MASTER

Remaining useful life prediction of lithium-ion batteries using machine learning

Bhargav, Kartik B.

Award date:
2021

Technische Universiteit Eindhoven University of Technology

Department of Mathematics and Computer Science
Uncertainty in Artificial Intelligence Research Group

Remaining useful life prediction of lithium-ion batteries using machine learning

Master Thesis

Kartik B Bhargav

Supervisor:

dr. ir. Erik Quaeghebeur

External Supervisor:

Subhajeet Rath

Other Committee Members:

Prof. dr. ir. Boudewijn van Dongen

Version 2.0

Eindhoven, October 2021

Abstract

Automobile vehicles are an integral part of our lives. They are used in a plethora of applications such as passenger commuting, goods transportation, travel services, emergency services, delivery services, etc. Traditional automobile vehicles use hydrocarbon fuels such as petrol or diesel as their energy source. These types of automobile vehicles are used by the vast majority of the population.

Over the years, usage of these vehicles has raised concerns about air pollution, global warming, fossil fuel depletion, etc., since these vehicles emit greenhouse gases such as carbon dioxide, nitrous oxide and methane which are byproducts of the combustion of hydrocarbon fuels that are required for the vehicles' movement.

This has led to a boom in the manufacturing of automobile vehicles that run on alternative green energy that is environmentally friendly. These types of automobile vehicles generally use electrical energy as the alternative energy which is generally supplied by lithium-ion batteries (LIB).

Lithium-ion batteries have a limited lifetime and can not be used after they have reached their end of life. Using a lithium-ion battery after it has reached its end of life can lead to serious consequences and can sometimes cause catastrophic battery explosion. Also, range anxiety among drivers and potential customers has become a barrier in the sales of electric vehicles. Hence, accurate prediction remaining useful life of lithium-ion batteries is of utmost importance in the domain of electric vehicles.

The above mentioned issues can be addressed using accurate RUL prediction models. Thus, as part of this research, machine learning models such as Support Vector Machine (SVM), Long Short Term Memory (LSTM) network and Similarity based Model (SBM) are implemented to predict the remaining useful life of a set of lithium-ion batteries. The effect of resting of batteries (calendar aging) is also considered. It is observed that the LSTM models perform better than the SVM models and they take much less time to train than the latter. SBM performs better than both LSTM models and SVM models and they can be trained faster than LSTM and SVM models. It is also observed that the resting of batteries impacts their remaining useful life.

Acknowledgments

Contents

List of Figures

List of Tables

Abbreviations and Acronyms

- **AED-SDA** - Average Euclidean Distance Stacked Denoising Autoencoder
- **ANN** - Artificial Neural Network
- **ARIMA** - Auto Regressive Integrated Moving Average
- **BATPFNN** - Bat based Particle Filter Neural Network
- **BOL** - Beginning of Life
- **BMS** - Battery Management System
- **BPSO-SVM** - Binary Particle Swarm Optimization - Support Vector Machine
- **CALCE** - Center for Advanced Life Cycle Engineering
- **CC** - Coulomb Counting
- **CCCV** - Constant Current Constant Voltage
- **CNN** - Convolutional Neural Network
- **CTGAN** - Conditional Generative Adversarial Network
- **DCNN** - Deep Convolutional Neural Network
- **DCNN-EL** - Deep Convolutional Neural Network with Ensemble Learning
- **DCNN-TL** - Deep Convolutional Neural Network with Transfer Learning
- **DOD** - Depth of Discharge
- **DNN** - Deep Neural Network
- **DTW** - Dynamic Time Warping
- **EDA** - Exploratory Data Analysis
- **EIS** - Electrochemical Impedance Spectroscopy
- **ELM** - Extreme Learning Machine
- **EOL** - End of Life
- **EV** - Electric Vehicle
- **GA** - Genetic Algorithm
- **GPR** - Gaussian Process Regression
- **GRU-GPR** - Gated Recurrent Unit kernel - Gaussian Process Regression

- **IBL** - Instance Based Learning

- **kNN** - k Nearest Neighbors

- **LCO** - Lithium Cobalt

- **LIB** - Lithium-ion Battery

- **LSSVM** - Least Squared Support Vector Machine

- **LSTM** - Long Short Term Memory

- **MLE** - Maximum Likelihood Estimation

- **NASA** - National Aeronautics and Space Administration

- **OCV** - Open Circuit voltage

- **OEM** - Original Equipment Manufacturer

- **PCoE** - Prognostics Center of Excellence

- **PDF** - Probability Density Function

- **PF** - Particle Filter

- **PHM** - Prognostics and Health Management

- **RDR** - Remaining Driving Range

- **RNN** - Recurrent Neural Network

- **RTF** - Run to Failure

- **RUL** - Remaining Useful Life

- **RVM** - Relevance Vector Machine

- **SEI** - Solid Electrolyte Interphase

- **SBM** - Similarity based Model

- **SOC** - State of Charge

- **SOH** - State of Health

- **SVM** - Support Vector Machine

Chapter 1

Introduction

The main area of focus of this research is to predict the remaining useful life (RUL) of a lithium-ion battery (LIB) onboard an electric vehicle (EV) in real-time. The remaining useful life of a lithium-ion battery is affected by various factors such as voltage, current, temperature, state of charge (SOC), state of health (SOH), resting duration, etc. In this research, we implement machine learning models that predict the RUL of a lithium-ion battery. Additionally, we also show that duration of rest of a battery has a significant effect on its RUL.

In Section 1.1, we motivate the importance of this study. In Section 1.2, we define the scope of this study. In Section 1.3, we briefly discuss the research approach and methodology used in this research. In Section 1.4, we present the outline of this thesis.

1.1 Motivation

Conventional automobile vehicles, also known as internal combustion engine vehicles (ICEV), use hydrocarbon fuels as their source of energy. These vehicles produce poisonous gases such as carbon dioxide, nitrous oxide and methane, which are the byproducts of hydrocarbon fuel combustion. The emission of these gases has raised concerns about air pollution and global warming. Extensive use of these vehicles has also resulted in fossil fuel depletion. Thus, there is huge demand for automobile vehicles that run on alternative green energy such as electricity.

Electric Vehicles (EVs) are becoming increasingly popular over the years. These vehicles are environmentally friendly as they run on green energy. Many automobile manufacturers have started manufacturing electric vehicles and over the years there is a growing demand for electric vehicles. EVs generally use lithium-ion batteries as their energy source.

When electric batteries are used in real-time, they are subjected to various factors such as temperature changes, load changes, etc., which directly affect their degradation. Lithium-ion batteries have a limited lifetime and cannot be used after they have reached their end of life. Using a lithium-ion battery after it has reached its end of life can be dangerous. It can lead to a battery fire incident or the battery may explode (P. Sun et al., 2020). While most of these fire incidents are reported to have occurred during the charging of the battery, recently it was reported that an electric vehicle caught fire while the vehicle was running (Herh, 2021). These battery fires are lethal to the drivers and the passengers. Additionally, the harmful gases released during these fire incidents are also poisonous (P. Sun et al., 2020).

Thus, to prevent catastrophic battery failure, accurate and timely prediction of remaining useful life of a battery in an EV can update the driver and then the driver can take necessary action.

1.2 Scope

In this research, the research problem is to predict the RUL of the LIB in an EV. In this study, we will implement the following RUL prediction models - Support Vector Regression (SVR), Long Short Term Memory (LSTM) and Similarity based model (SBM). In this research, we consider the effects of battery resting (calendar aging) on the RUL of LIBs. The data set used for this research is having the data of eight LIBs. In this research, we are interested in implementing RUL prediction models that return the point estimate of the RUL. The models developed in this research work only on the data set provided by TNO and it may not work for a different data set.

1.3 Research Approach and Methodology

To develop machine learning models to predict the RUL of an LIB onboard an EV in real-time, relevant battery health indicators and/or stress factors have to be selected from the data set. In prognostics and health management (PHM), the health indicators of a system, in general, are the properties of the system through which one can infer the present health of the system. The RUL of a battery is directly impacted by the health of the battery. In general, a battery with higher health would generally have a higher RUL as compared to a battery with lower health.

Also, appropriate evaluation metrics have to be selected to evaluate the performance of the machine learning models. The task of predicting the RUL of LIB on-board an EV in real-time can be categorized as a regression problem, as the predicted RUL is a continuous value. Hence, evaluation metrics such as mean absolute error (MAE), mean absolute percentage error (MAPE), root mean squared error (RMSE) and coefficient of determination (R^2) are used to evaluate the different RUL prediction models that will be implemented as part of this research. Also, the running time of a model is an extremely important factor as the RUL prediction model will be used in real time. An ideal RUL prediction model will typically take as less time as possible to generate the prediction.

1.4 Thesis Outline

This thesis report is organized as follows. Chapter 2 covers preliminaries where we discuss about the basic working principle of a lithium-ion battery. We also discuss briefly about battery aging, lithium-ion battery form factors and battery management systems. We also introduce the battery related terminology that are used throughout this report.

Chapter 3 comprises a literature review on some of the most commonly used lithium-ion battery (LIB) degradation data sets and basic exploratory data analysis (EDA) is performed on the data sets. Also, a detailed review on various methods and techniques to predict the RUL of LIBs is presented. In Chapter 4, we formulate the problem of Remaining Useful Life (RUL) prediction of lithium-ion battery (LIB) and we pose additional research questions that will be addressed as part of this research. In Chapter 5, the TNO data set is described in detail. Exploratory Data Analysis (EDA) is performed on the TNO data set, We then describe the various steps involved in data cleaning, data pre-processing and feature engineering. In Chapter 6, various machine learning models to predict the remaining useful life (RUL) of lithium-ion batteries (LIB) are explained. In Chapter 7, the results obtained from the implemented machine learning models are discussed. Finally, the conclusions, limitations and future work to be done in this research are discussed in Chapter 8.

Chapter 2

Preliminaries

A good understanding of the structure and working of a lithium-ion battery is crucial in understanding the various factors that affect the RUL of the battery. In this chapter, several definitions and terminologies related to LIBs are introduced which will be used throughout this thesis. In Section 2.1.1, we describe the basic working principle of a lithium-ion battery. In Section 2.1.2, we describe briefly the various LIB form factors. We define various terms and definitions related to battery in Section 2.1.3. In Section 2.2, we describe regarding battery management systems (BMS). In Section 2.3, we briefly describe about the aging in lithium-ion batteries.

2.1 Lithium-ion batteries

For an electric vehicle, the battery is the main source of energy. When EVs were first introduced, they used lead-acid batteries as their energy source. The energy density of a lead-acid battery is around 30 Wh/kg and in general, a lead-acid battery has a cycle life of around 300 cycles. The energy density of lead-acid battery is very less when compared to the energy density of gasoline used in internal combustion engine vehicles (ICEV) which is more than 10000 Wh/kg (Xiong, 2019). Hence, to match the performance of an ICEV, an EV with a lead-acid battery occupies more space and it also increases the mass of the vehicle. Also, lead-acid batteries result in lead emissions which are harmful to the environment.

The above shortcomings of a lead acid battery are overcome by a lithium-ion battery. The advantages of using lithium-ion batteries in EVs are as follows (Xiong, 2019).

1. Lithium-ion batteries (LIB) have very high specific energy (about 200 Wh/kg). It is much higher compared to lead-acid batteries.

2. Lithium-ion batteries have a longer cycle life (about 2000) than lead acid batteries. LIBs, when operated under a low depth of discharge, can be cycled more than 2000 cycles.

3. LIBs have a very low self-discharge rate (about 6% - 8%).

4. They are environmentally friendly. They are non-polluting batteries.

5. They have a high working voltage of (about 3.6 V).

6. LIBs can be manufactured in various form factors and thus, they easily meet the battery layout requirements in EVs. Further details about these form factors are discussed in Section 2.1.2.

7. They can be charged or discharged at any time and this does not affect their performance. Thus, lithium-ion batteries are memoryless.

The smallest packaged unit of lithium-ion battery that is manufactured is known as a *cell*. A single cell cannot power an EV. Hence, many cells are connected either in series or in parallel to form a *module*. Several modules are then connected either in parallel or in series to form a *battery pack*. A battery pack can supply enough power to power an EV (Team, 2008).

2.1.1 Functioning of a lithium-ion battery

The basic structure and working of a lithium-ion battery is illustrated in Figure 2.1. Both the positive electrode and the negative electrode of the battery are immersed in the electrolyte. The charging and discharging of the lithium-ion battery is achieved by inserting and releasing the lithium ions between the positive and negative electrodes.
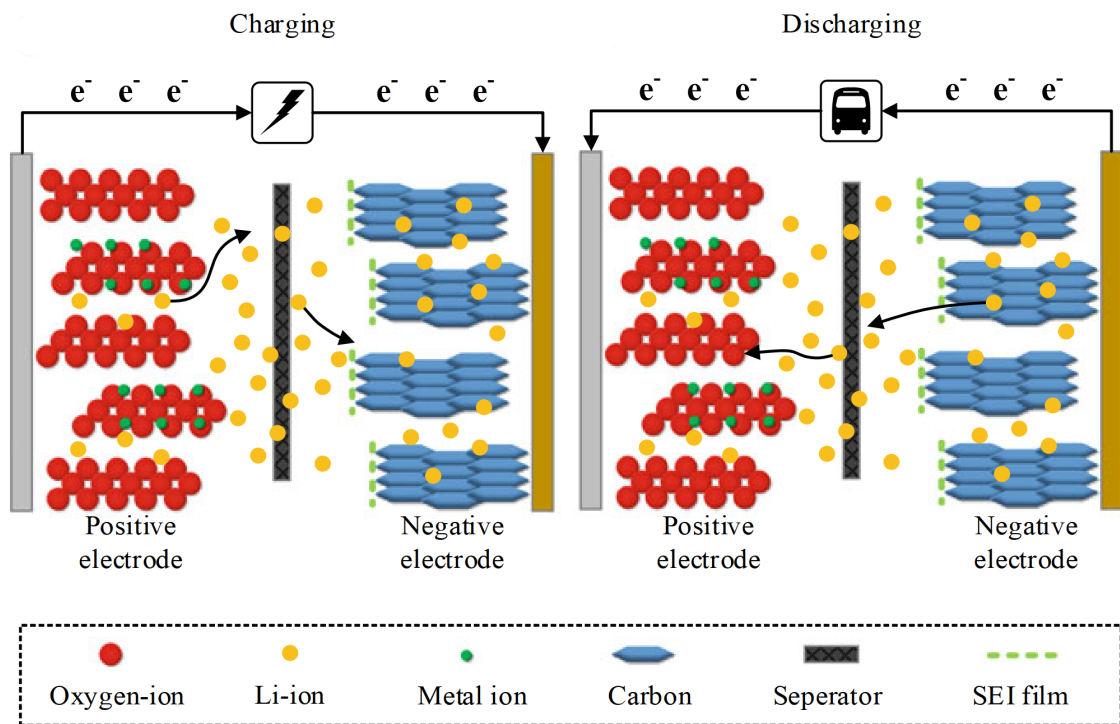
Figure 2.1: Illustration of the working principle of a Lithium-ion battery (Xiong, 2019)

During charging, the lithium ions move from the positive electrode into the electrolyte. The lithium ions present in the electrolyte are driven to move toward the negative electrode as the lithium ions in the electrolyte have a large concentration difference near the positive and negative electrodes. Finally, the lithium ions are embedded in the negative electrode through the separator. Meanwhile, the electrons in the external charging circuit move from the positive electrode to the negative electrode to form a current. Reduction occurs at the negative electrode as it receives electrons and oxidation occurs at the positive electrode as it loses electrons.

The discharging process is the opposite of the charging process. Lithium ions are released from the negative electrode and are embedded back into the positive electrode. The electrons in the external discharging circuit move from the negative electrode to the positive electrode through the load, thereby driving the EV. Oxidation occurs at the negative electrode as it loses electrons, whereas reduction occurs at the positive electrode since it receives the electrons.

When a LIB is charged for the very first time, a solid interface film is formed between in the electrolyte near the negative electrode. This film is called solid electrolyte interphase (SEI) film. It prevents the negative electrode from corrosion and it also prevents the reduction reaction between the electrolyte and the negative electrode. The SEI film constantly thickens or dissolves over time. This results in a decrease in the number of lithium-ions and active materials available for recycling. This is the primary reason for battery capacity decline.

2.1.2 Lithium-ion battery form factors

Lithium-ion batteries that are used in EVs are manufactured mainly in three form factors - *cylindrical cell, prismatic cell and pouch cell*. Each form factor has its own advantages and disadvantages. Different EV original equipment manufacturers (OEM) use different LIB form factors and no form factor satisfies all the requirements of the OEM. In this section, we briefly discuss about the different LIB form factors.

2.1.2.1 Cylindrical cell

A cylindrical cell is one of the commonly used lithium-ion battery form factors. It comprises sheet-shaped anodes, separators and cathodes. All these components are sandwiched, rolled and are packed within a cylinder-shaped container (Arar, 2020). This form factor of lithium-ion battery is one of the earliest form factors that was produced in bulk. The structure of a cylindrical cell is illustrated in Figure 2.2.
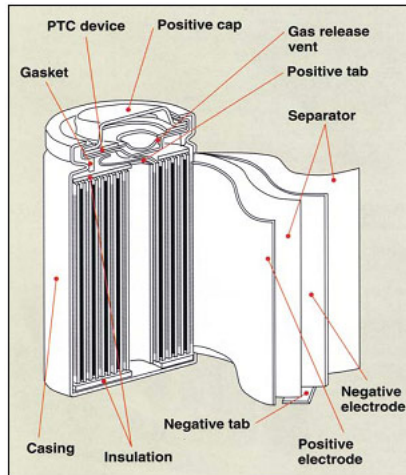
Figure 2.2: Structure of a cylindrical cell. Source: (Arar, 2020)

A Cylindrical cell has several advantages. This form factor is well suited for automated manufacturing (Arar, 2020). The cylindrical shape of the cell evenly distributes the internal pressure generated from the side reactions over the cell surface. This feature makes cylindrical cell form factor withstand a much higher internal pressure while still being intact (Arar, 2020).

The circular cross-section of the cylindrical form factor does not allow to completely utilize the battery space. Hence, the packing density of this form factor is low. However, because of the presence of cavities between the cells, this form factor allows for better thermal management.

2.1.2.2 Prismatic cell

A prismatic cell consists of large sheets of anodes, separators and cathodes sandwiched and rolled up similar to a cylindrical cell. However, the components of the prismatic cell are fit within a metallic or a plastic container in cubic form (Arar, 2020). The structure of a prismatic cell is illustrated in Figure 2.3.

The parts of the separator sheet and electrode that are close to the corners of the cubic container experience more stress than others. This can lead to damage of the electrode coating and may lead to uneven electrolyte distribution.

Since the prismatic cells are enclosed within a cubic container, cells of this form factor can be packed together compactly. Unlike cylindrical cells, prismatic cells completely utilize the battery space. However, this compact packing requires a complex thermal management system as there are no cavities between the cells in the pack.
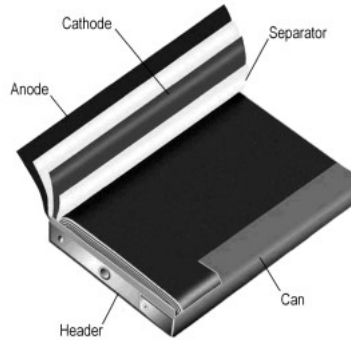
Figure 2.3: Structure of a prismatic cell. Source: (Arar, 2020)

2.1.2.3 Pouch cell

A pouch cell is another commonly used battery form factor. In a pouch cell, the separator layers and the electrode are stacked atop one another unlike cylindrical cells and prismatic cells where the anode, cathode and separator layers are sandwiched and rolled up. The contents of the pouch cell are enclosed within a container that is not rigid. The components of a pouch cell are packed and sealed within a flexible foil that acts as a cell container. The structure of a pouch cell is illustrated in Figure 2.4.

Figure 2.4: Structure of a pouch cell. Source: (Arar, 2020)

Since pouch cells do not have rigid containers, they weigh less as compared to cylindrical cells and prismatic cells. Pouch cells are prone to expansion during usage as the thin boundaries of the pouch cell can not withstand the internal pressure created by the gases that are released during cell operation. This prevents compact packing pouch cells.

2.1.3 Battery related terminology

In this section, we define some of the commonly used battery related terminology.

Beginning of Life (BOL) : The beginning of life (BOL) of a battery is the time point at which the battery has just started operating.

End of Life (EOL) : The end of life (EOL) of a battery is the time point since the beginning of life (BOL) of the battery after which the battery does not operate as per its specifications.

State of Charge (SOC) : SOC of a battery is the available capacity of the battery. SOC is generally expressed as a ratio of the available battery capacity to the maximum capacity. SOC is computed based on the values of current, temperature and voltage (Omariba, L. Zhang and D. Sun, 2018).

Depth of Discharge (DOD) : The percentage of the battery capacity that is discharged during a load cycle is known as depth of discharge (DOD). Typically, DOD is expressed as percentage of the maximum capacity (Omariba, L. Zhang and D. Sun, 2018).

State of Health (SOH) : SOH of a battery is the ratio of its present maximum charge capacity to the maximum charge capacity at its beginning of life (Omariba, L. Zhang and D. Sun, 2018). SOH is computed as shown in equation 2.1

$$SOH = \frac{Q_{act}}{Q_R} * 100\%$$ (2.1)

where Q_{act} is the actual capacity of the battery and Q_R is its rated capacity.

State of Power (SoP) : The state of power (SOP) of a battery is the ratio of peak power to nominal power (Xiang et al., 2018). The peak power of a battery is defined as the power that the battery can deliver persistently within the specified voltage, SoC, current and power constraints for T seconds (Plett, 2004).

Remaining useful life (RUL) : The remaining time or the remaining number of load cycles the battery has during which it will be able to maintain its operating requirements is known as the remaining useful life (RUL) of a battery. Alternatively, RUL is also the length of time from the present time to the end of life (Omariba, L. Zhang and D. Sun, 2018). When the battery is used in an electric vehicle (EV), the remaining useful life of the battery can also be expressed in terms of the remaining distance that the battery can power the electric vehicle. RUL of a battery when expressed as a remaining distance for which it can power an EV before reaching the end of life (EOL) is known as remaining driving range (RDR). RUL is computed as follows.

$$RUL = T_{EOL} - T_{present}$$ (2.2)

where RUL represents the current remaining useful life of the battery, T_{EOL} represents the time at end of life (EOL) of the battery, $T_{present}$ represents the current time instance at which the RUL is computed.

2.2 Battery management systems

A battery management system (BMS) is a system present in an electric vehicle (EV) that manages and ensures proper functioning of the battery. The schematic diagram of a BMS is shown in Figure 2.5. BMS verifies and ensures the reliability and safety of the battery. It also takes necessary actions on the battery when it detects irregular conditions. A battery management system contains a sampling unit that is responsible for the collection of battery temperature, voltage, current and other required information from the battery pack as well as from the individual cells within the battery pack. BMS also contains several computation algorithms in the control circuit which estimate the state of health (SOH), state of charge (SOC), state of power (SOP) and remaining useful life (RUL) of the battery (Xiong, 2019).
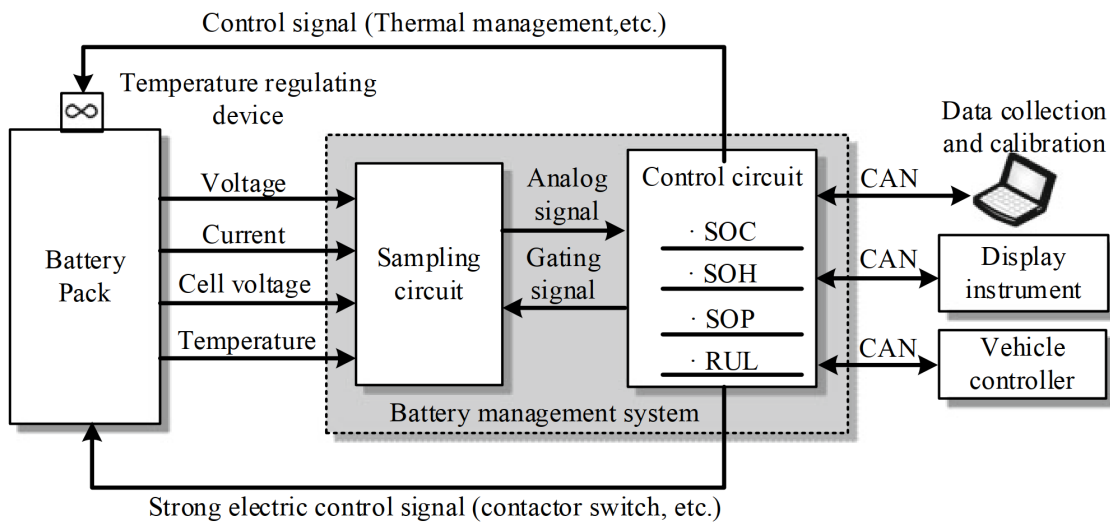
Figure 2.5: A schematic diagram of a Battery Management System (BMS) (Xiong, 2019)

Some of the basic functionalities of a BMS are battery state monitoring, data acquisition, energy management, thermal management, battery state monitoring and battery data management (Xiong, 2019).

2.3 Battery ageing

Lithium-ion batteries have a limited useful life and they undergo irreversible capacity loss when subjected to repeated charging and discharging and also when kept idle. The capacity loss of a lithium-ion battery due to repeated repeated charging and discharging is called as cyclic aging. The capacity loss of a lithium-ion battery when it is at rest is called calendar aging.

2.3.1 Cyclic Aging

The irreversible capacity loss that a lithium-ion battery undergoes when it is discharged and charged repeatedly is called cyclic aging. The capacity loss is dependent on the battery's state of charge (SOC), voltage, current and temperature. As a result, several factors play a role in this type of ageing. High operating temperatures speeds up the degradation process (Barré et al., 2013).

2.3.2 Calendar Aging

Calendar ageing refers to any ageing processes that cause a battery cell to degrade without being subjected to charge-discharge cycle. It is an important concern in EVs as most of the times the EVs will be in rest. The growth of passivation layers at the electrode–electrolyte interfaces is the major cause of calendar ageing (Keil et al., 2016).

Chapter 3

Literature Review

In this chapter, the literature review regarding this research presented. A brief overview of various remaining useful life (RUL) prediction methodologies is presented. In section, we review some of the commonly used publicly available lithium-ion battery degradation data sets.

3.1 Types of RUL prediction data sets

The data used to predict the RUL of a system can be classified into three categories as illustrated in Figure 3.1 - *lifetime data, run-to-failure data and threshold data.*

In lifetime data, the probability distribution of the failure times of the system are acquired and it is used to predict the RUL. Figure 3.2 shows the probability of a battery reaching its end of life (EoL) given the number of cycles the battery is in operation (Baru, 2018).
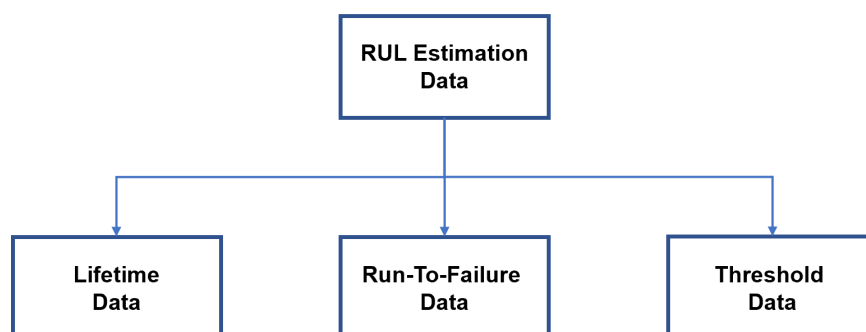
Figure 3.1: Plot showing SOH vs Cycles for the LIBs in NASA data set

In run-to-failure data, the entire degradation history from the beginning of life (BoL) to end of life (EoL) of similar systems are acquired. The RUL of a new test system can be predicted based on the RUL values of the degradation histories of the systems that resemble closely. Figure 3.3 illustrates the RUL prediction of a test system (highlighted in red) given the entire degradation histories of similar systems (highlighted in light blue).
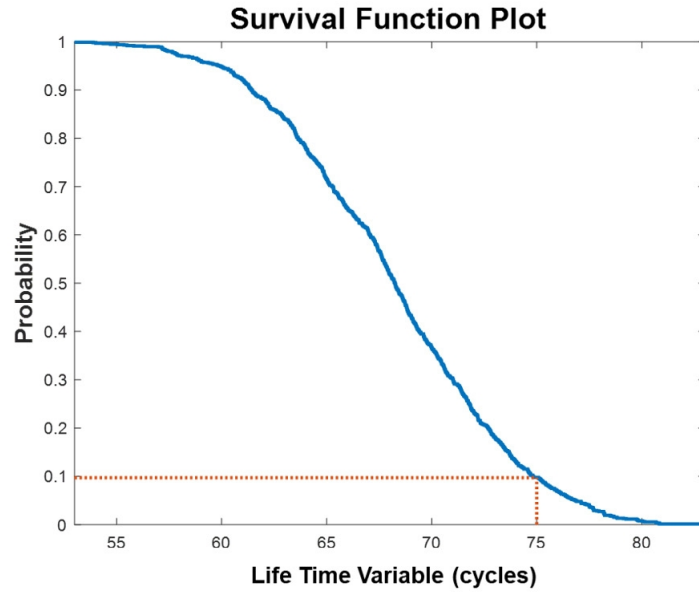
Figure 3.2: Illustration of RUL prediction using lifetime data. Source: (Baru, 2018)

When the run-to-failure data is unavailable and if the failure threshold of the system is known, then the RUL of such systems can be predicted when the system health indicator value crosses the known failure threshold. The RUL of a test system can be predicted by forecasting the health indicator and RUL can be computed as the time taken by the system to reach the failure threshold from its present state. Figure 3.4 illustrates the computation of RUL using threshold data.
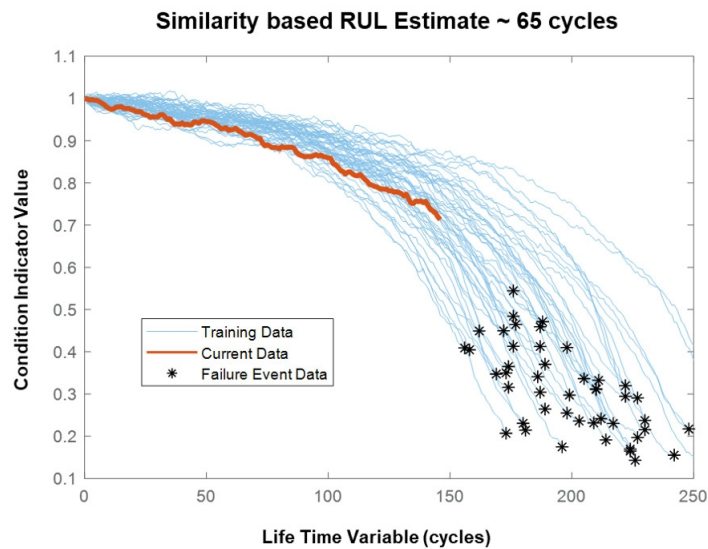
Figure 3.3: Illustration of RUL prediction using run-to-failure data. Source: (Baru, 2018)
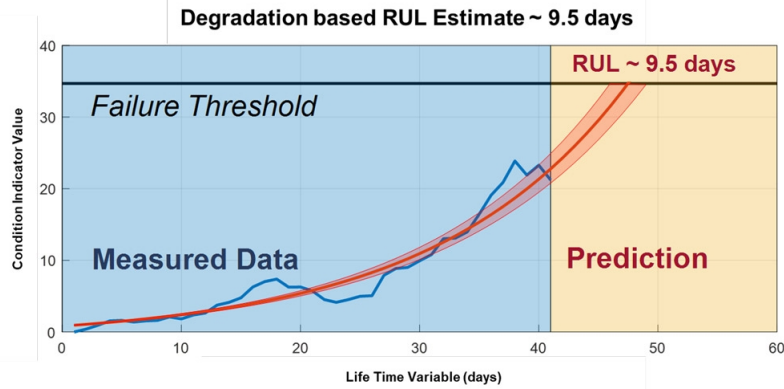
Figure 3.4: Illustration of RUL prediction using threshold data. Source: (Baru, 2018)

3.2 A review of publicly available data sets

There are many lithium-ion battery (LIB) degradation data sets that are publicly available. A list of publicly available lithium-ion battery data sets are aggregated and reviewed in a review paper published by Reis et al. (2021). In this section, we review some of the most commonly used publicly available LIB degradation data sets.

3.2.1 NASA Data set

The lithium-ion battery degradation data set published by NASA Ames Prognostics Center of Excellence (PCoE) is one of the most commonly used data sets to develop machine learning models for State of Charge (SOC), State of Health (SOH) and Remaining Useful Life (RUL) prediction. This data set contains the run-to-failure data of 4 LIBs, which are labeled as *B0005*, *B0006*, *B0007* and *B0018*. These batteries were subjected to charging in a constant current (CC) mode at 1.5A. The charging was performed until the battery voltage reached 4.2 V. Then the batteries were charged in a constant voltage (CV) mode until the charge current dropped to 20 mA.

This is then followed by discharging, where the batteries were discharged at a constant current (CC) level of 2A until the battery voltage fell to 2.7, 2.5 and 2.2 V for batteries *B0005*, *B0006* and *B0007* respectively. The impedance values of the batteries were measured through an Electrochemical Impedance Spectroscopy (EIS) frequency sweep from 0.1 Hz to 5 kHz. voltage, current and temperature values for each cycle for each of the batteries were acquired during the experiment.

Frequent charge and discharge cycles of the batteries resulted in accelerated aging of the batteries. The experiments and measuring battery parameters such as voltage, current and temperature were stopped when the batteries reached their end of life (EOL), .i.e., when the capacity of the batteries reached 70% of their initial capacity value (from 2Ahr to 1.4Ahr) (Saha and Goebel, 2007). Figure 3.5 shows the capacity degradation of 4 LIBs in the NASA data set.
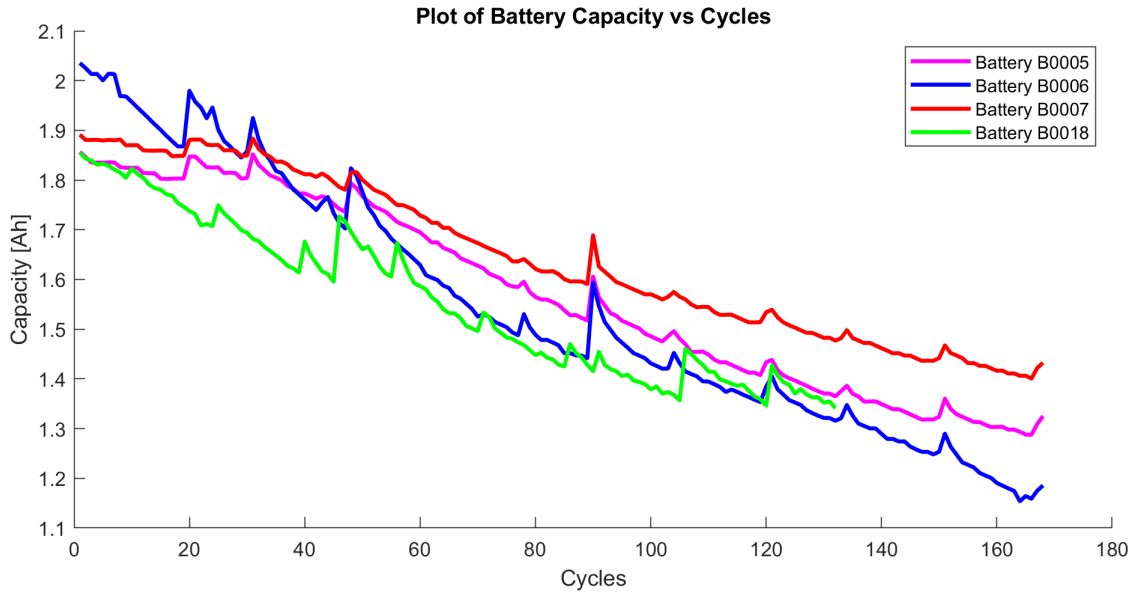
Figure 3.5: Plot showing SOH vs Cycles for the LIBs in NASA data set

3.2.2 CALCE Data set

The CALCE data set was published by the Center for Advanced Life Cycle Engineering (CALCE) at University of Maryland (Pecht, n.d.) and it contains data of several sets of batteries that were subjected to degradation by repeated charging and discharging.

The data for the first set of batteries were acquired by testing 15 Lithium Cobalt (LCO) prismatic cells. The cells are grouped into 6 groups based on the experimental conditions. The cells are at cycled at a room temperature of 23 °C. The cells are subjected to diverse range of partial charging and discharging. The data for this set of cells contains measurements of voltage, current, battery internal resistance, impedance, discharge capacity and charge capacity. The detailed measurement data for each battery cycle is stored in a separate file. The measurements from the batteries are acquired until the batteries end of life, i.e., until they reach 80% SOH.

The second set contains measurements of 12 LCO prismatic cells. The cells have a rated capacity of 1.35Ah. Similar to the first of cells used in this data set, the second set of cells are also grouped into 6 groups. The various parameters measured are the same as that of the first set of cells.

The third set contains measurements of 16 LCO pouch cells. The cells have a rated capacity of 1.5Ah. The cells are cycled at controlled temperature range of 23 °C to 27°C. The data for this set of cells contains the measurements of voltage, current and battery capacity.

3.2.3 Oxford Data set

The Oxford Battery Degradation data set contains measurements of battery aging data from 8 small lithium-ion pouch cells (Birkl, 2017). All the cells used for testing were tested at 40^oC. The required temperature during testing was maintained with the help of a thermal chamber. The cells under test were exposed to a constant current constant voltage (CCCV) charging profile, followed by a drive cycle discharging profile. The discharging profile was obtained from the urban Artemis profile. The measurements were taken every 100 cycles (Birkl, 2017).

Figure 3.6 illustrates the variation of battery temperature w.r.t time for various battery cycles for a battery in the Oxford data set. From this figure, it can be observed that the battery temperature increases when the battery is subjected to repeated charge discharge cycles. This increase in temperature is an indicator of battery aging (Nuhic et al., 2013). Figure 3.7 shows the graph of voltage vs time for different battery cycles for a battery in the Oxford data set.
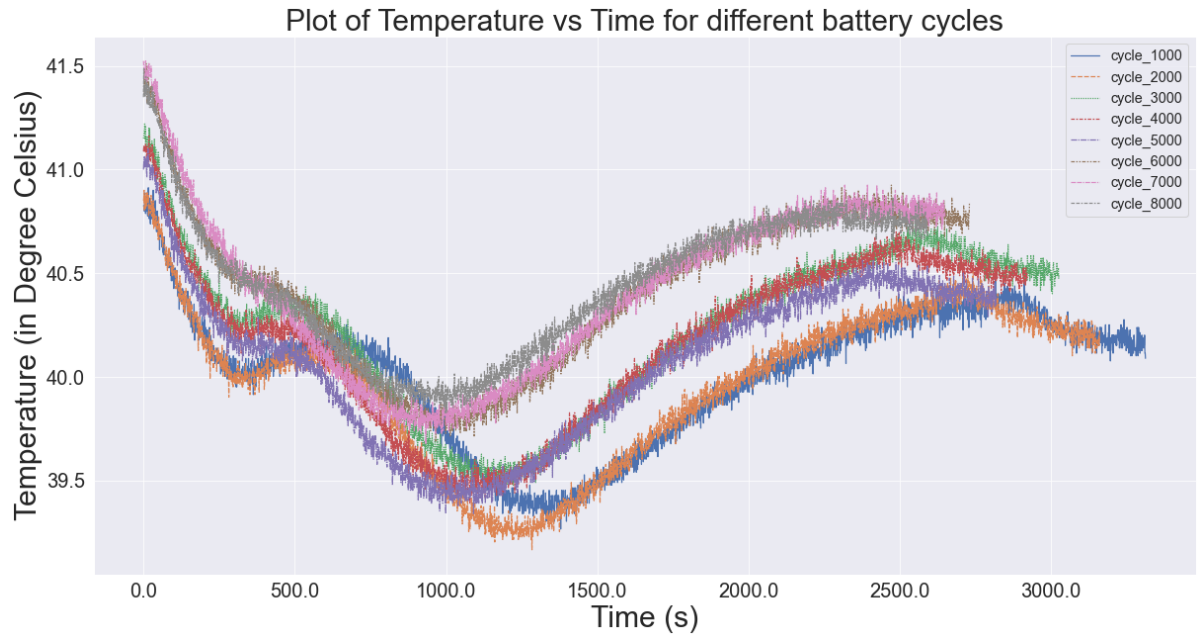
Figure 3.6: Plot of Battery Temperature vs Time of a battery in Oxford data set.
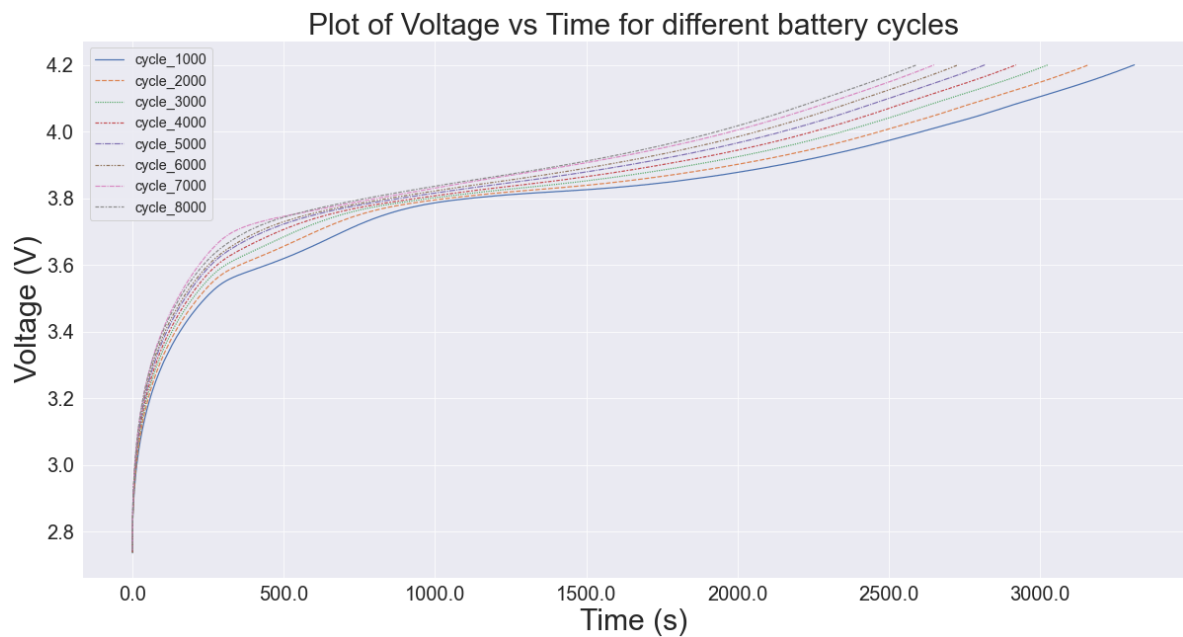
Figure 3.7: Plot of Voltage vs Time of a battery in Oxford data set.

3.2.4 Toyota Research Institute Data set

This lithium-ion battery degradation data set is published by Toyota Research Institute and it contains the run to failure (RTF) degradation data of 124 commercial lithium-ion batteries where each battery contains several lithium-ion phosphate (LFP) / graphite cells. The specifications of the cell used is shown in table B.2. The batteries were cycled at a constant temperature of 30° C and the temperature was controlled by a temperature chamber.

The main objective behind this data acquisition was to optimize the fast charging of the lithium-ion batteries. All the cells used to acquire the data are either charged with a single-step or two-step fast charging policy. The format of the charging policy is "C1(Q1)-C2", where C1 and C2 are the first and second constant current (CC) charging steps and Q1 is the state of charge (SOC) in percentage at which the currents switch. C2, which is the second current step in the fast charging policy, ends at 80% SOC followed by constant current constant voltage (CCCV) charging at the C-rate of 1C. All the cells discharge at a C-rate of 4C (Severson, 2019).

Data generated due to the cycling of the batteries were logged from cycle 2 until the end of life (EOL) of the batteries. A SOH level of 80% is considered to be the EOL of the battery. The acquired data contains in-cycle measurements of current, voltage, charge capacity, discharge capacity and temperature. In addition to the in-cycle measurements, the acquired data contains the per cycle measurements of internal resistance, capacity and charge time. A plot of discharge capacity vs cycles for the data set is shown in Figure 3.8
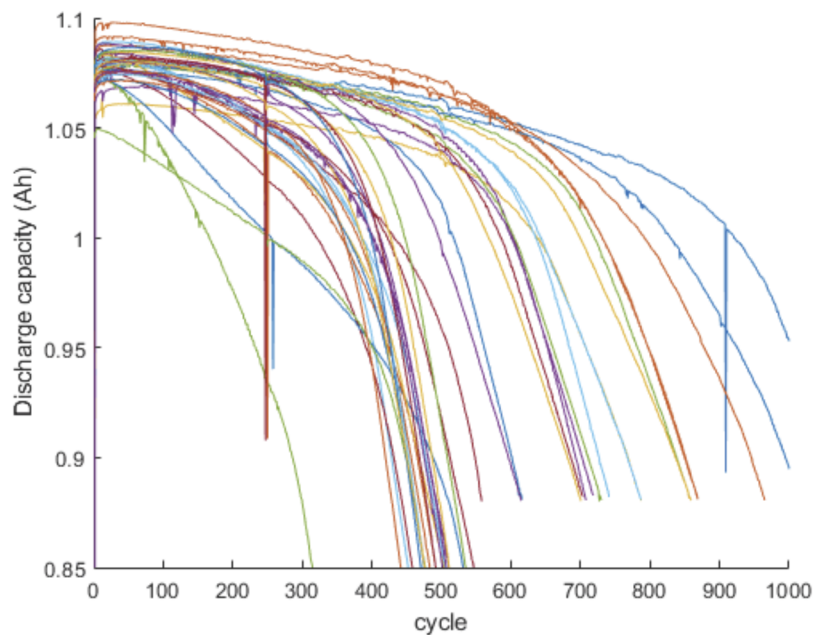
Figure 3.8: Plot of discharge capacity vs cycles of the lithium-ion battery degradation in the Toyota Research Institute (TRI) data set.

3.3 A review on RUL prediction methodologies

The RUL prediction methodology can be broadly classified into four methodologies: *physics-based methodology, experimental-based methodology, data-driven methodology and hybrid methodology.*

(Ahmadzadeh and Lundberg, 2014). This is illustrated in Figure 3.9.
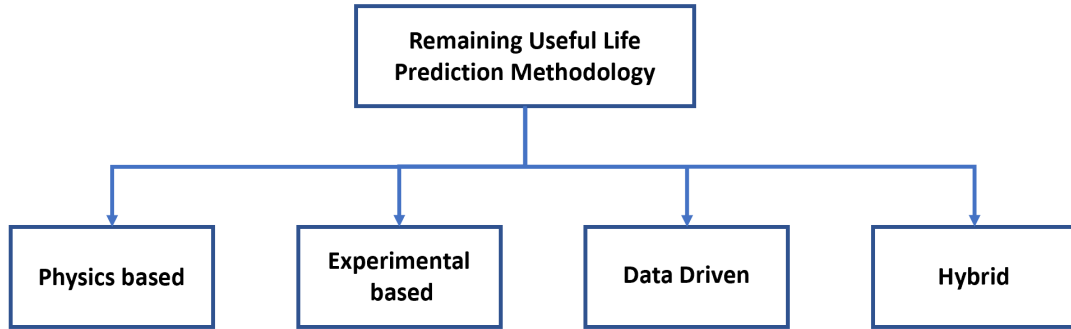
Figure 3.9: An overview of different RUL methodologies

Physics-based methodology for Remaining Useful Life(RUL) prediction of a system can be used, given that there is an accurate theoretical model of the system. With a better understanding of the system degradation, these models can be updated accordingly, which will, in turn, improve the overall accuracy of the RUL prediction. This approach for RUL prediction is typically used to model the crack propagation in gears and wear corrosion.

Experimental-based methodology for RUL prediction uses probabilistic models and/or stochastic models which model the life cycle or the degradation of the system under consideration. One of the key features of this methodology is that it also considers the data and knowledge regarding the system gathered by experience. This approach is used to predict the RUL life of systems over a wide range of domains such as chemical industries, energy industries, manufacturing industries, etc.

Data-driven methodology relies heavily on the data to predict the RUL of the system. This approach does not require product knowledge to predict the RUL. The performance and accuracy of the RUL prediction of a system using this approach depends on the quality and quantity of the available data. This approach leverages data analysis and machine learning techniques to generate predictions for the RUL of the system. Some of the machine learning techniques used in this methodology are Support Vector Machines (SVM), Gaussian Process Regressor (GPR), Artificial Neural Network (ANN), Auto-Regressive Integrated Moving Average (ARIMA) and Decision Tree Regressor. This methodology can be used to predict the RUL of a wide range of systems such as batteries, wind turbines. A detailed overview of several existing RUL prediction models based on the data-driven methodology is presented in Section 3.4

In hybrid methodology, the RUL prediction systems generally make use of more than one RUL prediction methodology for better performance. However, they are computationally expensive. A detailed overview of several existing RUL prediction models based on the hybrid methodology is presented in Section 3.5.

3.4 RUL prediction using data-driven methodology

3.4.1 Non-Probabilistic Models

Non-Probabilistic models are a type of machine learning model and they only return a point estimate when used for regression. Some of the non-probabilistic models commonly used for

Remaining Useful Life (RUL) prediction are ARIMA model, Artificial Neural Network (ANN) models and Support Vector Machine (SVM) models.

3.4.1.1 ARIMA Model

Auto-regressive (AR) based modeling is a non-probabilistic time series model. This model uses a linear combination of observations from previous time steps as input to predict the future time step values. The main advantages of the AR model is its low computational complexity and these models can be expressed using a small number of model parameters.

The AR model is linear while the battery capacity fading process is generally nonlinear. This can lead to under-fitting, especially for the long-term prediction. To solve this problem, an Auto-Regressive Integrated Moving Average (ARIMA) framework was proposed that combines the AR model and the Moving Average method. Instead of using past values of the forecast variable in a regression, the moving average uses the past forecast errors in a regression-like model.

A group of researchers proposed an RUL prediction for LIBs using ARIMA (Y. Zhou and Huang, 2016). The researchers applied Empirical mode decomposition (EMD) on the SOH time series to decouple the global trend and the capacity regeneration values. The forecasts were predicted on each of the decoupled series and the forecasts are then combined to obtain the final result.

3.4.1.2 Artificial Neural Networks

An Artificial Neural Network (ANN) is an ML model in which a group of computational units called neurons are stacked atop one another to form a layer. A neuron is the basic building block of a neural network. It typically accepts one or more inputs and produces one or more outputs. Many such layers are placed one after the other and all the layers are densely connected .i.e., there is a many to many connections between each neuron in a pair of layers.

In a research conducted by (Jorge et al., 2020), an RUL prediction system using Artificial Neural Network (ANN) was developed. The system was tested on the battery aging data made publicly available by the department of chemical engineering of the Massachusetts Institute of Technology (Severson, 2019).

In another research conducted by (D. Zhou et al., 2020), the authors implemented an RUL prediction system using Temporal Convolution Network (TCN) (Lea et al., 2016). They observed that their RUL system performed better than the existing systems.

(Khumprom and Yodo, 2019) and team implemented an RUL prediction system for Lithium io batteries using Deep Neural Networks. (Adib, Angela and Lim, 2020) proposed an RUL prediction system for lithium-ion batteries that uses a Long Short Term Memory (LSTM) (Hochreiter and Schmidhuber, 1997) to forecast the RUL of the batteries. (T. Sun et al., 2019) proposed a similar model that uses Particle Filter (PF) and Multi-Layer Perceptron (MLP), resulting in the PFMLP model. The researchers then compared the performance of PFELM model, the standard Extreme Learning Machine (ELM) model and the model using the Bat Particle Filter with Neural Network (BATPFNN). The researchers concluded that the proposed BATPFNN model outperforms the standard ELM model and the PFMLP model.

3.4.1.3 Support Vector Machines

Support Vector Machine (SVM) is a non-parametric supervised learning technique. When used in a classification problem, it performs classification by searching for the hyperplane separating classes of interest with a maximal margin. When dealing with non-linear features, kernel functions

are often used in SVM. Kernel functions transform the non-linear features in a low dimensional space into a linear features in a higher dimensional space. (X. Li et al., 2017) and team developed an RUL prediction system that uses the SVM model to forecast the RUL.

3.4.1.4 Similarity based models

In the domain of prognostics and health management (PHM), Similarity based models (SBM) are one the commonly used data-driven models to predict the remaining useful life of a system under consideration. SBM is widely used to predict the remaining useful life of a test system given that the run-to-failure (RTF) data of similar systems are available. The data from similar systems are utilized to create a database of degradation profiles. During the remaining useful life prediction of a system under test, the degradation profile of the system under test will be matched against the degradation profiles of the systems available in the database. The actual life of the most similar systems that match with the system under test will be used in the prediction of the RUL of the system under test. In this section, we review some of the available literature regarding the use of similarity based models (SBM) in remaining useful life prediction of lithium-ion batteries.

(S. Kim, N. H. Kim and Choi, 2020) used dynamic time warping (DTW) to augment the existing run-to-failure (RTF) data. The augmented data is then used to train a neural network model to predict the RUL of the system under test. When run to failure (RTF) data of similar systems with different degradation patterns under different failure modes are available, the results show that the proposed model predicts RUL with better certainty than the neural network models that predict RUL without data augmentation.

A group of researchers used similarity based data-driven prognostic methodology for predicting of the fatigue life or remaining life of the structures (Eker, Camci and Jennions, 2014). The method used was originally developed by (Zio and Maio, 2010). The model was tested on three data sets - Virkler fatigue crack growth data set (Virkler, Hillberry and Goel, 1979), a drilling process degradation data set and a sliding chair degradation of a turnout system data set. They evaluated the RMSE and found out that the model produced lesser RMSE for two out of three models considered for research.

(Pérez et al., 2018) proposed a similarity based approach for predicting the RUL of lithium-ion batteries. The research was conducted on the data obtained after cycling Sony US18650 1.4 Ah LIBs using different discharge rates. They also used Monte Carlo simulations to analyse the lifespan of batteries which helps to show the usage of the batteries in a more realistic way. The researchers claim that the suggested SBM technique can be extended to other types of LIBs as the results are normalized and scaling factors are used.

In another conducted by (Soons et al., 2020), the researchers used a similarity based model to predict the RUL of a filter in a chemical industry. The data for the research is gathered by Chemelot Industrial Site. The researchers claim that their model works well even if the operational conditions are unknown.

3.4.2 Probabilistic Models

Probabilistic models represent a set of machine learning models and they return a point estimate as well as confidence interval when used in regression. Some of the probabilistic models that are commonly used for RUL prediction are Gaussian Process Regression (GPR) model, Relevance Vector Machine (RVM) model.

3.4.2.1 Gaussian Process Regression

Gaussian Process Regressor (GPR) is a kernel based machine learning technique, which can realize prognostics combined with prior knowledge based on a Bayesian framework and provide variance around its mean prediction to describe the associated uncertainty. The Gaussian process can be seen as a collection of a limited number of random variables which have a joint multivariate Gaussian distribution.

In a research conducted by (Jia et al., 2020) and his colleagues, developed an RUL prediction system using Gaussian Process Regressor (GPR). The dataset used for this research is sourced by NASA Ames Prognostics Center of Excellence (PCoE) database (Saha and Goebel, 2007). (Liu and Z. Chen, 2019) also developed an RUL prediction system that uses Gaussian Process Regressor (GPR) to predict the RUL of LIBs.

3.4.2.2 Relevance Vector Machines

Relevance Vector Machine (RVM) is identical to Support Vector Machine (SVM). When used in a regression problem, a relevance vector machine returns a point estimate of the prediction as well as a confidence interval. The RVM employs a Bayesian framework to infer the weights with which the probability distribution functions (PDFs) of the outputs instead of point estimates can be obtained.

(Qin et al., 2017) conducted a research where they implemented an RUL prediction system using Relevance Vector Machine (RVM). The auhors used the Battery dataset published by NASA (Saha and Goebel, 2007).

3.5 RUL prediction using hybrid methodology

(Wei, Dong and Zonghai Chen, 2018) proposed an RUL prediction model for lithium-ion batteries (LIB) which uses Support Vector Regressor (SVR) and Particle Filter (PF). (Wu et al., 2019) developed an RUL prediction model for lithium-ion batteries and it uses Neural Network and Bat Based Particle Filter. The research shows that the accuracy obtained using the combined model exceeds the accuracy obtained using either only Neural Network or only Particle Filter(Moral, 1997).

(L. Li et al., 2018) developed an RUL prediction model for lithium-ion batteries and it uses Relevance Vector Machine (RVM) and Particle Filter (PF) and Auto Regression (AR) model. The implemented algorithm was tested on a private data set as well as on a public data set published by NASA (Saha and Goebel, 2007). The researchers conclude that the combined Hybrid model yields effective results.

(Y. Zhang et al., 2019) proposed an RUL prediction model which combines Relevance Vector Machine (RVM) and Particle Filer (PF). This procedure will significantly reduce the size of the training data required to train the model. Finally, the researchers use Monte Carlo method to calibrate the number of particles and to model the noise level of the Particle Filter.

(S. Wang et al., 2021) researched by analyzing, reviewing, classifying and comparing various adaptive mathematical models especially on deep learning algorithms for the task of remaining useful life prediction. Extreme Machine Learning (EML) models are used for the cycle life prediction of lithium-ion batteries. EML models possess a single hidden layer as compared to the feed-forward neural network. Different models such as Extreme Learning Machine (ELM), Deep Convolutional Neural Network (DCNN), Deep Neural Network (DNN), Recurrent Neural Network (RNN), Long Short-Term Memory (LSTM), DCNN - Ensemble Transfer Learning (DCNN-ETL), DCNN with Ensemble Learning (DCNN-EL), DCNN with Transfer Learning (DCNN-TL), Adaptive LSTM

(ALSTM), ALSTM with an attention mechanism, Average Euclidean distance Stacked Denoising Autoencoder (AED-SDA) and Gated Recurrent Unit kernel - Gaussian Process Regression (GRU-GPR) models were compared. Evaluation metrics such as Root Mean Squared Error, Max E, Speed and accuracy for all the above models are compared. They make a careful analysis of different DL modeling methods for accurate RUL prediction. Their experiments show that the DCNN-ELM algorithm gave better results than other models.

(D. Wang et al., 2017) proposed a model with non-linear drifted Brownian motion with many hidden states. The proposed method is more efficient than the models that used particle filtering. Private data of 26 rechargeable batteries were analysed. The researchers performed the comparisons of their proposed model with the standard PF (particle filter) based prognostic method, the spherical cubature PF based prognostic method and the classic Bayesian prognostic method. They finally concluded that the proposed model in their research, which is, nonlinear-drifted Brownian motion with the multiple hidden states has lower average prediction errors than the other models.

(H. Wang, Ma and Zhao, 2019) proposed a model where an improvised version of Wiener process model with adaptive drift and diffusion was developed for RUL prediction. They used an algorithm to eliminate the abnormal monitoring data based on the 3-sigma criterion. They built a complete framework for online RUL prediction.

(Shen et al., 2021) worked on predicting the RUL of lithium-ion batteries considering the random variable discharge current. They analysed the impact of the variable discharge current on Li-ion batteries. They designed a two stage Wiener model to explain the lithium-ion battery degradation. A widely known open source battery data set by Oxford has been used by the team for their research. The proposed method's performance metrics such as RMSE, MAPE, MAE, R-squared and Absolute Error are calculated and compared with the work conducted by (H. Wang, Ma and Zhao, 2019) and (D. Wang et al., 2017). The metrics were calculated for the three models taken for comparison and they were calculated for different failure thresholds of different batteries in the data set. It was concluded that the proposed model performed better than the models proposed by (H. Wang, Ma and Zhao, 2019) and (D. Wang et al., 2017).

(X. Xu et al., 2021) proposed an RUL prediction model considering the time varying temperature condition. The proposed model was developed using the Bayesian framework. The researchers observed that time-varying temperature condition has a major impact on the discharge capacity and aging of a lithium-ion battery. They started by proposing a stochastic degradation rate model based on the Arrhenius temperature model. Then, using Wiener process, an aging model of lithium-ion battery was developed. Then, an unbiased estimation method based on maximum likelihood estimation (MLE) combined with genetic algorithm (GA) is proposed. Under the Bayesian framework, the random parameter is updated. Probability Density Function (PDF) of the RUL for lithium-ion battery under given time-varying temperature condition is derived. Finally, the performance of the model thus built is evaluated and it was concluded that the model built had higher accuracy and lesser uncertainty. The data set used to check the performance of the model is CALCE (Pecht, n.d.).

(Hong et al., 2020) and team used deep learning model to predict the RUL of lithium-ion batteries in a much faster way. They claim that their model is the first end-to-end deep learning framework for RUL prediction. Their framework swiftly predicts the battery RUL with only four cycles. It is 25 times faster than the previous models. They analysed the temporal patterns of terminal voltage, temperature of the cell and current. The framework proposed is interpretable. They used dilated CNN for the prediction of RUL. They have used dilated CNN instead of using RNN and its variants such as LSTM, GRUs because of their limited ability to capture considerably long term relationships. Whereas dilated CNNs have proven examples where they have robustly captured long term relationships in a time series data. The researchers used the lithium-ion battery data set from MIT-Stanford. They built several DL based models such as shallow MLP, MLP, CNN,

CNN + LSTM and dilated CNN. Dilated CNN performs the best among other models and has given an error rate of 10.6

("Remaining useful life prediction of lithium-ion batteries based on Monte Carlo Dropout and gated recurrent unit" 2021) developed an RUL prediction model for lithium-ion batteries using Monte Carlo Dropout and GRU. To test the performance of this model, the battery data set provided by NASA PCoE Research Center is used (Saha and Goebel, 2007). The GRU is the model is used to overcome the gradient vanishing problem which is generally faced in Deep Neural Networks (DNN). The dropout is used in the proposed model to overcome the problem of over-fitting of the data. The performance of the proposed model is compared with two other models - Back Propogation Neural Network – Particle Swarm Optimization (BPNN-PSO) and Least Squares – Support Vector Machine (LS-SVM). RMSE, MAE and MAPE of all three models are calculated. It was found that the proposed model gave better results than BPNN-PSO and LS-SVM models.

(Zewang Chen et al., 2021) proposed a hybrid model which combines a broad learning system with a relevance vector machine (RVM) regressor. Empirical mode decomposition (EMD) is used to extract the features of the data. The training data is then sent into the BLS network, where multiple prediction starting points are chosen, and the associated prediction data is generated. To train the RVM, all prediction data is transformed into a matrix. The RVM is used as the prediction layer of the proposed model. The output generated by the RVM is the predicted RUL of the model. Three different lithium-ion battery data sets are used to check the performance of the model – NASA battery data set (Saha and Goebel, 2007), CALCE (Pecht, n.d.) and independent data set. The results suggest that this hybrid technique can forecast lithium-ion batteries with great precision. It has less errors, higher long-term prediction and generalization capabilities than a single model. RMSE, MAE and error in RUL are evaluated. The proposed hybrid model gave better results.

3.6 Conclusions

In this chapter, we reviewed some of the most commonly used publicly available lithium-ion battery (LIB) degradation data sets. We also reviewed four RUL prediction methodologies - *physics-based, experience-based, data-driven and hybrid methodologies.*

Physics-based methodology for RUL prediction of LIB gives accurate predictions provided there is an accurate model of an LIB capacity degradation. However, in real-time scenarios, the degradation of LIB in an electric vehicle is affected by various factors and various drive cycle patterns and a theoretical model can not does not take into account all the important factors that affect battery degradation. Also, physics-based models rely heavily on the model assumptions. Hence, we do not prefer physics-based methodology to predict RUL of LIB.

Experimental based methodology for RUL prediction relies on data and knowledge of the system gained by experience. However, models that belong to this methodology requires expert and the models verification is performed based on theoretical assumptions. Hence, we do not prefer experimental based methodology to predict the RUL of LIB.

Hybrid models generate better RUL predictions as they leverage the benefits of more than one RUL prediction model. However, they are computationally expensive. Hence, we do not prefer hybrid models.

Data-driven models rely only on the observed system data and it does not make any assumptions about the underlying system. The effects of various factors that affect the RUL of the system

are captured in the data and data-driven machine learning models can be developed accordingly to predict RUL with high accuracy. Based on the literature review, we implement SVR model, LSTM network and similarity based model to predict the remaining useful life.

Research Gap

Out of the many research articles published on RUL prediction of lithium-ion batteries, very rarely the articles address about implementing the RUL prediction algorithm in real-time in an EV. Ideally, an RUL prediction model should start predicting the RUL right after the first charge-discharge cycle of the battery. Hence, we try to address this research gap as part of this research.

Although, many researchers have used similarity based models to predict the RUL of a variety of systems, not many research has been published, where SBM is implemented to predict the RUL of LIBs using dynamic time warping (DTW) distance. In this research work, we also try to implement a SBM with DTW as the distance metric.

Chapter 4

Problem Formulation

The main research goal of this thesis is to predict the remaining useful life (RUL) of a lithium-ion battery (LIB) on board an EV in a real-time scenario using the available battery degradation data. Several factors affect the degradation of a lithium-ion battery.

The first step in predicting the remaining useful life prediction is to identify the factors that affect the aging of the battery. Temperature, voltage, current, battery internal resistance, rest duration, depth of discharge (DOD) of the cycle and ambient temperature of the battery are some of the features that are indicative of battery degradation.

Accurate prediction of remaining useful life (RUL) of lithium-ion battery (LIB) used in an electric vehicle (EV) can alert the driver and can help prevent possible catastrophic battery failure.

The main research question addressed in this thesis is mentioned below.

How to predict the RUL of a lithium-ion battery onboard an EV in a real-time scenario?

Some additional research questions that can be addressed are as follows:

- Will the prediction model gives better results when the impact of resting (calendar aging) on battery degradation is considered?

- Can the prediction model implemented return the remaining useful life of the battery in terms remaining range for which the EV can be driven?

- What are the different model performance criteria to compare different machine learning models?

- Can the prediction model predict the remaining driving range of electric vehicles that are subjected to a wide range of drive cycles?

Chapter 5

Feature Engineering

In this chapter, we describe, explore, pre-process and create new features from the TNO data set. Then, In section 5.1, we briefly discuss the data description of the TNO data set. In section 5.2, we perform an exploratory data analysis (EDA) on the data set. Finally, In section 5.3, we pre-process and create new features from the TNO data set. These new features, along with the existing features will be used to build machine learning (ML) models to predict the remaining useful life (RUL).

5.1 TNO Data set : Data Description

This is a private data set provided by TNO as part of this research. This is a lab-generated data set and it was acquired by subjecting eight lithium-ion batteries, namely *A1*, *A2*, *A3*, *A4*, *A9*, *A10*, *A11*, *A12*, to diverse charging (load) cycles and discharging (drive) cycles. Generally, batteries are said to have reached their end of life (EOL), when their capacity value reaches 80% of the initial capacity (80% SOH). Except for battery *A9*, all the remaining batteries are cycled starting from their beginning of life (BOL) until they reach the end of life (EOL). The degradation history data for each of the batteries in the data set consists of features described in Section 5.1.1.

5.1.1 Data Description

The following are the features that were acquired for each of the cells in the TNO data set.

1. *Time* is the recorded end time expressed as a time entry.

2. *SOC_init_p* is the initial State of Charge (SOC) value based on the SOC vs Open Circuit Voltage (OCV) relationship. It is expressed as a percentage.

3. *SOC_avg_p* is the average State of Charge (SOC) is based on the mean SOC value. It is expressed as a percentage.

4. *SOC_end_p* is the end State of Charge (SOC) value is based on Coulomb Counting (CC). It is expressed as a percentage.

5. *DOD_p* is the Depth of Discharge (DOD) and it is the difference between the initial SOC value and the end SOC value. It is expressed as a percentage.

6. *T_cell_min_C* is the minimum cell temperature of the cycle. It is expressed as degree Celsius.

7. *T_cell_avg_C* is the average cell temperature of the cycle. It is expressed as degree Celsius.

8. *T_cell_max_C* is the maximum cell temperature of the cycle. It is expressed as degree Celsius.

9. *T_amb_avg_C* is the average ambient temperature of the cycle. It is expressed as degree Celsius.

10. *Crate_avg* is the average current value of the cycle based on the initial capacity. It is expressed as per hour.

11. *Crate_RMS* is the root mean square (RMS) current value of the cycle based on the initial capacity. It is expressed as per hour.

12. *Crate_max* is the maximum current value of the cycle based on the initial capacity. It is expressed as per hour.

13. *Ctp_CC_As* is the constant current part of the charge throughput of the cycle based on coulomb counting. It is expressed as ampere second (As).

14. *Ctp_As* is the charge throughput of the cycle based on Coulomb Counting (CC). It is expressed as ampere second (As).

15. *Capacity_As* is the last recorded capacity of the battery. It is expressed as ampere second (As).

16. *FileName* is the name of the file that contains data of the cycle measured at a high sampling frequency.

17. *Resistance_Ohm* is the last recorded resistance at 50% SOC, where the cell temperature is between 22.5-25.5 $\circ C$. It is expressed as Ohm.

18. *TempCellResistance_C* is the recorded cell temperature of *Resistance_Ohm*. It is expressed as $\circ C$ (degree Celsius).

19. *Resistance_mea_Ohm* is the last recorded resistance at 50% SOC. Resistance measurement for LF Test might be extrapolated to get values at 50% SOC. It is expressed as Ohm.

20. *TempCellResistance_mea_C* is the recorded cell temperature for during resistance measurement. It is expressed as $\circ C$ (degree Celsius).

21. *dist_km* is the estimated distance traveled based on the charge throughput and total distance covered. It is expressed as a kilometer (km).

22. *Total_dist_km* is the cumulative distance. It is expressed as a kilometer (km).

23. *Total_Ctp_As* is the cumulative charge throughput. It is expressed as ampere second (As).

24. *duration* is the duration of the cycle. It is expressed as a time entry.

25. *Total_duration* is the cumulative duration. It is expressed as a time entry.

In addition to the degradation data of the 8 lithium-ion batteries (LIB), the TNO data set also contains data acquired at a very high sampling rate during charging and discharging. The acquired data for every charging instance and discharging instance for all the batteries are saved in separate .mat files. A .mat file is a binary file used to save and load MATLAB® workspace variables. There are around 44004 .mat files. Each .mat file contains the following features.

1. *Current_Cell* represents the name of the battery for which the data was acquired.

2. *Time* represents the time at which a particular sample was measured.

3. *Current* represents the value of the current measured. It is expressed as ampere (A).

4. *Voltage* represents the value of the voltage measured. It is expressed as volt (V).

5. *Temperature* represents the value of the temperature measured. It is expressed as degree Celsius (°C).

6. *Temperature_shunt* represents the temperature of the shunt. It is expressed as degree Celsius (°C).

5.2 Exploratory Data Analysis

In this section, we perform exploratory data analysis (EDA) on the TNO data set. As mentioned earlier, the data set contains the entire degradation history of 8 lithium-ion batteries (LIB), namely *A1, A2, A3, A4, A9, A10, A11 and A12*. These batteries are subjected to repeated charging, discharging and calendar aging until they reach the end of life (EOL). In most research regarding LIBs, a battery that has been degraded up to 80% SOH is considered to have reached it EOL. In this research, we consider that a battery has reached its EOL after it has degraded up to 81% SOH. The battery *A9* in the TNO data set is subjected to degradation up to 92% SOH and the battery has not been degraded up to its EOL. As seen from the Figure 5.1, the SOH of the battery *A9* (highlighted in bold red) is degraded only up to 92% of SOH. Hence, we do not consider the battery *A9* in our research.
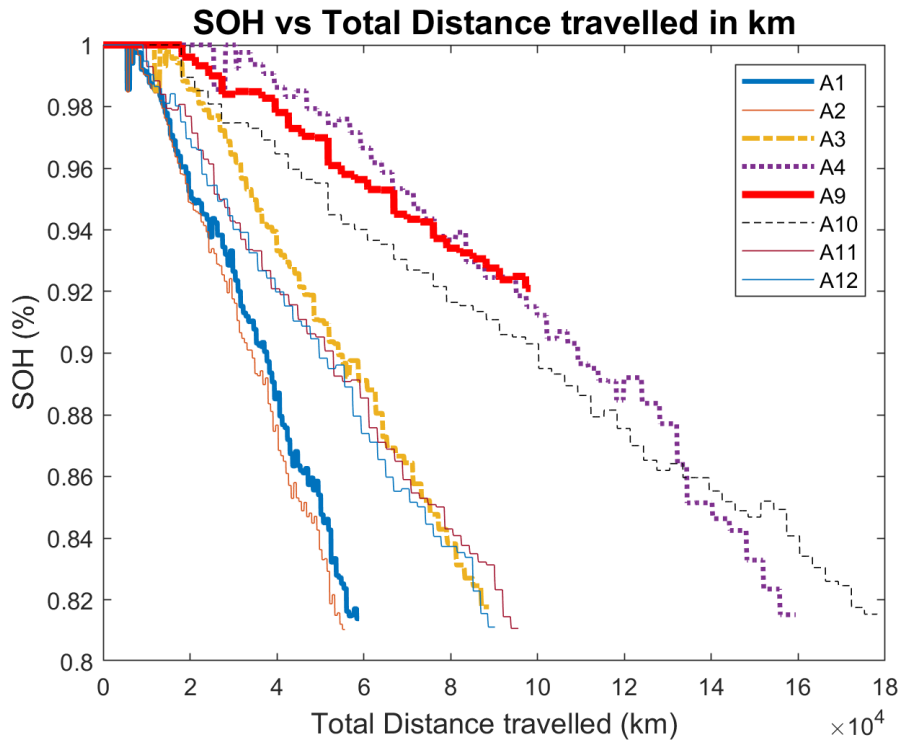
Figure 5.1: SOH (%) vs Total distance traveled (km)

Battery Id	A1	A2	A3	A4	A10	A11	A12
Maximum value of *Total_dist_km* **before the end of life**	58922	55615	88798	159410	178220	95573	90189

Table 5.1: Table showing the maximum *Total_dist_km* values of all the cells in the TNO data set.

5.3 Data Pre-processing and Feature Engineering

5.3.1 Removing unwanted columns

The degradation history of batteries *A10*, *A11* and *A12* contained some duplicate features. These duplicate columns were removed. After the removal of duplicate columns, the degradation histories of all the batteries contain the same number of features.

5.3.2 Handling the missing values

The data set from TNO does not have any missing values. However, there were some instances of `Inf` present in the data set. `Inf` in MATLAB represents Infinity. These values were replaced by the average values computed by a moving average window length of 3.

5.3.3 Normalizing the capacity values

The range of values of the feature *Capacity_As* in the degradation histories of different batteries present in the TNO data set are different. The maximum values of the feature *Capacity_As* for all the batteries in the TNO data set are shown in Table 5.2.

The *Capacity_As* feature of all the batteries in the TNO data set is normalized such that they range from 0 to 1. This is achieved using min-max scaling. Min-max scaling is a data pre-processing technique that scales a given feature such that the scaled feature ranges from 0 to 1. The equation for min-max scaling is shown in Equation 5.1.

$$X' = \frac{X - X_{min}}{X_{max} - X_{min}} \tag{5.1}$$

where *X'* is the scaled feature, *X* is the input feature, X_{max} is the maximum value of the input feature *X* and X_{min} is the minimum value of the input feature *X*. After the normalization the *Capacity_As* values of all cells, they range from [0, 1]. Since during normalization the *Capacity_As* values are divided by the maximum value, the scaled *Capacity_As* values represent the state of health (SOH) of the cells.

Battery Id	A1	A2	A3	A4	A10	A11	A12
Maximum value of *Capacity_As*	7231.33	7210.43	7062.24	7004.10	7106.28	7125.06	7047.30

Table 5.2: Table showing the maximum *Capacity_As* values of all the cells in the TNO data set.

5.3.4 Defining a battery cycle

The RUL of a battery is generally expressed in terms of the remaining distance that the battery can power an EV or in terms of the remaining number of charge-discharge cycles before the battery reaches the end of life. In this research, we define a battery cycle as the set of events recorded in the battery history between two consecutive discharge events. For every cell in the data set, each row in the battery degradation history is marked with its corresponding battery cycle identifier which is incremented after every cycle. Table 5.3 represents the marking of battery degradation history with battery cycle identifier. The different measurements in the history data can be grouped by the battery cycle identifier so that the predicted RUL of the batteries can be expressed in terms of cycles.

FileName	Battery cycle id
REST	1
Charge_1.mat	1
REST	1
Discharge_1.mat	1
REST	2
Charge_2.mat	2
REST	2
Discharge_2.mat	2
REST	3
Charge_3.mat	3
REST	3
Discharge_3.mat	3

Table 5.3: Table representing definition of a battery cycle defined in this research. (Entries in the *FileName* column are changed and only required columns are shown for brevity)

5.3.5 Adding new features

Three new columns are added to the degradation history of each cell. They are *voltage, current and temperature*. The values in these columns are filled by average values of voltage, current and temperature measurements recorded in the `.mat` files that are referred to in the *FileName* column of the degradation history.

Whenever each cell is subjected to rest, it is recorded in the *FileName* column as *REST*. The rest duration of a cell is computed using the *duration* and the *FileName* columns and a new feature *rest_duration* is created.
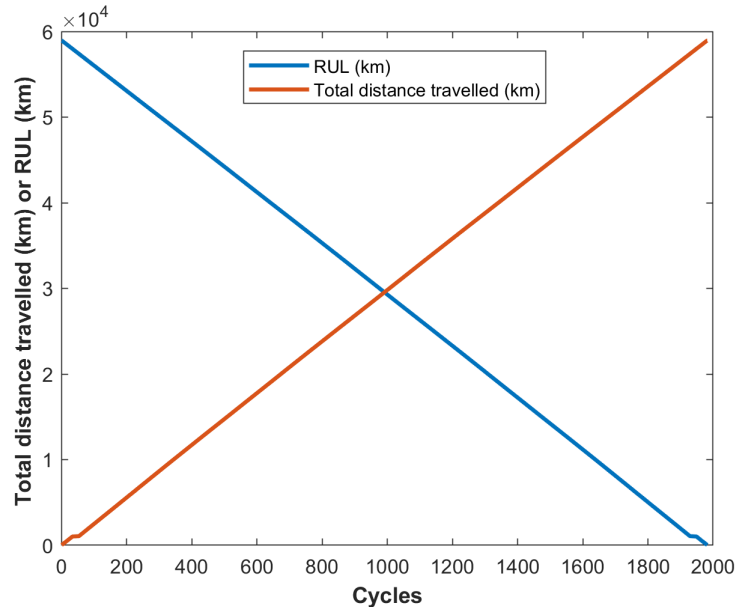
Figure 5.2: Total distance traveled (km) and RUL (km) for cell A1.

The *Total_dist_km* feature in the battery degradation history data set represents the simulated total distance in kilometers traveled by the EV. Sorting the *Total_dist_km* column in the descending

order gives the remaining distance in kilometers that the battery can power the EV before reaching the end of life. This is shown in Figure 5.2. The new column created by sorting the *Total_dist_km* is named as *rul_km*.

5.3.6   Feature selection and Aggregation

The RUL of a battery is influenced by various factors. The battery state of health (SOH) and state of charge (SOC) are some of the important factors that affect the RUL of a LIB (Nuhic et al., 2013). Hence, we also select *SOC* and *SOH* as the input features for our RUL prediction model. These values are generally computed by the battery management system (BMS). Battery internal resistance is one of the important health indicators by which we can infer the SOH of a battery. However, in an EV, battery internal resistance is measured occasionally by subjecting the battery to impulse tests. Thus the battery internal resistance is not computed frequently. Hence, we do not choose this feature to develop our RUL prediction model.

SOC of a LIB is dependent on the voltage, current and temperature of the battery (Omariba, L. Zhang and D. Sun, 2018). Also, we want to consider the effects of calendar aging on the RUL of the battery. Hence, we also select *voltage, current, temperature and resting_duration* as the input features for our RUL prediction model. We aggregate the following features by grouping them by the battery cycle identifier - *SOC, SOH, voltage, current, temperature, rest_duration and rul_km*. The *voltage, current, temperature, SOC* and *SOH* values of battery *A1* are grouped by the battery cycle identifier. The resulting features are shown in Figures 5.3, 5.4 and 5.5.

(a)   (b)
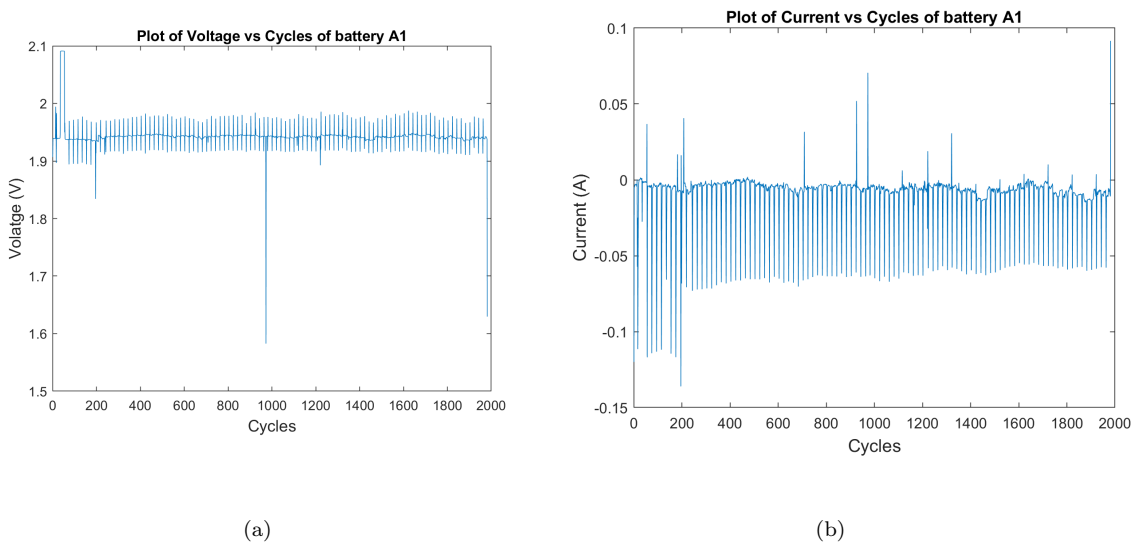
Figure 5.3: a) Plot of Voltage vs Cycles of battery A1 b) Plot of Current vs Cycles of battery A1.
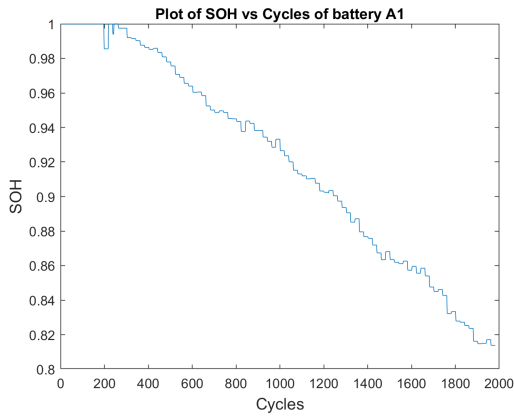
(a)
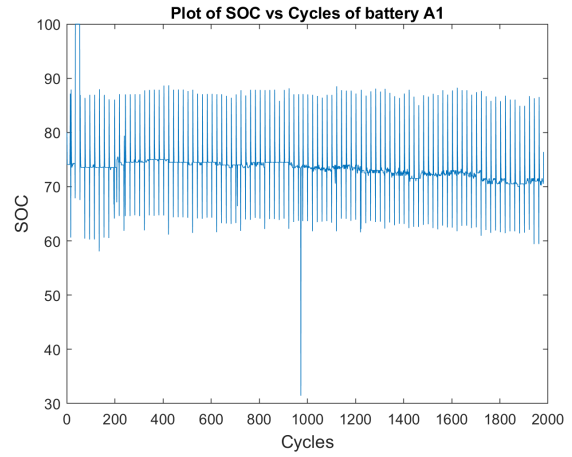
(b)

Figure 5.4: a) Plot of SOH vs Cycles of battery A1 b) Plot of SOC vs Cycles of battery A1.
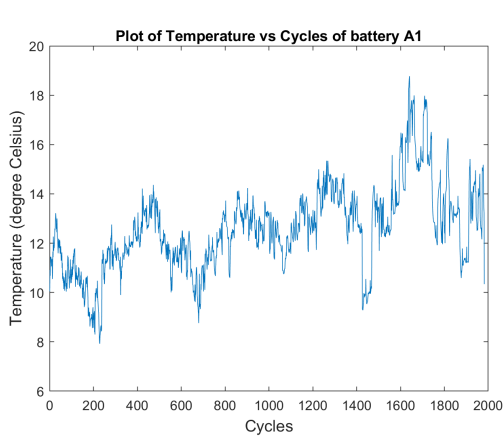
(a)
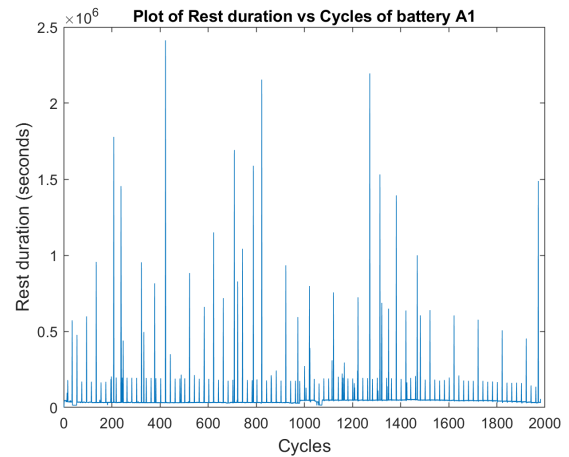
(b)

Figure 5.5: a) Plot of Temperature vs Cycles of battery A1 b) Plot of SOC vs Cycles of battery A1.

5.3.7 Standard Scaling the features

Standard scaling is a data preprocessing technique that scales a given feature such that the scaled feature has zero mean and unit standard deviation. The equation to perform standard scaling is shown in equation 5.2.

$$X' = \frac{X - \mu}{\sigma} \tag{5.2}$$

where X' is the scaled feature, X is the input feature, μ is the mean of X and σ is the standard deviation of X. When preparing the data to implement gradient descent based algorithms such as neural networks and distance based algorithms such as k-NN, SVM, etc., standard scaling of the features is an important step of data preprocessing.

In the case of gradient descent based models, standard scaling the input features ensures that all the input features have the same step size. This will also ensure that the gradient descends and converges smoothly and quickly towards the minima and the gradient descent steps for all the input features are updated at the same rate. The standard scaling of input features also makes model training faster and it avoids the gradient descent from getting stuck in local minima. In case of distance based models, standard scaling ensures that all the features have equal influence on the prediction (Vashisht, 2021).

5.3.8 Summary

After the data is preprocessed using the above preprocessing steps, the data can be fed to RUL prediction models. We want to test the effect of calendar aging on the batteries. Hence, we define two sets of inputs - X_{CA} and X_{no_CA}, where, X_{CA} and X_{no_CA} represent the sets of input features with and without calendar aging effects respectively. The y_RUL is the target feature. The input features present in X_{CA}, X_{no_CA} and y_RUL are as shown below.

$$X_{CA} = \{voltage, current, temperature, SOC, SOH, rest\_duration\} \tag{5.3}$$

$$X_{no\_CA} = \{voltage, current, temperature, SOC, SOH\} \tag{5.4}$$

$$y\_RUL = \{rul\_km\} \tag{5.5}$$

It is important to note that the input features are the aggregated features that are grouped by the battery cycle identifier. We use these sets of input features X_{CA} and X_{no_CA} and the target feature y_RUL in the later sections of this report.

Chapter 6

Machine Learning Models

In this chapter, we describe the details of Support Vector Regression (SVR), Long Short Term Memory (LSTM) Network and Similarity based Model (SBM) to predict the remaining useful life (RUL) of lithium-ion batteries (LIB). In Section 6.1, we describe the working principle and the implementation details of the SVR model. In Section 6.3, we elaborate on the LSTM networks and their implementation details. In Section 6.5, we briefly describe the SBM for RUL prediction.

6.1   Support Vector Machine Regression

Support Vector Machines (SVM) are one of the most often used machine learning models. They are supervised learning algorithms and hence they can be used to solve problems involving regression and classification. They were created by (Cortes and Vapnik, 2004). These models identify some input feature points, also known as support vectors, and try to fit a maximal margin classifier to categorize the linearly separable labels when used in a classification problem. When the labels are not linearly separable, the SVM classifier can be trained along with a kernel function that can classify non-linearly separable labels.

In Support Vector Regression, we construct a generalization bound and aim to optimize it in the same way as SVMs are optimized for classification. The SVR loss function is an ϵ-insensitive function that ignores errors that are within a predefined range of the real values. The variable in 6.1 represents the cost of errors for training observations. The error is zero for the observations within the ϵ band.

Given n observations of input and output data pairs belonging to the set $D(\mathbf{X}, Y)$, we want to predict an output value $\hat{y}_i$ (should be equal to y_i in the ideal case) corresponding to the input data vector $\mathbf{x}_i$ where $y_i \in Y$ and $\mathbf{x}_i$ is the input k dimensional vector representing k features $\mathbf{x_i} = (x_1, x_2, \ldots, x_k)$ such that $\mathbf{x_i} \in \mathbf{X}$.

The mathematical background and equations regarding SVR are obtained from the research work done by (Cortes and Vapnik, 2004), and (Smola and Schölkopf, 2004). The input $\mathbf{x}$ is mapped to an m dimensional feature space using non-linear mappings known as kernels. The SVR algorithm then constructs a linear model in this m dimensional space. This can be expressed as below.

$$f(\mathbf{x}, w) = \sum_{j=1}^{m} w_j g_j(\mathbf{x}) + b \tag{6.1}$$

where $g_j(\mathbf{x})$ for $j = 1, \ldots m$ is the set of non-linear transformations, w_j are the weights for each transformation and b represents the constant term. The loss function that will be minimized

during the training of SVR is an ϵ - insensitive loss function. This loss function is denoted by $L(y, f(\mathbf{x}, w))$ and is defined as:

$$L(y, f(\mathbf{x}, w)) = \begin{cases} 0 & \text{if } |y - f(\mathbf{x}, w)| \leq \epsilon \\ |y - f(\mathbf{x}, w)| - \epsilon & \text{otherwise} \end{cases} \tag{6.2}$$

When the input $\mathbf{x}$ is mapped to an m dimensional space, the SVR algorithm tries to fit a least squares regression model. Thus, the loss minimization function from the least squares regression can be used in this context along with extra variables that represent the slack variables. The loss function of the least squares regression is $\|w\|^2$ and the slack variables are ξ_i, ξ_i^*. The main purpose of using the slack variables is to calculate the deviation of training samples outside the ϵ - insensitive band. The SVR optimization problem becomes:

$$\begin{aligned} \text{minimize} \quad & \frac{1}{2}\|w\|^2 + C\sum_{i=1}^{n}(\xi_i + \xi_i^*) \\ \text{subject to} \quad & y - f(\mathbf{x}, w) \leq \epsilon + \xi_i^* \\ & f(\mathbf{x}, w) - y \leq \epsilon + \xi_i \\ & \xi_i, \xi_i^* \geq 0, i = 1, \ldots, n \end{aligned} \tag{6.3}$$

where the parameter C can be chosen to control the model complexity. It can also control the range of deviations that are allowed, greater than ϵ, in the objective function optimization. Using the theory of constrained optimization, this optimization problem can be transformed into a dual problem and by using the Lagrangian multiplier and the final solution is obtained as below.

$$f(\mathbf{x}) = \sum_{i=1}^{n*}(a_i - a_i^*)\langle \mathbf{x_i}, \mathbf{x}\rangle \quad \text{subject to } a_i, a_i^* \in [0, C] \tag{6.4}$$

where a_i, a_i^* are the Lagrange multipliers, n^* is the set of input samples that are support vectors, and $\langle \mathbf{x_i}, \mathbf{x}\rangle$ is the inner product of two observations $\mathbf{x_i}$ and $\mathbf{x}$. The input samples for which exactly one of the Lagrange multipliers is greater than zero are the support vectors. The inner product in equation 6.4 is mapped through the kernel function which is defined as:

$$K(\mathbf{x_i}, \mathbf{x}) = \sum_{j=1}^{m} g(\mathbf{x})g(\mathbf{x_i}) \tag{6.5}$$

The solution returned by the SVR algorithm is sparse, .i.e., only a few input samples are identified as the support vectors. a_i, a_i^* will be zero for all the samples which are inside the ϵ - insensitive band, as shown in Figure 6.1. The hyperparameters of a SVR model are C, ϵ and the type of kernel. Some of the commonly used kernels are linear kernel, polynomial kernel, hyperbolic tangent kernel and radial basis function (RBF) kernel.
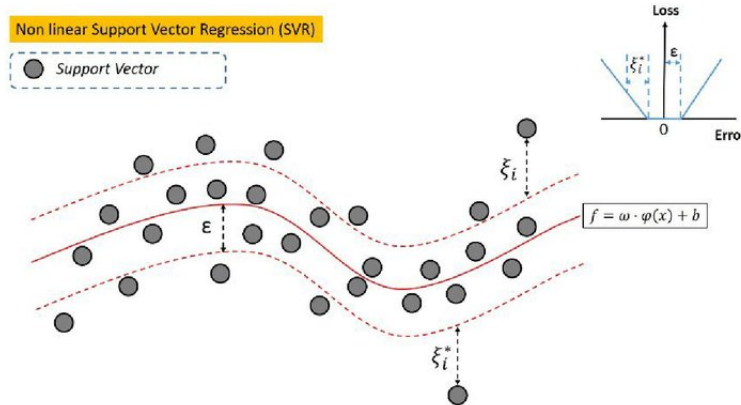
Figure 6.1: ϵ - insensitive band in an SVM. Recreated from (Gelfusa et al., 2015)
.

6.2 Support Vector Rergression - Implementation

Two sets of experiments were conducted for each cell (except cell A9) in the TNO data set. In the first experiment, X_{CA} (refer Section 5.3.8) was fed as the input and in the second experiment, X_{no_CA} was given as the input. In both experiments, separate SVR models were trained for each cell with y_{RUL} as the target variable. The SVR models were trained on 70% of the degradation history data of the cell and were tested on the remaining 30% of the data.

The SVR models were trained with automatic hyperparameter optimization. The optimal hyperparameters were selected with the help of Bayesian optimization (Snoek, Larochelle and Adams, 2012). Bayesian optimization is a commonly used optimization algorithm to find the global maximum or the minimum of a function (Agnihotri and Batra, 2020). The optimal hyperparameters for the SVR models are listed in Tables 6.1 and 6.2.

The predictions made by the LSTM network models are transformed back by multiplying the values with their corresponding standard deviation values and adding the mean value.

Battery Id	Kernel	Polynomial Order	Epsilon
A1	Polynomial	2	0.0026
A2	Linear	-	0.0955
A3	Polynomial	2	0.0024
A4	Linear	-	0.0008
A10	Polynomial	3	0.0009
A11	Polynomial	2	0.001
A12	Polynomial	2	0.04

Table 6.1: Optimal Hyperparameters of SVR models (with calendar aging)

Battery Id	Kernel	Polynomial Order	Epsilon
A1	Polynomial	2	0.0009
A2	Linear	-	0.0505
A3	Polynomial	3	0.0019
A4	Linear	-	0.0078
A10	Gaussian	-	0.0009
A11	Polynomial	3	0.0104
A12	Polynomial	2	0.0405

Table 6.2: Optimal Hyperparameters of SVR models (without calendar aging)

6.3 Long Short Term Memory Network

Long Short Term Memory (LSTM) network is a type of neural network that belongs to the family of neural networks. It consists of an input signal layer, one hidden layer of LSTM cell and an output layer that is fed by the output of the cells present in the hidden layer. An LSTM network typically contains many LSTM cells. The architecture of an LSTM cell is as shown in Figure 6.2.
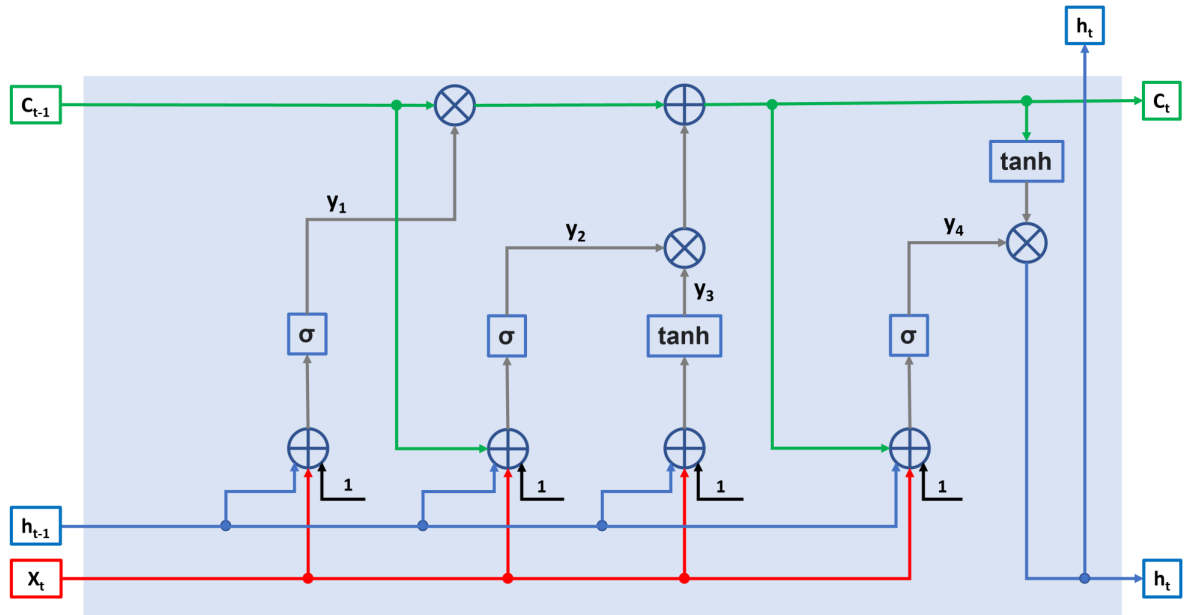
Figure 6.2: The architecture of an LSTM cell.

In Figure 6.2, $\mathbf{x}_t$ is the input signal vector at time point t, C_{t-1} and C_t represent the cell states at time points t-1 and t respectively, h_{t-1} and h_t are the output signals of the cell at time points t-1 and t respectively, 1 is the bias. The symbol $\times$ represents the multiplication of the signals and the symbol + represents the addition of the signals.

An LSTM cell operates at time point t which is the time point at which the output of the LSTM cell is generated and the cell also operates at the previous time point t-1 which represents the previous state of the cell. An LSTM can be mathematically modeled as a function L as shown in equation 6.6

$$(h_t, C_t) = L(h_{t-1}, C_{t-1}, \mathbf{x}_t) \tag{6.6}$$

where :

- $\mathbf{x}_t$ is the input signal vector at time point t. This is one of the inputs fed to the LSTM cell.

- C_{t-1} is the memory value of the cell at time point t-1. It is also the state of the cell at time point t-1. This is another input fed to the LSTM cell.

- h_{t-1} is the output signal of the cell at time point t-1. This is another input fed to the LSTM cell.

- C_t is the memory state of the cell at time point t. This is one of the outputs produced by the LSTM cell.

- h_t is the output signal of the cell at time point t. This is another output produced by the LSTM cell.

The outputs generated by the LSTM cell at time point t are again fed back to the same cell as inputs at time point t+1. The elements of the input vector $\mathbf{x}_t$ are also fed to the cell at any time point.

The multiplication gates in the LSTM cell, denoted by $\times$, play an important role in the control of information propagation and output generation. The first multiplication gate from the left computes the amount of memory from a previous time point to be transferred to the current time point. This amount of memory from the previous time point that will be propagated to the current time point is dependent on the values of signal y1. The second multiplication gate controls the amount of information c_t which represents the state of the LSTM cell at the current time point t. The third multiplication gate computes h_t, which is the product of hyperbolic tangent activated value of the current state of the cell c_t and y_4.

An LSTM cell can be subdivided into computational units based on the generated intermediate signals. As shown in Figure 6.2, an LSTM cell can be divided into four computational units. The intermediate signals y_1, y_2, y_3 and y_4 and the output signals c_t and h_t can be mathematically represented using weights and vectors. The weights can be expressed using double indices where the first index represents the identifier of the intermediate signals y_i, where $i \in [1,2,3,4]$. The second index is the symbol of the input signal. For example, w_{2h} represents the weight of the second gate that takes h_{t-1} as the input signal. With this notation, the intermediate signals y_1, y_2, y_3 and y_4 can be mathematically represented as below :

$$y_1 = \sigma(w_{1h}h_{t-1} + \mathbf{w}_{1x}\mathbf{x}_t + w_{10}) \tag{6.7}$$

$$y_2 = \sigma(w_{2c}C_{t-1} + w_{2h}h_{t-1} + \mathbf{w}_{2x}\mathbf{x}_t + w_{20}) \tag{6.8}$$

$$y_3 = tanh(w_{3c}h_{t-1} + \mathbf{w}_{3x}\mathbf{x}_t + w_{30}) \tag{6.9}$$

$$y_4 = \sigma(w_{4c}C_t + w_{4h}h_{t-1} + \mathbf{w}_{4x}\mathbf{x}_t + w_{40}) \tag{6.10}$$

where *tanh* represents the hyperbolic activation function and σ represents the sigmoid activation function. The output signal h_t and the memory state of the cell C_t are mathematically represented as below :

$$C_t = y_2 y_3 + C_{t-1} y_1 \tag{6.11}$$

$$h_t = (y_4)tanh(C_t) \tag{6.12}$$

Figure 6.3 represents the recurrent operation of the LSTM as a deep feed forward propagation of signals through time. When an LSTM network is trained, the gradient of the loss function is computed using back-propagation through time (BPTT). BPTT is realized within a time window T as illustrated in Figure 6.3 for a given input signal at different time points. The weights and biases of individual LSTM cells are adapted during network training.

An LSTM network is trained in a supervised mode by giving the network a set of features and their corresponding labels as inputs. Gradient descent algorithms such as Adam, AdaGrad or Stochastic Gradient Descent (SGD) are generally used in training. The input vectors are fed to the LSTM network, then they are processed within the hidden layers and output is computed. A loss function is generated based on the error between the computed output and the actual output and the error is then back-propagated through the LSTM network for gradient calculation as part of gradient descent.
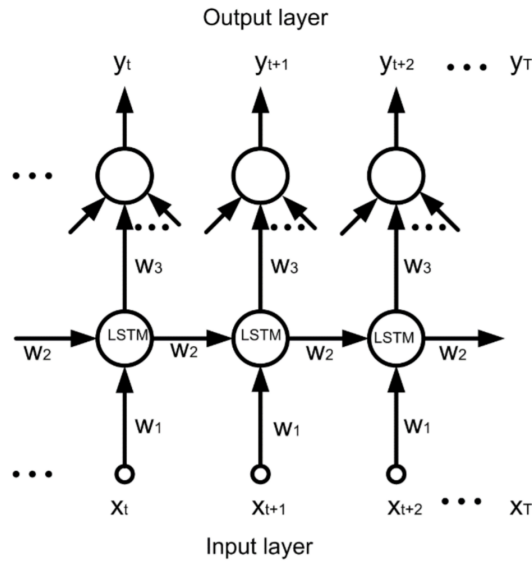
Figure 6.3: Representation of forward propagation through time in an LSTM network

6.4 LSTM network - Implementation

Two sets of experiments were conducted for each cell (except cell A9) in the TNO data set. In the first experiment, X_{CA} (refer Section 5.3.8) was fed as the input and in the second experiment, X_{no_CA} was given as the input. In both experiments, separate LSTM models were trained for each cell with y_{RUL} as the target variable. The SVR models were trained on 70% of the degradation history data of the cell and were tested on the remaining 30% of the data.

The architecture of the implemented LSTM network is shown in Figure 6.4. The same architecture is used for each LSTM model implemented. The LSTM network consists of an input layer. The input layer is connected to the first LSTM layer which consists of 150 LSTM cells. This LSTM layer is connected to a dropout layer. Dropout is regularization technique proposed by (Srivastava et al., 2014) and it is commonly used to prevent deep learning models from overfitting. Dropout probability is set to 0.1. The dropout layer is connected to a LSTM layer which has 300 LSTM cells. This LSTM layer is then connected to a dense layer, followed by a regression layer which generates the prediction.

Two experiments were conducted on the LSTM network. In both experiments, each model is run for 500 epochs. The learning rate of the model is set to 0.005. The learning rate is multiplied by a factor of 0.2 after every 250 epochs. The gradient descent during the training procedure of the models is optimized using the Adam optimizer (Kingma and Ba, 2015).

In the first experiment, which considers the effects of calendar aging, the number of trainable parameters in the different layers of the network are as follows - the first LSTM layer, the second LSTM layer and the dense layer had 94200, 541200 and 300 trainable parameters respectively. The total number of trainable parameters are 635700.
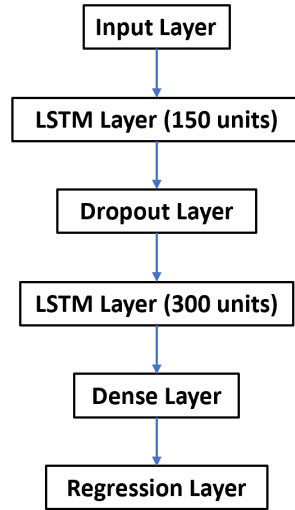
Figure 6.4: The architecture of the implemented LSTM model for RUL prediction.

In the second experiment without considering the effects of calendar aging, the number of trainable parameters in the different layers of the network are as follows - the first LSTM layer, the second LSTM layer and the dense layer had 93600, 541200 and 300 trainable parameters respectively. The total number of trainable parameters are 635100. The results and performance of the two LSTM models are discussed in detail in Section 7.3.

The predictions made by the LSTM network models are transformed back by multiplying the values with their corresponding standard deviation values and adding the mean value.

6.5 Similarity Based Model

There is a category of models that can predict the remaining useful life (RUL) of a test lithium-ion battery based on the degradation profiles of the lithium ion batteries that are similar. Such models are called similarity based models (SBM).

The main intuition behind similarity based model for RUL prediction is that systems that have similar degradation histories have similar RULs. This is evident from Figure 5.1 and Table 5.1. Cells *A1* and *A2* have very similar degradation profiles. Cell *A1* and *A2* reach EOL after powering the vehicle for 58922 km and 55615 km respectively. A similar pattern can be observed between cells *A11* and *A12*.

6.5.1 Working principle of a similarity based model (SBM)

In this section, we describe the basic working principle of a similarity based model for RUL prediction.

Let $TS = \{T_1, T_2, T_3, ...T_N\}$ be the library of time series representing the complete degradation histories of N similar systems namely $S_1, S_2, S_3, ...S_N$. In other words, TS represents the run-to-failure (RTF) data of N similar systems. Each time series in TS can be univariate or multivariate. In the context of this research, every element of the set TS represents the SOH degradation curve of a lithium-ion battery.

Let T_Q be the degradation history of a test system Q. The system Q has not reached its end of life. We want to predict the RUL of T_Q. Let val be the value of the present system health of the system Q.

For every degradation history T_i in TS where $i \in [1, N]$, let $T_i(val)$ represent the portion of T_i from its beginning of life (BOL) to val. Since, TS represents the RTF data of N systems, we know the RUL of these N systems at any value of their recorded system health. Let $y_{S_i}(val)$ represent the RUL of system S_i at the system health value of val, where $i \in [1, N]$.

The RUL of the test system Q can be computed using SBM as follows:

1. We have to define a function "$dist$" to compute the distance between two degradation histories. If two degradation histories are similar, the $dist$ function should return a smaller value and it should return a larger value otherwise.

2. Compute the distance between the test degradation history Q and all the degradation histories in the set TS, .i.e,

$$[d_i] \leftarrow dist(T_i(val), T_Q(val)), \forall i \in [1, N] \tag{6.13}$$

where $[d_i]$ represents the list of distances between the test system Q and N training systems.

3. Sort the distances in ascending order. Let $[d_{i'}]$ be the list of sorted distances where i' represents the sorted index and $i' \in [1,N]$. The

4. Select the k smallest distances. This implies that we are selecting k nearest degradation histories that are similar to the degradation history of Q.

$$[d_{i'}], \forall i' \in [1, k] \tag{6.14}$$

5. The RUL of the test system Q that has system health value of val is computed as follows:

$$y_Q(val) \leftarrow \frac{1}{k} \sum_{i'=1}^{k} y_{S_{i'}}(val) \tag{6.15}$$

Here, the RUL of the test system Q is calculated by taking the average of the RUL values of k systems that are similar to the test system Q. The RUL values of the k similar systems have an equal weight of 1. Therefore, in the rest of this research, we refer to this strategy of computing RUL as *SBM with uniform weights*.

Also, the RUL values of k similar systems that are similar to the test system Q can be weighted such that the RUL values of the systems that are the most similar to the test system are given higher weights and the RUL values of the systems that are the least similar to the test system can be given lower weights. The weighted average of these RUL values gives the RUL of the test system. This is mathematically expressed as follows:

$$y_Q(val) \leftarrow \frac{\sum_{i'=1}^{k} w_{i'} y_{S_{i'}}(val)}{\sum_{i'=1}^{k} w_{i'}(val)} \tag{6.16}$$

where,

$$w_{i'} = \frac{1}{d_{i'}} \tag{6.17}$$

In the rest of this research, we refer to this strategy of computing RUL as *SBM with inverse distance weights*. *SBM with uniform weights* is a special case of *SBM with inverse distance weights*, when all the k nearest degradation histories are at an exactly same distance from the query degradation history.

Earlier, while describing the steps to predict RUL using SBM, we defined a function *dist*, which is used to compute the distance between two degradation histories. If the two degradation histories are of equal length, then pointwise Euclidean distance or absolute distance can be easily calculated. However, the lengths of two degradation histories will not be equal in most cases. Hence, we have to define the *dist* function such that it can compute the distance between two degradation histories that have different lengths. One of the most commonly used distance functions to compute the distance between two time series (degradation histories) is Dynamic Time Warping (DTW). The working principle of DTW is explained in detail in Section 6.5.2

6.5.2 Dynamic Time Warping (DTW)

Dynamic Time Warping is one of the most commonly used algorithms to compute the similarity between two different asynchronous time series. DTW was first proposed by (Sakoe and Chiba, 1978), where the researchers used it for spoken word recognition. DTW compares two time series and measures the similarity between them by computing the optimal warping path between them. DTW has been commonly used in the field of speech recognition, human computer interaction, data mining and signal processing.

Let $X = \{x_1, x_2, ...x_N\}$ and $Y = \{y_1, y_2, ...y_M\}$ be two univariate time series of lengths N and M respectively. In the context of this research, X and Y represent the degradation profiles of two lithium-ion batteries. The main idea behind DTW is to compute the optimal warping path W, between X and Y. Once the warping path is computed, X and Y are extended by warping non-linearly with respect to time. The similarity between extended X and Y are can be computed easily as they have an equal number of points after warping.

Given X and Y, the optimal warping path W is expressed as below:

$$W = \begin{pmatrix} W_x(k) \\ W_y(k) \end{pmatrix}, k = 1, 2, \cdots, p \tag{6.18}$$

where $W_x(k)$ corresponds to an index of X and $W_y(k)$ corresponds to an index of Y. The length of the warping path W is p.

The computed warping path W satisfies the following conditions. Firstly, all the indices of X and Y should be used in the warping path W. Secondly, the warping path W should be continuous and it should be monotonically increasing. Given that the warping path W satisfies these conditions, the starting point of W has the indices $(1,1)^T$ and the ending point of W has the indices $(N,M)^T$. Along with these conditions, any two adjacent points $W(k)$ and $W(k+1)$ in the warping path W should satisfy the following inequality.

$$\begin{cases} W_x(k) \leq W_x(k+1) \leq W_x(k) + 1 \\ W_y(k) \leq W_y(k+1) \leq W_y(k) + 1 \end{cases} \tag{6.19}$$

Hence, W(k+1) can take only the following values : $(W_x(k), W_y(k+1))^T$, $(W_x(k+1), W_y(k))^T$ and $(W_x(k+1), W_y(k+1))^T$. This implies that the warping path W can move either horizontally or vertically or diagonally and it can never go backward. The DTW algorithm chooses the direction which has the lowest cost. Also, the length of W .i.e, p $\in [max(M, N), M + N]$.

By using the optimal warping path W, the two input time series X and Y can be extended to two new time series $\tilde{x}$ and $\tilde{y}$ respectively which are expressed as below :

$$\begin{cases} \tilde{X} = X\left(W_x(k)\right) \\ \tilde{Y} = Y\left(W_y(k)\right) \end{cases} \quad k = 1, 2, \cdots, p \tag{6.20}$$

Now that $\tilde{X}$ and $\tilde{Y}$ are of same length, the pointwise distance between them can be calculated as follows.

$$DTW(X,Y) = D(\tilde{X}, \tilde{Y}) = \sum_{k=1}^{p} D\left(X\left(W_x(k)\right), Y\left(W_y(k)\right)\right) \tag{6.21}$$

In Equation 6.21, D is a distance metric. The following distance metrics are commonly used.

- Euclidean distance, which is computed as the root sum of squared differences between the points.

$$D(\tilde{X}, \tilde{Y}) = \sqrt{\sum_{k=1}^{p}(\tilde{X}(k) - \tilde{Y}(k))(\tilde{X}(k) - \tilde{Y}(k))} \tag{6.22}$$

- Absolute distance, which is computed as the sum of absolute differences between the points. It is also known as the Manhattan distance or city block distance or taxicab distance metric.

$$D(\tilde{X}, \tilde{Y}) = \sum_{k=1}^{p} |\tilde{X}(k) - \tilde{Y}(k)| = \sum_{k=1}^{p} \sqrt{(\tilde{X}(k) - \tilde{Y}(k))(\tilde{X}(k) - \tilde{Y}(k))} \tag{6.23}$$

- Squared distance, which is computed as the square of the Euclidean metric.

$$D(\tilde{X}, \tilde{Y}) = \sum_{k=1}^{p}(\tilde{X}(k) - \tilde{Y}(k))(\tilde{X}(k) - \tilde{Y}(k))) \tag{6.24}$$

The warping path is computed as follows. Firstly, a cost matrix $C(i,j), where\, i \in 1, 2, ...N$ and $j \in 1, 2, ...M$ is created. Each element of $C(i,j)$ is computed using equation 6.25.

$$C(i,j) = D(X(i), Y(j)) + \min \begin{cases} C\,(i-1, j-1) \\ \quad C\,(i-1, j) \\ \quad C\,(i, j-1) \end{cases} \tag{6.25}$$

where $C(1,1) = D(X(1), Y(1))$. The path with the lowest cost that connects $C(1,1)$ and $C(M,N)$ is the optimal path W. The generation of the cost matrix and the computation of the warping path is usually computed using dynamic programming.

An illustration of DTW is shown in Figure 6.5 a) and Figure 6.5 b). Figure 6.5 a) consists of two input time series $X = [1, 5]$ and $Y = [1, 2, 2, 3, 3]$. We can observe that X and Y have different lengths. When the DTW algorithm is applied on X and Y, the time series X is warped (expanded in this case) to a time series of length 5 as shown in Figure 6.5 b). After warping, the length of X and length of Y are equal. Hence, pointwise distance can be calculated on the warped time series. The DTW distance between X and Y is 6, when Euclidean distance metric is used.

(a)   (b)
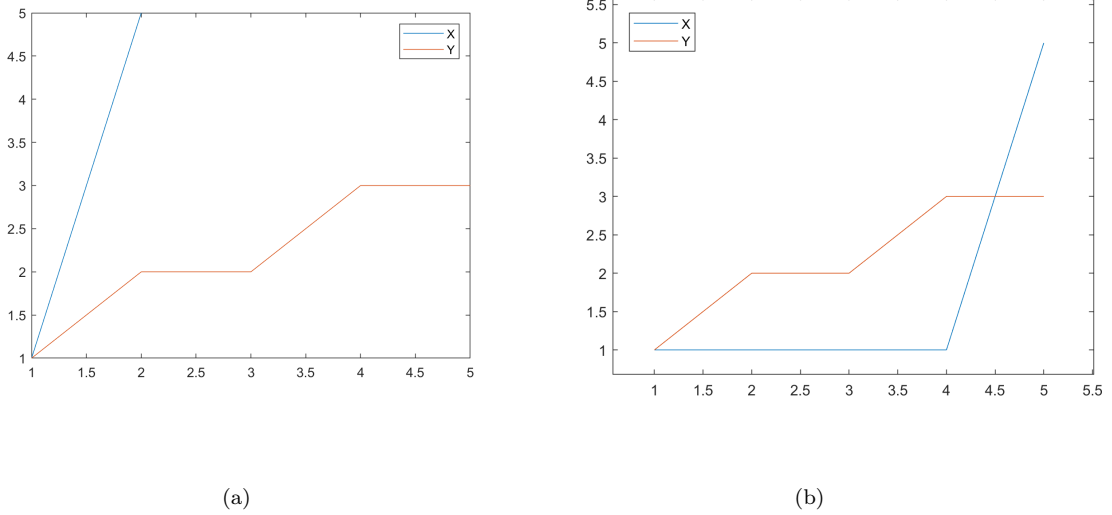
Figure 6.5: Plot of two sample time series X and Y a) before DTW b) after DTW.

6.6   Similarity based model - Implementation

Unlike the SVR and LSTM models, X_{CA} and X_{no_CA} are not used as the input features in the SBM model. We consider the aggregated SOH degradation histories of different batteries as the RTF data. In the TNO data set, we have 7 batteries that have a complete degradation history up to their EOL. We predict the RUL of each battery, by considering RTF data of the remaining batteries in the data set.

Given the degradation history of test battery and the RTF data of similar batteries, we first compute the DTW distance between the degradation history of the test battery from its BOL SOH up to 95% SOH value and all the other similar batteries in the RTF data whose degradation histories are also considered from their BOL SOH up to 95% SOH value.

The DTW distances are then sorted in ascending order. The DTW distance between two degradation histories are smaller if they are similar and the DTW distance is higher otherwise. Based on the sorted DTW distances, we select the target values of the k most similar degradation histories. Out of 8 degradation histories in the TNO data set, the RTF data used for SBM has the degradation histories of only 6 batteries as we excluded the degradation history of battery $A9$ and we select a degradation history for testing. Hence, in our SBM model, we can select k from 1,2,3,4,5,6.

Parameters	Possible values
k	1,2,3,4,5,6
SOH range	100% to 95%, 100% to 90%, 100%-85%
Weights	Uniform Weights, Inverse distance weights

Table 6.3: Parameters and possible values that can be used to obtain the optimal SBM model for the TNO data set

Based on the sorted DTW distances, we can get the RUL values of the k most similar degradation histories that have the same system health value as that of the test degradation history. The

predicted RUL can be computed using a uniform weighted average (equation 6.15) or an inverse distance weighted average (equation 6.16). The same procedure is repeated by considering each battery as the test battery. Table 6.3 shows the various hyperparameters that can be used to obtain the optimal SBM model for the TNO data set. The results and model evaluation metrics of the SBM model is discussed in Section 7.4.

Chapter 7

Results

In the last chapter, we implemented Support Vector Regressor (SVR), Long Short Term Memory (LSTM) and Similarity based Model (SBM) to predict the RUL of the batteries. In this chapter, we present the results obtained from the various machine learning models that were implemented and we discuss the results. In Section 7.1, we define evaluation metrics which we will use to evaluate the performance of the RUL prediction models. In Section 7.2, we discuss the results of the SVR model. In Section 7.3, we discuss the results of the LSTM model. The results of SBM model are discussed in Section 7.4.

7.1 Model performance metrics

In this section, we discuss some of the model evaluation metrics which are used to evaluate the performance of the implemented prediction models. There are numerous model evaluation metrics proposed in the literature, however, each metric has its own merits and demerits to interpret a model. Thus, we choose some of these model evaluation metrics based on their significance and explainability of the results.

7.1.1 Mean Absolute Error (MAE)

Mean Absolute Error is computed as the average of the absolute value of the error. MAE is calculated as follows:

$$MAE = \sum_{i=1}^{N} \frac{|y_i - \hat{y_i}|}{N} \qquad (7.1)$$

where y_i and $\hat{y_i}$ are the actual values of the target and the predicted values of the target respectively, at the ith sample.

7.1.2 Mean Absolute Percentage Error (MAPE)

Mean Absolute Percentage Error (MAPE) is one of the commonly used model evaluation metrics. MAPE is calculated as follows:

$$MAPE = \frac{100}{N} \sum_{i=1}^{N} \frac{|y_i - \hat{y_i}|}{|y_i|} \qquad (7.2)$$

where y_i and $\hat{y}_i$ are the actual values of the target and the predicted values of the target respectively, at the ith sample. MAPE is expressed as a percentage. We can observe that, if the actual value of a target is zero, then MAPE will be infinity.

7.1.3 Root Mean Squared Error (RMSE)

It is one of the most commonly used metric to evaluate the performance of regression and forecasting models. As the name suggests, RMSE is the square root of the square of the mean of the squared errors. The mathematical formula to compute RMSE is given below.

$$RMSE = \sqrt{\frac{\sum_{i=1}^{N} (y_i - \hat{y}_i)^2}{N}} \tag{7.3}$$

where y is the actual value of the target and $\hat{y}$ is the predicted value of the target. N is the length of $\hat{y}$ and also y.

7.1.4 Coefficient of Determination (R²)

This metric indicates the goodness of fit of a regression model. In a regression problem, R^2 metric is used to explain how well the regression line fits the input data. When comparing the performance of two or more regression models, a model with higher value of R^2, fits the input data better than the others. Typically, R^2 of a model is in the range $[0, 1]$. However, in some cases, the R^2 metric of a model is negative, which implies that the model has an opposite trend as the input data.

R^2 is computed as follows. Let y be the actual target of length N. Let $\hat{y}$ be the predicted target. The mean of the actual values is computed as below.

$$\bar{y} = \sum_{i=1}^{N} \frac{y_i}{N} \tag{7.4}$$

The total sum of squares is computed as $SS_{tot} = \sum_{i=1}^{N} (y_i - \bar{y}_i)^2$. The sum of squares of the residual is computed as $SS_{res} = \sum_{i=1}^{N} (y_i - \hat{y}_i)^2$. Using SS_{tot} and SS_{res}, R^2 is computed as below.

$$R^2 = 1 - \frac{SS_{res}}{SS_{tot}} \tag{7.5}$$

One of the main advantages of using R^2 as a model evaluation metric is that it is scale independent. However, it is sensitive to the variance in the input data.

7.1.5 Model Running Time

Model running time is an important model evaluation metric. Generally, most machine learning (ML) models take a lot of time and computational resources to generate predictions. In the context of RUL prediction of lithium-ion batteries in electric vehicles, the ML model to compute the RUL will be deployed in the control unit of the EV's battery management system (BMS). The BMS of an EV is an embedded hardware and has very less computing resources as compared to a personal computer. Hence the deployed ML model must make use of the computational resources of the BMS with high efficiency. Hence, an ideal RUL prediction model should take as less time and consume as less computational resources as possible.

7.2 Support Vector Regression model

In this section, we present the results obtained using the Support Vector Regression (SVR) model. Figures 7.1 a) to d) and Figures 7.2 a) to c) illustrate the RUL prediction made by the SVR model for all the batteries in the TNO data set (except battery A9). Table 7.1 shows the performance of SVR model to predict RUL for each cell when the effects of calendar aging is considered and Table 7.2 shows the performance of SVR model to predict RUL for each cell when the effects of calendar aging is not considered.

For cell *A1*, both the models with calendar aging and without calendar aging have a decent R^2 value of around 90%. However, for the cells *A3, A11 and A12*, the prediction made by the model without calendar aging effect do not follow the trend of the actual target values. Hence, the R^2 value of these models for cell *A3* are negative. The SVR model with calendar aging for cell *A3* also has a negative R^2 value.

For cell *A10*, the SVR model with calendar aging effects almost follows the trend of the actual target values. But the prediction made by the model during the last cycles deviates too much from the expected target. Hence, the R^2 of the SVR model with calendar aging for cell *A10* is lesser than that of the SVR model without calendar aging. For cell *A4*, the predicted RUL has a slightly similar trend to that of the actual target, but the predictions deviate too much from the actual target.

For the batteries *A3, A11* and *A12*, the SVR model without calendar aging effects has a negative R^2 score, which indicates that the model incorrectly fits the data. But, the SVR model with calendar aging effects has a negative R^2 score for only one out of 7 batteries. Thus, the SVR model with calendar aging effects is a better fitting model than the SVR model without calendar aging effects.

Battery Id	MAE (km)	MAPE	RMSE	R^2	MRT(s)
A1	1369.4	99.21	1580.8	0.9074	642.84
A2	3222.1	Inf	3439.5	0.4893	667.97
A3	7744.7	337.44	9785.6	-1.0052	580.56
A4	8630.6	82.11	9968.8	0.4495	534.96
A10	6589.4	518.61	10116	0.5803	1638.71
A11	4209.3	170.67	5255.1	0.4533	468.08
A12	4480.5	225.34	5800.4	0.1152	410.86

Table 7.1: Performance metrics of SVM regression models with calendar aging

Battery Id	MAE (km)	MAPE	RMSE	R^2	MRT(s)
A1	1447.1	123.30	1681.3	0.8953	721.80
A2	2848.8	Inf	2976.3	0.6176	820.67
A3	20942	750.80	26817	-14.05	593.62
A4	9365.3	94.70	10850	0.3478	650.78
A10	6859.6	144.94	8349.3	0.7141	981.35
A11	15285	625.01	19472	-6.5065	391.99
A12	6042.9	290.19	7644.7	-0.537	470.67

Table 7.2: Performance metrics of SVM regression models without calendar aging

(a)

(b)

(c)

(d)
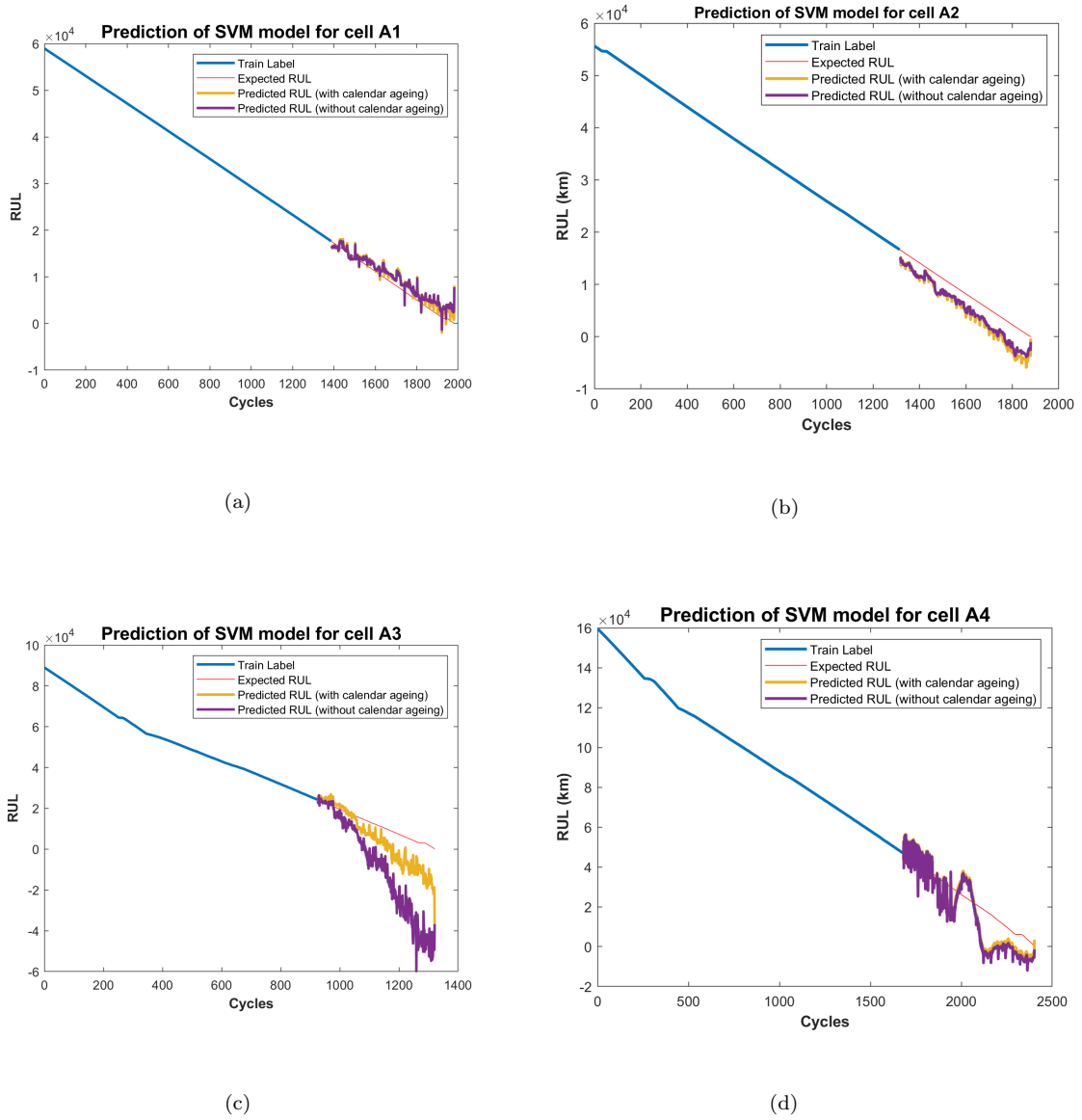
Figure 7.1: RUL prediction of LSTM model for cell a) A1 b) A2 and c) A3 and d) A4

Also, for most batteries (except batteries *A10* and *A11*), the model running time (MRT) of the SVR model with calendar aging effects is lower than the SVR model without calendar aging effects. Hence, we can prefer the former model over the latter.

In Tables 7.1 and 7.2, the MAPE value for battery A2 is infinity. This is because, the actual value of the RUL at the last cycle of the battery is zero and while computing the MAPE, the absolute error term in the numerator is being divided by zero.
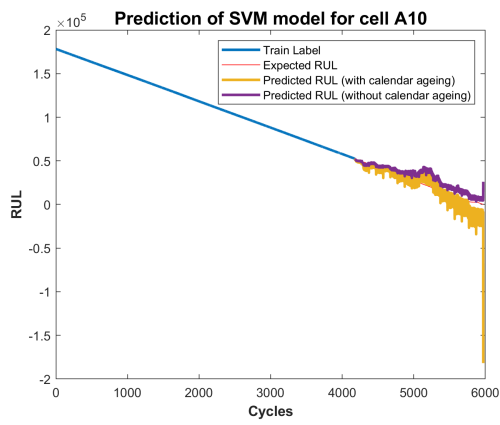
(a)
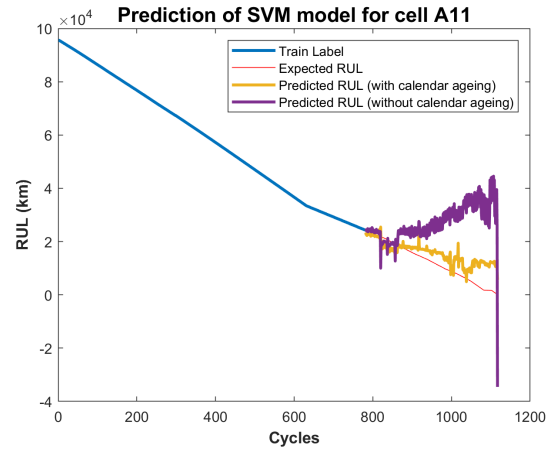
(b)
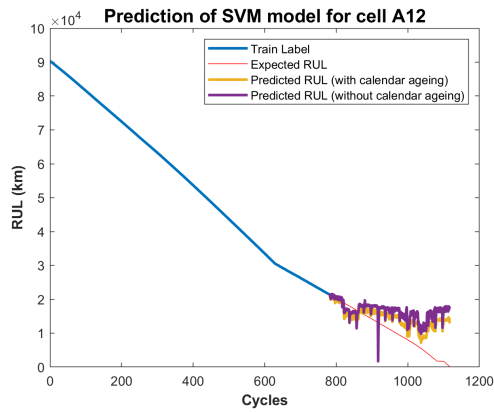
(c)

Figure 7.2: RUL prediction of SVM Regressor for cell a) A10 b) A11 and c) A12

7.3 Long Short Term Memory model

In this section, we present the results obtained using the LSTM model. Figures 7.3 a) to d) and Figures 7.4 a) to c) illustrate the RUL prediction made by the LSTM model for all the batteries in the TNO data set (except battery A9). Table 7.3 shows the performance of LSTM model to predict RUL for each cell when the effects of calendar aging is considered and Table 7.4 shows the performance of LSTM model to predict RUL for each cell when the effects of calendar aging is not considered.

The R^2 of each LSTM model is strictly positive indicating that the model is a good fit for the data. We can observe that when calendar aging is not considered, the models take longer time to run as compared when calendar aging is considered. The R^2 values for the models with calendar aging effects are a bit smaller than that of the models without calendar aging.

For battery *A3*, the MAPE value is 42.30%, which is significantly lesser as compared to the MAPE values of the LSTM models for other batteries. Overall, if we considering all the model evaluation metrics in Table 7.3, we can conclude that the LSTM model with calendar aging effects is a better model than the LSTM model without calendar aging. The RUL predictions made by the LSTM model are relatively good as compared to the predictions made by the SVR model. Hence, among SVR model and LSTM model, LSTM is a better performing model.

(a)

(b)

(c)

(d)
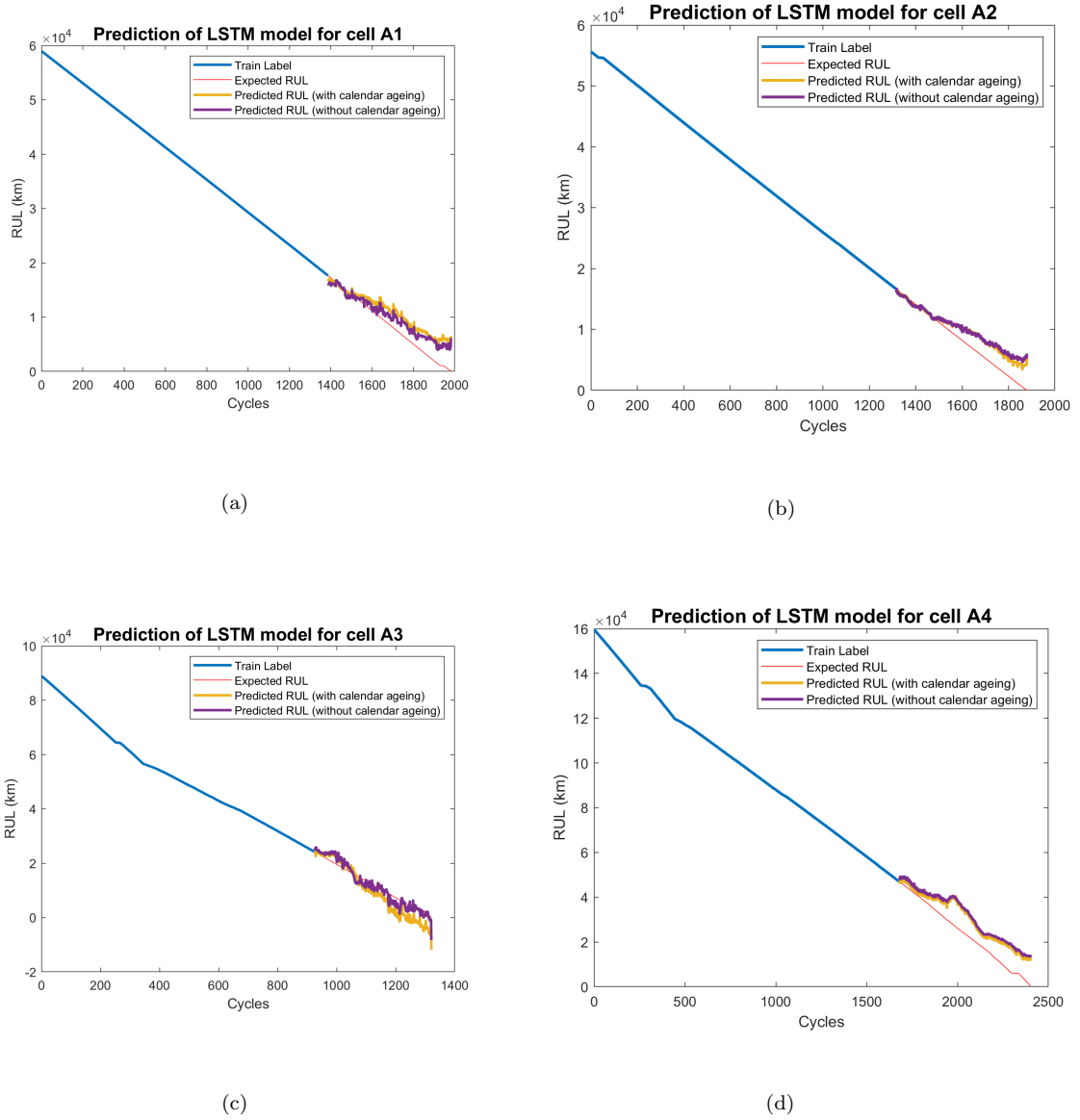
Figure 7.3: RUL prediction of LSTM model for cell a) A1 b) A2 and c) A3 and d) A4.

(a)

(b)

(c)
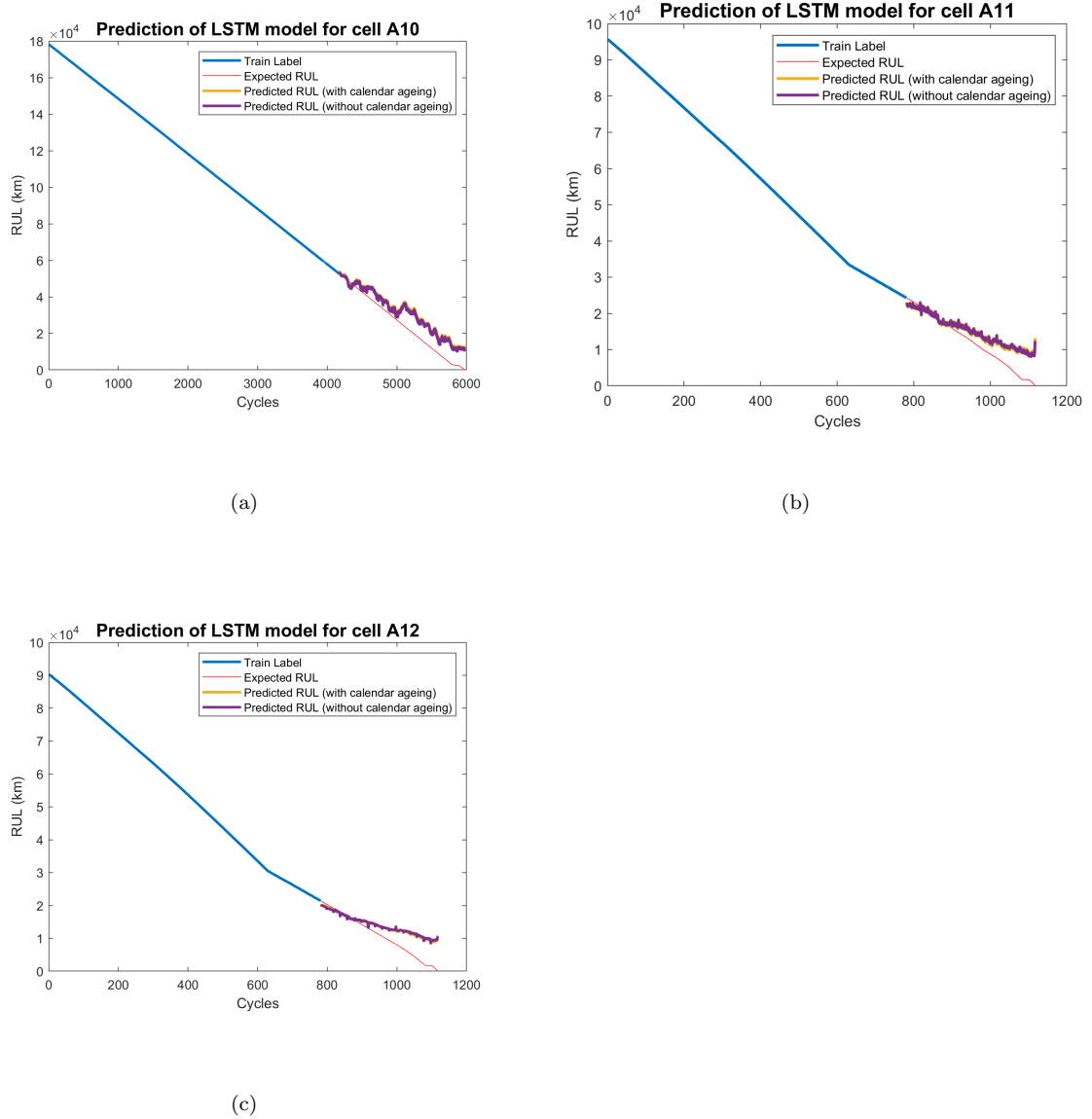
Figure 7.4: RUL prediction of LSTM model for cell a) A10 b) A11 and c) A12

Battery Id	MAE (km)	MAPE	RMSE	R^2	MRT(s)
A1	2512.9	194.71	2967.6	0.6737	113.10
A2	1746.4	Inf	2144.3	0.8015	116.26
A3	3152.7	112.13	3731.8	0.7084	84.91
A4	7065.6	112.90	7830.9	0.6603	154.95
A10	7471.3	174.85	8438.4	0.7080	376.51
A11	3260.1	159.65	4157.0	0.6579	77.28
A12	3357.1	163.59	4382.2	0.4950	75.79

Table 7.3: Performance metrics of LSTM models with calendar aging

Battery Id	MAE (km)	MAPE	RMSE	R^2	MRT(s)
A1	1817.5	155.70	2198.1	0.8210	125.76
A2	1954.3	Inf	2443.9	0.7422	141.59
A3	1571.3	42.30	1923.4	0.9225	162.76
A4	8246.2	126.09	8874.6	0.5637	319.82
A10	6678.1	161.21	7649.4	0.7600	836.42
A11	3254.4	155.09	4129.2	0.6625	181.22
A12	3468.9	167.16	4530.3	0.4602	196.23

Table 7.4: Performance metrics of LSTM models without calendar aging

7.4   Similarity based model

In this section, we present the results of the similarity based model (SBM). TNO data set is used to evaluate the performance of the SBM for RUL prediction. Except for battery *A9*, all the other batteries are used for training and testing.

In Table 7.7, given $k = 1$, the weighting scheme is inverse distance weights and the available range of SOH values is between 100% to 95%, i.e., if very less samples are available in the degradation history of the test data, the MAPE is higher and it reduces when the battery is nearing its EOL. The MAPE is 21.9% at 95% SOH and it reduces to 15.7% at 85% SOH. This can be observed for other values of k. The MAPE is the lowest at 85% SOH.

We can see that when $k = 1$, the values of MAPE in both tables 7.7 and 7.8, are equal. This is because, when $k = 1$, RUL predicted by using the inverse distance weight scheme and uniform weight scheme is equal. This can be verified from equations 6.15 and 6.16. From Tables 7.5, 7.6, 7.7 and 7.8, we can observe that for $k = 1$ and at 85% value of SOH of the test degradation battery, SBM model has the lowest values of MAE and MAPE. Thus, we can conclude that the optimal SBM model to predict the RUL in the TNO data set is a SBM with $k = 1$, and at 85% value of SOH value of the test degradation history.

From the above tables, we can also observe that the MAPE values are lower when the inverse distance weighting scheme is used as compared to uniform weights. This is because, when an inverse distance weighting scheme is used, the less similar degradation histories have lesser influence on the predicted RUL as compared to a uniform weighting scheme, where all the k degradation histories have equal influence on the predicted RUL.

Figures 7.5, 7.6 and 7.7 show the boxplots representing the spread of the MAPE for each battery in the data set, when inverse distance weighting scheme is used. Figures 7.8, 7.9 and 7.10 show the boxplots representing the spread of the MAPE for each battery in the data set, when uniform weighting scheme is used. From these figures, it can be observed that, in general, the spread of the MAPE of the SBM with inverse weighting scheme is less than the spread of the MAPE of the SBM with uniform weighting scheme.

From Figures 7.5, 7.6 and 7.7, we can observe that for the batteries *A11* and *A12*, the median value of the MAPE is very less (less than 10%) and the spread of the MAPE is very low. (The horizontal red line present inside the box of a boxplot, indicates the median value). This implies that different values of k of a SBM with inverse distance weighting scheme, predicts similar RUL for batteries *A11* and *A12*. This is also evident from Figure 5.1, as *A11* and *A12* have a very similar degradation pattern.

From Figures 7.5 to 7.7 , we can observe that for the batteries *A4* and *A10*, the median value of MAPE is around 40%. The main reason for this large MAPE is that there are no degradation histories in the data set that are very similar to the degradation histories of *A4* and *A10*.

From Figures 7.5 to 7.7 , we can observe that for the batteries *A1*, *A2* and *A3*, the median value of their MAPE is less than 15%. This is a reasonably good value of MAPE and since the degradation histories of batteries *A1*, *A2* and *A3* have degradation histories that are similar to them (Refer Figure 5.1).

One of the main takeaway from the boxplots shown in Figures 7.5 to 7.10 is that the *SBM with inverse distance weights* is better than *SBM with uniform weights*, as in the former, the extreme degradation histories has less influence on the predicted RUL and in the latter, all the *k* similar degradation histories have an equal influence on the predicted RUL.

SBM is an example of instance based learning algorithm (IBL), where the model generates predictions by comparing the test samples with train samples that are already known. Thus, when more samples are available, then the SBM model can generate better predictions. The MAPE of the predicted RUL can be reduced if more degradation histories are available in the RTF data.

The data preparation part for the SBM takes 4.25 seconds. The time taken to compute the DTW distance between two degradation histories is 9.3ms. The final RUL generation step takes around 2.5ms seconds. The SBM runs faster than the SVR model and the LSTM model.

The best model out of the SVR, LSTM and SBM model is the *SBM model with inverse distance weights* with *k*=1 and the MAPE of the model is 15.70% The optimal parameters *k* might change when the RUL is predicted with additional RTF data.

Range of SOH values considered	k=1	k=2	k=3	k=4	k=5	k=6
100% - 85%	2906.2	3560.6	3630.4	3520.6	3449.8	3387.3
100% - 90%	10643	11282	11301	11518	11303	10931
100% - 95%	17904	18367	19090	17218	17168	16930

Table 7.5: Mean Absolute Error (MAE) values of the similarity based model (SBM) with inverse distance weights for k=1 to k=6

Range of SOH values considered	k=1	k=2	k=3	k=4	k=5	k=6
100% - 85%	2906.2	4409.7	5270.0	5635.1	5085.6	4942.4
100% - 90%	10643	13592	16383	17520	15954	15720
100% - 95%	17904	21220	25919	24332	23838	26768

Table 7.6: Mean Absolute Error (MAE) values of the SBM with uniform weights for k=1 to k=6

Range of SOH values considered	k=1	k=2	k=3	k=4	k=5	k=6
100% - 85%	15.70643	18.98143	19.36586	18.99714	19.13771	18.88543
100% - 90%	20.40329	20.62886	20.387	21.50329	21.51429	20.91229
100% - 95%	21.98886	20.94329	21.566	19.23043	19.27286	19.04943

Table 7.7: Mean Absolute Percentage Error (MAPE) values of the SBM with inverse distance weights for k=1 to k=6

Range of SOH values considered	k=1	k=2	k=3	k=4	k=5	k=6
100% - 85%	15.70643	26.90471	34.587	38.36429	36.49271	38.37414
100% - 90%	20.40329	28.79286	38.55157	43.136	41.01271	46.01614
100% - 95%	21.98886	25.99314	34.49671	34.49929	36.04871	45.04986

Table 7.8: Mean Absolute Percentage Error (MAPE) values of the SBM with uniform weights for k=1 to k=6

Figure 7.5: Boxplot showing the MAPE grouped by batteries for k=1 to 6. Target is computed using inverse distance weights (100% - 85% of capacity values used for training)

Remaining Useful Life prediction of lithium-ion batteries using machine learning   57
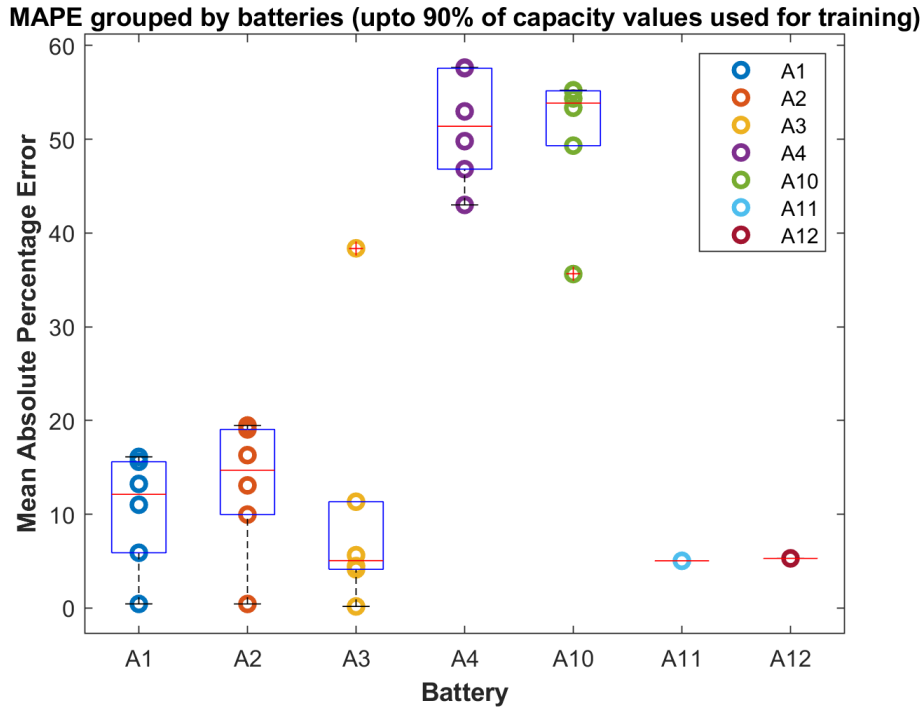
Figure 7.6: Boxplot showing the MAPE grouped by batteries for k=1 to 6. Target is computed using inverse distance weights (100% - 90% of capacity values used for training)
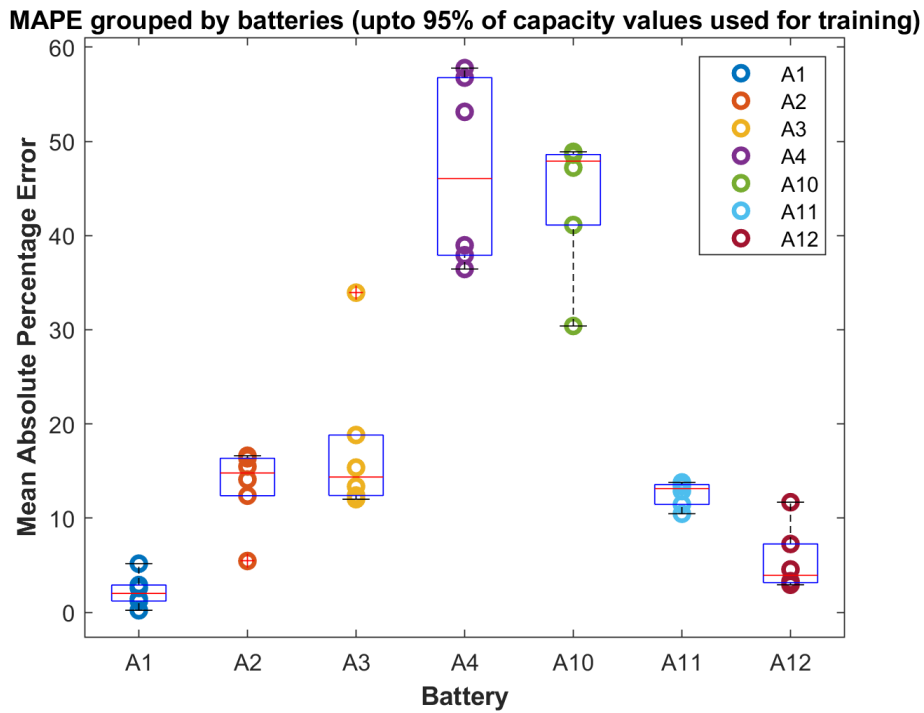
Figure 7.7: Boxplot showing the MAPE grouped by batteries for k=1 to 6. Target is computed using inverse distance weights (100% - 95% of capacity values used for training)

Figure 7.8: Boxplot showing the MAPE grouped by batteries for k=1 to 6. Target is computed using uniform weights (100% - 85% of capacity values used for training)

Figure 7.9: Boxplot showing the MAPE grouped by batteries for k=1 to 6. Target is computed using uniform weights (100% - 90% of capacity values used for training)
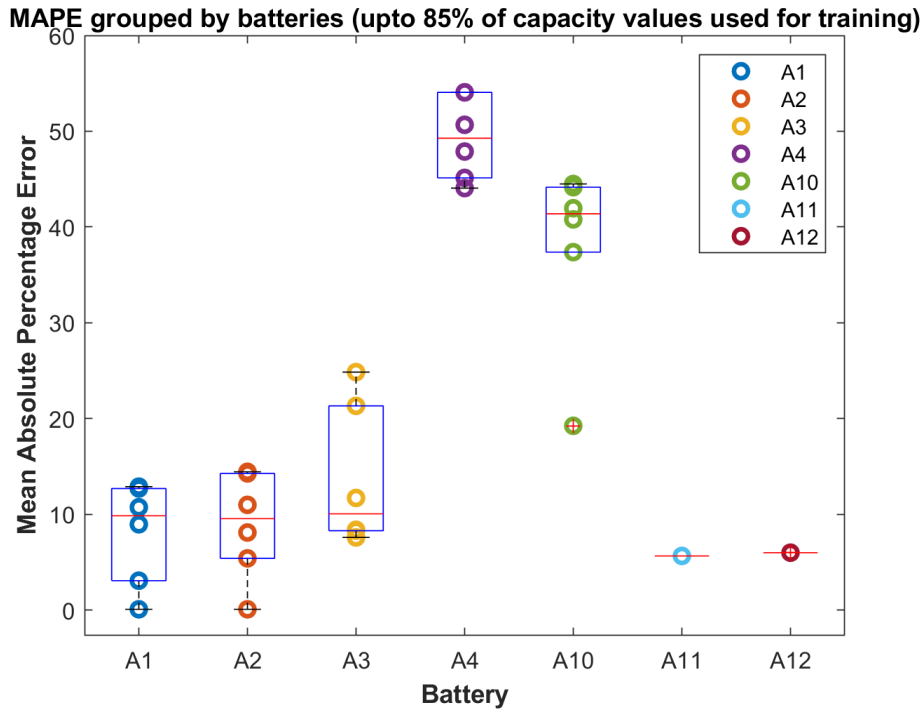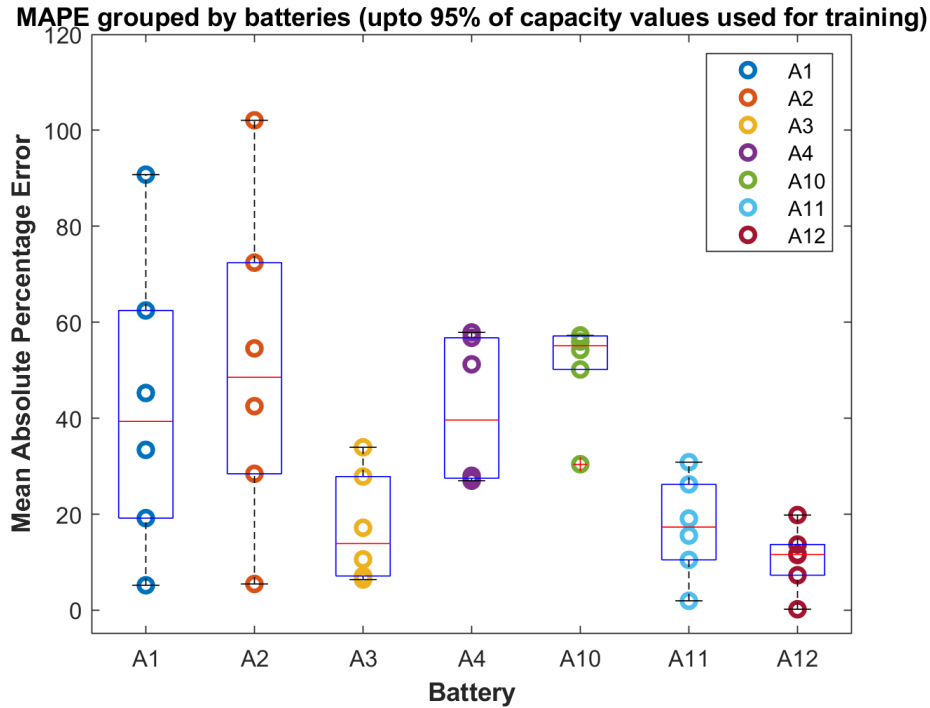
Figure 7.10: Boxplot showing the MAPE grouped by batteries for k=1 to 6. Target is computed using uniform weights (100% - 95% of capacity values used for training)

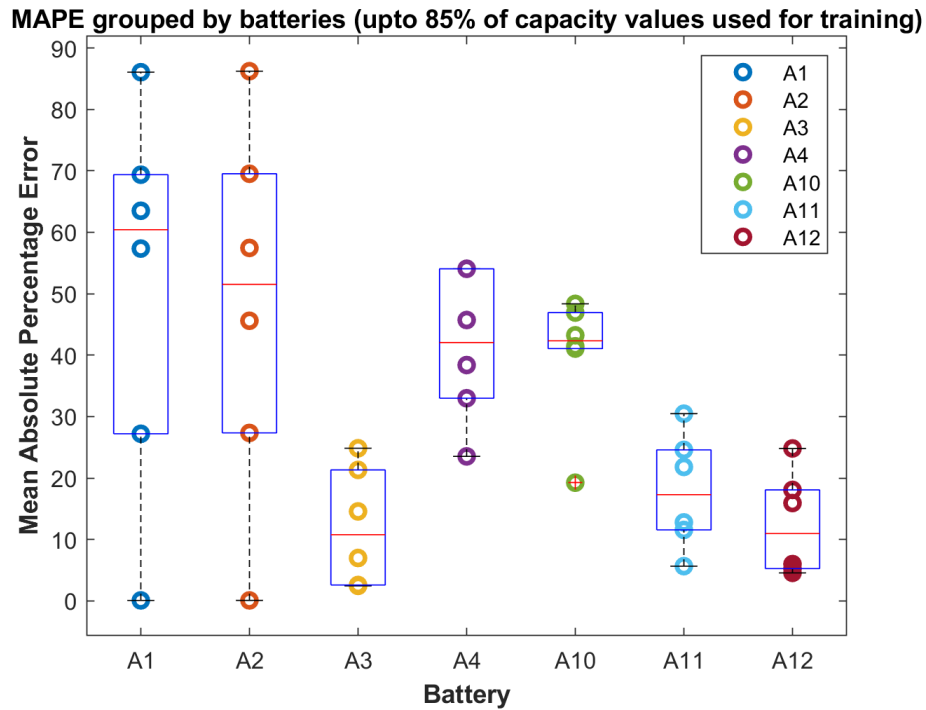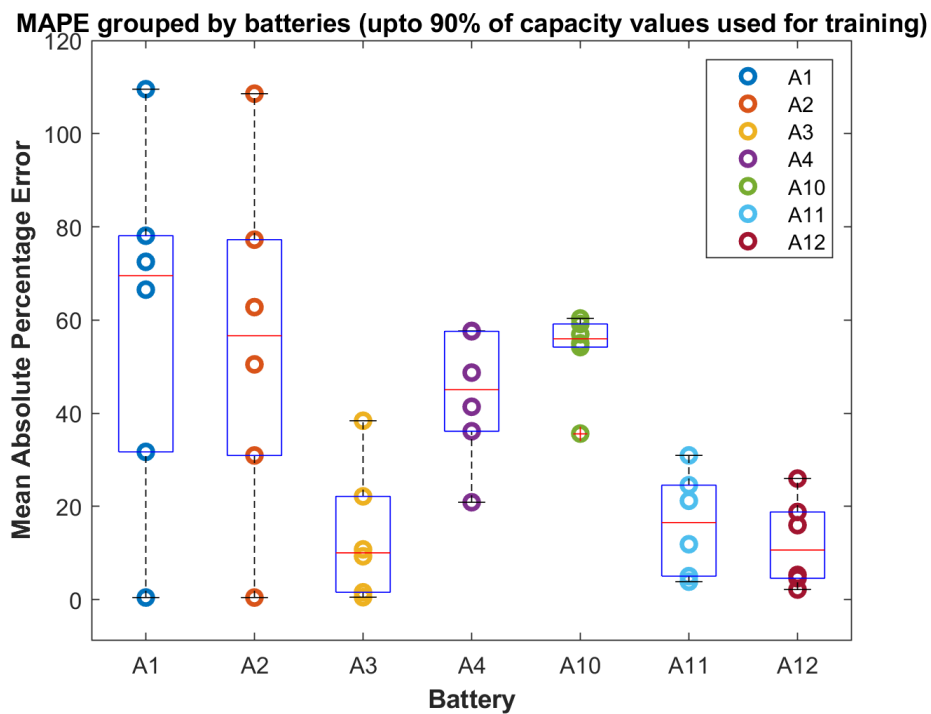## 7.5 Review of research questions

In the problem formulation section of Chapter 1, we posed our main research question and other possible research questions that can be addressed as part of this research. In this section, we present the research findings for each of these questions.

- *How to predict the RUL of a lithium-ion battery onboard an EV in a real-time scenario?*

  We implemented SVR models and LSTM models to predict the RUL of a LIB, given the degradation history of the battery. We split the degradation history of each battery in the data set into two subsets - the training subset and the testing subset. We trained a separate SVR model and LSTM model for each training subset and we also generated predictions from the models using the corresponding testing subsets.

  We also implemented SBMs to predict the RUL of a LIB. SBMs are instance based learning (IBL) models and they predict the RUL of a test battery by considering the degradation histories of similar batteries. This is unlike SVRs and LSTMs where we train using the training subset of degradation history data of a single battery.

- *Will the prediction model gives better results when the impact of resting (calendar aging) on battery degradation is considered?*

  We trained SVR models and LSTM models for RUL prediction. We trained the models without considering the effect of resting (calendar aging) and also with the effects of resting (calendar aging). We found that for some batteries, including the resting period of the battery(which is an indicative of calendar aging) as one of the input features, reduced the error. However, for some batteries, including the resting period increased the error.

- *Can the prediction model implemented return the remaining useful life of the battery in terms remaining range for which the EV can be driven?*

  Yes, the implemented models predict the RUL of the LIBs in terms of remaining driving range (RDR).

- *What are the different model performance criteria to compare different RUL prediction models implemented in this study?*

  We compared the different RUL prediction models based on the model evaluation metrics such as MAE, MAPE, RMSE, R^2 and model running time.

- *Can the prediction model predict the remaining driving range (RDR) of electric vehicles that are subjected to a wide range of driving profiles cycles?*

  The implemented RUL prediction model should be able to predict the RDR of an electric vehicle that is subjected to different driving profiles, given that the degradation history of the LIB used is preprocessed as suggested in this research. However, this was not tested and verified as part of this research as the data of LIBs subjected to diverse driving profiles were not present in the data set.

Chapter 8

Conclusions

8.1 Concluding Summary

In this section, we present the concluding summary of this research.

Preliminaries

In this chapter, we discussed about the basic working principle and advantages of a lithium-ion battery. We defined the necessary terminology regarding LIBs. We also discussed briefly about the various battery form factors, battery management systems and battery aging.

Literature Review

We reviewed some of the commonly used publicly available lithium-ion battery degradation data sets. We reviewed several research papers regarding RUL useful prediction of LIBs. We concluded that data-driven methodology based such as SVR, LSTM and SBM are suitable for our requirement. We also identified the research gap and we addressed the same.

Feature Engineering and EDA

In this chapter, we preprocessed the data by removing unwanted columns, created new features from the existing features in the TNO data set.

Machine Learning models

In this chapter, we implemented machine learning models such as SVR, LSTM and SBM. We optimized the hyperparameters of the SVR using Bayesian optimization technique. We implemented an LSTM network to predict the RUL of LIBs. We developed SVR and LSTM models for each battery in the data set, and we tested the effect of calendar aging on a battery's RUL. Finally, we implemented a SBM that uses dynamic time warping as a distance metric to compute the RUL of LIBs.

Results

In this chapter, we discussed the results of the three models that we implemented as part of this research.

8.2 Conclusions

Owing to the increasing popularity of electric vehicles, remaining useful life prediction of lithium-ion batteries is a crucial task. Predicting the RUL of a lithium-ion battery in an EV is a challenging problem, as the degradation of a battery is caused by multiple factors such as temperature, vehicle load, current, voltage, etc.

In this research, we analyzed the data, preprocessed the data and implemented three machine learning algorithms, namely, Support Vector Regression and Long Short Term Memory Network. We also tested the effect of calendar aging on the RUL of a battery. In this regard, we conducted two sets of experiment where in an experiment we considered the effect of calendar aging and in another experiment we ignored the effects of calendar aging. RUL prediction models that considered the calendar aging effects performed better than the models that did not consider it. We also implemented a Similarity based model to predict the RUL. The best SBM model predicted the RUL with an MAPE of 15.70%. This SBM can be used in an electric vehicle in real-time.

8.3 Limitations and Future Work

Similarity based Model (SBM) which we implemented in Section 6.5 will perform well when more run-to-failure (RTF) degradation profiles of Lithium-ion Batteries (LIB) are available. However, the data set used in this research has the run-to-failure degradation profiles of only 8 LIBs, out of which the degradation profiles of 7 LIBs were considered. This can be overcome by adding more run-to-failure degradation profiles of LIBs. Additional run-to-failure degradation profiles can be added either by acquiring data from EVs in real time or by generating synthetic data that resemble the original data using neural network models such as Conditional Generative Adversarial Networks (CTGAN) (L. Xu et al., 2019).

Another limitation regarding the data set is that the data used in this research is a lab generated data set. Additional data can be included in the data set by acquiring data from the battery within an actual electric vehicle (EV).

In this research, we have attempted to predict the remaining useful life (RUL) of the Lithium-ion Battery (LIB) in terms of remaining driving range. In a research conducted by Smuts, Scholtz and Wesson (2017), Remaining Driving Range (RDR) is influenced by various factors and they can be broadly classified into the following categories : battery modeling factors, vehicle modeling, driving behaviour, route and terrain and environmental factors. Each of these categories has several factors that affect RDR.

The data set used in this research, contains data only regarding the battery degradation, distance traveled and duration. The data does not contain a majority of the factors that affect the RDR as mentioned in (Smuts, Scholtz and Wesson, 2017). In the future, these factors can be measured from EVs in real time and the existing data set can be extended by adding new battery degradation profiles along with the data of the various factors that affect RDR.

Bibliography

Adib, K., C. Angela and W. Lim (2020). "SOH and RUL Prediction of Lithium-Ion Batteries Based on LSTM with Ensemble Health Indicators". In: (page 18).

Agnihotri, Apoorv and Nipun Batra (2020). "Exploring Bayesian Optimization". In: *Distill*. DOI: 10.23915/distill.00026. URL: https://distill.pub/2020/bayesian-optimization (page 37).

Ahmadzadeh, F and J Lundberg (2014). "Remaining useful life estimation: review". In: *International Journal of System Assurance Engineering and Management* 5, pp. 461–474 (page 17).

Arar, Dr. Steve (Oct. 2020). *The Three Major Li-ion Battery Form Factors: Cylindrical, Prismatic, and Pouch*. URL: https://www.allaboutcircuits.com/news/three-major-lithium-ion-battery-form-factors-cylindrical-prismatic-pouch (pages 5, 6).

Barré, Anthony et al. (2013). "A review on lithium-ion battery ageing mechanisms and estimations for automotive applications". In: *Journal of Power Sources* 241, pp. 680–689 (page 8).

Baru, Aditya (2018). *Three Ways to Estimate Remaining Useful Life for Predictive Maintenance*. URL: https://www.mathworks.com/company/newsletters/articles/three-ways-to-estimate-remaining-useful-life-for-predictive-maintenance.html (pages 11, 12, 13).

Birkl, Christoph (2017). "Diagnosis and prognosis of degradation in lithium-ion batteries". PhD thesis. University of Oxford. URL: https://doi.org/10.5287/bodleian:KO2kdmYGg (page 14).

Chen, Zewang et al. (2021). "Lithium-ion batteries remaining useful life prediction based on BLS-RVM". In: *Energy* 234, p. 121269. ISSN: 0360-5442. DOI: https://doi.org/10.1016/j.energy.2021.121269. URL: https://www.sciencedirect.com/science/article/pii/S0360544221015176 (page 22).

Cortes, Corinna and V. Vapnik (2004). "Support-Vector Networks". In: *Machine Learning* 20, pp. 273–297 (page 35).

Eker, Ömer Faruk, Faith Camci and Ian K. Jennions (2014). "A Similarity-Based Prognostics Approach for Remaining Useful Life Prediction". In: (page 19).

Gelfusa, M. et al. (2015). "Advanced signal processing based on support vector regression for LIDAR applications". In: *Image and Signal Processing for Remote Sensing XXI*. Ed. by Lorenzo Bruzzone. Vol. 9643. International Society for Optics and Photonics. SPIE, pp. 135–145. DOI: 10.1117/12.2194501. URL: https://doi.org/10.1117/12.2194501 (page 36).

Herh, Michael (July 2021). *Hyundai Motor's electric truck porter EV catches fire while running*. URL: http://www.businesskorea.co.kr/news/articleView.html?idxno=71926 (page 1).

Hochreiter, S. and J. Schmidhuber (1997). "Long Short-Term Memory". In: *Neural Computation* 9, pp. 1735–1780 (page 18).

Hong, Joonki et al. (2020). "Towards the swift prediction of the remaining useful life of lithium-ion batteries with end-to-end deep learning". In: *Applied Energy* 278, p. 115646. ISSN: 0306-2619. DOI: https://doi.org/10.1016/j.apenergy.2020.115646. URL: https://www.sciencedirect.com/science/article/pii/S0306261920311429 (page 21).

Jia, J. et al. (2020). "SOH and RUL Prediction of Lithium-Ion Batteries Based on Gaussian Process Regression with Indirect Health Indicators". In: *Energies* 13, p. 375 (page 20).

Jorge, I. et al. (2020). "New ANN results on a major benchmark for the prediction of RUL of Lithium Ion batteries in electric vehicles". In: *2020 19th IEEE International Conference on Machine Learning and Applications (ICMLA)*, pp. 1246–1253 (page 18).

Keil, Peter et al. (2016). "Calendar Aging of Lithium-Ion Batteries I. Impact of the Graphite Anode on Capacity Fade". In: *Journal of The Electrochemical Society* 163 (page 9).

Khumprom, Phattara and Nita Yodo (2019). "A Data-Driven Predictive Prognostic Model for Lithium-ion Batteries based on a Deep Learning Algorithm". In: *Energies* 12, p. 660 (page 18).

Kim, Seokgoo, Nam H. Kim and Joo-Ho Choi (2020). "Prediction of remaining useful life by data augmentation technique based on dynamic time warping". In: *Mechanical Systems and Signal Processing* 136, p. 106486 (page 19).

Kingma, Diederik P. and Jimmy Ba (2015). "Adam: A Method for Stochastic Optimization". In: *CoRR* abs/1412.6980 (page 40).

Lea, Colin S. et al. (2016). "Temporal Convolutional Networks: A Unified Approach to Action Segmentation". In: *ArXiv* abs/1608.08242 (page 18).

Li, Lianbing et al. (2018). "Indirect remaining useful life prognostics for lithium-ion batteries". In: *2018 24th International Conference on Automation and Computing (ICAC)*, pp. 1–5 (page 20).

Li, X. et al. (2017). "An On-Board Remaining Useful Life Estimation Algorithm for Lithium-Ion Batteries of Electric Vehicles". In: *Energies* 10, p. 691 (page 19).

Liu, J. and Z. Chen (2019). "Remaining Useful Life Prediction of Lithium-Ion Batteries Based on Health Indicator and Gaussian Process Regression Model". In: *IEEE Access* 7, pp. 39474–39484 (page 20).

Moral, P. (1997). "Nonlinear filtering : Interacting particle resolution". In: *Comptes Rendus De L' Academie Des Sciences Serie I-mathematique* 325, pp. 653–658 (page 20).

Nuhic, Adnan et al. (2013). "Health diagnosis and remaining useful life prognostics of lithium-ion batteries using data-driven methods". In: *Journal of Power Sources* 239, pp. 680–688. ISSN: 0378-7753. DOI: https://doi.org/10.1016/j.jpowsour.2012.11.146. URL: https://www.sciencedirect.com/science/article/pii/S0378775312018605 (pages 15, 32).

Omariba, Zachary Bosire, Lijun Zhang and Dongbai Sun (2018). "Review on Health Management System for Lithium-Ion Batteries of Electric Vehicles". In: *Electronics* 7, p. 72 (pages 7, 32).

Pecht, Michael (n.d.). *CALCE battery degradation data set*. URL: https://calce.umd.edu/battery-data (pages 14, 21, 22).

Pérez, Aramis et al. (2018). "Characterizing the Degradation Process of Lithium-ion Batteries Using a Similarity-Based-Modeling Approach". In: (page 19).

Plett, Gregory L. (2004). "High-performance battery-pack power estimation using a dynamic cell model". In: *IEEE Transactions on Vehicular Technology* 53, pp. 1586–1593 (page 7).

Qin, Xiaoli et al. (2017). "Prognostics of remaining useful life for lithium-ion batteries based on a feature vector selection and relevance vector machine approach". In: *2017 IEEE International Conference on Prognostics and Health Management (ICPHM)*, pp. 1–6 (page 20).

Reis, G. D. et al. (2021). "Lithium-ion battery data and where to find it". In: (page 13).

"Remaining useful life prediction of lithium-ion batteries based on Monte Carlo Dropout and gated recurrent unit" (2021). In: *Energy Reports* 7, pp. 2862–2871. ISSN: 2352-4847. DOI: https://doi.org/10.1016/j.egyr.2021.05.019. URL: https://www.sciencedirect.com/science/article/pii/S2352484721002973 (page 22).

Saha, B. and K. Goebel (2007). "NASA Ames Research Center, Battery Data Set, NASA Ames Prognostics Data Repository". In: URL: http://ti.arc.nasa.gov/project/prognostic-data-repository (pages 13, 20, 22).

Sakoe, Hiroaki and Seibi Chiba (1978). "Dynamic programming algorithm optimization for spoken word recognition". In: *IEEE Transactions on Acoustics, Speech, and Signal Processing* 26, pp. 159–165 (page 43).

Severson, K. A. (2019). "Data-driven prediction of battery cycle life before capacity degradation". In: *Nature Energy* 4, pp. 383–391 (pages 16, 18).

Shen, Dongxu et al. (2021). "A novel online method for predicting the remaining useful life of lithium-ion batteries considering random variable discharge current". In: *Energy* 218, p. 119490. ISSN: 0360-5442. DOI: https://doi.org/10.1016/j.energy.2020.119490. URL: https://www.sciencedirect.com/science/article/pii/S0360544220325974 (page 21).

Smola, Alex and B. Schölkopf (2004). "A tutorial on Support Vector Regression". In: *Statistics and Computing* 14, pp. 199–222 (page 35).

Smuts, M., B. Scholtz and J. Wesson (2017). "A critical review of factors influencing the remaining driving range of electric vehicles". In: *2017 1st International Conference on Next Generation Computing Applications (NextComp)*, pp. 196–201 (page 64).

Snoek, Jasper, H. Larochelle and Ryan P. Adams (2012). "Practical Bayesian Optimization of Machine Learning Algorithms". In: *NIPS* (page 37).

Soons, Youri et al. (2020). "Predicting Remaining Useful Life with Similarity-Based Priors". In: *IDA* (page 19).

Srivastava, Nitish et al. (2014). "Dropout: a simple way to prevent neural networks from overfitting". In: *J. Mach. Learn. Res.* 15, pp. 1929–1958 (page 40).

Sun, Peiyi et al. (Jan. 2020). "A Review of Battery Fires in Electric Vehicles". In: *Fire Technology*, pp. 1–50. DOI: 10.1007/s10694-019-00944-3 (page 1).

Sun, Tianfei et al. (2019). "A Novel Hybrid Prognostic Approach for Remaining Useful Life Estimation of Lithium-Ion Batteries". In: *Energies* 12, p. 3678 (page 18).

Team, MIT EV (Dec. 2008). *A Guide to Understanding Battery Specifications.* URL: http://web.mit.edu/evt/summary_battery_specifications.pdf (page 4).

Vashisht, Raghav (Sept. 2021). *Machine Learning: When to perform a Feature Scaling?* URL: https://www.atoti.io/when-to-perform-a-feature-scaling/ (page 34).

Virkler, Dennis A., B. M. Hillberry and Prem K. Goel (1979). "The Statistical Nature of Fatigue Crack Propagation". In: *Journal of Engineering Materials and Technology-transactions of The Asme* 101, pp. 148–153 (page 19).

Wang, Dong et al. (2017). "Nonlinear-drifted Brownian motion with multiple hidden states for remaining useful life prediction of rechargeable batteries". In: *Mechanical Systems and Signal Processing* 93, pp. 531–544. ISSN: 0888-3270. DOI: https://doi.org/10.1016/j.ymssp.2017.02.027. URL: https://www.sciencedirect.com/science/article/pii/S0888327017301012 (page 21).

Wang, Han, Xiaobing Ma and Yu Zhao (2019). "An improved Wiener process model with adaptive drift and diffusion for online remaining useful life prediction". In: *Mechanical Systems and Signal Processing* 127, pp. 370–387. ISSN: 0888-3270. DOI: https://doi.org/10.1016/j.ymssp.2019.03.019. URL: https://www.sciencedirect.com/science/article/pii/S0888327019301931 (page 21).

Wang, Shunli et al. (2021). "A critical review of improved deep learning methods for the remaining useful life prediction of lithium-ion batteries". In: *Energy Reports* 7, pp. 5562–5574. ISSN: 2352-4847. DOI: https://doi.org/10.1016/j.egyr.2021.08.182. URL: https://www.sciencedirect.com/science/article/pii/S235248472100785X (page 20).

Wei, Jingwen, Guangzhong Dong and Zonghai Chen (2018). "Remaining Useful Life Prediction and State of Health Diagnosis for Lithium-Ion Batteries Using Particle Filter and Support Vector Regression". In: *IEEE Transactions on Industrial Electronics* 65, pp. 5634–5643 (page 20).

Wu, Y. et al. (2019). "Remaining Useful Life Prediction of Lithium-Ion Batteries Using Neural Network and Bat-Based Particle Filter". In: *IEEE Access* 7, pp. 54843–54854 (page 20).

Xiang, Shun et al. (2018). "Lithium-Ion Battery Online Rapid State-of-Power Estimation under Multiple Constraints". In: *Energies* 11, p. 283 (page 7).

Xiong, Rui (2019). "Battery Management Algorithm for Electric Vehicles". In: (pages 3, 4, 8).

Xu, Lei et al. (2019). "Modeling Tabular data using Conditional GAN". In: *Advances in Neural Information Processing Systems* (page 64).

Xu, Xiaodong et al. (2021). "Remaining Useful Life Prediction of Lithium-ion Batteries Based on Wiener Process Under Time-Varying Temperature Condition". In: *Reliability Engineering & System Safety* 214, p. 107675. ISSN: 0951-8320. DOI: https://doi.org/10.1016/j.ress.2021.107675. URL: https://www.sciencedirect.com/science/article/pii/S0951832021002131 (page 21).

Zhang, Yongzhi et al. (2019). "Validation and verification of a hybrid method for remaining useful life prediction of lithium-ion batteries". In: *Journal of Cleaner Production* 212, pp. 240–249 (page 20).

Zhou, Danhua et al. (2020). "State of Health Monitoring and Remaining Useful Life Prediction of Lithium-Ion Batteries Based on Temporal Convolutional Network". In: *IEEE Access* 8, pp. 53307–53320 (page 18).

Zhou, Yapeng and Miaohua Huang (2016). "Lithium-ion batteries remaining useful life prediction based on a mixture of empirical mode decomposition and ARIMA model". In: *Microelectron. Reliab.* 65, pp. 265–273 (page 18).

Zio, Enrico and Francesco Di Maio (2010). "A data-driven fuzzy approach for predicting the remaining useful life in dynamic failure scenarios of a nuclear system". In: *Reliab. Eng. Syst. Saf.* 95, pp. 49–57 (page 19).

Appendix A

System Configuration

A.1 Hawrdware specifications

- **Processor** : Intel(R) Core(TM) i9-9980HK CPU

- **Operating System** : Windows 10 Home

- **Clock Speed** : 2.40GHz

- **Number of Cores** : 8

- **Total RAM** : 32 GB

A.2 Software Used

- **MATLAB R2021a - Student Edition**

Appendix B

Battery Specifications

B.1   NASA data set

Parameters	Values
Nominal Voltage	3.6V
Nominal Capacity	2,850 mAh
Minimum Discharge Voltage 3V	3V
Maximum Discharge current	1C
Charging Voltage	4.2V (maximum)
Charging current	0.5C
Charging Time	3 hours (approx)
Charging Method	CC and CV
Cell Weight	48g (approx)
Cell Dimension	18.4mm (dia) and 65mm (height)

B.2   CALCE data set

Parameters	Values
Cell Dimensions(mm):	25.85mm(D)*65.15mm(H)
Cell Weight(g):	70
Nominal Capacity:	2300mAh
Voltage(nominal,V):	3.3V
Internal impedance(1kHz AC typical, mΩ):	6
HPPC 10 Sec Discharge Pulse Power 50% SOC:	200W
Recommended standard charge method:	1C to 3.6V CCCV, 45 min
Recommended fast charge Method to 80% SOC:	4C to 3.6V CC,12 min
Maximum continuous discharge(A):	70
Maximum Pulse discharge(10 seconds,A):	120
Cycle life at 10C discharge, 100% DOD:	Over 1,000 cycles
Operating temperature range:	-30°C to +55°C
Storage temperature range: -40°C to +60°C	-40°C to +60°C

B.3 Oxford data set

Parameters	Values
Typical Capacity	740 mAh
Nominal Voltage	3.7 V
Max. Current (Charge Condition)	1.48 A
Max. Voltage (Charge Condition)	4.2 V ± 0.03 V
Continuous Current (Discharge Condition)	14.8 A
Peak Current (Discharge Condition)	29.60 A
Cut-off Voltage (Discharge Condition)	4.2 V ± 0.03 V
Cycle Life	> 500 cycles
Operating Temperature (Charge)	0 to 40°C
Operating Temperature (Discharge)	-20 to 60°C

Table B.1: Battery specifications of the battery used in Oxford data set

B.4 Toyota Research Institute data set

Parameters	Values
Typical Capacity	1200 mAh
Nominal Voltage	3.3 V
Max. Current (Charge Condition)	2.0 A
Charge Time	2.0hrs
Max. Voltage (Charge Condition)	4.2 V ± 0.05 V
Max. Continuous Current (Charge Condition)	4 A
Max. Continuous Current (Discharge Condition)	30 A
Min. Voltage (Discharge Condition)	2V
Cut-off Voltage (Discharge Condition)	2.5 V
Temperature (Charge)	0 to 55°C
Temperature (Discharge)	-30 to 55°C
Temperature (Storage)	-40°C to 60°C

Table B.2: Battery specifications of the battery used in TNO data set

B.5   TNO data set

Parameters	Values
Typical Capacity	2000 mAh
Nominal Voltage	3.6 V
Max. Current (Charge Condition)	2.0 A
Charge Time	2.0 hrs
Max. Voltage (Charge Condition)	4.2 V $\pm$ 0.05 V
Continuous Current (Discharge Condition)	14.8 A
Peak Current (Discharge Condition)	6.0 A at 45°C, 4.0 A at 60°C
Cut-off Voltage (Discharge Condition)	2.5 V
Temperature (Charge)	0 to 45°C
Temperature (Discharge)	-20 to 60°C
Temperature (Storage)	$< 35^o$C

Table B.3: Battery specifications of the battery used in TNO data set