DEVELOPMENT OF INTELLIGENT SOC ESTIMATION ALGORITHM FOR LITHIUM POLYMER BATTERIES

by Bb10 Bb10

ABSTRACT

Rise of air pollution, Exploitation of fossil fuels, increase in population demands more usage of vehicles, new government restrictions on emissions from vehicles had a great impact on Electric Vehicles. The above-mentioned reasons accelerated the usage of EV's. Battery is the key element in EV it can also be considered as heart of EV. Battery parameters should be accessed carefully time to time.

To estimate the health of battery, SOC has to be determined which is a tricky and difficult parameter to find. Knowing the parameters helps us to exchange the batteries with transparency.

The proposed work "DEVELOPMENT OF INTELLIGENT SOC ESTIMATION ALGORITHM FOR LITHIUM POLYMER BATTERIES" helps us to estimate the SOC in a better way. In this work we had taken the data of the battery from the data sheet and developed an algorithm "curve fitting algorithm" and to reduce complexity in calculation we went with the code. The same data from data sheet is trained to ANN and validated the curve fitting algorithm with WEKA. By determining the battery health every person purchasing the batteries for electric vehicles can know the health and state of the battery even before purchasing so that they cannot be cheated by the vendors or any other individuals.

List of Figures

FIG.NO	NAME	PAGE NO
1.1.0	Conversion from conventional to electric vehicle	6
1.2.0	Methodology	8
2.1.0	Lithium battery pack	9
2.6.1	Dual RC battery model	12
3.2.1	Classification of SOC estimation methods	14
3.4.1	Curve fit for Ro vs SOC	17
3.4.2	Curve fit for R1 vs SOC	18
3.4.3	Curve fit for R2 vs SOC	18
3.4.4	Curve fit for C1 vs SOC	19
3.4.5	Curve fit for C2 vs SOC	19
3.4.6	Curve fit for Z vs SOC	20
3.4.7	Curve fit for SOC vs. Ro	20
3.4.8	Curve fit for SOC vs. R1	21
3.4.9	Curve fit for SOC vs. R2	21
3.4.10	Curve fit for SOC vs. C1	22
3.4.11	Curve fit for SOC vs. C2	22
3.4.12	Curve fit for SOC vs. Z	23
3.4.13	Curve fit for SOC vs. R	23
3.4.14	Code for SOC vs. R	24
4.1.0	Architecture of ANN	26
4.4.1	Layers in MLP	29
4.4.2	Confusion matrix	30
4.4.3	GUI output	30
4.4.4	WEKA Result	31

List of tables

TABLE NO	NAME	PAGE NO
2.2.1	Comparison of batteries	10
3.3.1	Comparison of estimation accuracy	15
3.3.2	Comparison of tracking speed ability with initial SOC error	15
3.3.3	Comparison of computational complexity	15
3.4.1	Curve fit status	24
4.4.1	Validation of curve fit with ANN	31

List of abbreviations

1. EV- Electric Vehicle
2. SOC- State of Charge
3. SOH- State of Health
4. EKF- Extended Kalman Filter
5. UKF- Unscented Kalman Filter
6. NLO- Non- Linear Observer
7. SMO- Sliding Mode Observer
8. SPEMT-Single Particle Model with Electrolyte and Thermal
9. CC-Constant Current
10. CCCV-Constant Current Constant Voltage
11. ANN- Artificial Neural Network
12. DOD- Depth of Discharge
13. GUI- Graphical User Interface
14. WEKA- Waikato Environment for Knowledge Analysis
15. ROC-Receiver Operating Characteristics
16. MLP-Multilayer Perceptron

CHAPTER 1
INTRODUCTION

1.1 Introduction

The demand for electric vehicles (EVs) has increased rapidly in the last decades, with rise in air pollution and declining fossil fuels. In most of the countries, electric vehicles had already replaced the conventional ones. The battery, as a crucial component of the EVs, has a major effect on the parameters of EVs like durability, speed and lifetime. Due to their high energy and power density, long service life, eco-friendly nature and low self-discharge characteristics, lithium-ion batteries are currently widely used in energy storage systems for EVs.

The battery must be carefully monitored to ensure its dependency and protection. SOC is one of the crucial parameter that need to be frequently monitored. Accurate SOC estimation is important as an indicator of the remaining useful capacity to prevent overcharging / discharging of the battery as well as to predict the remaining electric vehicle driving range. SOC has to be known for knowing the battery capacity. SOC is a tricky parameter that isn't easily found. As SOC is an important parameter it should be determined efficiently since any error in the SOC calculation could result in damage to the battery. SOH and SOC are interlinked. Finding SOC helps us to find SOH which enables us to know the current status of a battery. SOC determination increases the transparency when there is a need of exchanging of batteries. Usually this exchange of batteries come into picture because of high charging time and medium driving range of a vehicle.

SOC estimation methods are categorized into several categories such as observer-based models, fuzzy logic and data driven models which include Kalman filter, nonlinear observer (NLO), unscented Kalman filter (UKF), extended Kalman filter (EKF) and sliding mode observer (SMO). Such algorithms evaluate accuracy of estimation, report speed to initial SOC error and difficulty of computations. These filters provide appropriate and stable results of input values at low current. However, the estimates show large errors at high current rates which lead to maximum or minimum concentrations of near-electrodes.

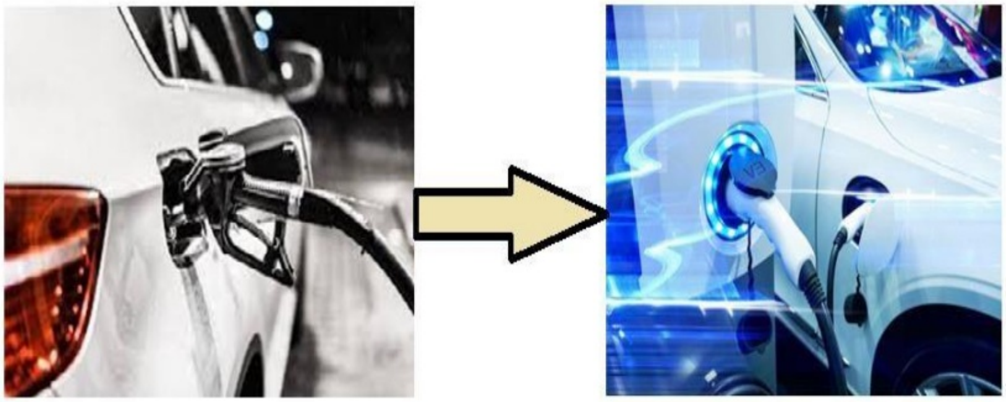


Fig 1.1 Conversion from Conventional to EV

The charging schemes also affect the battery performance, there are many charging strategies such as multi-stage CCCV, constant-current (CC), pulse charging techniques and CC constant voltage (CCCV). Choosing the best one from these also effect the lifespan and the performance of the battery. Charging strategies basically use empirical models like equivalent circuit-based models and neural network models. These charging strategies forecast the state of the battery and calculate electrical parameters using past practical data.

There are many battery models such as RC model, single particle model, electro chemical battery model and a combination of single particle model with electrolyte and thermal (SPMET). This project is to develop an algorithm which estimates SOC precisely. Another new approach for estimation of SOC is piece wise curve fitting algorithm. This algorithm uses the RC battery model to reduce computational complexity.

This curve fitting is a mathematical tool which gives the relation between the parameters.

1.2Literature Survey

As population is increasing day by day, the usages of vehicles also increase which may lead to the exploitation of fossil fuels. This laid a path towards alternate energy sources. Here comes the concept of electric vehicle. Battery is the main component of electric vehicles without which this EV’s couldn’t have gained that importance.

The design of battery and monitoring its parameters are very important. SOC is the crucial parameter and a tricky one to find. There are many types of batteries in which lithium battery gained its popularity because of its specific features like cell voltage, self-discharge, high energy density, less maintenance, transportation, cost, developing technologies, load characteristics and portability.

Finding a SOC for lead acid battery is not a big task because the battery gives us the lithium polymer battery it's a quite difficult task since the parameters that can be known are only the temperature, impedance, current and so on. But any of these parameters doesn't give the SOC directly.

There are many ways to find SOC such as observer model technique (nothing but filters like extended Kalman filter, nonlinear observer, sliding mode observer, dual extended Kalman filter), fuzzy networks, neural networks .out of which ANN gives us the best results compared to observer model techniques. Not only SOC, SOH is a crucial parameter which tells performance of the battery. Knowing SOC and SOH increases the reliability on the battery and also helps to estimate its driving range, life cycles and maintenance schedule.

1.3 Problem Identification

- In EVs batteries are used for the running electric motor, therefore there is a need to monitor its state continuously. But finding its state is difficult i.e. exact its charge whether it's charging, discharging and so on.
- SOC estimation should be accurate any errors in it may cause huge damage.
- To increase the driving range, batteries are connected in parallel or series so that it forms a battery pack. In this scenario, Fault location is a major task.
- Charging a battery requires much time, these made EV's usage limited to short range. Even if they take the risk for travelling long distances then most of the time is required for charging at battery charging stations.
- The battery performance should be known, this can be achieved with the help of SOH. Determining SOH can be done by monitoring the battery parameters time to time. But the battery parameters also vary with time to time.

1.4 Aim of thesis

- There are many algorithms and methods to for SOC estimation.
- Main objective in this thesis to develop a simple algorithm and accurate algorithm to estimate SOC.
- Validating the developed algorithm with existing accurate SOC estimation method i.e. ANN.

1.5 Structure of thesis:

- In chapter 1 the importance of electric vehicles, battery, battery parameters, basic introduction to various SOC estimation algorithms, charging schemes are discussed. Literature survey, problem formulation which motivated us to take over the project and finally the aim of the project.

- In chapter 2 different types of batteries, majorly focuses on lithium ion and lithium polymer battery, its characteristics and advantages of lithium batteries are discussed. In this chapter we also discuss about the battery which we had used in this project and finally we end with the disadvantages of lithium polymer battery.
- Chapter 3 deals with the SOC estimation methods and also discuss about a new simple SOC estimation algorithm which we have developed and also the problems we encountered while developing an algorithm. The results of developed algorithm are also included in this chapter.
- The data driven model, the algorithms which aare used and the explanation of the algorithms are discussed in chapter 4 . In this chapter we also discuss about the training of Weka and analysis of the results obtained through it.
- Finally, we conclude the thesis and also discuss about the future scope.

1.6Methodology:

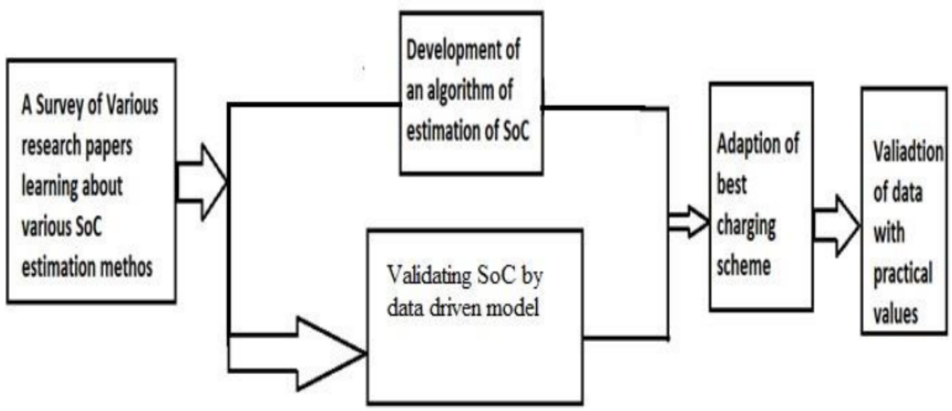


Fig 1.2 Methodology

CHAPTER 2

BATTERIES

2.1 Introduction to batteries

Batteries produce electricity using liquids or metals that are good conductors of electricity. Essentially, batteries store chemical energy and it converts chemical energy into electrical energy. Basically, batteries are classified into 2 types. Primary batteries and secondary batteries. Primary batteries are of use and throw type which means they are non-rechargeable batteries so these batteries cannot be used in electric vehicles. Whereas, secondary batteries are rechargeable batteries which enabled the use of these batteries in vehicles and in many devices. Lead acid battery, nickel cadmium battery and lithium ion battery are some of the examples of secondary batteries.

Fig: 2.1 lithium ion battery

The Lead-acid battery is one of the oldest **types** of rechargeable batteries. It has been in operation for more than 150 years, and remains one of the cornerstones of the automotive industry. The lead acid battery has a high current capability, low cost and is resistant to abuse. This feature make it optimally suited for other applications. Electric vehicles now tend to be the future with manufacturing companies and regulation leading to the phasing out of the internal combustion engine, with the transition to more environmentally sustainable power sources. The Lithium Ion technology gives better performance for EVs, is more eco-friendly and has potential to make EVs competitive.

2.2 Comparative study of different batteries

Parameters	Lead acid battery	Li-ion battery
Life span	Less	More
Efficiency of work	Less efficient	More efficient
Charge rate	Over heats When charged quickly so it has limited charge current	Has better charge rate
Density of energy	Low	High
Depth of discharge	Total capacity in a single cycle should only be discharged up to 50 percent. And beyond 50% it effects the battery lifetime	Li-ion batteries have higher capacity and can handle discharges of 80%

Table 2.1 Comparative study of lead acid with Li-ion battery

Lithium ion batteries are adopted by many electronic companies from early 1990's since they are rechargeable. But, in recent years lithium polymers have become popular in less time because of their vast applications and advantages.

2.3 LiFePO4 Characteristics

- Higher energy-weight-rate, energy-volume-rate.
- High voltage. The voltage of each cell is 3.6V equivalent to three times of nickel-cadmium battery.
- Because of the low self-discharge rate, keep for long.
- No effect on memories. For LiFePO4 battery it need not be discharged before charging.
- High life span. The cycle times for loading and discharging are over 500 in the normal working state.
- Fast charging. Current range of 0.5 to 1 can be used for charging process and the time required for charging can be limited to 1-2 hours.
- It is the most advanced "organic" battery in the world, since it does not harm the atmosphere and does not contain heavy metal cadmium, lead or mercury.
- High price. The LiFePO4 battery is more costly, compared to other types of batteries.

2.4 DESIGN OF LiFePO4 BATTERY

There are two structures for LiFePO4 battery namely rectangular and cylindrical. It is in spiral structure inside which has high permeable polyethylene film to partition cathode and anode. The anode has Li-ion collector which is made with Lithium and CoO2 and aluminium film collector. The cathode has Li-ion collector which is made from Carbon sheets and current bronze film collector. The battery is filled with a solution of organic electrolytes. In addition, there is a safety valve. It has PTC component that protects the battery in the time of faults or short out. Single cell voltage is 3.6V.

2.5 Complete cycle of LiFePO4

1. Charging cycle of LiFePO4: Based on the design characteristics the highest charging voltage for the li battery is 4.2V. Over-charging is discouraged as the battery will get damaged if the Li-ion is taken too far from anode. The charging and discharging method is complex, but we can use the constant voltage adapter with qualified constant current. As the battery voltage exceeds 4.2V in the constant current charging mode the battery will be charged in the constant voltage mode. So, if the current decreases below 100 mA, the charging will cease.

2.LiFePO4 battery discharge: The lithium-ion cannot completely move to anode during the discharging process due to its internal structure. To ensure that the Lithium-ion can be freely embedded in the channel, a part of the Li-ion should be kept in cathode. The total ultimate voltage of the discharge will be reduced to ensure that the graphite layer retains enough Li-ion after discharge. In other words, they cannot discharge the Lifepo4 battery over. The standard voltage for discharging is 3.0V / cell, cannot be less than 2.5V / cell. The time of discharge depends on the battery capacity and the discharging current. Duration of unloading = capacity of battery/ discharging current. The discharging current from the LiFePO4 battery(mA) cannot exceed thrice its battery capacity.

2.6Equivalent Battery models

There are various battery models present for LiFePo4 battery such as single particle model, dual RC model, single particle model with electrolyte and thermal (SPMET)., etc. Here after analysing these battery models. We found dual RC network model with less computational complexity. The equivalent circuit of dual RC model can be found below.

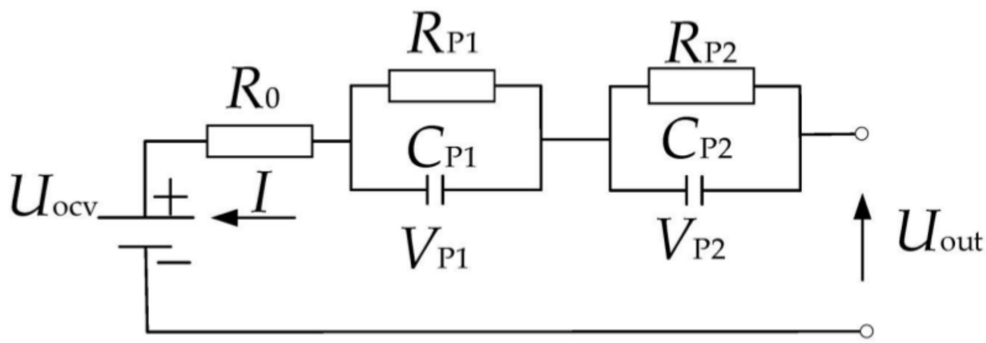


Fig 2.6.1Dual RC model

The data taken for the curve fit is from this dual RC network at constant room temperature by using constant current constant voltage charging strategy (CCCV).

CHAPTER 3
SOC ESTIMATION METHODS

3.1 Introduction to SOC

SOC estimation is the most challenging task in recent years. The SOC of a battery describes its left-over capacity which is a very supreme parameter. As SOC is an important parameter, which reflects the battery performance, so accurate estimation of SOC not only protects the battery but also prevents it from over discharging and over charging. The classical methods for estimation of SOC are direct measurement (i.e., open circuit voltage method, terminal voltage method, terminal voltage method, impedance method, impedance spectroscopy method), Book estimation methods (coulomb counting method, modified coulomb counting). Before going in detail let us familiarize with the terms.

SOC(State of charge): It is the level of charge of an electric battery relative to its capacity. Usually it is represented in percentage (0% = empty; 100% = full. SoC is usually used when discussing about the current state of a battery in use.

Depth of Discharge (DOD): This is an alternate approach for signalling the charging state of a battery. The discharge depth is the counterpart to charging status: the one increases and the other decreases. Usually, the discharge depth is expressed using units of Ampere hour or percentage points. Capacity of a battery may be higher than its nominal weight. And by nominal value the depth of the discharge value can be reached.

State of health (SOH): The State of Health is a "standard measure" that shows the general state and ability of a battery to deliver the specified value in contrast with a fresh battery. It helps determine considerations like charge recognition, internal resistance, self-discharge and voltage.

This is an indicator of the long-term capacity of the battery and provides an “indication" not an actual estimate but indicates how much of the battery's total "lifetime energy output" has been used, and how much remains.

3.2. An overview of estimation methods of SOC

SOC estimation methods are categorized as follows:

- Fuzzy logic
- Data driven models.
- observer -based methods.

Observer based techniques provide appropriate and stable results of input values at low current. However, the estimates show large errors at high current rates which lead to maximum or minimum concentrations of near-electrodes.

Fuzzy logic and artificial neural networks provide better results when compared to observer-based techniques.

Most popular algorithms are Extended Kalman filter, Sliding mode observer, Nonlinear observer and Unscented Kalman filter. Kalman filter yields the better results for a linear system but it is limited for linear systems only. To overcome the drawbacks of Kalman filter, extended Kalman filter (EKF) is proposed. Extended Kalman filter is also applicable for nonlinear systems. Even though the Extended Kalman filter overcomes the drawbacks of Kalman filter but it has its own drawbacks. EKF includes linearization and calculation of Jacobian matrix. To overcome problems in EKF, unscented Kalman filter (UKF) has been developed. Sliding mode observer (SMO) approach is built by adding a discontinuous feedback based on the Luenberger observer. When SMO is compared to the nonlinear observer (NLO), non-linear observer is best suited for nonlinear output equations.

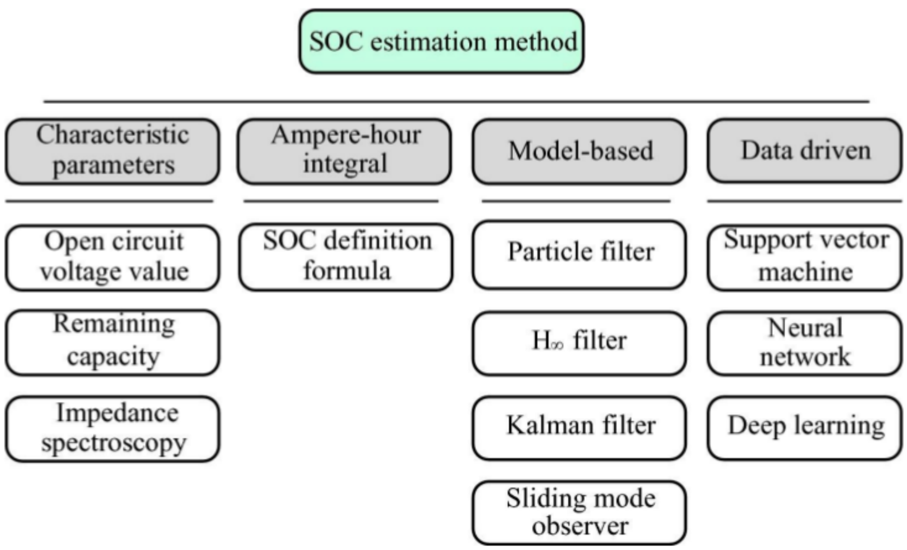


Fig 3.2.1 Classification of SOC estimation methods

3.3. Comparison of observer model-based techniques:

The important model-based estimation methods for SOC are NLO, SMO, EKF and UKF. These are analysed in terms of predictive error, initial SOC error tracking capability and computational complexity. All the four algorithms forecast the maximum error in SOC in the range of 7.5% and has range of 4% for root mean squared error. From the below table, we can summarize that UKF is best in terms of estimating SOC accurately.

	EKF (%)	UKF (%)	SMO(%)	NLO (%)
Maximum SOC error	5.90	4.20	7.20	4.30
Root mean square error	2.70	1.60	3.70	2

Table 3.3.1 Comparison of estimation accuracy

From the below table, referring to tracking ability NLO is best when compared to other methods.

Estimation techniques	EKF	UKF	SMO	NLO
Time of convergence	5min 10s	2min 20s	2min 30s	1min 35sec

Table 3.3.2 Comparison of tracking speed ability with initial SOC error

Computational difficulty is a factor which influences practical application of estimation algorithms.

Estimation Methods	EKF	UKF	SMO	NLO
Mathematical difficulty	0.40	1	0.440	0.440

Table 3.3.3 comparison of computational complexity

According to table UKF is complex in calculation compared to other three. Even though EKF, SMO and NLO have same computational complexity, EKF is quite difficult in matrix processing where SMO has high complexity in calculating the gain matrix.

Conclusions from comparison:

- UKF performs better in terms of accuracy but takes more time to calculate the initial SOC error compared to other three.
- In terms of tracking speed capability with initial SOC error, NLO is best and has comparable accuracy with UKF.
- Estimation error for SMO is higher but it has more tracking efficiency compared to UKF and EKF.

3.4 Developed algorithm (curve fitting algorithm):

Curve fitting is a mathematical tool which gives the relationships between dependent and independent variables in a form of an equation for a given collection of data or a set of data. Curve fitting uses least square approach to find the relation between the parameters. The 'least squares' method is a type of mathematical analysis used to determine the best fit line for data collection, providing a visual representation of the relation of data points. Each data point gives the relationship between a known independent variable and an unknown variable which is dependent on independent variable.

Mathematical analysis of curve fitting:

Let y = f(x) be the equation, where f(x)=ax+b, a and b are constants which are to be determined. **err** $=\sum(d_i)^2 = (y_1 - f(x_1))^2 + (y_2-f(x_2))^2 + (y_3-f(x_3))^2 + \ldots + (y_n-f(x_n))^2$. Put f(x)=ax+b, **err** $==\sum_{\text{data points}=n}^{i=1} (y_i-(f(x_i))^2 = \sum_{\text{data points}=n}^{i=1} (y_i-(ax_i+b))^2$.

To reduce the error err $=\sum_{\text{data points}=n}^{i=1} (y_i-(ax_i+b))^2$.

The error can be reduced by finding

1. Function's derivative which represents the slope of function.
2. To minimize the error the slope should be equated to zero.

   Error is to be dervated w.r.t unknowns and equated to 0.

$$\frac{\partial err}{\partial a} = -2\sum_{i=1}^{n} x_i(y_i - ax_i - b) = 0$$

$$\frac{\partial err}{\partial b} = -2\sum_{i=1}^{n} (y_i - ax_i - b) = 0$$

By solving the above equations we get,

$$a\sum x_i^2 + b\sum x_i = \sum(x_i y_i)$$
$$a\sum x_i + b*n = \sum y_i$$

put these into **matrix form**

$$\begin{bmatrix} n & \sum x_i \\ \sum x_i & \sum x_i^2 \end{bmatrix} \begin{bmatrix} b \\ a \end{bmatrix} = \begin{bmatrix} \sum y_i \\ \sum(x_i y_i) \end{bmatrix}$$

So , the data points(x_i y_i) for i=1,…,n are known.We have all the summation terms in the matrix form. So unknowns are a and b .

$$A = \begin{bmatrix} n & \sum x_i \\ \sum x_i & \sum x_i^2 \end{bmatrix}, \quad X = \begin{bmatrix} b \\ a \end{bmatrix}, \quad B = \begin{bmatrix} \sum y_i \\ \sum (x_i y_i) \end{bmatrix}$$

so

$$AX = B$$

using built in Mathcad matrix inversion, the coefficients a and b are solved

```
>> X = A^-1*B
```

So this is how the least square approach finds the relationship between the dependent term and independent terms and gives us the equation.

Curve fit for R_0 vs SOC:

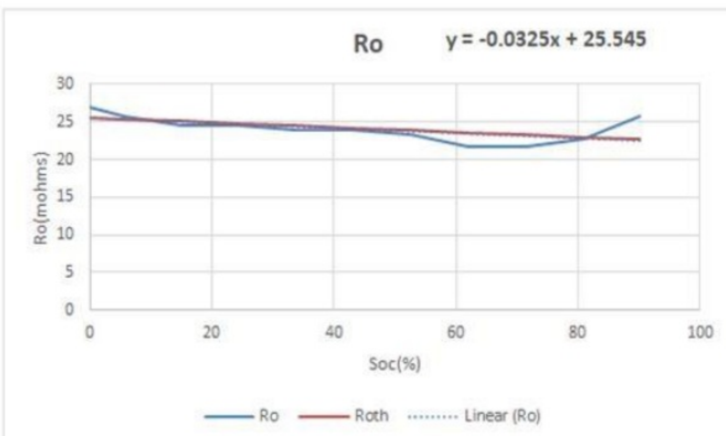


Fig 3.4.1 Curve fit for Ro vs SOC

Curve fit for R1 vs SOC:

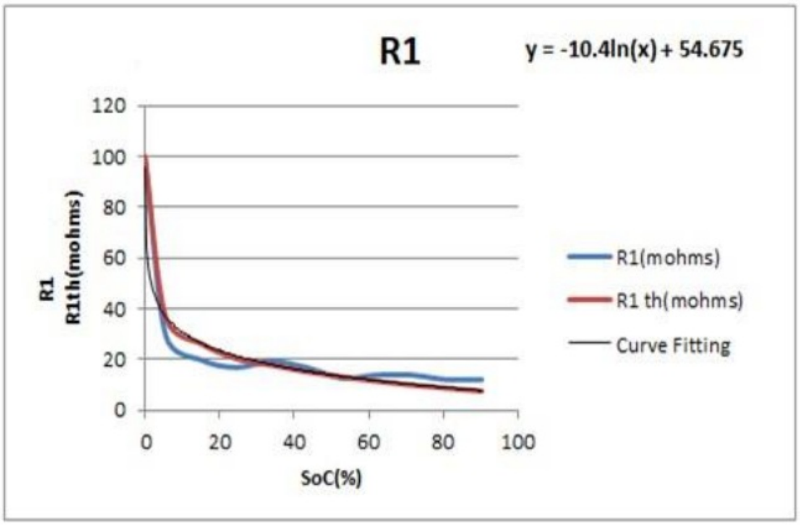


Fig 3.4.2 Curve fit for R1 vs SOC

Curve fit for R2 vs SOC:

SOC(%)	R2(mohms)	R2th(exp)	R2th(lin)
0.019	14.43197	12.36909	8.4686
5.59582	7.26811	1.247939	8.0726
15.01412	4.28661	-153.392	7.4039
24.6549	5.44608	-701.949	6.7195
33.86555	5.44608	-1826.38	6.0655
43.28385	3.66547	-3817.55	5.3585
52.71699	3.08573	-6897.3	4.72709
62.13529	3.08573	-11295.5	4.0583
71.34594	4.86635	-17101.6	3.4044
80.77907	4.86635	-24823.4	2.7346
90.19737	2.46459	-34560.4	2.0659

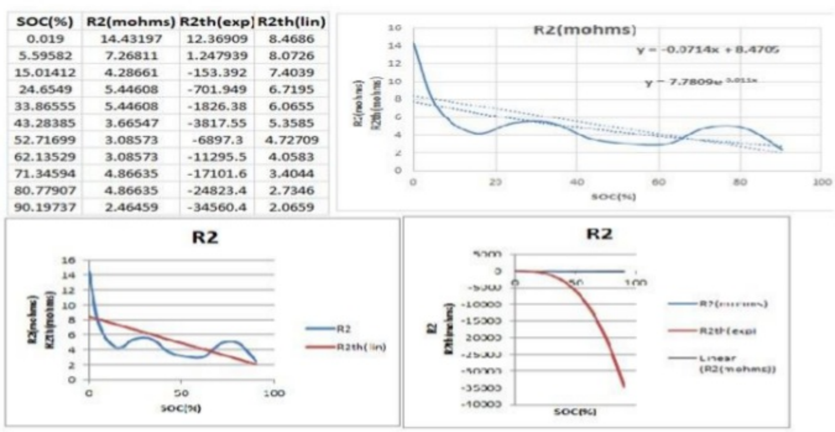
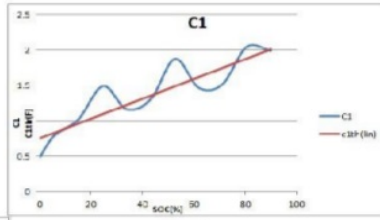
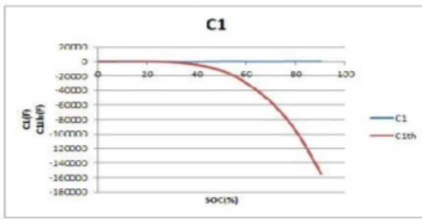


Fig 3.4.3 Curve fit for R2 vs SOC

Curve fit for C1 vs SOC:

SOC(%)	C1(F)	C1th(F)	C1th(lin)(F)
0.019	0.4904	0.4921	0.7502
5.59582	0.7908	0.843	0.8283
15.01412	1.02005	-72.5817	0.9601
24.6549	1.49238	-664.3971	1.0951
33.86555	1.16333	-2563.9865	1.22411
43.28385	1.30562	-7148.3345	1.35597
52.71699	1.87775	-16161.0769	1.488
62.13529	1.46372	-35040.8389	1.61989
71.34594	1.52005	-60843.70766	1.74884
80.77907	2.04969	-99985.888	1.8809
90.19737	1.99238	-155425.054	2.0127





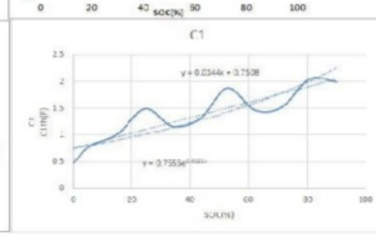


Fig 3.4.4 Curve fit for C1 vs SOC

Curve fit for C2 vs SOC:

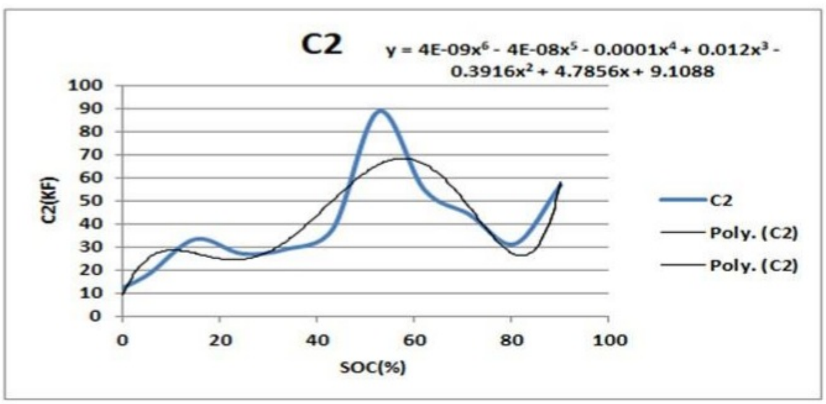
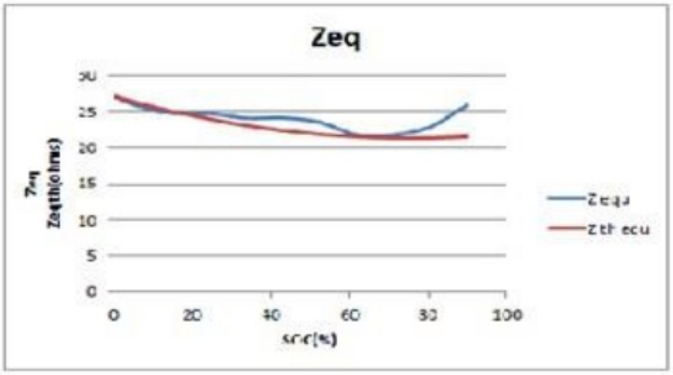


Fig 3.4.5 Curve fit for C2 vs SOC

Curve fit for Z vs SOC:



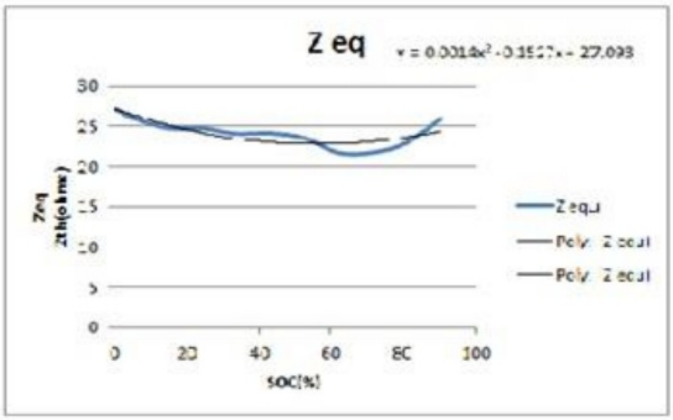
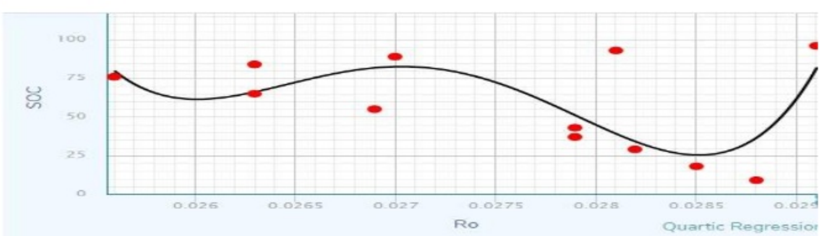


Fig 3.4.6 Curve fit for Z vs SOC

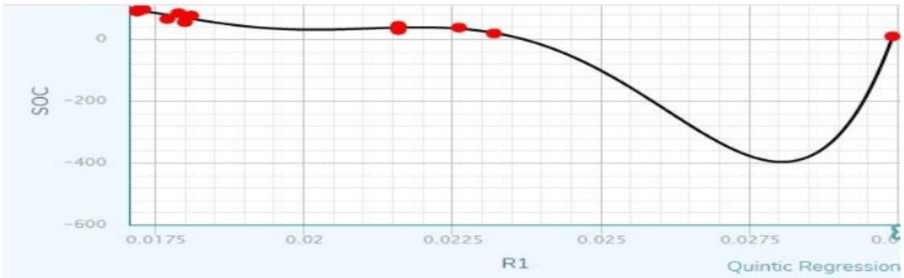
The below are the curves which we got when SoC is on y axis and parameters on x axis **Curve fit for SOC vs Ro:**



Goodness Measures What's this?		
R^2		0.4809
aR^2		0.1843
P		0.2704
SE		27.5
F		1.621
AIC		117.1
BIC		119.5
DWS		2.521
DoF		7
AICc		127.1
Coefficients		
a	8138287	± 5485000
b	−1200049000	± 803800000
c	66312940000	± 44150000000
d	−1627456000000	± 1077000000000
e	14967220000000	± 9843000000000
Equation		
$y = 8138287 - 1200049000x + 66312940000.00001x^2 - 1627456000000x^3 + 14967220000000x^4$		

Fig 3.4.7 Curve fit for SOC vs Ro

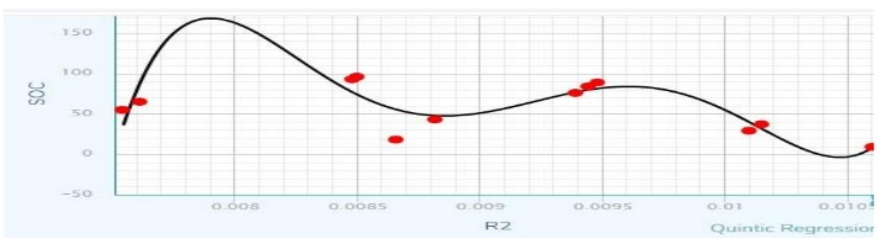
Curve fit for SOC vs R1:



Goodness Measures What's this?		
R^2		0.9232
aR^2		0.8592
P		0.002719
SE		11.42
F		14.43
AIC		96.19
BIC		99.1
DWS		3.698
DoF		6
AICc		113
Coefficients		
a	−235085.7	± 32.02
b	56728400	± 41.67
c	−5413847000	± 177.4
d	255449700000	± 124.7
e	−5957627000000	± 212.3
f	54909300000000	± 90.42
Equation		
$y = -235085.7 + 56728400x - 5413847000x^2 + 255449700000.00003x^3 - 5957627000000x^4 + 54909300000000x^5$		

Fig 3.4.8 Curve fit for SOC vs R1

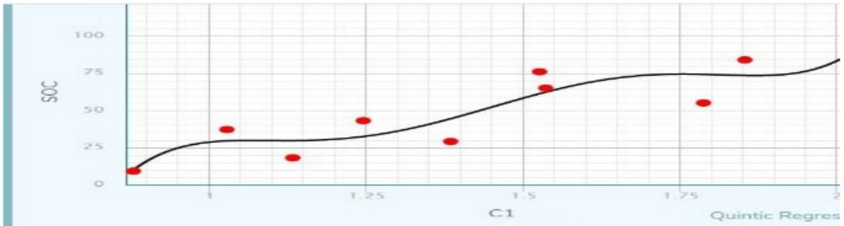
Curve fit for SOC vs R2:



Goodness Measures What's this?		
R^2		0.6807
aR^2		0.4146
P		0.142
SE		23.29
F		2.558
AIC		113.3
BIC		116.2
DWS		2.625
DoF		6
AICc		130.1
Coefficients		
a	−4081497	± 15.12
b	2249155000	± 31.85
c	−493916400000	± 59.66
d	54034500000000	± 70.05
e	−2945108000000000	± 23.43
f	63982310000000000	± 24.66
Equation		
$y = -4081497 + 2249155000x - 493916400000.00006x^2 + 54034500000000x^2 - 2945108000000000x^4 + 63982310000000000x^5$		

Fig 3.4.9 Curve fit for SOC vs R2

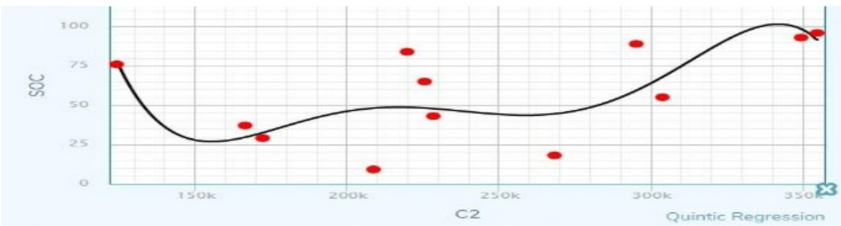
Curve fit for SOC vs C1:



Goodness Measures What's this?		
R^2		0.8623
aR^2		0.7476
P		0.01455
SE		15.3
F		7.515
AIC		103.2
BIC		106.1
DWS		3.115
DoF		6
AICc		120
Coefficients		
a	−8188.021	± 9.94
b	30670.12	± 26.71
c	−44945.47	± 35.98
d	32262.52	± 48.17
e	−11329.42	± 22.09
f	1558.87	± 20.62
Equation		
$y = -8188.021 + 30670.12x - 44945.47x^2 + 32262.52x^3 - 11329.42x^4 + 1558.87x^5$		

Fig 3.4.10 Curve fit for SOC vs C1

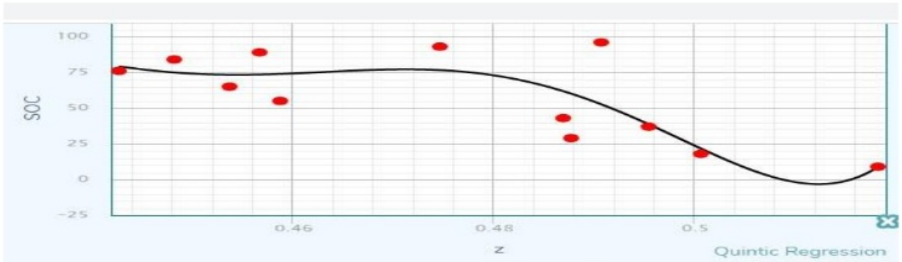
Curve fit for SOC vs C2:



Goodness Measures What's this?		
R^2		0.515
aR^2		0.1109
P		0.383
SE		28.71
F		1.274
AIC		118.3
BIC		121.2
DWS		2.704
DoF		6
AICc		135.1
Coefficients		
a	6364.776	± 17.9
b	−0.1454325	± 50.24
c	0.000001298025	± 51.47
d	$-5.631163 \cdot 10^{-12}$	± 88.47
e	$1.189223 \cdot 10^{-17}$	± 20.93
f	$-9.785498 \cdot 10^{-24}$	± 28.85
Equation		
$y = 6364.776 - 0.1454325x + 0.000001298025x^2 - 5.631163 \cdot 10^{-12}x^3 + 1.1892230000000001 \cdot 10^{-17}x^4 - 9.785498 \cdot 10^{-24}x^5$		

Fig 3.4.11 Curve fit for SOC vs C2

Curve fit for SOC vs Z:



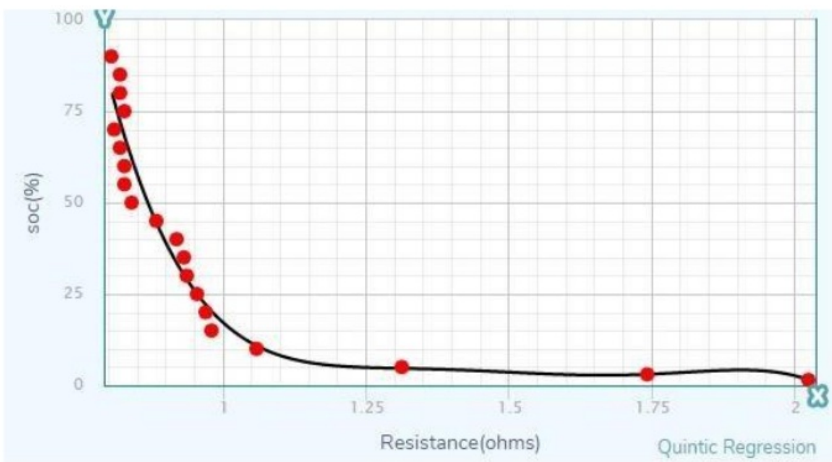
Goodness Measures What's this?		
R^2		0.5854
aR^2		0.24
P		0.2688
SE		26.54
F		1.695
AIC		116.4
BIC		119.3
DWS		2.927
DoF		6
AICc		133.2
Coefficients		
a	−12076260	± 20.31
b	130230300	± 44.11
c	−560972700	± 46.1
d	1206500000	± 75.48
e	−1295568000	± 21.17
f	555673500	± 24.9

Equation

$$y = -12076260 + 130230300x - 560972700x^2 + 1206500000x^3 - 1295568000x^4 + 555673500x^5$$

Fig 3.4.12 Curve fit for SOC vs Z

Curve fit for SOC vs R:



Coefficients		
a	3924.643	± 4.468
b	−13140.78	± 21.11
c	17501.82	± 14.71
d	−11565.77	± 44.56
e	3789.156	± 32.86
f	−492.129	± 5.944

Equation

$$y = 3924.643 - 13140.78x + 17501.82x^2 - 11565.77x^3 + 3789.156x^4 - 492.129x^5$$

Fig 3.4.13 Curve fit for SOC vs R

The below are the conclusions which are drawn by observing the curve fits

PARAMETERS	CURVE FIT STATUS	CONCLUSIONS
R0	MORE DEVIATION	NOT USED
R1	LESS DEVIATION	NOT USED
R2	LESS DEVIATION	NOT USED
R	LESS DEVIATION	USED
C1	MORE DEVIATION	NOT USED
C2	MORE DEVIATION	NOT USED
Z	MORE DEVIATION	NOT USED

Table 3.4.1 Curve fit status

Code:

```
#include <stdio.h>
#include<conio.h>
#include<math.h>

int main()
{
    double res,SOC;
    printf("enter the value of resistance");
    scanf("%lf",&res);
    SOC=(3924.643-(13140.78*res)+(17501.82*pow(res,2))-(11565.77*pow(res,3))+(3789.156*pow(res,4))-(492.129*pow(res,5)));
    printf("SOC=%lf",SOC);

    return 0;
}
```

input

```
enter the value of resistance0.8199
SOC=71.192152
```

Fig 3.4.14 Code for SOC vs R

Steps to be followed for curve fit

- Data should be taken practically i.e. by experimental setup. Any deviation in observations may affect the entire result.
- Choose the appropriate x and y parameters i.e. in which terms the equation should be

- Choose plot the curve using any of the tools which uses least square fit approach.

Problems encountered while developing this algorithm

- Data should be accurate; it should take experimentally. But in our case, we didn't have the possibility of taking the data practically due to unavailability of charger.
- Choosing the appropriate axis parameters. First, we had chosen resistance on y axis and SOC on x axis. So, we got the equation in terms of resistance. Here the challenging task is we need to find SOC which is in terms of powers. So, for finding SOC we need to find roots. Here the question rises is, if we have the 4^{th} order polynomial then, we will get 4 roots that means we get 4 SOC values. Which SOC should be considered? No answer for this which made us to go for next approach.
- SOC on y axis and resistance on x axis resulted the equation in terms of soc.so it will be easy to calculate the value. But we didn't get the exact curve fit. so, this laid a foundation for new idea i.e. piecewise curve fitting.
- Even the piecewise curve fit has some deviations, here the task is we need to write a program to find SOC.As we did piecewise curve fit, we have many equations but for writing code the resistance should be in specific order. But the value of resistance is not in specific order.
- Finally, we didn't consider the piecewise curve fit due to the mentioned reason. We went with SOC in terms of resistance and write the code for the same.

CHAPTER 4

DATA DRIVEN MODEL

4.1 Introduction to data driven model

Artificial neural networks are computer - based systems that provide the (often unknown) connection between input and output data by artificially replicating the anatomical structure and operation of human brain structures. Rather, the normal neural network includes of a very great amount of nerve cells, named neurons, and interconnected together in a vast network. ANN 's intelligent behaviour is the product of intensive cooperation between interconnected units. Output of a neuron consists of the feedback signals from the associated neurons.

In other words, the Artificial Neural Networks (ANN) are entirely linked with several layers of neural networks. They are comprised of one input layer, multiple hidden layers and one output layer. By increasing the number of hidden layers, we are making the network deeper.

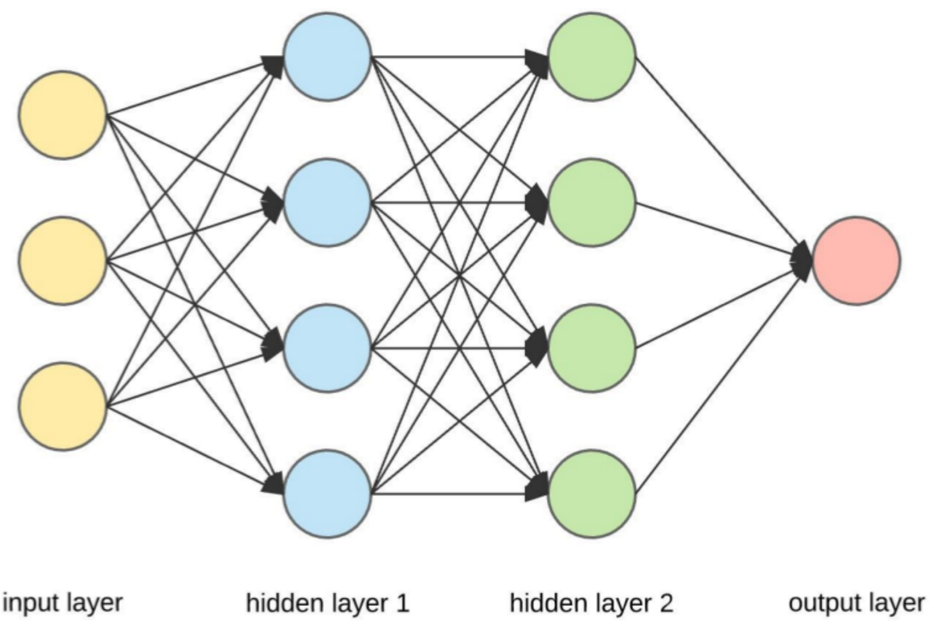


Fig 4.1 architecture of artificial neural network.

Input Layer: As the name implies, it accepts inputs made available by the programmer in many different formats.

Hidden Layer: The hidden layer shows the input and output layers in between. To find hidden features and patterns it performs all the calculations.

Output Layer: The input goes through a series of transformations using the hidden layer which ultimately leads to the output being transmitted using this layer.

The artificial neural network takes input and determines the weighted sum of the inputs and provides a bias. This calculation is in the form of a transfer function.

$$\sum_{i=1}^{n} Wi * Xi + b$$

It determines weighted total is passed to produce the output as an input to an activation function. Activation functions choose whether or not a node will start fire. Only fired ones make it into the output layer. There are distinctive activation mechanisms that can be applied to the type of task that we do.

4.2 WEKA Software

WEKA stands for Waikato Environment for Knowledge Analysis. Weka is a set of machine learning algorithms that are used to solve real world data mining problems. It is developed in java, and runs on almost any machine. You also can add the algorithms directly to a dataset, or use your own Java code to call them.

Features of weka:

- machine learning
- data mining
- pre-processing
- classification
- regression
- clustering
- association rules
- attribute selection
- experiments
- workflow
- visualization

4.3. Training and testing in WEKA

Steps to be followed for training ANN in weka:

- The data should be in arff format Attribute-Relation File Format.

Example of the format

```
@relation battery
@attribute r real
@attribute z real
@attribute soc{1,2,3,4,5,6,7,8,9,10}
@data
2.3664,0.586704,1
2.1536,0.5768,1
1.7968,0.55216,1
1.6016,0.549088,1
1.4448,0.54538,1
1.1984,0.567632,1
1.1184,0.52492,1
```

- Open weka
- Open explorer
- Open file (choose our file which should be in arff format)
- Choose filter
- Go to classify
- For training and testing the number samples should be divided in the ratio 4:1 that means if we have 100 instances, the training set consists of 80 instances and the test set consists of 20 instances.
- For training set click use training set, choose the filter and start. In our case we had used functions (multilayer perceptron and j48 for verifying).
- For obtaining neural network architecture (GUI), click on selected filter then the following window appears. Select GUI as true
- The output window appears.

The terms in the output screen are
In weka the method which we are going to use is multilayer perceptron to train the weka. Basically, in this method a certain collection of data is given for training purpose and remaining is given for testing. We get the weight matrix by using this method. There are certain terms which are to be known before knowing about the result obtained by using MLP.

RMSE: The root mean square error gives the difference between the values estimated by model and the values that are observed. This should be of very less value for accuracy.

Mean Square Error: It is the average of absolute errors.

True Positive: It says about the instances correctly classified by the algorithm.

Matthews correlation coefficients: It helps to known how well the classified model is performing. It has a range of -1 to +1. -1 indicates a completely wrong binary classifier and vice versa.

ROC Area (Receiver Operating Characteristics): It gives us an idea of how the classifiers are performing in general.

Confusion Matrix: In this matrix, the diagonal elements are the instances which are correctly classified by the algorithm. The other elements are incorrectly classified by the algorithm.

4.4 Multilayer Perceptron

Multilayer perceptron are networks of perceptron's, linear classification networks. In addition, using "hidden layers" they can enforce arbitrary decision boundaries Weka has a graphical interface that helps you to build as many perceptron and links as you want with your own network structure.

Back Propagation has 3 layers namely input layer, multiple hidden layers and output layer. The input from the input layer goes to the hidden layer as input and then the output of hidden layer goes to the output layer as input. From this we get the actual output. The error is calculated by back propagating and is given to the input via this output layer so that weights get updated.

Consider the neural network below

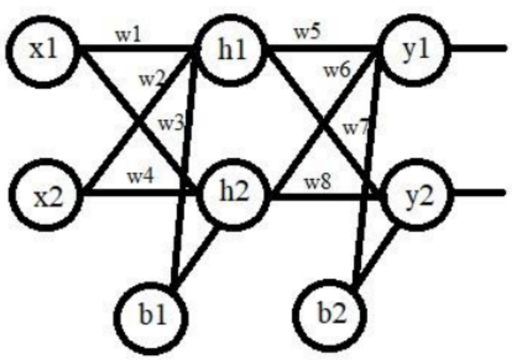


Fig 4.4.1 layers in mlp

Hidden layer can be calculated as,

H1=x1*w1+x2*w2+b1

Assuming the weights, by using the activation function the output of hidden layers is calculated. The same way the output of output layer is calculated as the hidden layer. These values differ with the target values. The error is calculated, and then back propagation is done to update the weights. For example, if we want to calculate the error at w5, then we should partially differentiate the error with respect to w5. The updated w5 is then calculated as W5= w5-(learning rate) *error calculated at w5.

The same process is done at all the layers and the weights are calculated. Then forward propagate to know the output. This process is repeated until the output values are close to the target values.

Confusion Matrix:

	TP Rate	FP Rate	Precision	Recall	F-Measure	MCC	ROC Area	PRC Area	Class
	1.000	0.000	1.000	1.000	1.000	1.000	1.000	1.000	1
	0.875	0.000	1.000	0.875	0.933	0.929	0.995	0.946	2
	0.875	0.041	0.700	0.875	0.778	0.756	0.979	0.781	3
	0.750	0.014	0.857	0.750	0.800	0.782	0.983	0.786	4
	0.875	0.014	0.875	0.875	0.875	0.861	0.987	0.832	5
	1.000	0.027	0.800	1.000	0.889	0.882	0.986	0.800	6
	1.000	0.000	1.000	1.000	1.000	1.000	1.000	1.000	7
	1.000	0.014	0.889	1.000	0.941	0.936	0.993	0.889	8
	0.875	0.000	1.000	0.875	0.933	0.929	0.993	0.938	9
	0.778	0.000	1.000	0.778	0.875	0.870	0.988	0.895	10
Weighted Avg.	0.901	0.011	0.913	0.901	0.902	0.894	0.990	0.887	

```
=== Confusion Matrix ===

 a b c d e f g h i j   <-- classified as
 8 0 0 0 0 0 0 0 0 0 | a = 1
 0 7 1 0 0 0 0 0 0 0 | b = 2
 0 0 7 0 1 0 0 0 0 0 | c = 3
 0 0 2 6 0 0 0 0 0 0 | d = 4
 0 0 0 1 7 0 0 0 0 0 | e = 5
 0 0 0 0 0 8 0 0 0 0 | f = 6
 0 0 0 0 0 0 8 0 0 0 | g = 7
 0 0 0 0 0 0 0 8 0 0 | h = 8
 0 0 0 0 0 0 0 1 7 0 | i = 9
 0 0 0 0 0 2 0 0 0 7 | j = 10
```

Fig 4.4.2 Confusion matrix

GUI OUTPUT:

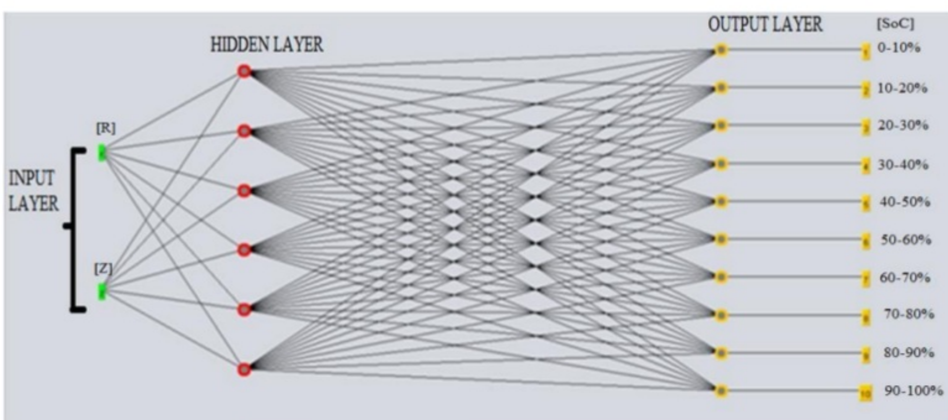


Fig:4.4.3 GUI Output

WEKA Result:

```
=== Summary ===

Correctly Classified Instances          73               90.1235 %
Incorrectly Classified Instances         8                9.8765 %
Kappa statistic                          0.8903
Mean absolute error                      0.0307
Root mean squared error                  0.1239
Relative absolute error                 17.0696 %
Root relative squared error             41.3157 %
Total Number of Instances               81

=== Detailed Accuracy By Class ===

                 TP Rate  FP Rate  Precision  Recall   F-Measure  MCC      ROC Area  PRC Area  Class
                 1.000    0.000    1.000      1.000    1.000      1.000    1.000     1.000     1
                 0.875    0.000    1.000      0.875    0.933      0.929    0.995     0.946     2
                 0.875    0.041    0.700      0.875    0.778      0.756    0.979     0.781     3
                 0.750    0.014    0.857      0.750    0.800      0.782    0.983     0.786     4
                 0.875    0.014    0.875      0.875    0.875      0.861    0.987     0.832     5
                 1.000    0.027    0.800      1.000    0.889      0.882    0.986     0.800     6
                 1.000    0.000    1.000      1.000    1.000      1.000    1.000     1.000     7
                 1.000    0.014    0.889      1.000    0.941      0.936    0.993     0.889     8
                 0.875    0.000    1.000      0.875    0.933      0.929    0.993     0.938     9
                 0.778    0.000    1.000      0.778    0.875      0.870    0.988     0.895     10
Weighted Avg.    0.901    0.011    0.913      0.901    0.902      0.894    0.990     0.887
```

Fig 4.4.4Weka Result

Validation of curve fit with ANN:

The developed curve fit model is validated with ANN and below are the conclusions drawn
.

CURVE FIT	ANN
Plot curve between SOC and R	Divide the data into test set and training set
Obtain equation for SOC in terms of R	Train the ANN using MLP
Result: For R=0.8199 SOC=71.192	Result: For R=0.8199 SOC=8th stage. RMS error=0.1239
Can be further extended to piecewise curve fit	It displays graphical user interface

Table 4.4.1 Validation of curve fit with ANN

CONCLUSION

Recent trends show that the Electric vehicles had almost replaced the conventional vehicles. As we further move on to deep analysis of electric vehicle it was known that battery parameters are crucial and they are the parameters which need frequent monitoring. Out of all the parameters, SOC and SOH are important and these are the parameters which we rely on for the performance of the battery, this made us to work on the estimation methods for SOC. A detailed analysis of different SOC estimation methods are done, also compared all the estimation methods and we came out with an idea to develop a new algorithm for SOC estimation. For developing an algorithm, we did an analysis of practical data of the $LIFEPO_4$ battery and made the data to fit on a curve I.e., curve fitting. After curve fit, we wrote a code for determining SOC which takes the input as resistance of the battery and gives displays SOC.

For validating the developed algorithm, we went for an artificial neural network (ANN) as it has high accuracy. WEKA is used for building data driven model. And the estimates of the WEKA is analysed and it has RMSE of 0.1239 & mean absolute error of 0.0307 % which is inacceptable range. The developed algorithm can be used for SOC estimation and its implementation helps us to reduce the complexity and also helps the individual to understand better about the battery.

FUTURE SCOPE

- As this developed algorithm is simple and easily understandable, it can be implemented and can be used in wide range.
- The same analysis can also be done for SOH estimation as the SOC and SOH are correlated.
- Analysing the developed SOC estimation method with different charging strategies and finding the best charging strategy yields a better result.
- Life time of the battery can be increased by choosing the best charging strategy and the method which gives us the accurate SOC.
- Further efforts are to be done for decreasing the charging time of the battery by using appropriate charging scheme.

DEVELOPMENT OF INTELLIGENT SOC ESTIMATION ALGORITHM FOR LITHIUM POLYMER BATTERIES

ORIGINALITY REPORT

11%	6%	4%	7%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	www.genstattu.com Internet Source	3%
2	Submitted to University of West London Student Paper	<1%
3	Submitted to Central Queensland University Student Paper	<1%
4	Submitted to Liverpool John Moores University Student Paper	<1%
5	Submitted to University of Nottingham Student Paper	<1%
6	Submitted to South Dakota Board of Regents Student Paper	<1%
7	Submitted to University of Wolverhampton Student Paper	<1%
8	Submitted to Raha International School Student Paper	<1%
9	Hedra Saleeb, Khairy Sayed, Ahmed Kassem,	

	Ramadan Mostafa. "Power Management Strategy for Battery Electric Vehicles", IET Electrical Systems in Transportation, 2019 Publication	<1%
10	Submitted to Westminster Academy School Student Paper	<1%
11	Submitted to University of Hertfordshire Student Paper	<1%
12	doc.lagout.org Internet Source	<1%
13	www.mdpi.com Internet Source	<1%
14	byjus.com Internet Source	<1%
15	Submitted to Queen's University of Belfast Student Paper	<1%
16	Submitted to Coventry University Student Paper	<1%
17	worldwidescience.org Internet Source	<1%
18	lutpub.lut.fi Internet Source	<1%
19	Submitted to Island School Student Paper	<1%

20	docshare.tips Internet Source	<1%
21	Submitted to University of Southern California Student Paper	<1%
22	www.archive.org Internet Source	<1%
23	www.supplydemandmarketresearch.com Internet Source	<1%
24	G. Raghotham Reddy. "A Neuro-Fuzzy System for Automatic Multi-Level Image Segmentation using KFCM and Exponential Entropy", IFIP International Federation for Information Processing, 2007 Publication	<1%
25	Changkuan Zhang, Hongwu Tang. "Advances in Water Resources and Hydraulic Engineering", Springer Science and Business Media LLC, 2009 Publication	<1%
26	eprints.maynoothuniversity.ie Internet Source	<1%
27	Submitted to Queensland Academy for Health Sciences Student Paper	<1%
28	Submitted to Heriot-Watt University	

Student Paper

<1%

29	M. A. Si Mohammed, A. Bellar, A. Adnane, H. Boussadia. "Performance analysis of attitude determination and estimation algorithms applied to low earth orbit satellites", 2016 UKACC 11th International Conference on Control (CONTROL), 2016 Publication	<1%
30	hdl.handle.net Internet Source	<1%
31	Submitted to University of Warwick Student Paper	<1%
32	Submitted to Universiti Putra Malaysia Student Paper	<1%
33	Anirudh Allam, Simona Onori. "An Interconnected Observer for Concurrent Estimation of Bulk and Surface Concentration in Cathode and Anode of a Lithium-ion Battery", IEEE Transactions on Industrial Electronics, 2018 Publication	<1%
34	Max Bramer. "Principles of Data Mining", Springer Science and Business Media LLC, 2020 Publication	<1%

35	Submitted to The University of Manchester Student Paper	<1%
36	Submitted to Cranfield University Student Paper	<1%

Exclude quotes	Off	Exclude matches	Off
Exclude bibliography	Off		