincing prediction.

The rest of the paper is organized as followed. Section 2 introduces the concept of the Relevance Vector Machine (RVM) algorithm. In Section 3, a practical method is proposed to determine the minimum embedding dimension and get the recurrence formula, with which a capacity model is built for the prediction. Then, we describe the optimized RVM algorithm in detail and verify the algorithm based on NASA battery data in Section 4 before making the conclusion in Section 5.

INTRODUCTION OF RELEVANCE VECTOR MACHINE FOR REGRESSION

Consider a data set of input-output pairs $\{\mathbf{x}_j, y_j\}_{j=1}^N$, and the mathematic relation between them is described as:

$$\begin{cases} y_j = f(\mathbf{x}_j) + \varepsilon_j \\ \varepsilon_j \sim N(0, \sigma^2) \end{cases} (j = 1, 2, 3, \dots, N) \quad (1)$$

$\{\varepsilon_j\}_{j=1}^N$ are independent samples from noise process, which is taken as mean-zero Gaussian with variance σ^2 . Then the conditional probability distribution of the output target is achieved as:

$$p(y_j | \mathbf{x}_j) = N(y_j | f(\mathbf{x}_j), \sigma^2) \quad (2)$$

We always wish to learn a model to reach the objective of making accurate prediction for unknown value of $\mathbf{x}$ based on the training of inputs. And the key is to figure out the function form of $f(\cdot)$ defined over the input space. A flexible and popular set of candidates is to utilize the kernel function to define one basis function for every element of the input-output pair [11], which is obtained as:

$$f_j(\cdot) = \sum_{i=1}^N \omega_i K(\mathbf{x}_j, \mathbf{x}_i) \quad (3)$$

Then, the regression model can be expressed as:

$$\mathbf{y} = \Phi \mathbf{w} + \varepsilon \quad (4)$$

where

$$\begin{cases} \mathbf{y} = [y_1 & y_2 & \dots & y_N]^T \\ \Phi = [\boldsymbol{\varphi}_1 & \boldsymbol{\varphi}_2 & \dots & \boldsymbol{\varphi}_N]^T \\ \mathbf{w} = [\omega_1 & \omega_2 & \dots & \omega_N]^T \\ \varepsilon = [\varepsilon_1 & \varepsilon_2 & \dots & \varepsilon_N]^T \\ \boldsymbol{\varphi}_j = [K(\mathbf{x}_j, \mathbf{x}_1) & \dots & K(\mathbf{x}_j, \mathbf{x}_N)]^T \end{cases} \quad (5)$$

Spare Bayesian Learning Framework

As stated by Equation (2), the likelihood function of all the output samples with the assumption that they are also independent with each other can be written as:

$$p(\mathbf{y} | \mathbf{w}, \sigma^2) = (2\pi\sigma^2)^{-N/2} \exp\left(-\frac{1}{2\sigma^2} \|\mathbf{y} - \Phi \mathbf{w}\|^2\right) \quad (6)$$

From the perspective of likelihood probability, we can get the maximum-likelihood estimation of $\mathbf{w}$ and σ^2 by calculating the maximum value of $p(\mathbf{y} | \mathbf{w}, \sigma^2)$ in Equation (6). However, the growth of data volume can lead to a severe over-fitting, which may be avoided by putting constraints on the unknown parameters. A Bayesian method is proposed to define a prior probability distribution for the parameters. And a very simple choice of the prior probability distribution is the zero-mean Gaussian, which can guarantee that only a few weights is not zero and the RVM model is sparse. Many distribution functions can be approximated as the zero-mean Gaussian in certain conditions, so the Gaussian distribution is always

the most suitable choice. Although we utilize the distribution of ω_i as prior information, which may not conform to the reality of batteries, lots of references [12, 13, 14] has verified that this can fit the experimental data well.

$$p(\mathbf{w}|\mathbf{a}) = \prod_{j=1}^N N(\omega_j | 0, \alpha_j^{-1}) \quad (7)$$

Hyper-priors should be defined over $\mathbf{a}$ and σ^2 to form the specification of the hierarchical prior. As explained in [15], the suitable priors is Gamma distribution, the parameters of which may be set to small values to ensure that these priors are non-informative, as shown in Equation (8).

$$\left\{ \begin{array}{l} p(\mathbf{a}) = \prod_{j=1}^N \text{Gamma}(\alpha_j | a, b) \\ p(\beta) = \prod_{j=1}^N \text{Gamma}(\beta | c, d) \\ \beta = \sigma^{-2} \\ a = b = c = d = 10^{-4} \end{array} \right. \quad (8)$$

Based on the limitations mentioned above, the hyper-parameter $\mathbf{w}$ should be a sparse matrix, where lots of its elements are close to zero. And the input-output pairs with non-zero weights ω_i are called Relevance Vectors (RVs), which indicate the core and essential character of the original data set.

When a new output sample is observed, its prior probability, i.e., the spare Bayesian learning can be calculated as:

$$p(y_{N+1} | \mathbf{y}) = \iiint p(y_{N+1} | \mathbf{w}, \mathbf{a}, \sigma^2) p(\mathbf{w}, \mathbf{a}, \sigma^2 | \mathbf{y}) d\mathbf{w} d\mathbf{a} d\sigma^2 \quad (9)$$

TRADITIONAL RVM LEARNING ALGORITHM

It seems that we can directly compute the probability in Equation (9) on the left-hand-side. However, the posterior probability $p(\mathbf{w}, \mathbf{a}, \sigma^2 | \mathbf{y})$ is unable to be computed, leading to that we cannot perform full analytically in the computation. So we must look for an effective approximation for $\mathbf{a}$ and σ^2 . We realize the approximation by a delta-function at the most probable values $\mathbf{a}_{MP}$, σ_{MP}^2 . As stated by [11], this approximation does not necessitate that the entire mass of the posterior probability is accurately approximated by the delta-function. And we only want to make the following equation true.

$$\begin{aligned} & \int p(y_{N+1} | \mathbf{a}, \sigma^2) \delta(\mathbf{a}_{MP}, \sigma_{MP}^2) d\mathbf{a} d\sigma^2 \\ & \approx \int p(y_{N+1} | \mathbf{a}, \sigma^2) p(\mathbf{a}, \sigma^2 | \mathbf{y}) d\mathbf{a} d\sigma^2 \end{aligned} \quad (10)$$

Bayesian inference is a method of statistical inference where Bayes' rule is used to update the probability for a hypothesis as evidence is acquired. So the posterior probability of $\mathbf{w}$ can be updated as:

$$p(\mathbf{w} | \mathbf{y}, \mathbf{a}, \sigma^2) = \frac{p(\mathbf{y} | \mathbf{w}, \sigma^2) p(\mathbf{w} | \mathbf{a})}{p(\mathbf{y} | \mathbf{a}, \sigma^2)} = N(\mathbf{w} | \boldsymbol{\mu}, \boldsymbol{\Sigma}) \quad (11)$$

with

$$\left\{ \begin{array}{l} \boldsymbol{\mu} = \sigma^{-2} \boldsymbol{\Sigma} \boldsymbol{\Phi}^T \mathbf{y} \\ \boldsymbol{\Sigma} = (\sigma^{-2} \boldsymbol{\Phi}^T \boldsymbol{\Phi} + \mathbf{A})^{-1} \\ \mathbf{A} = \text{diag}(\alpha_1, \alpha_1, \dots, \alpha_N) \end{array} \right. \quad (12)$$

Then, the posterior probability, which is unable to be computed can may be expressed as:

$$\begin{aligned} p(\mathbf{w}, \mathbf{a}, \sigma^2 | \mathbf{y}) &= p(\mathbf{w} | \mathbf{y}, \mathbf{a}, \sigma^2) \cdot p(\mathbf{a}, \sigma^2 | \mathbf{y}) \\ &\approx p(\mathbf{w} | \mathbf{y}, \mathbf{a}, \sigma^2) \cdot \delta(\mathbf{a}_{MP}, \sigma_{MP}^2) \end{aligned} \quad (13)$$

Give more insight into all these equations, and we can realize that the key part is to seek a good approximation of $\mathbf{a}_{MP}$ and σ_{MP}^2 , which is verified later to be very effective in generalization performance. Then, our mission becomes to search for the values of unknown parameters to maximize $p(\mathbf{a}, \sigma^2 | \mathbf{y})$, in direct proportion to $p(\mathbf{y} | \mathbf{a}, \sigma^2)$. Combined with Equation (6) and Equation (7), it can be computed and given by:

$$\begin{aligned} p(\mathbf{y} | \mathbf{a}, \sigma^2) &= \int p(\mathbf{y} | \mathbf{w}, \sigma^2) p(\mathbf{w} | \mathbf{a}) d\mathbf{w} \\ &= \frac{\exp\{-0.5 \mathbf{y}^T (\sigma^2 \mathbf{I} + \boldsymbol{\Phi} \mathbf{A}^{-1} \boldsymbol{\Phi}^T)^{-1} \mathbf{y}\}}{(2\pi)^{N/2} |\sigma^2 \mathbf{I} + \boldsymbol{\Phi} \mathbf{A}^{-1} \boldsymbol{\Phi}^T|^{1/2}} \end{aligned} \quad (14)$$

With differentiation of Equation (14), this marginal likelihood is also referred to as the “evidence for hyper-parameters” by Mackay, which gives the iterative re-evaluation algorithm as followed [16].

$$\left\{ \begin{array}{l} \alpha_j^{new} = \frac{1 - \alpha_j \boldsymbol{\Sigma}_{jj}}{\mu_j^2} \\ (\sigma^2)^{new} = \frac{\|\mathbf{y} - \boldsymbol{\Phi} \boldsymbol{\mu}\|^2}{N - (1 - \alpha_j \boldsymbol{\Sigma}_{jj}) \boldsymbol{\Sigma}_j} \end{array} \right. \quad (15)$$

This algorithm shows a clear flow, which is easy to realize. But there are two obvious defects in it. When the number of basic functions (M) is large, the computation complexity will increase by orders of magnitude of $O(M^3)$. Furthermore, inexistence of inverse of matrix in Equation (15) will leads to no solution for the question.

OPTIMIZED RVM LEARNING ALGORITHM

Then, to increase the computation speed while avoiding the disadvantages mentioned, the fast marginal likelihood maximization algorithm is used in this paper. As explained in [17], we evaluate the logarithm of Equation (14).

$$\begin{cases} L(\boldsymbol{\alpha}) = -0.5 \left(N \ln 2\pi + \ln |\mathbf{C}| + \mathbf{y}^T \mathbf{C}^{-1} \mathbf{y} \right) \\ \mathbf{C} = \sigma^2 \mathbf{I} + \boldsymbol{\Phi} \mathbf{A}^{-1} \boldsymbol{\Phi}^T \end{cases} \quad (16)$$

It is obvious that the value of $L(\boldsymbol{\alpha})$ depend on every single hyper-parameter α_j by decomposing $\mathbf{C}$ as:

$$\begin{cases} \mathbf{C} = \mathbf{C}_{-j} + \alpha_j^{-1} \boldsymbol{\phi}_j \boldsymbol{\phi}_j^T \\ \mathbf{C}_{-j} = \sigma^2 \mathbf{I} + \sum_{m \neq j} \alpha_m^{-1} \boldsymbol{\phi}_m \boldsymbol{\phi}_m^T \end{cases} \quad (17)$$

$\mathbf{C}_{-j}$ is the matrix that has nothing to do with the basis vector, whose subscript is j . Derived from Equation (16) and Equation (17), we can express $L(\boldsymbol{\alpha})$ as:

$$L(\boldsymbol{\alpha}) = L(\boldsymbol{\alpha}_{-j}) + L(\alpha_j) \quad (18)$$

where

$$\begin{cases} L(\boldsymbol{\alpha}_{-j}) = -0.5 \left(N \ln 2\pi + \ln |\mathbf{C}_{-j}| + \mathbf{y}^T \mathbf{C}_{-j}^{-1} \mathbf{y} \right) \\ L(\alpha_j) = 0.5 \left(\frac{q_j^2}{\alpha_j + s_j} + \ln \left(\frac{\alpha_j}{\alpha_j + s_j} \right) \right) \\ s_j = \boldsymbol{\phi}_j^T \mathbf{C}_{-j}^{-1} \boldsymbol{\phi}_j, q_j = \boldsymbol{\phi}_j^T \mathbf{C}_{-j}^{-1} \mathbf{y} \end{cases} \quad (19)$$

Based on the analysis of Equation (18), we can figure out that $L(\boldsymbol{\alpha})$ can have a unique maximum with respect to α_j . Combined with Equation (18) and Equation (19), then we can calculate the gradient of $L(\boldsymbol{\alpha})$ for each α_j :

$$\frac{\partial L(\boldsymbol{\alpha})}{\partial \alpha_j} = \frac{\alpha_j^{-1} s_j^2 - (q_j^2 - s_j)}{2(\alpha_j + s_j)^2} \quad (20)$$

Setting Equation (20) to zero, the stationary point, where $L(\boldsymbol{\alpha})$ reaches the maximum, can be achieved:

$$\alpha_j = \frac{s_j^2}{q_j^2 - s_j} \quad (21)$$

However, we have defined that α_j must be positive because it is a parameter of probability function. Thereby there are two cases, where the maximum of $L(\boldsymbol{\alpha})$ can be obtained:

$$\begin{cases} \alpha_j = \frac{s_j^2}{q_j^2 - s_j} & (q_j^2 - s_j > 0) \\ \alpha_j = +\infty & (q_j^2 - s_j \leq 0) \end{cases} \quad (22)$$

The term q_j is explained as the quality factor, while the term s_j is explained as the sparsity factor. The value of q_j^2 indicates that the error between the estimated value and the real value without the corresponding basis function $\boldsymbol{\phi}_j$. And the larger q_j^2 is, the larger the error is. The value of s_j indicates that the sparsity with the corresponding basis function $\boldsymbol{\phi}_j$ included in the model. And the less s_j is, the sparser the model is. However we must put more emphasis on the error. Then we utilize the value of $q_j^2 - s_j$ to determine whether to keep the corresponding basis function $\boldsymbol{\phi}_j$. If the value of $q_j^2 - s_j$ is larger than zero, which means that the error has more influence over the model without the corresponding basis function $\boldsymbol{\phi}_j$, we should keep the corresponding basis function $\boldsymbol{\phi}_j$ to reduce errors. So the value of q_j^2 indicates how well the basis vector $\boldsymbol{\phi}_j$ helps to maximize $L(\boldsymbol{\alpha})$, and the value of s_j indicates how the existence of the basis vector $\boldsymbol{\phi}_j$ serves to decrease $L(\boldsymbol{\alpha})$, respectively [18].

In a word, this optimized algorithm has two merits besides avoiding the defects of traditional method. One is that it provide indicators to depict the degree of correlation of the basis vector, which is the candidate of relevance vector. Another is the efficient computation, which is improved by adding the basis vector into computation one by one. And the steps of the fast marginal likelihood maximization algorithm are as followed.

Step 1. Initialization: initialize σ^2 and α_j to some sensible values, as:

$$\begin{cases} \sigma^2 = 0.1 \times \text{var}[\mathbf{y}] \\ \alpha_j = \frac{\|\boldsymbol{\phi}_j\|^2}{\|\boldsymbol{\phi}_j^T \mathbf{y}\|^2 / \|\boldsymbol{\phi}_j\|^2 - \sigma^2} \end{cases} \quad (23)$$

Step 2. Computation: compute $\boldsymbol{\Sigma}$ and $\boldsymbol{\mu}$ in Equation (12), along with s_j and q_j .

Step 3. Decision: decide that whether the basis vector should be included in the model and how to set the value of α_j after selecting $\boldsymbol{\phi}_j$ as a candidate basis vector.

Case 1. $q_j^2 - s_j > 0$ and $\alpha_j \leq +\infty$

Add $\boldsymbol{\phi}_j$ to the model with setting $\alpha_j = \frac{s_j^2}{q_j^2 - s_j}$