Machine learning systems based on xgBoost and MLP neural network applied in satellite lithium-ion battery sets impedance estimation

Thiago H. R. Donato and Marcos G. Quiles

Department of Applied Computation, National Space Research Institute,
Sao Jose dos Campos, P.0 Box 12227-010, Brazil
Institute of Science and Technology, Federal University of Sao Paulo,
Sao Jose dos Campos, P.0 Box 12231-280, Brazil

Abstract. In this work, the internal impedance of the lithium-ion battery pack (important measure of the degradation level of the batteries) is estimated by means of machine learning systems based on supervised learning techniques MLP - Multi Layer Perceptron - neural network and xgBoost - Gradient Tree Boosting. Therefore, characteristics of the electric power system, in which the battery pack is inserted, are extracted and used in the construction of supervised models through the application of two different techniques based on Gradient Tree Boosting and Multi Layer Perceptron neural network. Finally, with the application of statistical validation techniques, the accuracy of both models are calculated and used for the comparison between them and the feasibility analysis regarding the use of such models in real systems.

Keywords: Lithium-ion battery, Internal impedance, State of charge, Multi Layer Perceptron, Gradient Tree Boosting, xgBoost

1 Introduction

The choice for the technology to be applied in the electrical power system (EPS) is important to the success of a satellite mission since it represents around 20 to 30 percent of a satellite total mass.

For outer space applications, lithium-ion batteries have less than one half of the mass of nickel hydrogen batteries for the same stored energy [6] and, for this reason, can reduce the system weight of a spacecraft, thus improving the load efficiency of satellites. As a result, lithium-ion batteries are highly adopted in the satellites of the United States and European Space Agency (ESA) [17]. In addition, lithium-ion batteries are expected to become the third generation of satellite power storage batteries for Chinas future space platform instead of NiMH batteries and NiCd batteries.

The reliability of Li-ion batteries is an important issues due to the fact that failures of Li-ion battery not only result in serious inconvenience and enormous replacement/repair costs, but also can cause overheating and short circuiting which

can lead to catastrophic consequences, such as explosion. In order to prevent severe failures from occurring, and to optimize Li-ion battery maintenance schedules, breakthroughs in prognostics and health monitoring of Li-ion batteries, with an emphasis on fault detection, correction and remaining-useful-life prediction, must be achieved [20].

Battery PHM has a wide variety of meaning, ranging from irregular manual measurements of voltage and electrolyte specific characteristics to fully automated online observation of various measured and estimated battery parameters. An important battery PHM analysis consists of the measurement of its state of charge (SOC). The state of charge (SOC) of Li-ion battery sets is a function of its internal impedance. Therefore, the state of charge (SOC) can be estimated through the methods described on Section 2 or determined by the battery internal impedance measurement or estimation along the operating cycles, as proposed in this study.

2 Li-ion battery state of charge (SOC) estimation

State of charge estimation has always been a big concern for all battery driven devices but its definition presents many different issues [3]. In general, the SOC of a battery is defined as the ratio of its current capacity (Q_t) to the nominal capacity (Q_n). The nominal capacity is given by the manufacturer and represents the maximum amount of charge that can be stored in the battery. The SOC can be defined as follows:

$$SOC_t = \frac{Q_t}{Q_n}. \tag{1}$$

Therefore, once the nominal capacity is already defined, the methods described below intend to estimate the actual battery capacity after charge/discharge cycles.

The various mathematical methods of estimation are classified according to methodology. The classification of these SOC estimation methods is different in the various literatures. One approach is according to the following categories [18]:

2.1 Direct measurements

Direct measurement methods refer to some physical battery properties such as the terminal voltage and impedance. Many different direct methods have been employed: open circuit voltage method, terminal voltage method, impedance measurement method, and impedance spectroscopy method.

Open circuit voltage method There is approximately a linear relationship between the SOC of the lead-acid battery and its open circuit voltage (OCV) given by:

$$OCV(t) = a_1 * SOC(t) + a_0. \tag{2}$$

Where $SOC(t)$ is the SOC of the battery at t, a_0 is the battery terminal voltage when $SOC = 0$, and a_1 is obtained from knowing the value of a_0 and $OCV(t)$ at $SOC = 1$. Unlike the lead-acid battery, the Li-ion battery does not have a linear relationship between the OCV and SOC [4].

Terminal voltage method The terminal voltage method is based on the terminal voltage drops because of the internal impedances when the battery is discharging, so the electromotive force (EMF) of battery is proportional to the terminal voltage. Since the EMF of battery is approximately linear proportional to the SOC, the terminal voltage of battery is also approximately linear proportional to the SOC [15]. The terminal voltage method has been employed at different discharge currents and temperatures but, at the end of battery discharge, the estimated error of terminal voltage method is large, because the terminal voltage of battery suddenly drops at the end of discharge.

Once the Li-ion battery does not have a linear relationship between the $V_O C$ and SOC, this method is not indicated due to the estimation error.

Impedance spectroscopy method The impedance spectroscopy method measures battery impedances over a wide range of ac frequencies at different charge and discharge currents. The values of the model impedances are found by least-squares fitting to measured impedance values. SOC may be indirectly inferred by measuring present battery impedances and correlating them with known impedances at various SOC levels [10].

Coulomb counting method The Coulomb counting method measures the discharging current of a battery and integrates the discharging current over time in order to estimate SOC. Coulomb counting method is done to estimate the $SOC(t)$ which is estimated from the discharging current, $I(t)$ and previously estimated SOC values, $SOC(t-1)$. SOC is calculated by the following equation:

$$SOC(t) = SOC(t-1) + \frac{I(t)}{Q_n} * \Delta_t. \tag{3}$$

2.2 Machine learning systems

Machine learning system consists of an approach which uses pattern recognition and machine learning techniques to detect changes in system states [12]. With this approach, few information regarding to the analyzed system is necessary in order to build prognostic models due to the fact that only the monitored data itself is

necessary to elaborate them. For this reason, machine learning systems are adequate to EPS systems which are sufficiently complex such that developing an accurate physical model is prohibitively expensive. As batteries have been affected by many chemical factors and have nonlinear SOC, machine learning systems offer good solution for SOC estimation. However, this approach implies in wider confidence intervals than other approaches and requires a substantial amount of data for training.

After the data preparation, a machine learning algorithm shall be selected in order to build the machine learning system. The selection of a proper algorithm for a specific application is a challenging factor in applying data driven prognostics methods. Examples of machine learning algorithms applied in machine learning systems to estimate SOC: back propagation Multi Layer Perceptron neural network (MLP), gradient boosting ($xgBoost$), radial basis function (RBF), fuzzy logic methods, support vector machine (SVM), fuzzy neural network, and Kalman filter [16].

In this study, two machine learning systems were applied: MLP neural network (Section 2.2) and gradient boosting (Section 2.3)

Multi Layer Perceptron In order to build an algorithm capable of classifying a label attribute, the Multi Layer Perceptron (MLP) neural network can be applied. The MLP consists of a feedforward artificial neural network model that can be used in classification or prognostic issues.

The Multi Layer Perceptron network contains three or more layers (an input and an output layer with one or more hidden layers) of nonlinearly-activating neurons. Each neuron combine the inputs multiplied by their correspondent weights and apply an activation function which output is delivered as input of neurons of the following layer [8, Chapter 4].

In many practical applications of artificial neural networks (ANN), there exist natural constraints on the model such as monotonic relations between inputs and outputs that are known in advance. It is advantageous to incorporate these constraints into the ANN structure [19]. The monotonic Multi Layer Perceptron network (MONMLP) is an approach for multi-dimensional function approximation ensuring monotonicity for selected input-output relations. Moreover, we determine the requirements for the network structure regarding universal approximation capabilities.

2.3 Gradient Tree Boosting

Gradient tree boosting is typically used with decision trees (especially CART trees) of a fixed size as base learners. For this special case Friedman proposes a modification to gradient boosting method which improves the quality of fit of each base learner.

Generic gradient boosting at the $m - th$ step would fit a decision tree $h_m(x)$ to pseudo-residuals. Let J_m be the number of its leaves. The tree partitions the input space into J_m disjoint regions $R_{1m}, \ldots, R_{J_m m}$ and predicts a constant value in each region. Using the indicator notation, the output of $h_m(x)$ for input x can be written as the sum:

$$h_m(x) = \sum_{j=1}^{J_m} b_{jm} I(x \in R_{jm}), \tag{4}$$

where b_{jm} is the value predicted in the region R_{jm}.

Then the coefficients b_{jm} are multiplied by some value γ_m, chosen using line search so as to minimize the loss function, and the model is updated as follows:

$$F_m(x) = F_{m-1}(x) + \gamma_m h_m(x), \quad \gamma_m = \gamma arg\,min \sum_{i=1}^{n} L(y_i, F_{m-1}(x_i) + \gamma h_m(x_i)) \tag{5}$$

Friedman proposes to modify this algorithm so that it chooses a separate optimal value γ_{jm} for each of the tree's regions, instead of a single γ_m for the whole tree. He calls the modified algorithm "TreeBoost" [7]. The coefficients b_{jm} from the tree-fitting procedure can be then simply discarded and the model update rule becomes:

$$F_m(x) = F_{m-1}(x) + \sum_{j=1}^{J_m} \gamma_{jm} I(x \in R_{jm}), \quad \gamma_{jm} = \gamma arg\,min \sum_{x_i \in R_{jm}} L(y_i, F_{m-1}(x_i) + \gamma)$$

$$\tag{6}$$

Root Mean Square Error Machine learning systems apply machine learning techniques in a supervised approach. Considering a numeric label attribute in the estimative of the battery set impedance, each observed value can be compared with the predicted one. This individual deviation is called a residual and the aggregation of all the residuals is denominated the Root Mean Square Error (RMSE), obtained as follows [9]:

$$RMSE = \sqrt{\frac{\sum_{t=1}^{n}(\hat{y}_t - y_t)^2}{n}} \tag{7}$$

Where:

- $\hat{y}_t$: predicted instance
- y_t: observed instance
- n: number of instances

2.4 Li-ion battery state of charge (SOC) based on battery internal impedance

According to the definition of the SOC, the function which provides the battery state of charge based on battery internal impedance is obtained as follows:

$$SOC_t = SOC_{t0} + \int_{t0}^{t} (\frac{\eta \cdot I}{C_t}) \cdot d_t \tag{8}$$

Where:

- SOC_{t0}: estimated SOC at time $t0$, when the estimation process starts
- SOC_t: estimated SOC at time t
- η: current efficiency
- I: current - assumed to be positive when charging
- C_n: capacity of the battery at time t

In this study, the battery impedance is obtained through the machine learning systems (see Section 2.2) which can be applied in the determination of battery state of charge (SOC).

Bagging optimization According to bagging optimization method, a training set $D of size n$ is divided into m new training sets $D_i, each of size n$, by sampling from D uniformly and with replacement. Sampling the m new training sets with replacement, implies that some observations may be repeated in each D_i[1].

This kind of sample is also known as bootstrap sample. In order to obtain the resulting ensemble of models, the m models are fitted using the above m bootstrap samples. Finally, all the models are applied to the scoring set and the labels are combined by averaging the output (for regression) or voting (for classification).

3 Machine learning systems for Li-ion battery impedance estimation

This study applies two machine learning systems (MLP neural network - Section 2.2 and gradient boosting - Section 2.3) in order to estimate the battery internal impedance and compares the obtained results.

To perform the comparison between the two machine learning systems, a battery testing database provided by NASA Ames Research Center [14] was used as data set. The database comprises sensor monitoring data of Li-ion batteries mounted in batches of 4 and running through 3 different operational profiles (charge, discharge and impedance) at ambient temperatures of 4, 24 and 44 Celsius degrees (see Section 4).

From the analyzed data set, features were selected to characterize the battery condition during a certain period of time (see Section 5). After the development

of the data set, machine learning systems are trained (see Section 6), Root Mean Square Error (RMSE) performance metric is used to evaluate the obtained machine learning systems (see Section 9) and, if the performance criteria are achieved, they can be considered feasible solutions to estimate the battery internal impedance in an online basis.

4 Data preparation for Li-ion battery impedance estimation

Battery impedance, which decreases over the working time of a battery, is an important and direct indicator for estimating battery state of charge (SOC). In online or in-orbit applications, such as electric vehicles and satellites, the battery internal impedance measurement or monitoring is difficult [11]. It can be used charge transfer resistance and electrolyte resistance extracted from EIS to estimate battery capacity [14]. However, these features can only be obtained via offline tests under the optimal measuring conditions and by using specialized and expensive equipment for EIS measurements [5]. The results of the aging experiment showed that increase in battery capacity loss or resistance in a lifetime is related to operating conditions, such as voltage, current, and temperature. However, in practical applications, several characteristics, such as current and voltage, are controlled to meet the load requirements of an associated circuit and thus cannot represent battery aging [13].

4.1 Li-ion battery testing set information

The following sections apply machine learning techniques in Li-ion battery impedance estimation using battery data provided by National Aeronautics and Space Administration (NASA) Ames Prognostics Center of Excellence [14], where 134 rechargeable lithium-ion batteries were tested.

The laboratory setup and data recording were conducted by National Aeronautics and Space Administration (NASA) Ames Prognostics Center of Excellence [14]. According to NASA experiment, the 134 rechargeable lithium-ion batteries are organized in 34 battery sets. Each battery set contains the test data organized according to Figure 2.

4.2 Laboratory setup

The experimental setup primarily consists of a set of Li-ion cells (which may reside either inside or outside an environmental chamber), chargers, loads, EIS equipment for battery health monitoring (BHM), a suite of sensors (voltage, current and temperature), some custom switching circuitry, data acquisition system and a computer for control and analysis. Figure 1 details the assembly of the equipment.

7
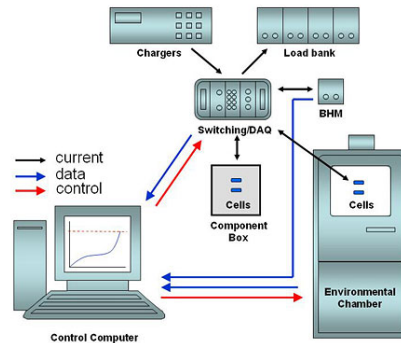
Fig. 1. Laboratory setup [14]

The cells are cycled through charge and discharge cycles under different load and environmental conditions set by the electronic load and environmental chamber respectively. Periodically EIS measurements will be taken to monitor the internal condition of the battery using the BHM module. The DAQ system collects the externally observable parameters from the sensors. The switching circuitry enables the cells to be in the charge, discharge or EIS health monitoring state as dictated by the algorithms running on the control computer [14].

4.3   Li-ion battery testing set information

The Li-ion batteries are organized in batches of 4 are run through 3 different operational profiles (charge, discharge and impedance) at ambient temperatures of 4, 24 and 44 oC [14]:

1. Charge step: charging was carried out in a constant current (CC) mode at 1.5A until the battery voltage reached 4.2V and then continued in a constant voltage (CV) mode until the charge current dropped to 20mA
2. Discharge step: discharging was conducted in CC mode until the discharge voltage reached a predefined cutoff voltage. Fixed and variable load currents at 1, 2, and 4 Amps were used and the discharge runs were stopped at 2V, 2.2V, 2.5V or 2.7V
3. Impedance measurement: measurement was performed through an electrochemical impedance spectroscopy (EIS) frequency sweep from 0.1 Hz to 5 kHz

   Figure 2 details the battery data structure of the operational profiles.

4.4   Li-ion battery impedance measurement rectifier

In order to eliminate the noise generated by time-varying current passing through an electro-chemical cell or battery due to load fluctuation, a filtering approach or an
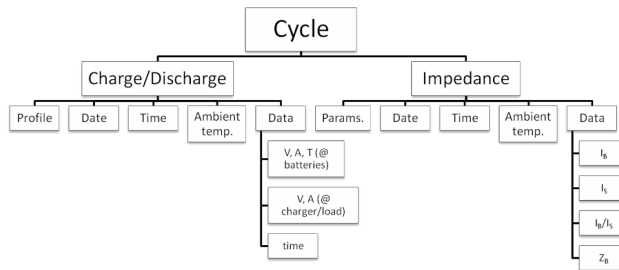
Fig. 2. Data structure [14]

electronic cancellation technique shall be applied. In the laboratory setup described above, it was used an electronic device.

Time-varying current flowing in a circuit which includes the cell/battery is sensed externally to the cell/battery with a magnetically-coupled ac current probe thereby producing an induced time-varying signal. This induced signal is amplified to the level of the original time-varying current and applied to the cell/battery's terminals in phase-opposition to the original current. As a result, the component of time-varying current flowing in the cell/battery's external leads assumes an alternate path around the cell/battery and is effectively canceled within the cell/battery itself [2].

5 Features selection for Li-ion battery impedance estimation

In this study, the 134 rechargeable lithium-ion batteries are organized in 34 battery sets. Each battery set contains the test data according to Figure 2. For each C-D (charge and discharge) cycle, the following features were extracted [20]:

- F1: during charge cycle, time interval between the nominal voltage and the cutoff voltage
- F2: during charge cycle, time interval between the nominal current and the cutoff current
- F3: during discharge cycle, time interval between two predefined voltages
- F4: average temperature during the time interval F1
- F5: average temperature during the time interval F2
- F6: during discharge cycle, cutoff voltage

The historical set applied in the machine learning systems modeling includes the six features (F1, F2, F3, F4, F5 and F6) and the label attribute which corresponds to the rectified battery impedance. For each machine learning system, two

regression models are built in order to estimate the real and imaginary components of the rectified impedance. Tables 1 and 2 are extractions of the historical sets corresponding to the real and imaginary components which contain 20 records (C-D cycles) for each one of the 34 battery sets (684 records - C-D cycles, in total).

Table 1. Historical set - impedance real component

cycle	F1	F2	F3	F4	F5	F6	imp_re
2008_4_1	9966.407	6422.609	472.313	27.0074	26.3008	2.4556	0.06175
2008_4_2	10226.375	6627.891	472.125	25.6742	26.5323	2.6321	0.05989
2008_4_3	10635.968	6528.063	472.344	25.6754	26.3325	2.5010	0.05919

Table 2. Historical set - impedance imaginary component

cycle	F1	F2	F3	F4	F5	F6	imp_img
2008_4_1	9966.407	6422.609	472.313	27.0074	26.3008	2.4556	-0.00096
2008_4_2	10226.375	6627.891	472.125	25.6742	26.5323	2.6321	-0.00112
2008_4_3	10635.968	6528.063	472.344	25.6754	26.3325	2.5010	-0.00105

6 Li-ion battery impedance data modeling

After the data preparation, two machine learning techniques (MLP neural network - Section 2.2 and gradient boosting - Section 2.3) are applied in order to estimate the battery internal impedance. For each applied technique, a few training cycles are executed with the variation of the technique hyperparameters. Therefore, each training cycle generates a regression model which can be compared with the other obtained regression models by the application of the RMSE (Root Mean Square Error) validation technique, according to Section 2.3.

7 MLP neural network with monotonicity constraints modeling

Tables 1 and 2 are used to build two regression models based on MLP neural network technique (Section 2.2).

Multi Layer Perceptron neural network with monotonicity constraints implements one hidden-layer Multi Layer Perceptron neural network (MLP) models that enforces monotonic relations on designated input variables. Each training cycle applies 10 or 20 ensemble members to fit and 1, 2, 3 or 4 hidden nodes in the hidden layer.

The ensemble members to fit in each training cycle are obtained according to Section 2.4. Each ensemble member contains a random subset of 70 percent of the

training set and, after generating the 10 or 20 regression models (depending on the number of ensemble members to fit), the result corresponds to the mean value obtained through the application of all regression models.

These training cycles with different combinations result into different MLP neural networks with monotonicity constraints. These different combinations of the hyperparameters are described in Table 3.

Table 3. MLP neural network with monotonicity constraints hyperparameters

| Hyperparameter	Description	Possible values
hidden1	number of hidden nodes in the first hidden layer	1, 2, 3, 4
n.ensemble	number of ensemble members to fit	10, 20
monotone	column indices of covariates for which the monotonicity constraint should hold	1
bag	logical variable indicating whether or not bootstrap aggregation (bagging) should be used	TRUE
iter.max	maximum number of iterations of the optimization algorithm	500

7.1 MLP neural network modeling with monotonicity constraints and 10 ensemble members to fit

Estimative of real component of battery set impedance Using the backpropagation algorithm to update the MLP neural network weights, monotonicity constraints, 10 ensemble members to fit and using different numbers of hidden nodes in the hidden layer (1, 2, 3 or 4), the real component of the battery impedance through the cycles is according the Figure 3.
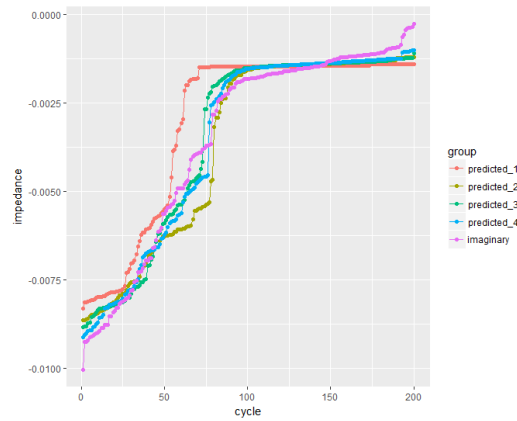
Estimative of imaginary component of battery set impedance Using the backpropagation algorithm to update the MLP neural network weights, monotonicity constraints, 10 ensemble members to fit and using different numbers of hidden nodes in the hidden layer (1, 2, 3 or 4), the imaginary component of the battery impedance through the cycles is according the Figure 4.

7.2 MLP neural network modeling with monotonicity constraints and 20 ensemble members to fit

Estimative of real component of battery set impedance Using the backpropagation algorithm to update the MLP neural network weights, monotonicity constraints, 20 ensemble members to fit and using different numbers of hidden nodes in the hidden layer (1, 2, 3 or 4), the real component of the battery impedance through the cycles is according the Figure 5.
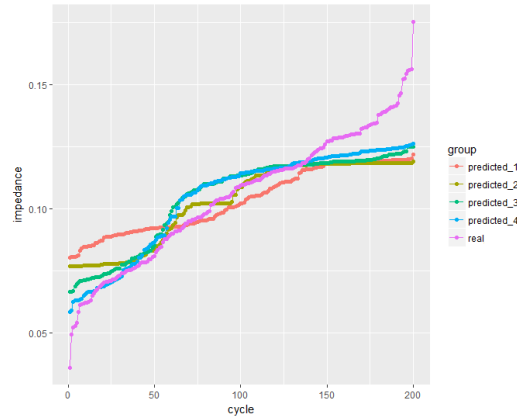
Fig. 3. Estimated real component of battery set impedance using MLP neural network modeling with monotonicity constraints and 10 ensemble members to fit
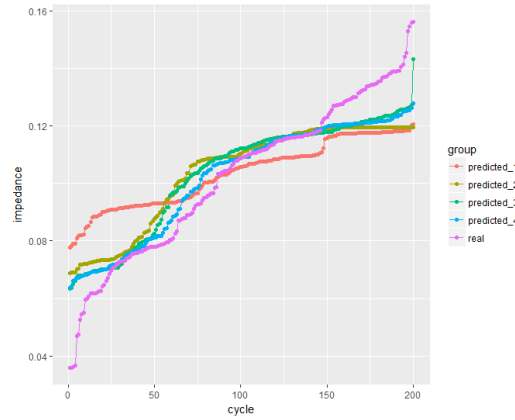

Fig. 4. Estimated imaginary component of battery set impedance using MLP neural network modeling with monotonicity constraints and 10 ensemble members to fit
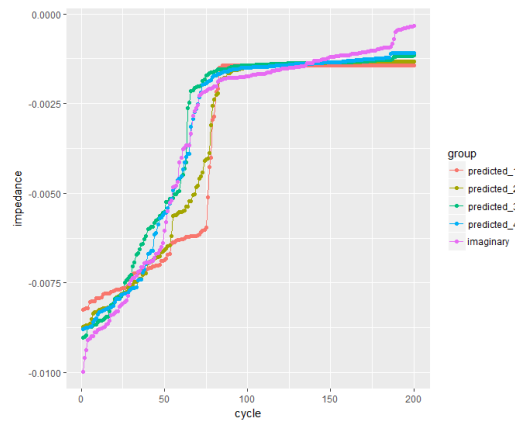
Estimative of imaginary component of battery set impedance Using the backpropagation algorithm to update the MLP neural network weights, monotonicity constraints, 20 ensemble members to fit and using different numbers of hidden nodes in the hidden layer (1, 2, 3 or 4), the imaginary component of the battery impedance through the cycles is according the Figure 6.

7.3 MLP neural network with monotonicity constraints model validation

Applying the RMSE (Root Mean Square Error) validation technique, according to Section 2.3, each model developed with the application of Multi Layer Perceptron

Fig. 5. Estimated real component of battery set impedance using MLP neural network modeling with monotonicity constraints and 20 ensemble members to fit

Fig. 6. Estimated imaginary component of battery set impedance using MLP neural network modeling with monotonicity constraints and 20 ensemble members to fit

technique in the estimative of real and imaginary components of the battery set impedance was evaluated and the corresponding RMSE values are presented in Table 4.

Increasing the number of ensemble members to fit, there was no impact on the root square mean error for estimating the real component of battery set impedance. However, a higher number of ensemble members to fit minimized the root square mean error for estimating the imaginary component of battery set impedance.

According to Table 4, a higher number of hidden nodes on the hidden layer minimizes the root square mean error for estimating the real and imaginary components of battery set impedance.

Table 4. MLP neural network with monotonicity constraints model validation

Battery impedance	Ensemble members to fit	Hidden nodes on layer	RMSE
real	20	1	0.0161220600
real	20	2	0.0124101700
real	20	3	0.0100222800
real	20	4	0.0094165090
imaginary	20	1	0.0011236520
imaginary	20	2	0.0007225610
imaginary	20	3	0.0004410664
imaginary	20	4	0.0003257367

The MLP models with the lower root square mean error in estimating the real and imaginary components of battery set impedance have the configuration of 20 ensemble members to fit and 4 hidden nodes on the hidden layer and an RMSE error of 0.0094165090 and 0.0003257367, respectively.

8   Gradient boosting modeling

Tables 1 and 2 are used to build two regression models based on gradient boosting technique (Section 2.3). The training set is divided into training and validation sets. The validation set is used to evolve the regression models during each gradient boosting iteration. Each training cycle applies an 'eta' equal to 0.20 or 0.60 to control the learning rate and a subsample ratio of the training instances of 0.80, 0.85, 0.90 or 0.95.

These training cycles with different combinations result into different gradient boosting models. These different combinations of the hyperparameters are described in Table 5.

Table 5. Gradient boosting hyperparameters

Hyperparameter	Description	Possible values
objective	objective function	reg:linear
max_depth	maximum depth of a tree	10
eta	control the learning rate: scale contribution of each tree by a factor of $0 <$eta<1	0.20, 0.60
col_sample	subsample ratio of columns when constructing each tree	0.80
ss_sample	subsample ratio of the training instance. 0.5 means that xgboost randomly collected half of the data to grow trees and this will prevent overfitting	0.80, 0.85, 0.90, 0.95
eval_metric	evaluation metric per validation cycle	root mean square error
nrounds	the max number of validation cycles	200

The hyperparameter 'eta' has an important role in the gradient boosting modeling since it controls the learning rate. This parameter scales the contribution of each tree by a factor between 0 and 1 when it is added to the current approximation. It is used to prevent overfitting by making the boosting process more conservative. Lower value for 'eta' implies larger value for 'nrounds': low 'eta' value means model more robust to overfitting but slower to compute.

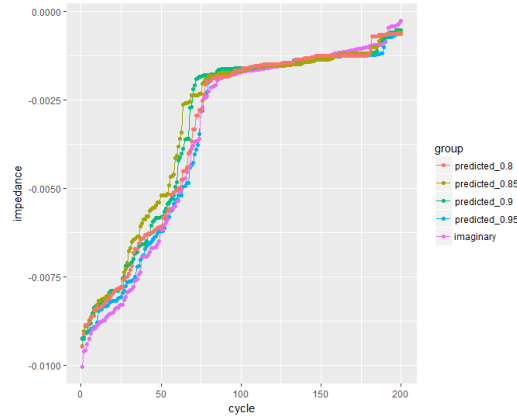## 8.1   Gradient boosting modeling with eta = 0.20

Estimating real component of battery set impedance Building an ensemble of decision trees in which each decision tree is built in order to minimize the error of the previous one (gradient boosting method), setting the maximum depth of the trees to 10, applying the root mean square error method as a metric to evolve the model with the validation set, controlling the learning rate through 'eta' of 0.20 and subsetting the training set instances with distinct ratios (0.80, 0.85, 0.90 or 0.95), the real component of the battery impedance through the cycles is according the Figure 7.
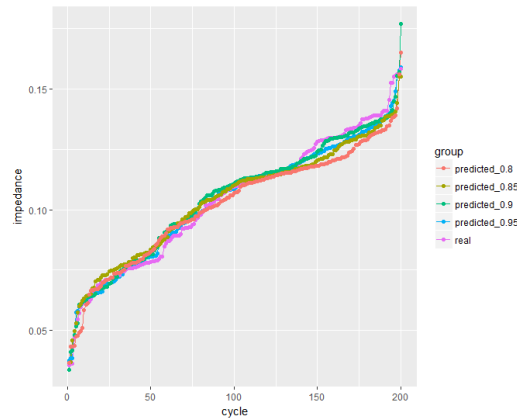
Fig. 7. Estimated real component of battery set impedance using gradient boosting modeling with eta = 0.20

Estimating imaginary component of battery set impedance Building an ensemble of decision trees in which each decision tree is built in order to minimize the error of the previous one (gradient boosting method), setting the maximum depth of the trees to 10, applying the root mean square error method as a metric to evolve the model with the validation set, controlling the learning rate through 'eta' of 0.20 and subsetting the training set instances with distinct ratios (0.80,

0.85, 0.90 or 0.95), the imaginary component of the battery impedance through the cycles is according the Figure 8.
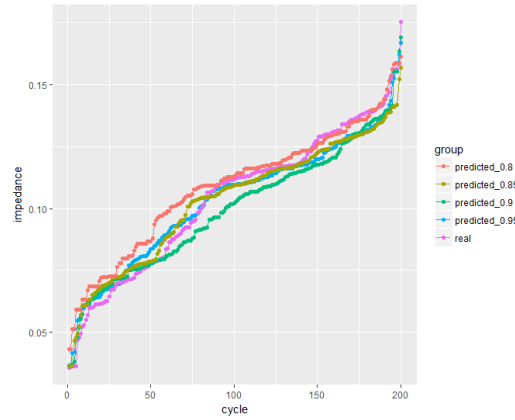

Fig. 8. Estimated imaginary component of battery set impedance using gradient boosting modeling with eta = 0.20
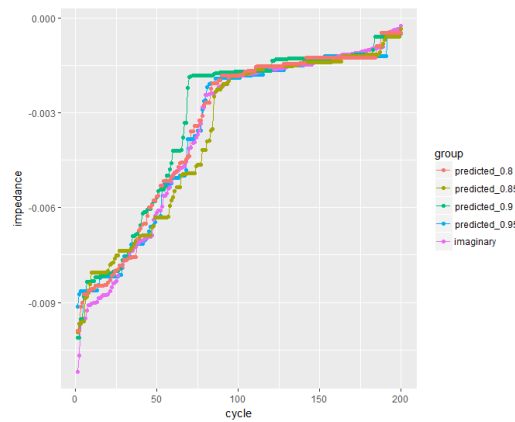
8.2 Gradient boosting modeling with eta = 0.60

Estimating real component of battery set impedance Building an ensemble of decision trees in which each decision tree is built in order to minimize the error of the previous one (gradient boosting method), setting the maximum depth of the trees to 10, applying the root mean square error method as a metric to evolve the model with the validation set, controlling the learning rate through 'eta' of 0.60 and subsetting the training set instances with distinct ratios (0.80, 0.85, 0.90 or 0.95), the real component of the battery impedance through the cycles is according the Figure 9.

Estimating imaginary component of battery set impedance Building an ensemble of decision trees in which each decision tree is built in order to minimize the error of the previous one (gradient boosting method), setting the maximum depth of the trees to 10, applying the root mean square error method as a metric to evolve the model with the validation set, controlling the learning rate through 'eta' of 0.60 and subsetting the training set instances with distinct ratios (0.80, 0.85, 0.90 or 0.95), the imaginary component of the battery impedance through the cycles is according the Figure 10.

Fig. 9. Estimated real component of battery set impedance using gradient boosting modeling with eta = 0.60

Fig. 10. Estimated imaginary component of battery set impedance using gradient boosting modeling with eta = 0.60

8.3   Gradient boosting model validation

Applying the RMSE (Root Mean Square Error) validation technique, according to Section 2.3, each model developed with the application of Gradient boosting technique in the estimating real and imaginary components of the battery set impedance was evaluated and the corresponding RMSE values are presented in Table 6.

A lower 'eta' minimizes the root square mean error for estimating the real and imaginary components of battery set impedance. According to Table 6, increasing the number of instances to be used in the training process, there was a decrease on the root square mean error for estimating the real component of battery set impedance. However, a higher number of instances to be used in the training process

Table 6. Gradient boosting model validation

Battery impedance	'eta' - Learning rate	Subsample ratio	RMSE
real	0.20	0.80	0.0057317940
real	0.20	0.85	0.0044927730
real	0.20	0.90	0.0034616720
real	0.20	0.95	0.0028836380
imaginary	0.20	0.80	0.0003310718
imaginary	0.20	0.85	0.0006632989
imaginary	0.20	0.90	0.0005046742
imaginary	0.20	0.95	0.0001962272

maximized the root square mean error for estimating the imaginary component of battery set impedance.

The Gradient boosting models with the lower root square mean error in estimating the real and imaginary components of battery set impedance have the configuration of 'eta'=0.20 and a subsample ratio of 0.95 and an RMSE error of 0.0028836380 and 0.0001962272, respectively.

9 Li-ion battery impedance model validation

According to RMSE validation technique (see Section 2.3), the Multi Layer Perceptron model with the higher performance (20 ensemble members to fit and 4 hidden nodes on the hidden layer) and the Gradient Tree Boosting model with the higher performance ('eta'=0.20 and a subsample ratio of 0.95) are compared in 11 regarding the estimation of the real and imaginary components of the battery set impedance.
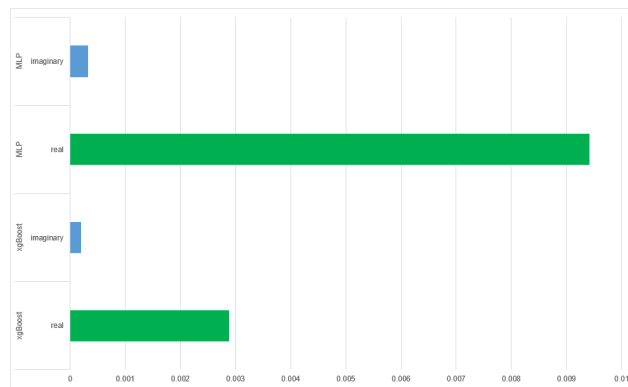
Fig. 11. RMSE State of charge (SOC) model validation

10   Conclusion

In this study, two distinct machine learning approaches were applied in order to estimate the battery impedance of satellite Li-ion battery sets which is used in the determination of their state of charge (SOC) (see Section 2.4). The obtained RMSE (Root Mean Square Error) of both approaches has demonstrated the feasibility of such machine learning systems in estimating Li-ion battery sets battery impedance in cases where an error below 0.10 is acceptable.

However, once the obtained RMSE (Root Mean Square Error) of the Gradient Tree Boosting model is lower in estimating both real and imaginary components of the battery impedance, this approach is recommended over the Multi Layer Perceptron model.

Acknowledgements

References

1. J. A. Aslam, R. A. Popa, and R. L. Rivest. On estimating the size and confidence of a statistical audit. *Proceedings of the Electronic Voting Technology Workshop*, 7, 2007.
2. K. S. Champlin. Method and apparatus for suppressing time-varying signals in batteries undergoing charging or discharging, 11 1992.
3. Y. Chang, W. The state of charge estimating methods for battery: A review. *International Scholarly Research Notices Applied Mathematics*, 2013:1–7, 2013.
4. M. Coleman, C. K. Lee, C. Zhu, and W. G. Hurley. State of-charge determination from emf voltage estimation: using impedance, terminal voltage, and current for lead-acid and lithium-ion batteries. *IEEE Transactions on Industrial Electronics*, 54:25502557, 2007.
5. M. Dalal, J. Ma, and D. He. Lithium-ion battery life prognostic health management system using particle filtering framework. *Proc. Inst. Mech. Eng. Part O J. Risk*, 225:8190, 2011.
6. G. J. Dudley. Lithium-ion batteries for space. *Proceedings of the Fifth European Space Power Conference*, page 17, 1998.
7. T. Hastie, R. Tibshirani, and J. H. Friedman. *The elements of statistical learning: data mining, inference, and prediction*, volume 02. Springer, USA, 2009.
8. S. Haykin. *Neural networks - A comprehensive foundation*, volume 02. Cambridge University Press, USA, 1999.
9. R. J. Hyndman and A. B. Koehler. Another look at measures of forecast accuracy. *International Journal of Forecasting*, 04:679688, 2006.
10. R. Li, J. F. Wu, H. Y. Wang, and G. C. Li. Prediction of state of charge of lithium-ion rechargeable battery with electrochemical impedance spectroscopy theory. *Proceedings of the 5th IEEE Conference on Industrial Electronics and Applications*, page 684688, 2010.
11. D. Liu, H. Wang, Y. Peng, W. Xie, and H. Liao. Satellite lithium-ion battery remaining cycle life prediction with novel indirect health indicator extraction. *Energies*, 6:36543668, 2013.
12. J. Liu and G. Wang. A multi-step predictor with a variable input pattern for system state forecasting. *Mechanical Systems and Signal Processing*, 23:15861599, 2009.
13. M. Parviz and S. Moin. Boosting approach for score level fusion in multimodal biometrics based on auc maximization. *J. Inf. Hiding Multimed. Signal Process*, 2:5159, 2011.

14. B. Saha and K. Goebel. Battery data set, 2007. NASA Ames Research Center, Moffett Field, CA.
15. S. Sato and A. Kawamura. A new estimation method of state of charge using terminal voltage and internal resistance for lead acid battery. *Proceedings of the Power Conversion Conference*, page 565570, 2002.
16. M. Schwabacher. A survey of data-driven prognostics. *AIAA Meeting Papers*, 2005.
17. D. Wang, G. Li, and Y. Pan. The technology of lithium-ion batteries for spacecraft application. *Aerospace Shanghai*, 4:5459, 2000.
18. N. Watrin, B. Blunier, and A. Miraoui. Review of adaptive systems for lithium batteries state-of-charge and state-of-health estimation. *Proceedings of IEEE Transportation Electrification Conference and Expo*, pages 1–6, 2012.
19. H. Zhang and Z. Zhang. Feedforward networks with monotone constraints. *International Joint Conference on Neural Networks*, 03:1820–1823, 1999.
20. J. Zhang and J. Lee. A review on prognostics and health monitoring of li-ion battery. *Journal of Power Sources*, 196:60076014, 2011.

Authors

Thiago Donato graduated at Electrical Engineering from Federal University of Itajub - UNIFEI (2006) and has worked during five years in private companies (EMBRAER - aircraft manufacturer company - and TOTVS - software company) developing solutions which apply machine learning techniques in the resolution of major issues. Donato is enrolled in master's at Computer Science from National Space Research Institute - INPE (2018). Has experience in Machine Learning, acting on the following subjects: data and text preparation and mining, neural network and other machine learning classification techniques.
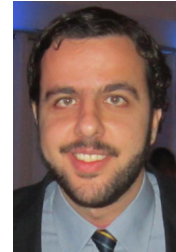
Marcos G. Quiles is an Associate Professor at the Department of Science and Technology, Federal University of So Paulo, Brazil. He received the BS degree, with honors, in 2003 from the State University of Londrina, Brazil, and a Ph.D. degree from the University of So Paulo, Brazil, in 2009, both in Computer Science. From January to July of 2008, Quiles was a Visiting Scholar in the Perception and Neurodynamics Lab at The Ohio State University. From January to December of 2017, Quiles was an Academic Visitor at the University of York, York-UK. He was awarded a Brazilian research productivity fellowship from the Brazilian National Research Council (CNPq). His research interests include nature-inspired computing, machine learning, complex networks, and their applications in interdisciplinary problems.