Applying Machine Learning to predict the lifetime of Li-Ion Batteries

Submitted by

Akanksha Priya	**(19115014)**
Gursimran	**(19115057)**
Jaiyash Agrawal	**(19115063)**
Neha Gupta	**(19115078)**
Nischal Jain	**(19115081)**
Rishika Atal	**(19115101)**

Supervisor

Prof. Premalata Jena

Department of Electrical Engineering

Indian Institute of Technology Roorkee

2022

DECLARATION

I hereby declare that the work which is presented here, entitled **Applying Machine Learning to predict the lifetime of Li-Ion Batteries**, submitted for the completion, of course, EEN 300: Industry Oriented Problem. I also declare that I have been doing my work under the supervision and guidance of **Prof. Premalata Jena, Electrical Department, Indian Institute of Technology Roorkee**. The matter presented in this report is not submitted for the award of any other degree of institute or any other institutes.

Date: 19th April 2022

Akanksha Priya	**Gursimran**	**Jaiyash Agrawal**	**Neha Gupta**	**Nischal Jain**	**Rishika Atal**
19115014	**19115057**	**19115063**	**19115078**	**19115081**	**19115101**

CERTIFICATE

This is to certify that the above statement made by the candidate is true to the best of my knowledge and belief.

Prof. Premalata Jena

Associate Professor

Department of Electrical Engineering

Indian Institute of Technology Roorkee

ACKNOWLEDGEMENT

It is our privilege to express our sincerest thanks to our project supervisor, **Prof. Premalata Jena** for her valuable input, able guidance, encouragement, whole-hearted cooperation, and constructive criticism throughout the duration of our project.

We express our sincere gratitude to our Head of Department **Prof. G.N Pillai** for encouraging us and allowing us to present the project on the topic “**Applying Machine Learning to predict the lifetime of Li-Ion batteries**“ at our department premises.

We are also very grateful to the course coordinator **Prof. Abhishek Kumar Behra**, for his constant efforts and able guidance. We would like to thank all the teaching faculty in the Electrical Engineering Department.

We take this opportunity to thank all Teaching Assistants who have directly or indirectly helped our project. We pay our respects and love to our family members and friends for their encouragement throughout our careers. Last but not least we express our thanks to everyone whose input has directly or indirectly contributed to the success of this project.

ABSTRACT

The project aims to build machine learning models that can accurately predict lithium-ion batteries' lifetime. We used a publicly available dataset to develop accurate and easily interpretable models. Using discharge voltage curves from early-stage cycles that have yet to undergo degradation, we have applied Random Forest, AdaBoost, ElasticNet, and Artificial Neural Network algorithms to predict cycle life.

With each model, we have attempted to draw out as high accuracy as possible, and have seen impressive accuracy in the Artificial Neural Network Model (ANN). Our model can quantitatively predict cycle life using the first 100 cycles with an accuracy of 75-85%.

Our model ensures the reliable operation of lithium-ion batteries and contributes to accelerated technology development. Accurate prediction of a lifetime using early-cycle data would unlock new opportunities in battery production, use, and optimization. For example, manufacturers can accelerate the cell development cycle, perform rapid validation of new manufacturing processes, and sort/ grade new cells by their expected lifetime.

TABLE OF CONTENTS

List of Figures

CHAPTER 1

INTRODUCTION

1.1 Introduction

A Lithium-ion battery is a rechargeable type of battery that is composed of cells. In these cells, the lithium ions navigate from the negative electrode to the positive electrode through an electrolyte during the discharging period and vice-versa during the charging period. These batteries have a long life span and high energy density. These batteries' uses have been increasing extremely rapidly in the last few decades and their prices have been continuously decreasing, which makes them even more extensively used.

The degradation of lithium-ion batteries is one of the significant issues faced by industries. Uncertainties in the environment make it challenging to predict the degradation rate. The operating life of batteries constitutes a key component in the reliability and cost of energy systems.

We aimed at predicting the lifetime of Li-ion batteries by using various Machine Learning algorithms such as Random Forest, AdaBoost, ElasticNet, and Deep Learning Algorithm like Artificial Neural Network algorithms. In our models, we used a publicly available dataset which consists of training data of 41 cells and test data of 43 cells of the first 99 cycles, at 1000 voltage points ranging from 2 - 3.5 V. For this, at the same discharging conditions, these batteries were cycled keeping the temperature-controlled environment chamber around 30º C under varied fast-charging conditions.

With the coming of the E-vehicles era, it has become critical to predict battery life to avoid any potentially hazardous impact. The lithium-ion battery has broad utility in industries and many areas which makes the prediction of its lifetime critically challenging however at the same time it is challenging because of its non-linear degradation. Moreover, information about the remaining useful life of batteries is essential for its second-life applications.

CHAPTER 2

DATASET AND METHODOLOGY

2.1 Dataset

We have used a three-dimensional dataset containing discharging capacities of cells. The distribution of training data is as follows: 41 cells on the x-axis, 1000 voltage points ranging from 2 - 3.5 V on the y-axis, and 99 cycles on the z-axis.

According to the paper referenced[1], the dataset was generated in controlled environmental conditions in a chamber at a constant temperature of 30 degrees Celsius. However, it should be noted that since there is a lot of heat generated due to the continued charging and discharging, the temperature of the cells has varied up to 10 degrees. The cell used was commercial Lithium Iron Phosphate/graphite cells. To generate different types of data, during the charging the rate was varied from the recommended fast rate (by manufacturer) of about 3.6 C to about 6 C, which is extremely fast. This is important because an extremely fast rate of charging can be of commercial interest as well. However, The conditions have been kept identical during the process of discharging.

2.2 Data Preprocessing

Before feeding the data as an input to the various models, we preprocess the data. We used the ‘variance’ model, wherein we use the variance of $\Delta Q_{100\text{-}10}(V)$ for prediction, where Q_x represents the discharging capacity in the x^{th} cycle. It should be noted that summary statistics, for example, mathematical quantities like mean and variance, are used for their predictive abilities and do not have a physical significance here. It is found that there emerges a clear and strong relationship between the cycle life of the cell and the variance of $\Delta Q_{100\text{-}10}(V)$, which has been justified in figure 2.2 also. Therefore, we have taken the 100th and 10th cycle, because on analyzing the data, the predictive power of features based on the variance of the difference of these cycles was maximum.

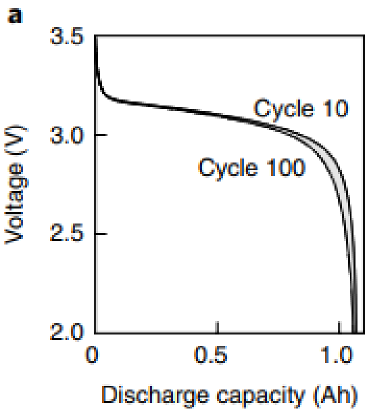


Fig 2.1 Discharge capacity curves for 100th and 10th cycles for a representative cell.

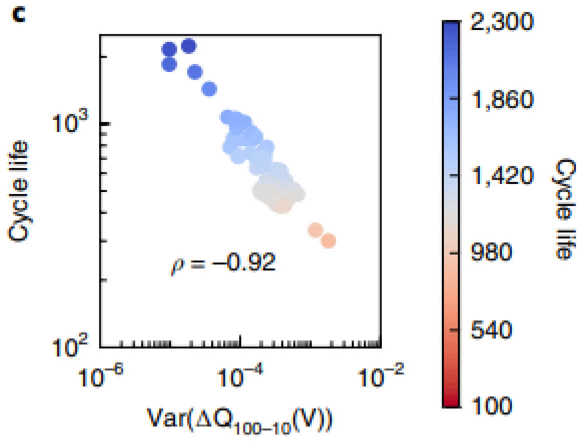


Fig 2.2 Cycle life plotted as a function of the variance of ΔQ100-10(V) on a log-log axis, with a correlation coefficient of −0.93.

The correlation coefficient is a measure that specifies how strongly are two or more variables in a dataset linearly interdependent on each other. Therefore a correlation coefficient close to -1 signifies a very strong negative correlation.

2.3 Random Forest

It is an **ensemble-based supervised learning** algorithm for regression, which means it combines predictions from various decision trees to make a more precise prediction than a single tree. Random Forest is an improvement over the decision trees classifier because it prevents overfitting by using the **bagging** technique. Bagging means creating different subsets from the training data with replacement and finally choosing the output based on the majority. The newly created dataset formed from the subset of the original dataset is called **Bootstrapped dataset.**

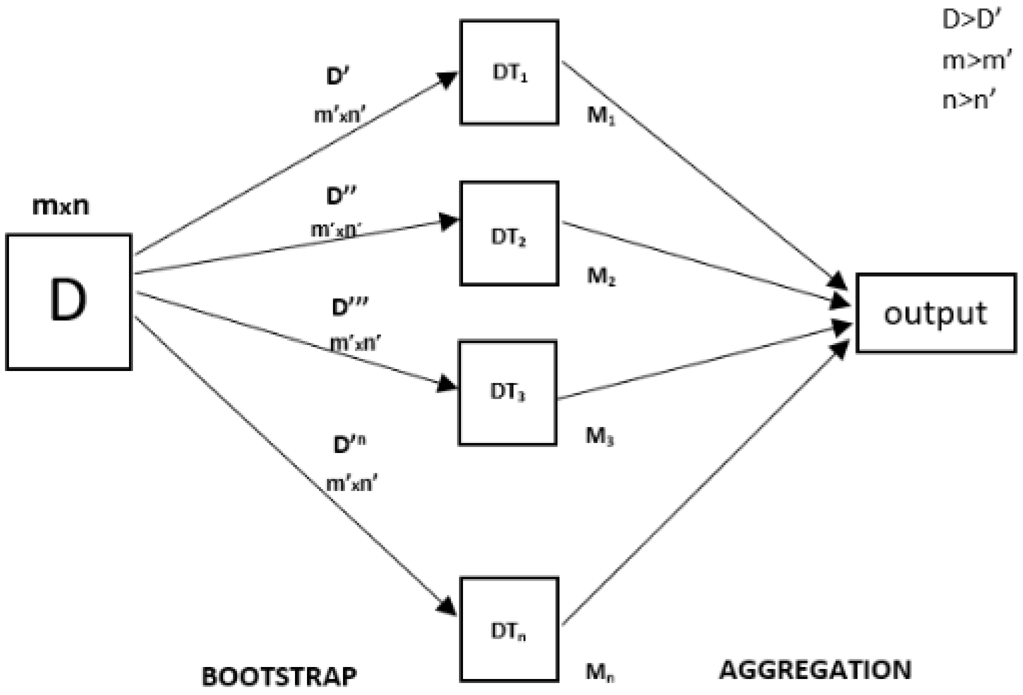


Fig 2.3 Working of Random Forest

2.4 AdaBoost

AdaBoost or **Adaptive Boosting** is an **Ensemble-based** machine learning algorithm. An ensemble-based method is basically a machine learning technique that combines several base models in order to produce one optimal and precise model.

Adaboost uses decision trees with single splits. Initially bagging is performed to reduce the variance, followed by boosting which primarily reduces the bias and may also reduce any additional variance, in turn converting several weak learners into a strong one. In AdaBoost, once the sampling is done by one regressor, its loss is calculated and is in turn fed to the next regressor depending upon this loss as shown in the diagram as well.

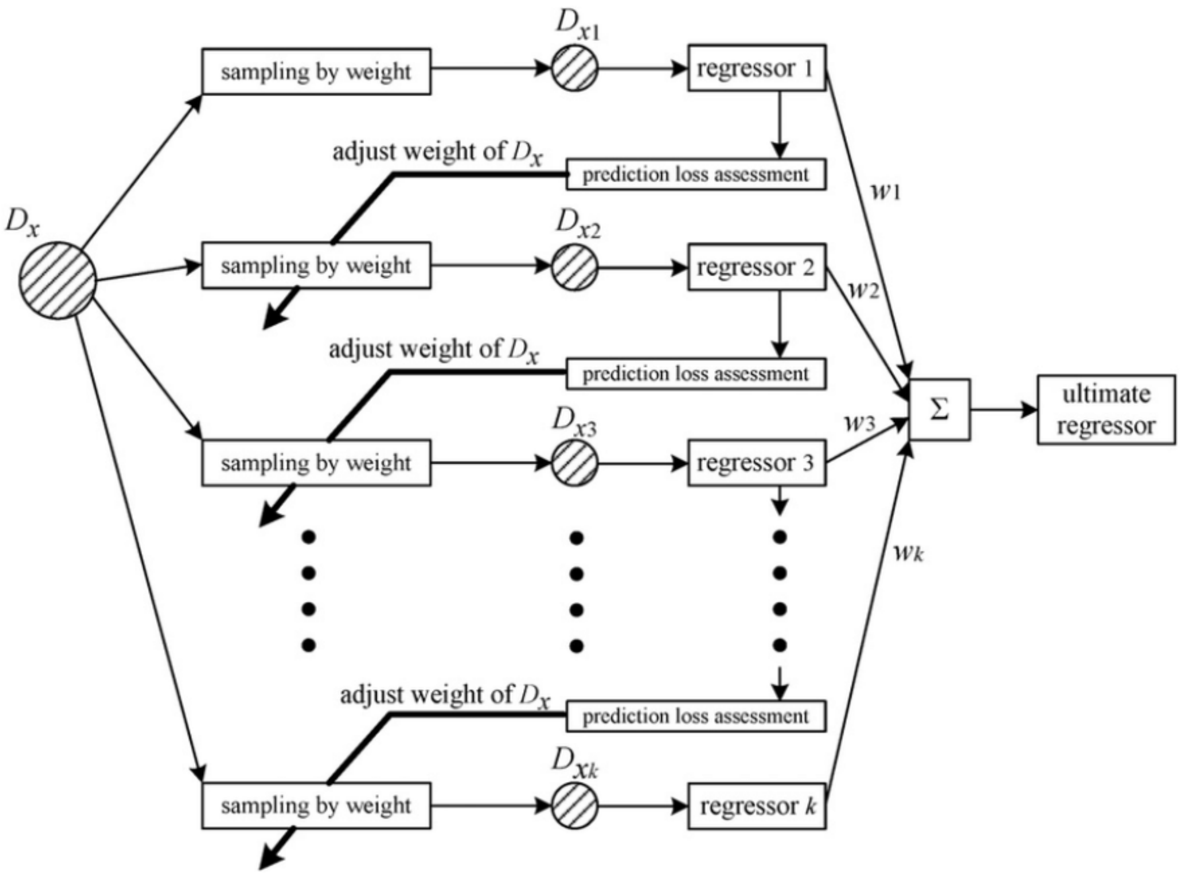


Fig 2.4 Working of AdaBoost

The formula to calculate the sample weights is:

$$w(x_i, y_i) = \frac{1}{N},\ i = 1, 2, \ldots . n$$

where N is the total number of data points.

α i.e. **the amount of say** for a classifier is given by:

$$Performance\ of\ the\ stump\ =\ \frac{1}{2}\log_e\left(\frac{1 - Total\ Error}{Total\ Error}\right)$$

For wrongly classified samples the updated weights are given by:

$$New\ sample\ weight\ =\ old\ weight\ *\ e^{\pm Amount\ of\ say\ (\alpha)}$$

2.5 Elastic Net

Ridge Regression adds a term of the sum of squares of coefficients multiplied by the parameter in the ordinary least square error function that regularises the value of coefficients of variables. The new term added is called **L2** regularisation.

Upon minimizing the below objective function, for $||\boldsymbol{\beta}||^2<t$, Lasso Regression can be interpreted.

$$\min_{\beta_0,\beta}\left\{\frac{1}{N}\|y-\beta_0-X\beta\|_2^2\right\}$$

Lasso(least absolute shrinkage and selection operator) Regression adds a term of the mean absolute value of coefficients and can completely eliminate the variable by reducing its coefficient value to 0 due to the difference in the shape of their constraint boundaries. The new term added to Ordinary Least Square(OLS) is called **L1** Regularisation.

Upon minimizing the below objective function, for $||\boldsymbol{\beta}||<t$, Lasso Regression can be interpreted.

$$\min_{\beta_0,\beta}\left\{\frac{1}{N}\|y-\beta_0-X\beta\|_2^2\right\}$$

Elastic Net is an evolved **amalgam** Regression method of **Lasso and Ridge Regression.** Elastic Net includes an “N” number of variables until the saturation is achieved, unlike Lasso which takes a few samples of high dimensional data. Lasso chooses a single variable out of highly correlated groups. In order to overcome these shortcomings from Lasso Regression, Elastic Net introduces a quadratic expression: $||\boldsymbol{\beta}||^2$ in the error equation which further elevates the loss function to convex. Ridge regression coefficients are found prior which are followed by lasso shrinkage of coefficients. The coefficients are rescaled further in order to increase efficiency and reduce bias.

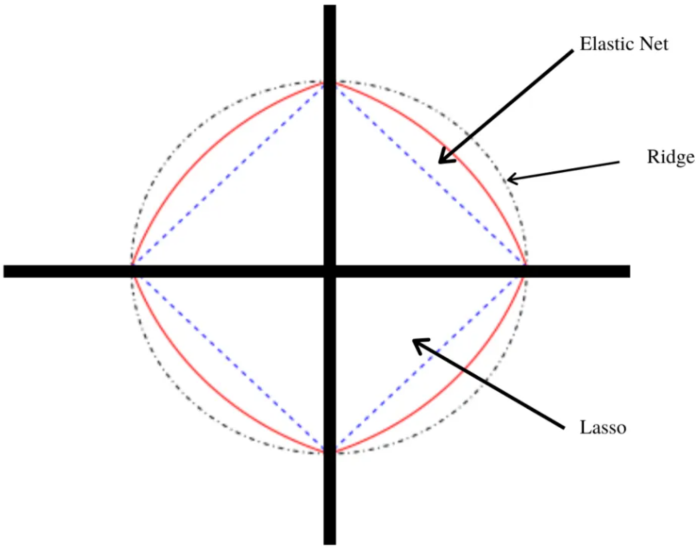


Fig 2.5 Characteristic Plots of Ridge, Lasso, and Elastic Net Algorithms

As can be seen from the above figure that the characteristic plot of Elastic Net lies in the middle of the Lasso and Ridge plots. Elastic Net plot manifests singularity at the vertices. Its convexity depends on **α**. **Grouping effect** also plays an important role in deciding the convexity of the model i.e. the correlation of variables. As the correlation increases, the grouping effect also increases, in turn increasing the number of variables included in the sample.

The process of reducing Variance upon introducing some Bias is called regularisation. During regularisation, the l_1 section of the penalty forms the sparse model while the quadratic section of the penalty makes the l_1 part more steady, further eliminating the limit of variables and hence enhancing the grouping effect. Hence, correlation can be used in order to identify the variables.

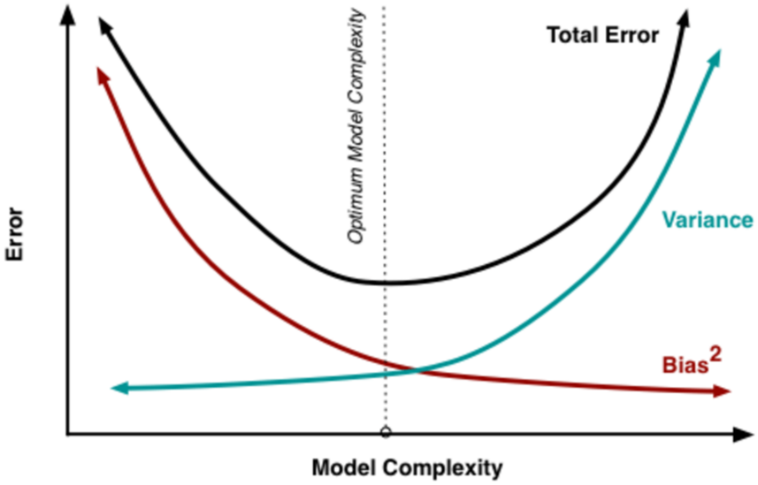


Fig 2.6 Plot of Model Complexity vs Error

The **Complexity** of any Model is an important measure for the model fitting, which is determined by effective **degrees of freedom** in the case of Elastic Net Algorithm.

The algorithm aims at minimizing the below loss function:

$$\frac{\sum_{i=1}^{n}(y_i - x_i^J \hat{\beta})^2}{2n} + \lambda\left(\frac{1-\alpha}{2}\sum_{j=1}^{m}\hat{\beta}_j^2 + \alpha\sum_{j=1}^{m}|\hat{\beta}_j|\right)$$

2.6 Artificial Neural Networks

An **Artificial Neuron Network (ANN)** is a type of computational network or simulation which can be used as a function approximator for a particular practical data. **ANN** is trained using learning algorithms that are capable of updating the weights of the network as new inputs are received. An ANN consists of several nodes and each node is a weighted sum of the outputs from the previous layer with some activation function applied to increase the complexity of the model.

ANN primarily consists of 3 layers:

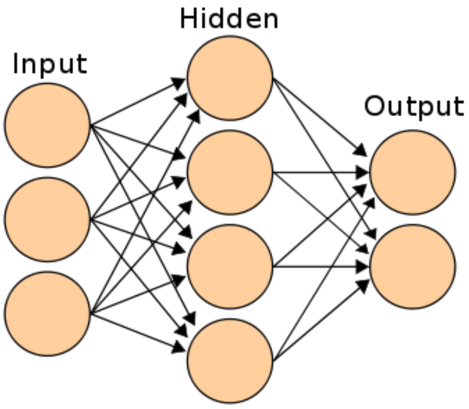


Fig 2.7 Layers in ANN

- **Input layer:** This layer contains artificial neurons (units) that obtain inputs consisting of independent variables in different formats. This actual process of learning or recognition on the network happens in this layer.

- **Hidden layer:** The inner layers or Hidden layers present between input and output layers adaptively modify the information received from layer to layer. Each layer gets its input from a previous layer and returns output to the next layer allowing the network to understand complex features. These layers perform necessary calculations to extract any hidden features or patterns. These hidden layers are also called neural layers. The units present in the hidden layer learn about the information received by weighing it as per the internal system. The processed data is then provided as input to the next stacked layer.

- **Output layer:** The input after being transformed by the neural layers, finally comes out as an output conveyed through this layer. Final activation is applied to this layer as the required

Computation is represented by a transfer function.

$$y_{in} = x_1 . w_1 + x_2 . w_2 + x_3 . w_3 \ldots x_m . w_m$$

i.e., Net input $y_{in} = \sum_i^m x_i . w_i$

The activation function receives a weighted sum as an input. Activation functions serve the purpose of increasing complexity by decreasing linearity of the model. Those who are fired reach the output layer. Different kinds of activation functions are used depending upon the kind of task being performed.

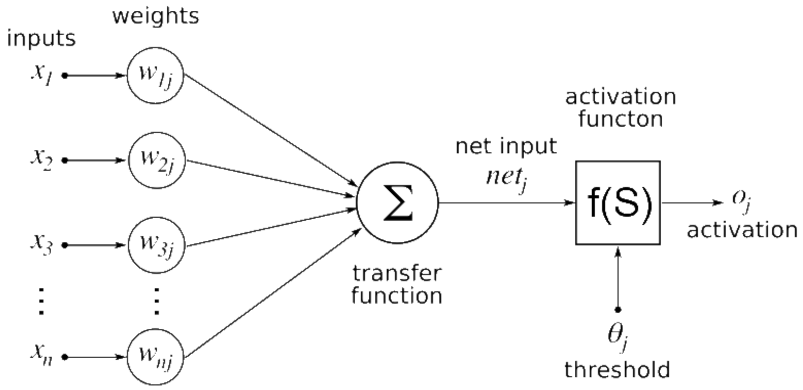


Fig 2.8 Working of ANN

A weighted directed graph consisting of nodes formed by artificial neurons can be used to describe an Artificial Neural Network. Here, the directed edges with weights represent the association between the neuron input and neuron output. These weights usually signify the strength of the interconnections among neurons inside the ANN.

ANN receives input signals in the form of patterns and images as vectors; which are then denoted by x(n) for 'n' inputs. Each of the inputs is further multiplied by its corresponding weight(weights refer to the details used by ANN to solve a particular problem). These weighted inputs are then summarised inside the computing unit.

In case the weighted sum of the above becomes zero, then an extra bias needs to be added because the system's response needs to be scaled up which requires a non-zero quantity. In bias, the input is kept the same as earlier, and weight is made 1. The total sum of the weighted inputs can have any value between zero and infinity. A particular maximum value is kept as a benchmark and then an activation function is applied to the total of all weighted inputs and then the neuron takes the further decision whether this signal should be passed to the next layer or stopped there.

An activation function is a transfer function used to get the desired output for a particular problem. The activation function serves two main purposes, it captures the non-linear relationship between inputs and helps to convert input into a more useful output.

A few examples of activation functions include

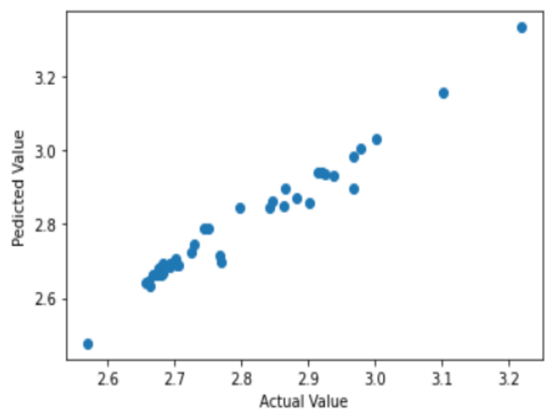
- **Binary:** Binary function gives an output of either 0 or 1. A comparison of the net weighted input of neurons with a threshold value is done to accordingly assign value to them.
- **Bipolar Sigmoidal function:** Input editing between -1 and 1 is done by an activation function. It can either come out to be positive or negative by nature. Output lies within -1 and 1 and is strictly increasing.

Among the many advantages of ANN, one of the most prominent is the fact that it is capable of learning from observing data sets. Hence, sometimes it is used as a random approximation tool, helping in estimating the most ideal and cost-effective method for arriving at solutions while dealing with computation functions or distributions.

CHAPTER 3

RESULTS

3.1 Random Forest



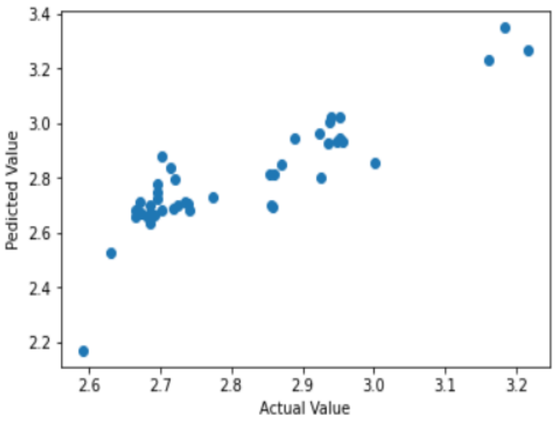


Fig 3.1 Train Data Prediction

using Random Forest

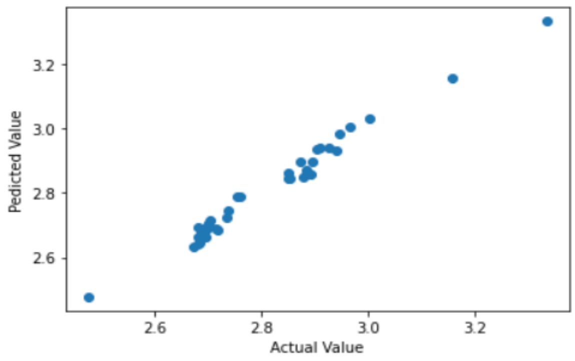
(r2_score = 0.9313)

Fig 3.2 Test Data Prediction

using Random Forest

(r2_score = 0.7425)

3.2 Adaboost Regressor



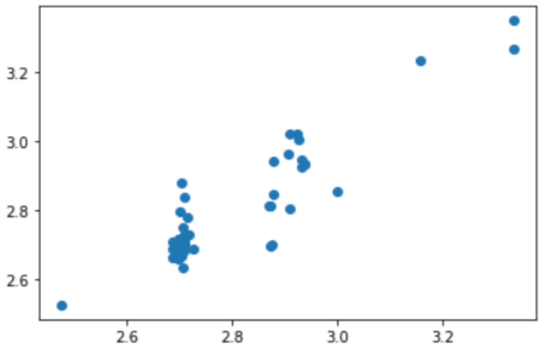


Fig 3.3 Train Data Prediction

using AdaBoost

(r2_score = 0.9710)

Fig 3.4 Test Data Prediction

using AdaBoost

(r2_score = 0.7993)

3.3 Elastic Net

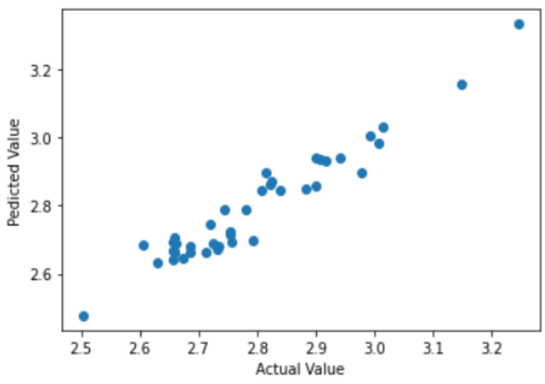


Fig 3.5 Train Data Prediction

using Elastic Net

(r2_score = 0.9177)

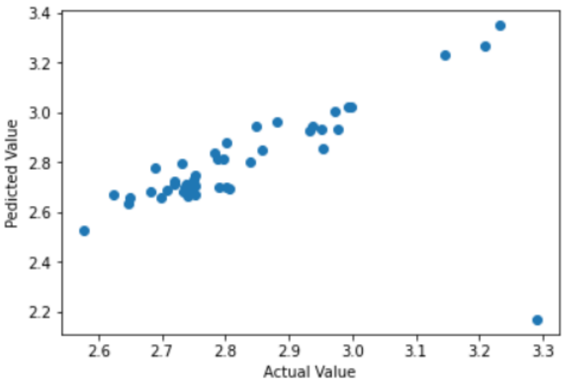


Fig 3.6 Test Data Prediction

using Elastic Net

(r2_score = 0.8440)

3.4 Artificial Neural Network

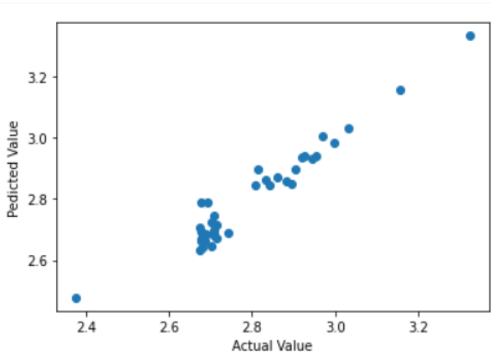


Fig 3.7 Train Data Prediction

using ANN

(r2_score = 0.9402)

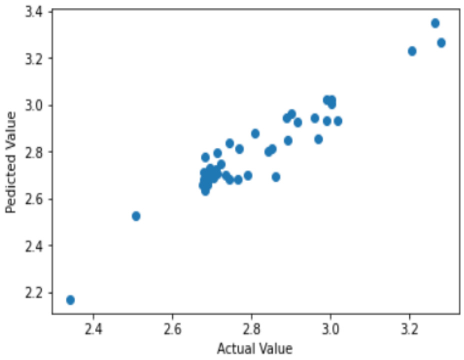


Fig 3.8 Test Data Prediction

using ANN

(r2_score = 0.8846)

CONCLUSION & FUTURE SCOPES

Predicting the lifetime of Li-ion batteries is a crucial factor in achieving sustainable battery performance. Approaches based on statistical and ML techniques to predict cycle life tend to have high accuracy as it adopts a multi-feature approach or broad view of different parameters while applying empirical data to deduce results over a theoretical system, it considers real-life disturbances into account and works as a universal approximator. A well-built training dataset drives the quality of the overall ML model. However, this model also faces a few limitations. ANN is the most accurate, and lacks trust in the network as it does not give a detailed understanding of why and how. It's hardware-dependent and it lacks a proper specific rule for determining the structure of its network. AdaBoost needs a quality dataset, hence noisy data has to be avoided before applying it. Random Forest is not suitable for linear methods with high sparse features, its training is slow and is biased when dealing with categorical variables. Nevertheless, no model is perfect, but the best one can still be put to use.

So, in the future, an enhanced dataset can be deployed to increase the model accuracy by exploring or creating new features out of existing ones. Since the model has to calculate and update a large number of parameters during run time, consumer hardware finds it difficult to do extensive computations quickly. Thus, going forward, a hardware version of the model can also be implemented.

REFERENCES

[1] Severson, K.A., Attia, P.M., Jin, N. et al. Data-driven prediction of battery cycle life before capacity degradation. Nat Energy 4, 383–391 (2019).