Battery Prognostics: Estimation of Remaining Operational Time of batteries using convolution and temporal-correlation

A Project Report

Submitted in partial fulfillment of the requirements for the award of the degree of

BACHELOR OF TECHNOLOGY

in

MECHANICAL ENGINEERING

by

SHRISUDHAN G

ME17B122

Department of Mechanical Engineering

INDIAN INSTITUTE OF TECHNOLOGY MADRAS

May 2022

CERTIFICATE

This is to certify that the project report titled "**Battery Prognostics: Estimation of remaining operational time of batteries using convolution and temporal-correlation**" submitted by **Shrisudhan G (ME17B122)** for the award of the degree of **Bachelor of Technology in Mechanical Engineering** is a bonafide record of the work carried out by him in the Department of **Mechanical Engineering, IIT Madras**. The contents of this project report, in full or in parts, have not been submitted to any other University or Institute for the award of any degree or diploma.

Dr. N. Arunachalam

Associate Professor

Department of Mechanical Engineering

Indian Institute of Technology Madras

Chennai - 600036

India

Date:

Dr. A. Seshadri Sekhar

Head of the Department

Department of Mechanical Engineering

Indian Institute of Technology Madras

Chennai - 600036

India

Date:

ACKNOWLEDGEMENTS

I would like to take this opportunity to express my gratitude to **Dr. N. Arunachalam** for his guidance during the course of the project. He has been a constant source of inspiration and motivation to me. Without his expert guidance, eye for detail and scrutiny of the results, it would have been impossible for me to accomplish this task.

I would also like to express my sincere gratitude to **Deep Singh**, for guiding me during the course of the project. The knowledge I gained from him helped me immensely in the completion of this project. Without his constant aid, patience and eagerness to help during testing times, this project would not have been completed

Lastly, I would like to acknowledge the ever present support of my **family and friends** in completing this project. Their moral support helped immensely in keeping me calm and composed.

IIT Madras

SHRISUDHAN G

Date: 27 May 2022

ABSTRACT

Electricity is an essential part of modern life for running various appliances of daily use. We are currently in the transition phase, in which we are reducing our reliance on fossil fuels and increasing our use of renewable energy sources such as wind, tidal, and solar energy. As the demand for the electrical vehicles, self-driving vehicles and unmanned aerial vehicles (UAV) is increasing, the demand for battery is going high. Hence, the operation of these vehicles depends on the battery and the capability to store energy efficiently. Battery is one of the important unit for these vehicles and devices as it decides the overall operational time. In order to plan a transportation it is important to estimate the remaining operational time (ROT) based on the rate of battery discharge considering various affecting parameters at any instance of time. The present study propose a novel deep neural network (DNN) based architecture which uses temporal information to learn the changes in the battery across time and utilizes that in-coherence with the available parameters for estimating accurate remaining operational time. The proposed architecture has been implemented on two different types of batteries from NASA Ames Prognostics Center of Excellence (PCoE). Furthermore, same architecture has been tested with the experimental data obtained by flying the DJI Mavic Mini 2 drone in an open environment. The proposed algorithm is shown to produce accurate and robust estimation of ROT as it outperforms existing techniques for various batteries including the battery used in DJI Mini 2.

Keywords: Battery prognostics, Remaining operational time estimation, NASA dataset, Drone battery prognostics

Contents

List of Tables

List of Figures

NOMENCLATURE

SoC	State of Charge
SoH	State of Health
LIB	Li-ion Battery
Li-Po	Lithium-Polonium
CC	Constant Current
CV	Constant Voltage
EIS	Electrochemical Impedance Spectroscopy
EOL	End-of-Life
RUL	Remaining Useful Life
ROT	Remaining Operational Time
SVM	Support Vector Machine
k-NN	k-Nearest Neighbors
DT	Decision Tree
RF	Random Forest
GBT	Gradient Boosting Trees
MAE	Mean Absolute Error
MSE	Mean Square Error

1 Introduction

Energy storage is a very critical hub connecting the entire entire energy generation and consumption for the entire grid by augmenting renewable resources like wind, hydro and solar as well as fossil and nuclear fuels, demand side resources and system efficiency assets. It also has the potential to serve as a generation, transmission, or distribution asset all at the same time. Because most consumer gadgets are powered by battery-like technologies and humanity is striving to increase electricity generation from renewable resources, energy storage has emerged as one of the most essential fields. The present energy storage technology can be categorised as[8]:

1. Electrochemical storage, such as battery technology, is typically utilised for short storage period and great efficiency.

2. Mechanical storage, such as compressed air energy storage, is generally utilised for high capacity and power.

3. Chemical storage, such as hydrogen and methane, is employed in situations where low efficiency is acceptable but long-term storage is required.

4. Thermal storage is commonly utilised for high efficiency and long lifespan, such as sensible heat storage, thermochemical energy storage, and latent heat storage.

According to the inventory of U.S. Greenhouse Gas Emissions and Sinks, during the period 1990-2019, transportation was the major source of total U.S. greenhouse gas emissions in 2019, contributing for 29 percent of total U.S. greenhouse emissions[9]. The gradual depletion of fossil fuels has revolutionized the transportation by shifting from chemical energy storage in the form of fossil fuels to electrochemical energy storage in the form of batteries. This has led to the introduction of electric vehicles and the rapid growth of its market. Among the electric vehicles, Battery Electric Vehicles (BEVs) accounts for two-thirds of the new electric car registrations in 2020. Similar to the proliferation observed in BEVs, Electirc-Unmanned Aerial Vehicles have also been subject to rapid growth of late[10]. The E-UAV market size is expected to grow at a CAGR of 14.36% from 2021 to 2026 as per the latest market report[11]. The significant increase in the procurement of small E-UAVs owing to the rapid deployment of tactical Unmanned Aerial Vehicles (UAVs) in military and defense applications have led to a exponential growth in the E-UAVs market. With the increasing production and demand for electric

vehicles and UAVs, energy storage has emerged as one of the most vital points, as the bulk of these locomotives are powered by rechargeable batteries technology and electricity production from renewable energy sources is rapidly expanding. With the increasing demand for the electric vehicles and UAVs, it becomes increasingly critical to perform battery management and prognostics.

As electric vehicles, unmanned vehicles and E-UAVs are the new futures of transportation, the battery is one of the critical unit as the performance and durability of these vehicles depends on it. Due to usage and ageing, the battery's capacity to store charge steadily declines over time, limiting total journey duration. The ageing of batteries can be caused due to a variety of conditions, including high-rate cycling, over-charge and over-discharge, and variation in the ambient temperature[12]. Hence, battery monitoring and health management are critical for mitigating faults, planning vehicle functioning and missions, and having a safe, dependable solution[13]. Multi-rotor drones are one form of the UAVs for which the prominent uses are aerial photography and videography, real estate photography, mapping and surveying, asset inspection, package delivery, agriculture, crop spraying, crop monitoring, live streaming events, inspection using multispectral or thermal cameras, search and rescue missions, marine rescue, disaster zone mapping, mining, firefighting, meterorology etc. Multi-rotor unmanned drones are customized for its uses in military for variety of purposes like supervison, combat mission and target decoys.

Given the importance of multi-rotor drone in military operations and equivalent fields it becomes important to estimate the journey duration in order to formulate mission related decision based on the estimated operatable time. This estimation of the remaining journey duration should be robust to various weather condition, should also have the ability to assess the drones current operating conditions and adapt to these changes inherently.

Consequently, the focal point of this study is to develop a general and robust machine learning architecture to estimate the operational time for various types of battery. To develop the framework in this project, the battery dataset provided by the NASA's Ames Prognostics Centre of Excellence (PCoE)[1]. Further the same developed architecture has been trained and validated with the battery of DJI Mavic Mini 2 drone upon experimentation.

The order of the subsequent sections of this report are as follows. In section 2 a brief discussion on the research related to the battery health monitoring and management that has been conducted till date are discussed. A summary of battery management systems and the objective of this research is given in section 3. The data acquisition systems used by our drone data collection process and the system used in NASA battery dataset is explained in section 4. Data analysis and visualizations of these features

over various run are discussed in section 5. section 6 explain some of the famous techniques that can be applied for the estimation of remaining operational time(ROT). In section 7, a novel convolution and temporal-attention based Deep Neural Network(DNN) is proposed which understand the temporal correlation between the time-series data points and estimates ROT based on these information. Finally, section 8 and section 9 presents results and discussion of the performance of the system respectively.

2 Literature Survey

Battery prognostics is a very common field in prognostics and health management. The extensive use of electric batteries presently has led to increase in research publication on battery modelling and prognostics. Most of the research works concentrate on State of Charge(SoC) and State of Health(SoH) estimation.

2.1 State of Charge and State of Health estimation

Six techniques is used to determine a battery's state of charge: 1. Look-up tables procedure, 2. Coulomb counting procedure[14], 3. Artificial neural network procedure[15], 4. Support vector regression procedure[16], 5. Electro-chemical procedure[17], and 6. Equivalent circuit procedure[18]. The SOH, like the SOC estimate method, may be divided into four categories: 1.Models that are physics-dependent, 2. Empirical processes, 3. Models based on (a) Differential Voltage Analysis (DVA), (b) Incremental Current Analysis (ICA), and (c) Data-Driven Method (DDM) [19]. However, different ways to estimating SOC and SOH have been presented, and much study has been done in this topic. In [20], accurate monitoring of SOH and SOC was introduced using an acoustic-ultrasonic stress wave was proposed. In [21], a built-in piezoelectric sensor was used in the time-domain analysis to discover an inherent co-relation between battery SOH and SOC with waveform signal categories. A strain gauge type very accurate displacement sensor was utilised to calculate the produced force of battery cell expansion at the time of charging as a result of the bumping of battery electrode dynamic material [22]. The SOH and SOC estimation time update is first determined, after which the result is supplied into an adjustable controller and the SOC and SOH updates are observed. The ability of batteries to store energy and their lifetime are reduced when electrodes age [23]. Because the variety of factors that explain battery health is so broad, any features that describe cell health could be used to characterise SOH.

3 Objectives of the current study

As explained in section 1, the objective of this project is to estimate the Remaining Operational Time(ROT) of the battery. The ROT of the battery is defined as the operable time of the battery under discharge condition from the current scenario without reaching over-discharge condition. The two major factors that affect the operational time of the battery are:

1. State of Charge
2. State of Health

At any given time, the State of Charge (SoC)[24] is the percentage of charge available in the battery relative to the total charge available in the battery when fully charged. A battery's SoC at any given time varies from 0 to 1, without 1 indicates that the battery is fully charged and 0 indicates that the battery is completely discharged. The SoC can be mathematically expressed as:

$$SoC = \frac{Q_{available}}{C_N} \tag{1}$$

where $Q_{available}$ represents the available amount of charge and C_N represents the rated capacity from battery manufacturers.

The State of Health (SoH)[24] is a method of determining the battery's age. At any given cycle, the SoH of a battery represents the battery's actual condition in comparison to its nominal circumstances. The major feature that characterizes the battery SoH is the real capacity of the battery, which decreases as the number of charge/discharge cycles increases. SoH can be generally defined as:

$$SoH = \frac{C_{max,p}}{C_N} \tag{2}$$

where $C_{max,p}$ denotes the maximum practical capacity as determined from the operational battery at the moment of measurement. $C_{max,p}$ may fade over time as a result of battery ageing.

The following factors can be used for the Neural Network framework to intrinsically calculate SoH and SoC:

1. Battery percentage
2. Current dissipation from battery

3. Remaining voltage of battery
4. Battery temperature, Ambient temperature
5. Rotor speed(in case of drone data acquisition)
6. Rotor signals(in case of drone data acquisition)

4 Data acquisition and available Li-ion battery

The framework for the Remaining Operational Time(ROT) estimation need to be robust for successful application in different ambient conditions and should be prone to any defects in other parts of the system. To evaluate the robustness of various algorithms for ROT estimation, the algorithms are evaluated in two different varieties of dataset namely, **NASA battery dataset[1]** and **Esperimental Drone battery dataset**(this dataset is acquired for this study).

4.1 NASA Battery dataset

This data set was obtained using a customised battery auguring benchmark at NASA's Ames Prognostics Centre of Excellence (PCoE)[1]. Li-ion battery (LIB) cells were charged and discharged, and Electrochemical Impedance Spectroscopy was used to capture those operational profiles at various temperatures. The data was gathered from commercial Li-ion 18650 rechargeable batteries. Charging was done in constant current (CC) mode until the battery voltage reached a base voltage, then in constant voltage (CV) mode until the charge current dropped to 20mA. Discharge was performed at a continuous current level of 2A until a rated threshold voltage for the specified battery was attained. The impedance was measured using an electrochemical impedance spectroscopy (EIS) frequency sweep from 0.1Hz to 5kHz. Repeated charge and discharge cycles accelerate battery ageing, whereas impedance measurements provide insight into internal battery properties that change as the battery ages. The time at which the threshold voltage is reached and the constant current discharge process is terminated is specified as the threshold operating condition for this data set. The estimate algorithms are tested using three different types of batteries from the NASA dataset.

Figure 1: DJI Mavic Mini 2 drone

4.2 Drone Battery dataset

This data set was acquired using DJI Mavic Mini 2 drone where the drone battery is a Lithium-Polonium(Li-Po) 2S type battery. The battery has a ratged capacity of 2250 mAh, rated voltage of 7.7 V and maximum charging power of 29W. The Li-Po battery cells were run in a charge and discharge operational profiles and various features like battery current dissipation, battery voltage, battery temperature, etc., were recorded. Charging was carried out using a fast charger at 2A and 9V with power input rated at 18W. Discharge was carried out at a by running the drones at various ambient conditions. The data was recorded at a frequency of 10Hz during the discharge cycle. The threshold operational condition for this data set is defined as the time when the battery percentage reaches a threshold value pre-set for the battery.

5 Data Analysis

As section 4 explains the setup process for the data acquisition process for the two data sets, in this section we look at the data acquired and perform deep analysis on the features available in the dataset.

5.1 NASA Battery dataset[1]

The discussed dataset on the Li-ion battery obtained by the NASA Ames consists of data for various different batteries in regards to battery capacity. For the purpose of experimentation and evaluation, two different sets of data are considered in this research.

5.1.1 Set 1: B0045, B0046, B0047, B0048

The discharge was performed at a constant current (CC) level of 1.5A until the threshold voltage was attained. For batteries B0045, B0046, B0047, and B0048, the threshold voltage for determining the threshold operational condition is pre-set at 2.0V, 2.2V, 2.5V, and 2.7V, respectively. When the batteries hit end-of-life (EOL) standards, which is specified as a 30% decrease in rated capacity, the tests were halted (from 2Ahr to 1.4Ahr). During the discharge operational profile, the following features are collected to enable remaining operational time(ROT) estimation:

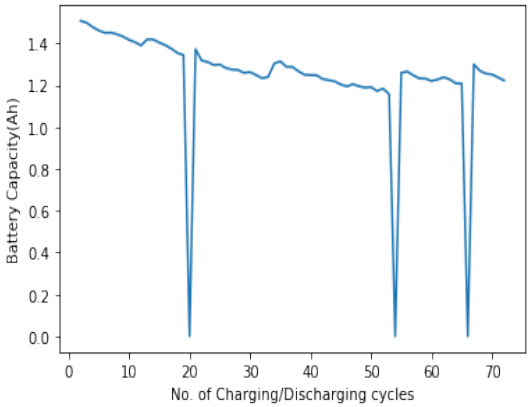
- Voltage at the battery terminals (Volts)
- Current drawn from the battery (Amps)
- Temperature of the battery (degree C)
- Current monitored at load (Amps)
- Voltage monitored at load (Volts)
- Cycle's time vector (secs)
- Capacity(Ahr) of the battery
- Temperature of the ambient settings(degree C)

The plots depicting the variation in the **Battery voltage, Battery current, , Battery temperature and Battery capacity** over a discharge run and across various discharge runs of battery B0048 are shown in fig. 2.

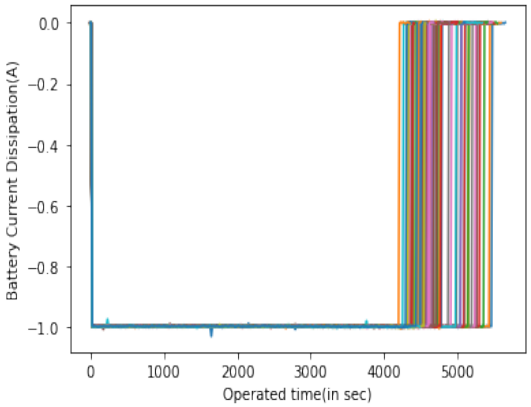
The Pearson's correlation matrix[25] of the data denoting the linear relationship between each of the individual input features and the prediction, i.e., remaining operation time is shown in the correlation matrix in fig. 3

5.1.2 Set 2: B0033, B0034, B0036

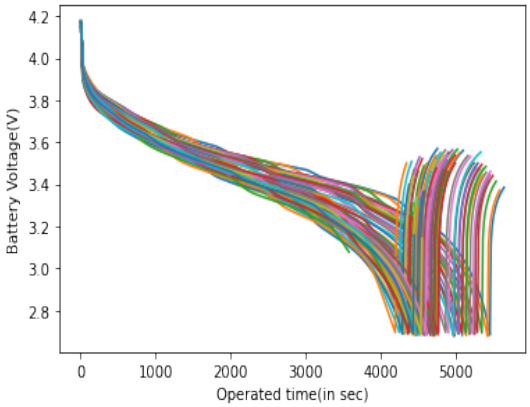
The discharge was performed at a constant current (CC) level of 4A until the threshold voltage was attained for batteries B0033 and B0034, and at 2A for battery B0036. For batteries B0033, B0034, and B0036, the threshold voltage for determining the threshold operational condition is pre-set at 2.0V, 2.2V, and 2.7V, respectively. When the batteries hit end-of-life (EOL) standards, which is specified as a 20%



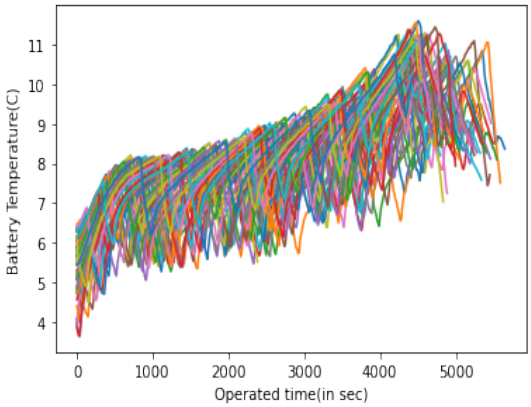
(a) Battery capacity of discharging cycles



(b) Battery Current of different discharge cycles



(c) Battery Voltage of different discharge cycles



(d) Battery Temp of different discharge cycles

Figure 2: The trend of different features for various discharge runs for battery B0048

decrease in rated capacity, the tests were halted (from 2Ahr to 1.6Ahr). The extracted features remain the same as discussed for Set 1.

The plots depicting the variation in the **Battery voltage, Battery current, Battery temperature and Battery capacity** over a discharge run and across various discharge runs of battery B0033 are shown in fig. 4.

The Pearson's correlation matrix[25] of the data denoting the linear relationship between each of the individual input features and the prediction, i.e., remaining operation time is shown in the correlation matrix in fig. 5

5.2 Drone dataset

This is an experimental dataset, that was acquired as a part of this project for battery analysis. For the drone dataset, the charge operational profile was carried out with a 11V and 3A rated fast charger with

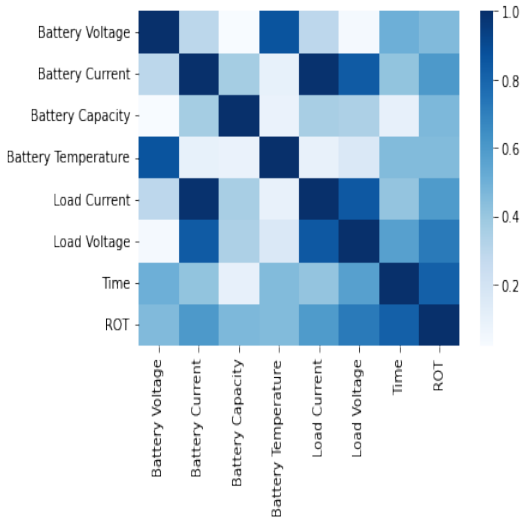
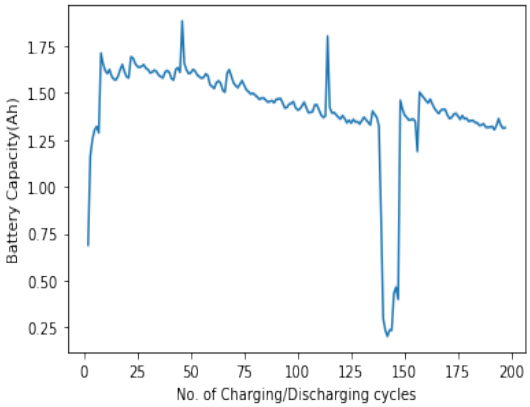


Figure 3: Correlation matrix between input features and ROT for NASA Set 1

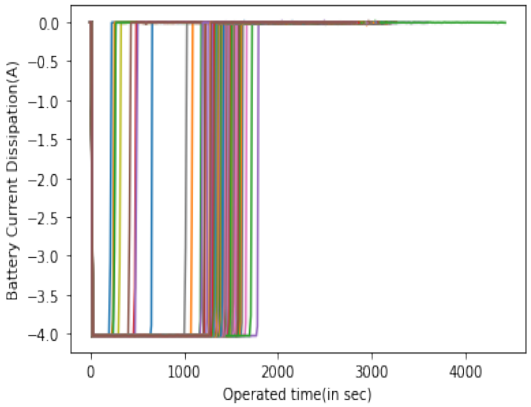
a power rating of 33W. The charge and discharge operations are carried out for a series of 12 runs and the data collection process is carried out at a frequency of 10Hz. The threshold for determining the operational time of each discharge run is calculated using battery percentage. The remaining operational time is defined as the operable amount of time from the given instant before the battery percentage becomes 10%. During the discharge operational profile, the following features are extracted to enable remaining operational time(ROT) estimation:

- Percentage of battery unconsumed
- Battery temperature
- Battery voltage
- Current dissipation from battery
- Pitch and Roll angle of the drone
- Height from takeoff position
- Speed along X, Y and Z directions

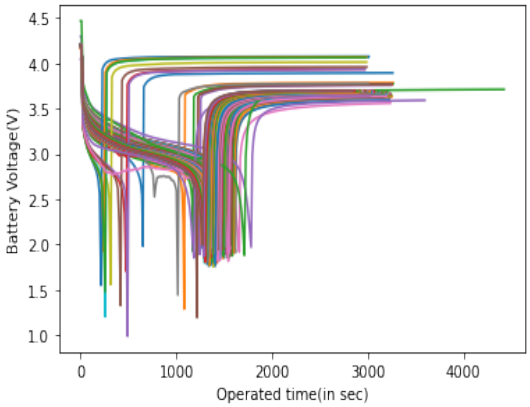
Note: *The above mentioned features are extracted from the metadata available in the drone using the software provided by Airdata UAV[26].*



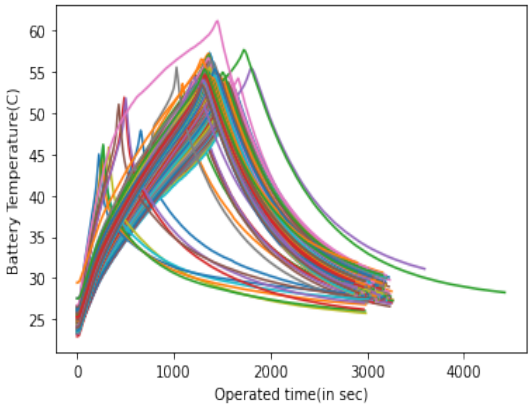
(a) Battery capacity of discharging cycles



(b) Battery Current of different discharge cycles



(c) Battery Voltage of different discharge cycles



(d) Battery Temp of different discharge cycles

Figure 4: The trend of different features for various discharge runs for battery B0033

The plots depicting the variation in the **Voltage measured, Current measured, Battery temperature and Battery percentage** over a discharge run and across various discharge runs of drone's battery are shown in fig. 6.

The Pearson's correlation matrix[25] of the data denoting the linear relationship between each of the individual input features and the prediction, i.e., remaining operation time is shown in the correlation matrix in fig. 7

6 Existing Frameworks

For estimation of remaining operational time(ROT) of the battery during a discharge run, various regression estimators/networks can be trained for the different sets of data acquired as explained in section 4. In this section, a famous regression estimators are described. These estimators will then be compared with

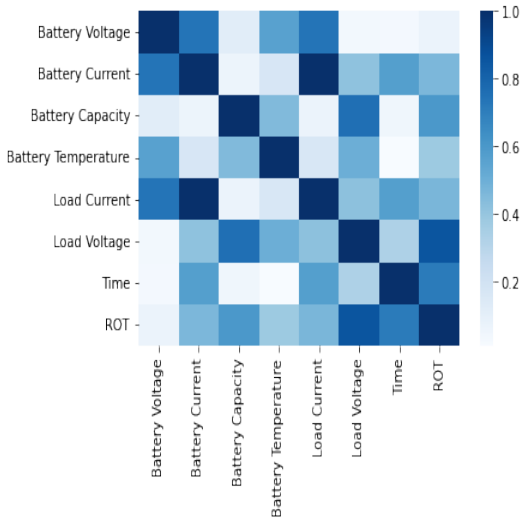
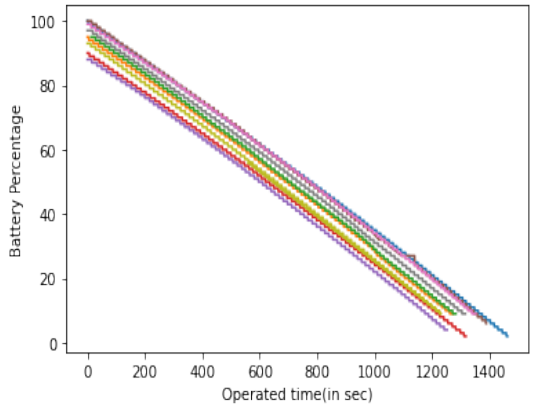


Figure 5: Correlation matrix between input features and ROT for NASA Set 2

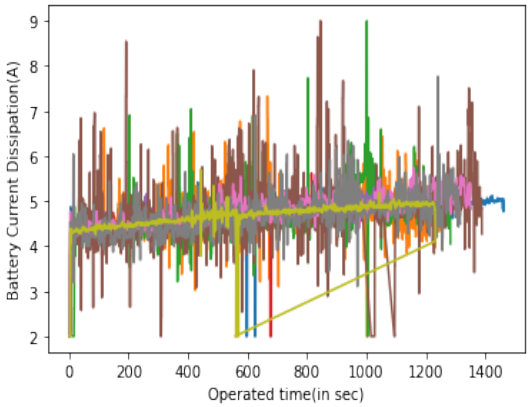
a novel estimator proposed in section 7 which was designed as a part of this research work. Comparisons of results between each of these models with the proposed framework is shown in section 8

6.1 Support Vector Machine(SVM)[2]

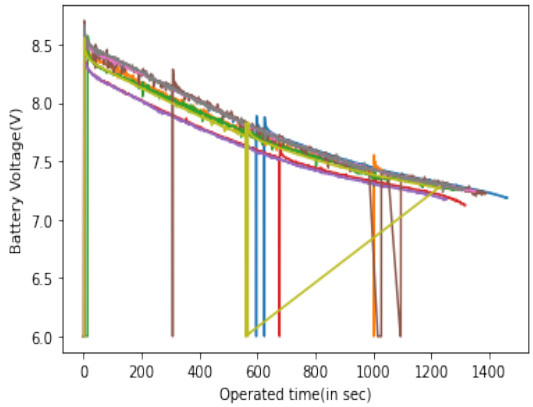
SVMs are supervised learning models that are used to accomplish classification and regression analysis. SVMs, which are based on statistical learning frameworks, was first developed by Vladimir Vapnik and colleagues at ATT Bell Laboratories[27]. An SVM training algorithm, similar to a non-probabilistic binary linear classification technique, trains a model that learns to assign samples to one of two classes, given a set of training samples, where each is categorised into one of the two classes. Training samples are assigned to Cartesian coordinates in space by SVM, which is then utilised to achieve maximum the distance between the two classes. Support Vector Regression operates on the same principles as SVMs. The fundamental notion of SVR is to determine the best fit line. The hyperplane that contains the majority of the sample points in the training dataset is considered the best fit line in SVR. Soft-margin SVMs are favoured over hard-margin SVMs in practice because they try to match the best line within a threshold value(C), making the algorithm immune to outliers and random noise[28]. SVR's threshold value is the distance between the hyperplane and the boundary line. SVRs may do nonlinear regression efficiently in addition to linear regression while using the kernel approach, which entails implicitly mapping their inputs into high-dimensional feature spaces.



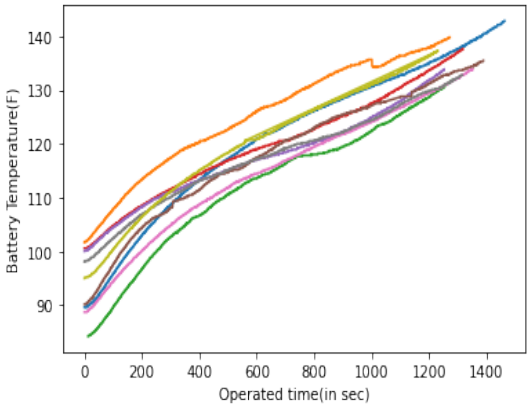
(a) Battery percentage of different discharging cycles



(b) Battery Current of different discharge cycles



(c) Battery Voltage of different discharge cycles



(d) Battery Temp of different discharge cycles

Figure 6: The trend of different features for various discharge runs for drone's battery
Note: The zones where current/voltage abruptly lowers are where the drone is landed and promptly lifted off. This is done to get diverse variety of data with visit points in the travel plan.

6.2 k-Nearest Neighbors(k-NN)[3]

The k-nearest neighbours algorithm (k-NN) is a non-parametric supervised learning approach invented in 1951 by Evelyn Fix and Joseph Hodges and later worked upon by Thomas Cover[29]. The k-NN algorithm learns the k nearest neighbours of the input from the given dataset and uses this to predict the output, which is defined as the average of the k nearest neighbours' values. An drawback with this averaging approach is that it treats all neighbours equally. This can be improved by assigning weights to the contributions of the neighbours, such that the farther neighbors contribute less to the average than the farther neighbors. A popular weighting strategy used when implementing k-NN is to assign a weight of $1/d$ to each neighbour, where d is the distance from the considered sample to the neighbour[30].

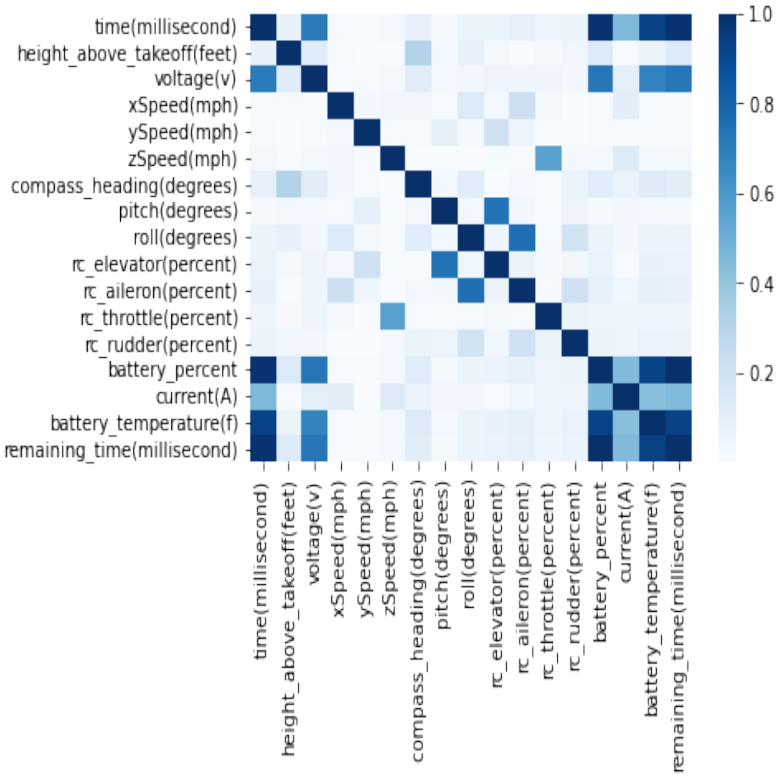


Figure 7: Correlation matrix between input features and ROT for Drone dataset

6.3 Decision Tree(DT)[4][5]

Decision tree uses a tree structure to develop regression or classification models. The algorithm creates a hierarchical structure that incrementally splits the training dataset into smaller and smaller groups of similar samples, while simultaneously creating an accompanying decision tree. Using this method, a tree data structure containing leaf nodes and decision nodes is constructed. A decision node typically includes two or more branches that each learn a characteristic value that represents the data points under consideration. A leaf node represents a numerical target choice that is assigned as the sample's output. The root node at the top is the tree's first decision node. Decision trees can handle both category and numerical data. The heuristic algorithm and the least squares approach are used in the heart of the regression tree to identify the best splitting node. Because the regression tree's basic method is the least squares method, the final fitting function is a piece-wise constant function; thus, the regression tree is also known as the least squares regression tree.

6.4 Random Forest(RF)[6]

Random Forest Regression is a supervised learning technique that does regression using the ensemble learning method. Tin Kam Ho developed the first random forest algorithm in 1995, utilising the random subspace method[31]. Random forest regression is a regression ensemble learning method that operates by creates a large number of decision trees during training. The random forest output is generated by averaging the predictions of individual trees for regression problems[32]. Random forests corrects the over-fitting tendency of the decision tree by decreasing the variance of the individual trees. Random forests outperform decision trees in general, but their accuracy is lower than that of gradient boosted trees. However, the properties of data characteristics can have an effect on performance. It often excels at a wide range of problems, particularly those with non-linear relationships. However, there is no interpret-ability, over-fitting is easily conceivable, and the number of trees to include in the model must be chosen.

6.5 Gradient Boosting Tree(GBT)[7]

The concept of gradient boosting arose from Leo Breiman's realisation that boosting can be viewed as an optimization technique on an appropriate cost function. Jerome H. Friedman later created explicit regression gradient boosting methods[33]. It uses multiple weak prediction models, generally decision trees which are employed as an ensemble to return a prediction model. When a decision tree is employed as the weak learner, the resulting algorithm is called gradient-boosted trees, and it usually outperforms random forest. A gradient-boosted tree model, like other boosting strategies, is generated iteratively while also generalising the previous approaches by permitting the optimization of a viable differentiable loss function. Fitting the training set too closely can reduce the model's capacity to generalise. Several regularisation strategies, by restricting the fitting operation, mitigate the over-fitting impact. The number of trees (M) in the model is a natural regularisation parameter[34]. Increasing M minimises training set error, but setting it too high may result in over-fitting. Monitoring prediction error on a different validation data set is frequently used to determine an ideal value of M. The depth of the trees is another regularisation parameter. The greater this value, the greater the likelihood that the model will over-fit the training data.

7 Novel Framework

In this section, a novel Attention based-Dense Convolutional Neural Network is proposed for the estimation of Remaining Operational Time(ROT). Various famous regression algorithms that can be implemented for the estimation of Remaining Operational Time(ROT) are explained in section 6. One major drawback that all the algorithms proposed in section 6 face is that those do not consider the temporal variation in the data. The regression techniques detailed in section 6 try to find a relationship between input features and the ground truth prediction. The inability to model the temporal variation of these features across time limits the performance of these algorithms and make them prone to outliers in the data which can be seen due to a variety of internal conditions of the battery or the ambient variation. The algorithm proposed in this research take into consideration the temporal variation of the features and estimates the ROT based on both the feature values and their variation over time.

7.1 Model architecture

The proposed algorithm aims to model the inter-relationship between the features, learn the local and global variation of the features across time. The ability of the algorithm to learn both global and local temporal variations in the feature maps is essential for estimation of ROT as the algorithm needs to inherently adapt the value of ROT when there are sharp changes absorbed due to battery/component defects in the system. In order to understand the time series data, the proposed network takes features from 64-consecutive timestamps as input for predicting ROT. Considering the above mentioned points, the algorithm designed can be split into three parts(as shown in fig. 8):

- **Dense layers** to learn inter-feature relationships
- **Convolutional layers(1D)** to learn the local variation of features across time
- **Attention Block** to learn the global variation of features across time

7.1.1 Dense layers

As depicted in fig. 3, fig. 5 and fig. 7 features like battery voltage, battery current, battery temperature, time and battery percentage have high linear correlation with the output, i.e., remaining operational time(ROT). For learning this relationship between the features and ROT, we deploy a two-layer dense

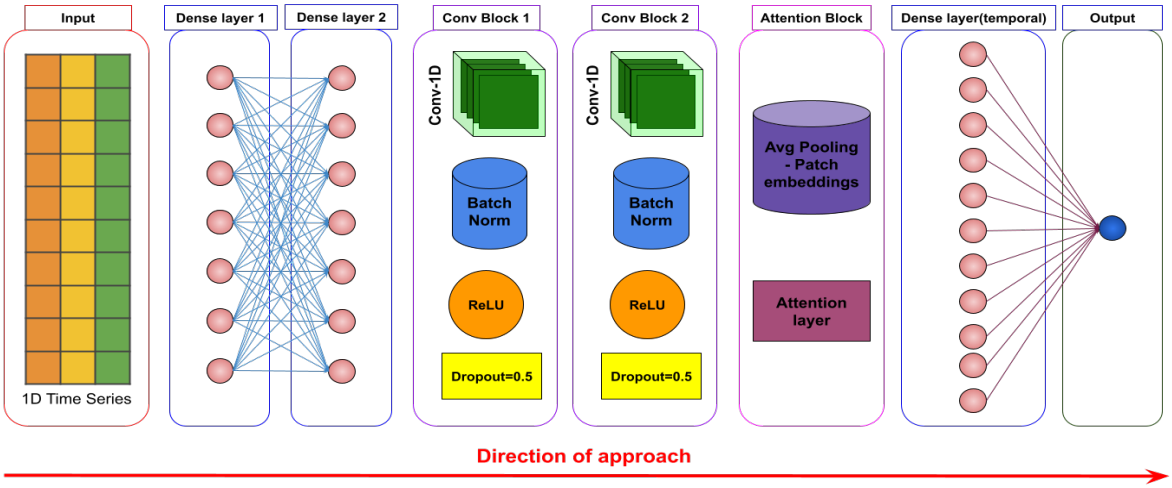


Figure 8: Architecture of the proposed model

neural network(also called Artificial Neural Network(ANN)). For the case of NASA dataset, the input layer takes in 8 features and outputs 32 different nodes. These nodes are then used as input to the second dense layer and this layer outputs 64 nodes. ReLU activation is used at the end of each of the dense layers. For the case of drone dataset, since the input has 31 features, the input layers outputs 64 nodes and the second layer outputs 128 nodes. The output at the end of these two dense layer is expected to have modelled the relationship between the features and the expected output. The output from the second layer is then passed through the 1D convolutional network for learning the temporal information from the time-series data.

7.1.2 Convolutional layer

As explained at the beginning of the section, it is important to analyze the temporal variation of the features in order to accurately estimate the remaining operational time(ROT). After obtaining the feature relationships from dense layer as explained in section 7.1.1, the network will now need to learn the temporal structure of the data. Convolutional layers can be used to model the temporal variation of these features as convolutional layers are basically sliding filter computed over the whole input sequence. Hence, convolutional layers are implemented with the input extracted from the dense layers. The sliding filter operation is performed in temporal direction on the time series data as this enables the algorithm to learn temporal variations in the data, i.e., slope of the features, concavity of the variation over time, etc. For this purpose, the output from the dense layer is passed to a series of two convolutional layers where each of the convolutional layers perform sliding operation in 1D(along time) with a filter size of

3 while maintaining the number of nodes. The output neurons at the end of the convolutional operation is expected to have learned the short-term temporal variations in the dataset and hence will be able to better estimate the remaining operational time.

7.1.3 Attention Block

The use of convolutional layers extracted temporal variations only within the filter size for features obtained at each timestamp. For example, with a filter size of 3, the use of two convolutional layer spans across the features from two timestamps before and after the chosen timestamp. Hence, the use of convolutional layer alone limits the degree of temporal information that the network can utilize as the convolutional layers have a local receptive field. So, in order to develop an improved understanding of the temporal variation in the given 64-timestamp input, we use an attention aided techniques to understand the correlation of temporal variation in each sub-patches(containing 8 timestamps) and provide the global temporal variation. The output from convolutional layer is given as input to the attention block. The attention block splits the temporal data into sub-patches as shown in fig. 9 where each patch contains feature data from 8 consecutive timestamp. The features from sub-patches are passed through an adaptive average pooling layer from which we get one feature representation for each sub-patch, named as $F_1, F_2, ..., F_p$ as shown in fig. 9. These sub-patch representation are expected to contain the input feature relationship extracted using dense layer as explained in section 7.1.1 and local temporal information extracted using convolutional layers as explained in section 7.1.2. Now these sub-patch features are passed as input to the attention layer. The attention mechanism followed in this study is similar to the attention mechanism used in [35]. The foundation of the attention mechanism, as depicted in fig. 9, is a simple feed-forward neural network (FNN) with one hidden layer and an output layer consisting of one output neuron, followed by a softmax layer. The FNN takes the feature map f_i from the CNN and attends to those feature maps and outputs a reward value v_i for each of these feature maps which represents the battery information extent of the latter. Subsequently, the reward values v_i are passed through a softmax function that normalizes the rewards to output a probability distributiona_i. The output of the attention mechanism corresponds to the probability distribution a_i of the importance of the corresponding feature vector f_i in the estimating ROT and indicates the significance of the corresponding feature mapping f_i as it estimated from the attention mechanism. Finally, an output representation with high level temporal information is obtained by performing weighted summation of the sub-patch features f_i using the estimated attention a_i.

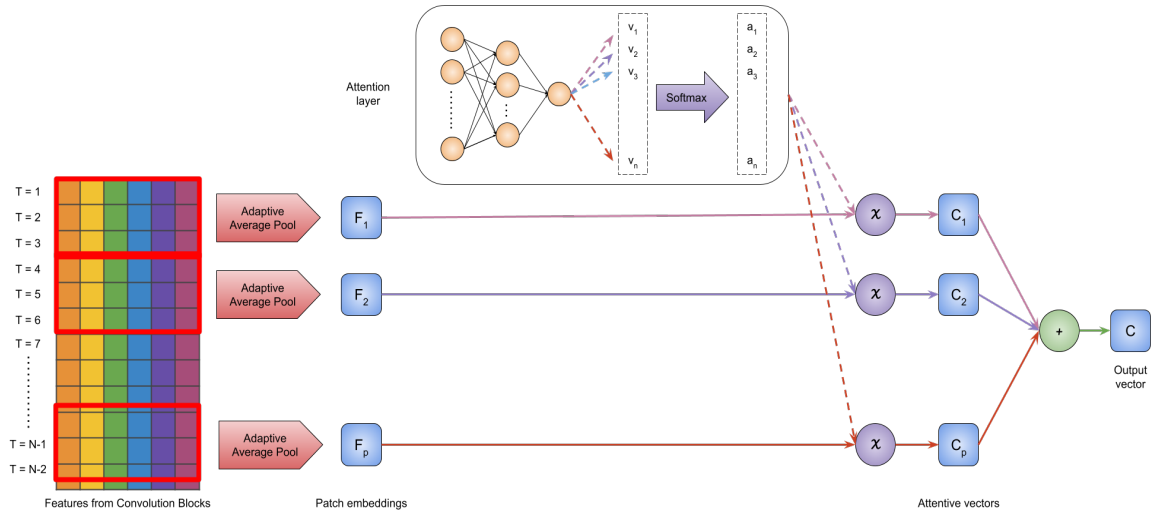


Figure 9: Attention mechanism based on feed-forward NN

The output of the attention block is then passed through a dense layer to estimate the ROT of the battery.

7.2 Loss function

An optimal loss function is necessary for training the proposed algorithm to estimate accurate ROT. The most popular loss functions used for regression problems are Mean Square Error(MSE) and Mean Absolute Error(MAE). MAE is used to minimise the error, which is calculated as the sum of all the absolute discrepancies between the true and predicted values. The MAE loss can be mathematically represented as,

$$L_{MAE} = \sum_{i=1}^{N} |y_{true} - y_{pred}| \tag{3}$$

MSE is used to minimise error, which is defined as the sum of all squared deviations between the true and predicted values. The MSE loss can be mathematically represented as,

$$L_{MSE} = \sum_{i=1}^{N} (y_{true} - y_{pred})^2 \tag{4}$$

In generally, MSE is preferred in most of the cases to train the algorithm. But when the outliers are present in the dataset, then the MSE does not perform well. However, when outliers are present in the dataset, the MSE performs poorly. The reason for this poor performance is that if the dataset contains outliers, the evaluation of squared differences results in a considerably greater inaccuracy. In addition

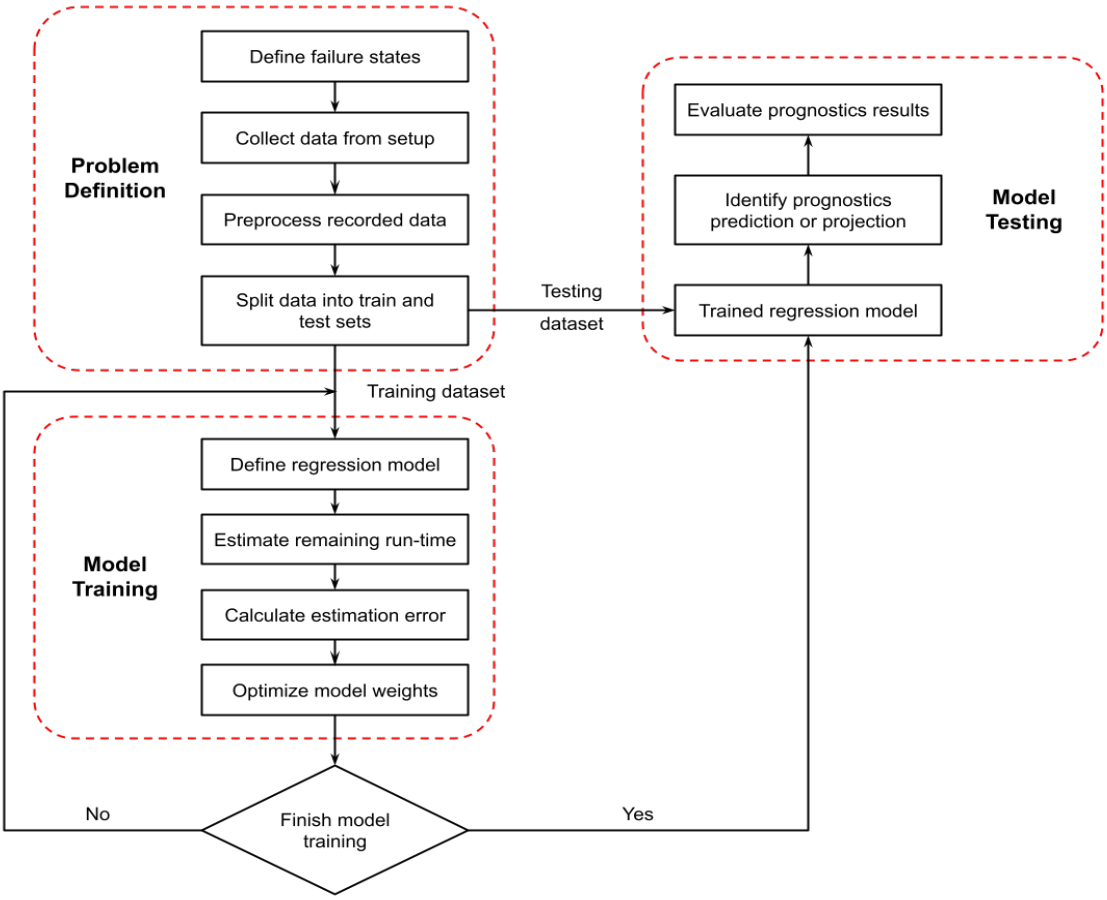


Figure 10: Overall implementation flow-chart of the algorithm

to that MSE tends to care more about higher loss values and does not operate with small losses with magnitude less than 1. This becomes essential as the network should be robust enough to predict accurate ROT even when the ground truth is of smaller values. In contrast, MAE loss gives equal weightage to all samples and hence performs better with small magnitude losses as well and enable the network to learn better by pushing the loss close to zero value. To benefit from the advantages of both the loss functions, a weighted combination of these two loss functions is used to train the algorithm. The final loss function used to train the algorithm is defined as,

$$L_{final} = \lambda_1 L_{MAE} + \lambda_2 L_{MSE} \tag{5}$$

where λ_1 and λ_2 are given values 2 and 1 respectively.

7.3 Implementation procedure

In this section, details about the implementation of the algorithm are detailed, refer fig. 10. The complete process can be divided into three parts, problem formulation + data analysis, training the algorithm and evaluation of the algorithm. The process of data collection and data analysis are carried out as detailed in section 4 and section 5. For evaluation of the various algorithms, each of the data sets are split in train and test sets. For NASA dataset's set 1, we use battery B0006, B0007 and B0018 as the training set and later test the model on battery B0005. For set 2, battery B0034 and B0036 are used for training and B0033 is used for testing purpose. While for set 3, we used battery B0048 for testing and B0045, B0046, B0047 for training the algorithm. Finally, for the drone dataset, we chose 3 of the 12 runs for testing the algorithm and the other 9 runs are used for training the algorithm. The three runs are chosen considering the batteries aging in order to ensure that the algorithms are robust at estimation of ROT.

After the problem formulation and splitting of dataset, the proposed algorithm details in section 7.1 will be trained on the training dataset using the loss functions defined in section 7.2. At every training iteration, the algorithm estimates the predicted ROT using the feature provided as inputs. This prediction is then compared with the ground truth and the estimation error is computed. The estimated error is then used to fine-tune the algorithm based on Gradient Descent technique. This process of training the algorithm is continued until the error between the predicted and ground truth ROT saturates. We select as optimizer the stochastic gradient descent(Adam with decoupled weight decay). The learning rate is set to 0.001, and the Nesterov momentum is activated, with weight decay and momentum parameters set to $5 \cdot 10^{-4}$ and 0.9, respectively. The model is trained for 200 epochs with a batch size of 16.

Once the training of the algorithm is saturates, the algorithm will then be used to predict the remaining operational time(ROT) in the test set. The prediction in the test set will be used to evaluate the algorithm and also to aid the decision making process.

The proposed models are written using Python 3.7 programming language, using the Pytorch framework for ANN and Sklearn framework for ML algorithms, and the experiments are run on Windows 10 with Nvidia RTX 3080 GPU.

8 Results and discussion

In this section, we take a look at the results for each of the data sets explained in section 4 and compare the proposed algorithm with the existing algorithms. As explained in section 4, we use four different

dataset for training and evaluating eacch algorithm. We test each of the exisiting algorithm and the algorithm proposed in section 7 on each of these datasets and analyze the generalizing ability of these algorithm across varieties of data. For evaluating the performance and accuracy of these techniques, three common scoring metrics are considered namely,

- **Mean Absolute Error(MAE)** is a popular basic error measurement. MAE is the average of absolute error, which is mathematically represented as,

$$E_{MAE} = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i| \tag{6}$$

  When comparing data of different scales, MAE provides little insight into the differences. It also treats large errors and small errors the same way.

- **Root Mean Square Error(RMSE)** is an other widely used standard measure of error. RMSE is the root of mean of quadratic error which can be mathematically represented as,

$$E_{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2} \tag{7}$$

  RMSE treats large errors with high weightage and hence gives low weightage to small scale data and their corresponding error.

- **Mean Absolute Percentage Error(MAPE)** is the absolute error normalized over the data and is mathematically represented as

$$E_{MAPE} = \frac{1}{n}\sum_{i=1}^{n}100 * \frac{|y_i - \hat{y}_i|}{y_i} \tag{8}$$

  MAPE is computed and averaged over each data point, capturing additional errors and outliers. MAPE compares error across data with different scales since it gives equal weightage to large and small scale data.

Note: *Small and large scale here implies the values of ROT. When we say small scale ROT, we imply that the remaining operational time is low(like 1-5 mins) and large scale ROT means we have 20-40 mins of operational time.*

Algorithm	Model description
k-NN	25-Nearest Neighbor model is used with uniform weights. The model is trained with 60,000 examples of 16 dimensions
SVM	SVM regression model with Radial Basis Function(RBF) kernel is used with inverse regularization parameter of 10. The model is trained with 60,000 examples of 16 dimensions
RF	Random Forest model is made up of 100 decision trees with the maximum depth of each tree restricted to 9. The model is trained with 60,000 examples of 16 dimensions
GBT	XGBoost model is made up of 100 decision trees with the maximum depth of each tree restricted to 6. In addition, each tree learns with 80% of the samples from the training set and performs split based on 80% of the features. The model is trained with 60,000 examples of 16 dimensions
Proposed model	The proposed model is made up of two dense layers with outputs a 64-feature representation. The feature representation is then passed through two CNN layers and Attention Block for final estimation.

Table 1: Models description for each algorithm trained on Drone dataset

In case of this study, the focus is on estimating ROT to enable easy planning of travel. For this, we require accurate ROT calculations on large and, more importantly, small scales, because travel must be carefully scheduled before the battery runs out.

Since, the comparison of the algorithms are carried out on four different sets of data, each with different scales, MAPE is the preferred evaluation metric chosen for this study. The scale of the values varying with a discharge cycle and also across discharge cycles also makes an compelling argument for using MAPE as the main evaluation metric.

8.1 Drone dataset

In this subsection, we compare the performance of the proposed algorithm with the exisiting algorithms on three test runs(refer to section 4.2) of the Drone dataset. The parameter for each of the models that were observed to give the best performance is denoted in table 1.

The estimated ROT and the actual ROT, i.e., ground truth(GT) as shown in fig. 11 for the first validation run. As can be seen from the figure, the estimation from the proposed algorithm is more accurate in comparison to the prediction from the existing model, namely, SVM, k-NN, RF and GBT. The regions where the proposed algorithm is consistent with ground truth while the existing algorithms

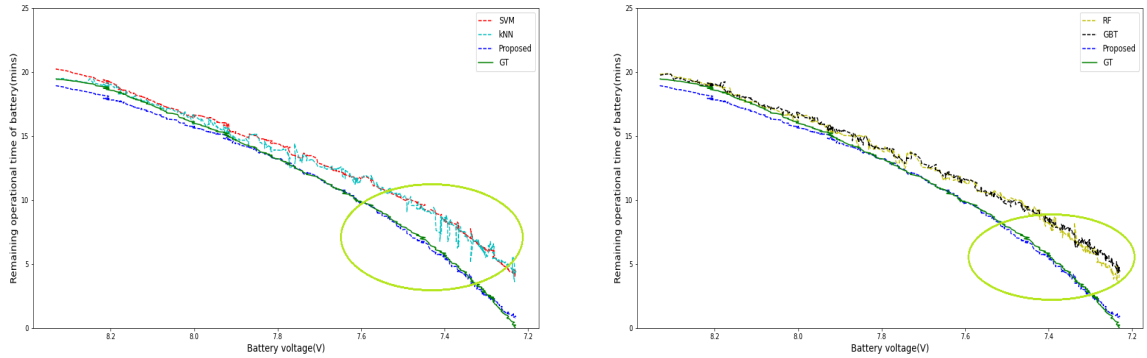


Figure 11: Comparison of ROT estimation for validation run-1

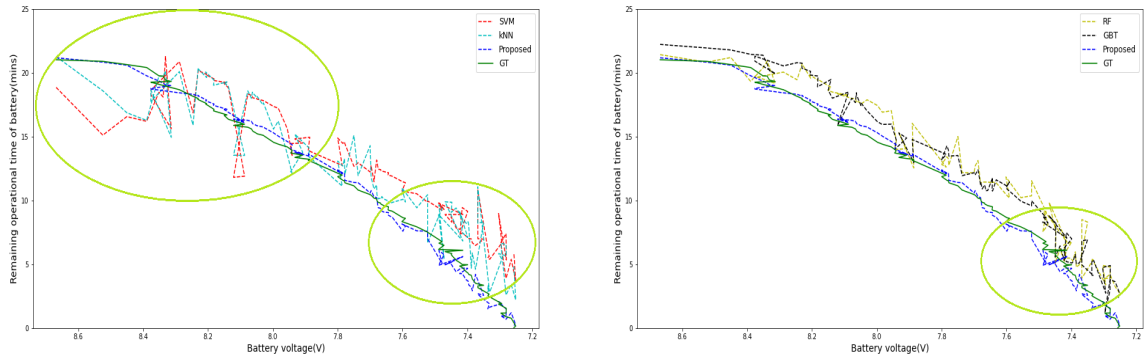


Figure 12: Comparison of ROT estimation for validation run-2

highly deviate are highlighted in the figure. The fact that the proposed algorithm outperform the existing algorithms in different scales(for small as well as large actual ROTs) can be noticed in fig. 11. The ability to perform accurate estimation for small scales in a crucial factor for battery ROT as this defines the time in which the battery becomes inoperable.

Similar observations are observed in the other two validation runs as well, refer to fig. 12, fig. 13. Like in the first run, the proposed algorithm is able to perform accurate estimations for different scales of ROT and also predicts smoother outputs which is not observed in case of SVM, k-NN, Random Forest and Gradient Boosting Trees.

The performance matrices for quantitative analysis has be reported in table 2, which also supports the observations noted in the plots, i.e, the proposed algorithm outperforms the existing methods and produces accurate estimation in different scales.

Error	SVM	k-NN	RF	GBT	Proposed
MAPE	222.504%	202.4%	**162.028%**	165.642%	**23.849%**

Table 2: Performance metrics for ROT estimation in drone dataset

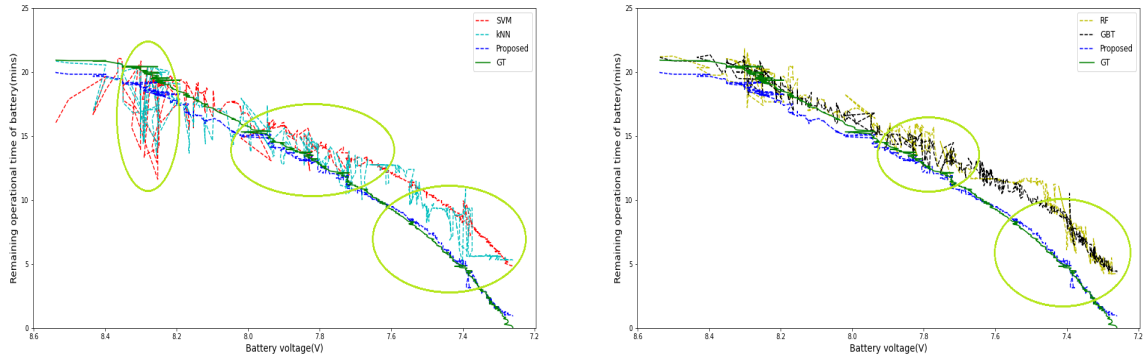


Figure 13: Comparison of ROT estimation for validation run-3

8.2 NASA Set 1

Similar to the previous section, in this subsection we compare the estimations of the different models on validation set of NASA dataset, set-3, i.e., battery B0048. The parameter for each of the models that were observed to give the best performance is denoted in table 3.

As can be seen from fig. 14, the proposed algorithm consistently outperforms the other existing estimation techniques and is robust to sharp changes in actual ROT due to external and internal parameters varied in the battery. The regions where the proposed algorithm produces high accuracy estimation while the existing methods lack accuracy are highlighted in fig. 14.

Further in depth analysis on how the performance of the algorithm gets affected on aging of the battery can be obtained by estimation of ROT for different cycles where each of these cycles are chose at different stages of battery age. For this purpose, estimation of ROT on three different cycles are shown in fig. 15, fig. 16 and fig. 17. Here, fig. 15 denotes the estimation of ROT during the first discharge cycle of the battery, fig. 16 denotes the estimation of ROT during the middle discharge cycle and fig. 17 denotes the estimation of ROT during the final discharge cycle. As can be seen from these figures, the proposed model is more robust towards various stages of battery aging when compared to other algorithms.

The performance matrices for quantitative analysis has be reported in table 4, which also supports the observations noted in the plots.

8.3 NASA Set 2

Here we compare the estimations of the different models on validation set of NASA dataset, set-2, i.e., battery B0033. The parameter for each of the models that were observed to give the best performance is denoted in table 5.

Algorithm	Model description
k-NN	25-Nearest Neighbor model is used with uniform weights. The model is trained with 60,000 examples of 8 dimensions
SVM	SVM regression model with Radial Basis Function(RBF) kernel is used with inverse regularization parameter of 10. The model is trained with 60,000 examples of 8 dimensions
RF	Random Forest model is made up of 50 decision trees with the maximum depth of each tree restricted to 10. The model is trained with 60,000 examples of 8 dimensions
GBT	XGBoost model is made up of 10 decision trees with the maximum depth of each tree restricted to 6. In addition, each tree learns with 80% of the samples from the training set and performs split based on 80% of the features. The model is trained with 60,000 examples of 8 dimensions
Proposed model	The proposed model is made up of two dense layers with outputs a 32-feature representation. The feature representation is then passed through two CNN layers and Attention Block for final estimation.

Table 3: Models description for each algorithm trained on NASA dataset, set-1

Error	SVM	k-NN	RF	GBT	Proposed
MAPE	3.904%	**2.758%**	4.858%	3.144%	**2.245%**

Table 4: Performance metrics for ROT estimation in NASA set-1 dataset

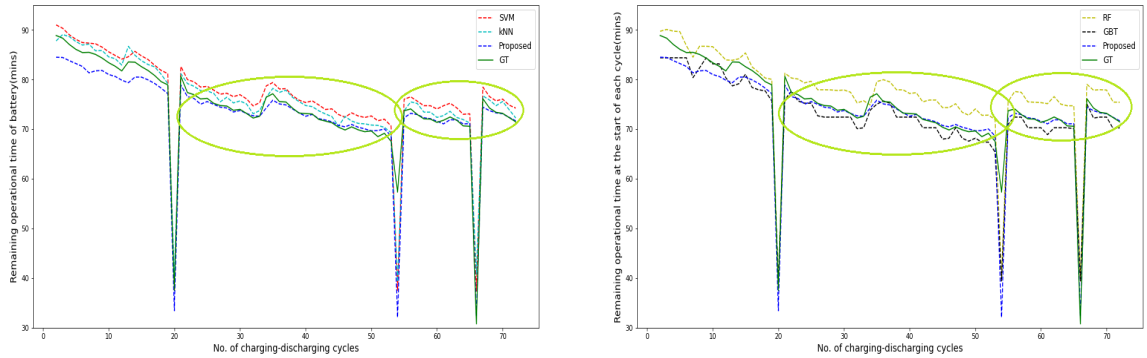


Figure 14: Comparison of ROT estimation for B0048 at the start of each discharge cycle as battery ages

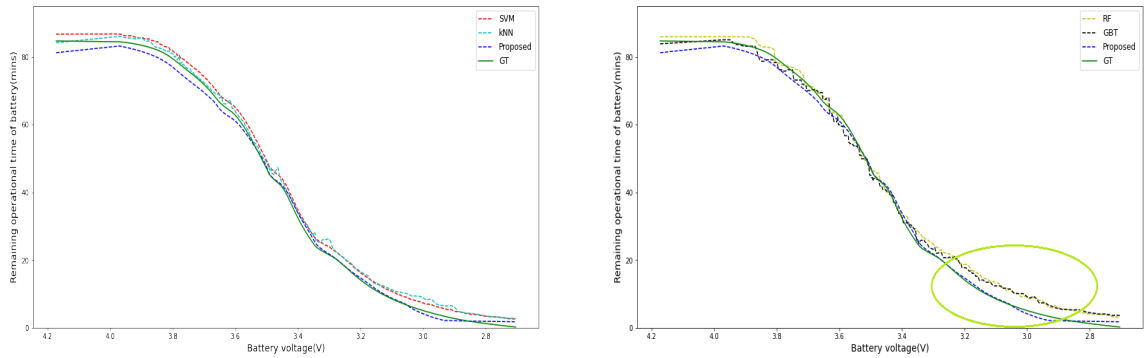


Figure 15: Estimation of ROT as the first discharge cycle progress for battery B0048

As can be seen from fig. 18, the proposed algorithm is robust to sharp changes in actual ROT due to external and internal parameters varied in the battery. The regions where the proposed algorithm produces high accuracy estimation while the existing methods lack accuracy are highlighted in fig. 18.

Further in depth analysis on how the performance of the algorithm gets affected on aging of the battery can be obtained by estimation of ROT for different cycles where each of these cycles are chose at different stages of battery age. For this purpose, estimation of ROT on three different cycles are shown in fig. 19 and fig. 20. Here, fig. 19 denotes the estimation of ROT during the first discharge cycle of the battery and fig. 20 denotes the estimation of ROT during the final discharge cycle. As can be seen from these figures, the proposed model is more robust towards various stages of battery aging when compared to other algorithms.

The performance metrics based on quantitative analysis has be reported in table 6, which also supports the observations noted in the plots.

Algorithm	Model description
k-NN	10-Nearest Neighbor model is used with uniform weights. The model is trained with 40,000 examples of 8 dimensions
SVM	SVM regression model with Radial Basis Function(RBF) kernel is used with inverse regularization parameter of 100. The model is trained with 40,000 examples of 8 dimensions
RF	Random Forest model is made up of 25 decision trees with the maximum depth of each tree restricted to 10. The model is trained with 40,000 examples of 8 dimensions
GBT	XGBoost model is made up of 100 decision trees with the maximum depth of each tree restricted to 9. In addition, each tree learns with 80% of the samples from the training set and performs split based on 80% of the features. The model is trained with 40,000 examples of 8 dimensions
Proposed model	The proposed model is made up of two dense layers with outputs a 32-feature representation. The feature representation is then passed through two CNN layers and Attention Block for final estimation.

Table 5: Models description for each algorithm trained on NASA dataset, set-2

Error	SVM	k-NN	RF	GBT	Proposed
MAPE	35.647%	**18.811%**	22.155%	19.978%	**10.518%**

Table 6: Performance metrics for ROT estimation in NASA set-2 dataset

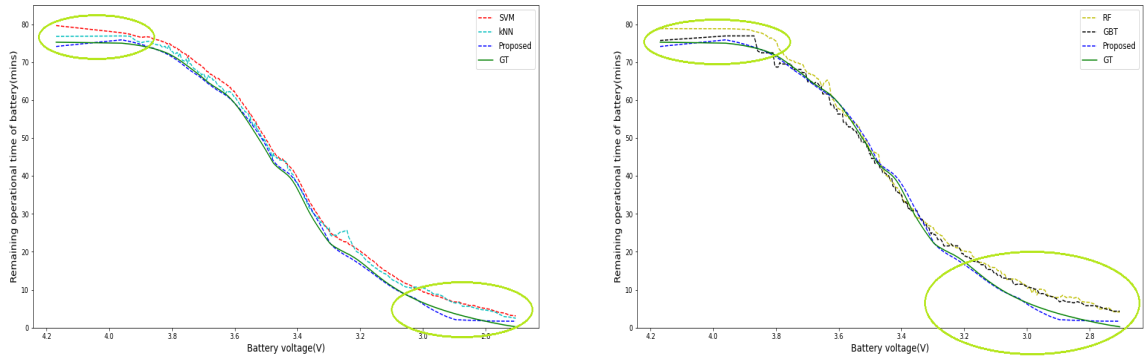


Figure 16: Estimation of ROT as the middle discharge cycle progress for battery B0048

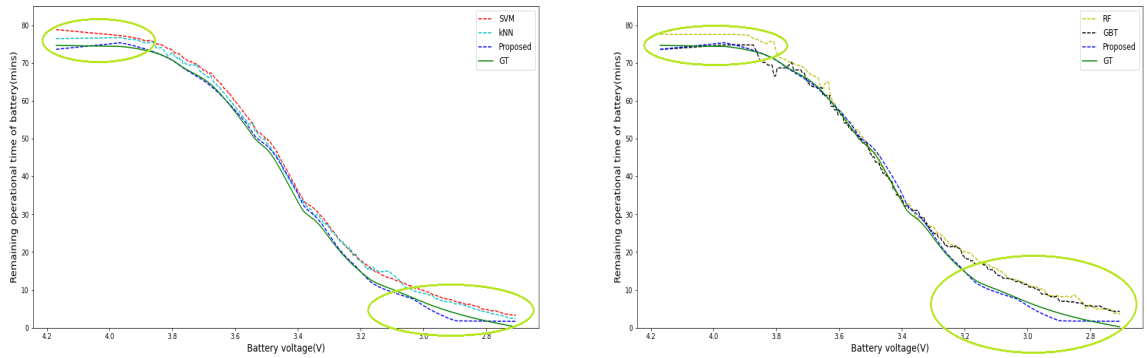


Figure 17: Estimation of ROT as the final discharge cycle progress for battery B0048

9 Conclusion

One of the most important part of prognostics is decision making. In the present work, the estimation of remaining operational time of the battery is used to take decision as to whether the desired task/mission could be completed before battery run-out or not. For this the estimation algorithms are expected to predict ROT accurately irrespective of available battery percentage, meaning the algorithm is supposed to be robust for different SoC and SoH of the battery. As discussed in section 8, the proposed algorithm estimates the ROT accurately for different scales, under varied SoC and SoH conditions. This improves the predictability of the battery condition and aids in making suitable decisions regarding the task in hand. Hence, the proposed algorithm in this research, uses temporal information to learn the changes in the battery across time and utilizes that in-coherence with the available features for estimating accurate remaining operational time. The proposed architecture is robust for estimation of ROT for varied batteries from the NASA battery dataset and Drone battery dataset. Hence this architecture can be generalized and used for varied battery types.

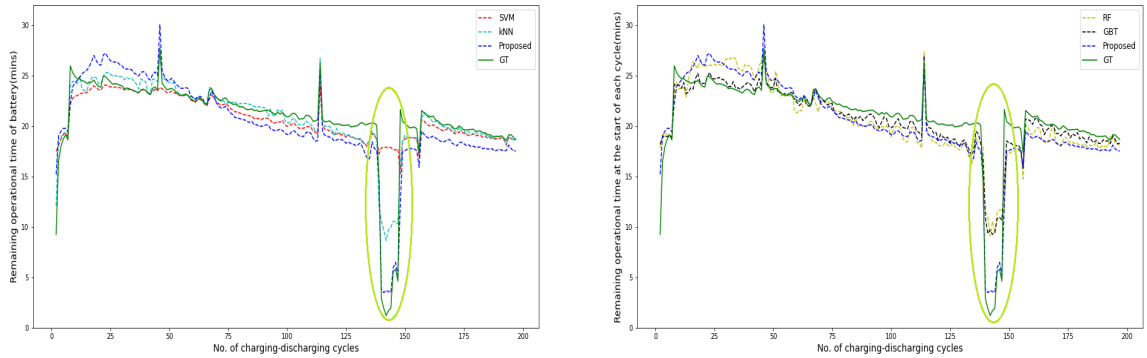


Figure 18: Comparison of ROT estimation for B0033 at the start of each discharge cycle as battery ages

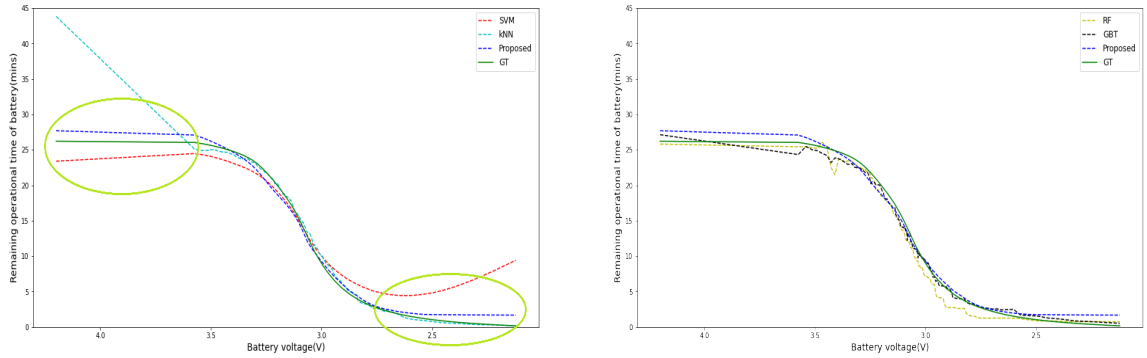


Figure 19: Estimation of ROT as the first discharge cycle progress for battery B0033

References

[1] Nasa's open data portal: Li-ion battery aging datasets. `https://data.nasa.gov/dataset/Li-ion-Battery-Aging-Datasets/uj5r-zjdb`.

[2] Evgeniou, T., and Pontil, M., 2001. *Support Vector Machines: Theory and Applications*. Springer Berlin Heidelberg, Berlin, Heidelberg, pp. 249–257.

[3] Guo, G., Wang, H., Bell, D., and Bi, Y., 2004. "Knn model-based approach in classification".

[4] Quinlan, J., 1985. "Induction of decision trees.".

[5] Wikipedia: Decision tree. `https://en.wikipedia.org/wiki/Decision_tree`.

[6] Ali, J., Khan, R., Ahmad, N., and Maqsood, I., 2012. "Random forests and decision trees". *International Journal of Computer Science Issues(IJCSI),* ***9***, 09.

[7] An introduction to gradient boosting decision trees. `https://www.machinelearningplus.com/machine-learning/an-introduction-to-gradient-boosting-decision-trees/`.

[8] Hasib, S. A., Islam, S., Chakrabortty, R. K., Ryan, M. J., Saha, D. K., Ahamed, M. H., Moyeen, S. I., Das, S. K., Ali, M. F., Islam, M. R., Tasneem, Z., and Badal, F. R., 2021. "A comprehensive review of available battery datasets, rul prediction approaches, and advanced battery management". *IEEE Access,* ***9***, pp. 86166–86193.

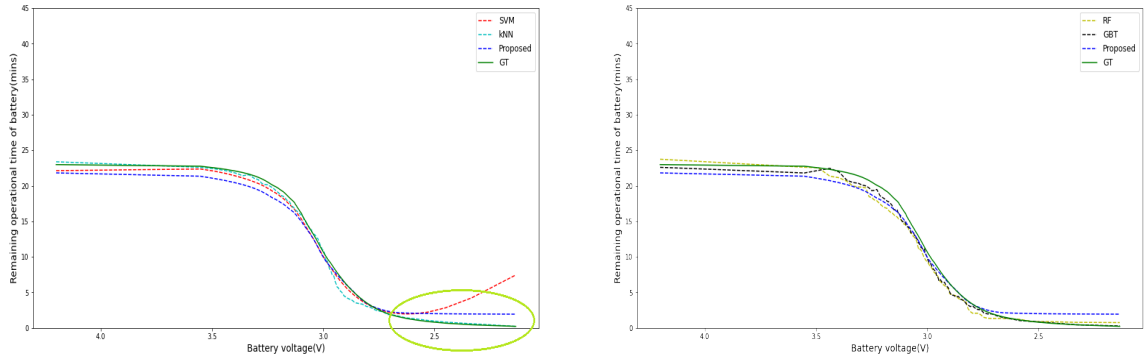


Figure 20: Estimation of ROT as the final discharge cycle progress for battery B0033

[9] Vehicles and greenhouse gases - epa's role. https://www.epa.gov/greenvehicles/vehicles-and-greenhouse-gases-epas-role.

[10] Trends and developments in electric vehicle markets. https://www.iea.org/reports/global-ev-outlook-2021/trends-and-developments-in-electric-vehicle-markets.

[11] Technavio: Electric unmanned aerial vehicle (e-uav) market. https://www.prnewswire.com/news-releases/electric-unmanned-aerial-vehicle-e-uav-market\OT1\textemdash35-of-growth-to-originate-from-north-america\OT1\textendashevolving-opportunities-with-aerovironment-inc\OT1\textendashairbus-se-17000-technavio-reports-301509007.html.

[12] Watrin, N., Blunier, B., and Miraoui, A., 2012. "Review of adaptive systems for lithium batteries state-of-charge and state-of-health estimation". In 2012 IEEE Transportation Electrification Conference and Expo (ITEC), pp. 1–6.

[13] G. K. Sierra Paez, M. Orchard, C. K., and Goebel, K., 2018. ""a hybrid battery model for prognostics in small-size electric uavs"". *PHM$_C$ONF*.

[14] Meng, J., Luo, G., and Gao, F., 2016. "Lithium polymer battery state-of-charge estimation based on adaptive unscented kalman filter and support vector machine". *IEEE Transactions on Power Electronics,* ***31***(3), pp. 2226–2238.

[15] Zheng, F., Xing, Y., Jiang, J., Sun, B., Kim, J., and Pecht, M., 2016. "Influence of different open circuit voltage tests on state of charge online estimation for lithium-ion batteries". *Applied Energy,* ***183***, pp. 513–525.

[16] Hu, X., Cao, D., and Egardt, B., 2018. "Condition monitoring in advanced battery management systems: Moving horizon estimation using a reduced electrochemical model". *IEEE/ASME Transactions on Mechatronics,* ***23***(1), pp. 167–178.

[17] Hu, X., Li, S., and Peng, H., 2012. "A comparative study of equivalent circuit models for li-ion batteries". *Journal of Power Sources,* ***198***, pp. 359–367.

[18] Zou, C., Hu, X., Dey, S., Zhang, L., and Tang, X., 2018. "Nonlinear fractional-order estimator with guaranteed robustness and stability for lithium-ion batteries". *IEEE Transactions on Industrial Electronics,* ***65***(7), pp. 5951–5961.

[19] Hu, X., Feng, F., Liu, K., Zhang, L., Xie, J., and Liu, B., 2019. "State estimation for advanced battery management: Key challenges and future trends". *Renewable and Sustainable Energy Reviews,* ***114***, p. 109334.

[20] Ladpli, P., Kopsaftopoulos, F., Nardari, R., and Chang, F.-K., 2017. "Battery charge and health state monitoring via ultrasonic guided-wave-based methods using built-in piezoelectric transducers". In Smart Materials and Nondestructive Evaluation for Energy Systems 2017, N. G. Meyendorf, ed., Vol. 10171, International Society for Optics and Photonics, SPIE, pp. 53 – 64.

[21] , 2017. "Probing lithium-ion batteries' state-of-charge using ultrasonic transmission – concept and laboratory testing". *Journal of Power Sources,* ***343***, pp. 536–544.

[22] Kim, Y., Samad, N. A., Oh, K.-Y., Siegel, J. B., Epureanu, B. I., and Stefanopoulou, A. G., 2016. "Estimating state-of-charge imbalance of batteries using force measurements". In 2016 American Control Conference (ACC), pp. 1500–1505.

[23] Xiong, R., Li, L., and Tian, J., 2018. "Towards a smarter battery management system: A critical review on battery state of health monitoring methods". *Journal of Power Sources,* ***405***, pp. 18–29.

[24] Khumprom, P., and Yodo, N., 2019. "A data-driven predictive prognostic model for lithium-ion batteries based on a deep learning algorithm". *Energies,* ***12***(4).

[25] Kirch, W., ed., 2008. *Pearson's Correlation Coefficient*. Springer Netherlands, Dordrecht, pp. 1090–1091.

[26] Airdata uav - drone data management and flight analysis. `https://airdata.com/`.

[27] Wikipedia: Support-vector machine. `https://en.wikipedia.org/wiki/Support-vector_machine`.

[28] Scikit learn: Support vector regression. `https://scikit-learn.org/stable/modules/generated/sklearn.svm.SVR.html`.

[29] Wikipedia: k-nearest neighbors. `https://en.wikipedia.org/wiki/K-nearest_neighbors_algorithm`.

[30] Scikit learn: k-nearest neighbors regression. `https://scikit-learn.org/stable/modules/generated/sklearn.neighbors.KNeighborsRegressor.html`.

[31] Wikipedia: Random forest. `https://en.wikipedia.org/wiki/Random_forest`.

[32] Scikit learn: Random forest regression. `https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.RandomForestRegressor.html`.

[33] Wikipedia: Gradient boosting. `https://en.wikipedia.org/wiki/Gradient_boosting`.

[34] Xgboost documentation. `https://xgboost.readthedocs.io/en/stable/`.

[35] Plakias, S., and Boutalis, Y. S., 2020. "Fault detection and identification of rolling element bearings with attentive dense cnn". *Neurocomputing,* ***405***, pp. 208–217.