

# **Entwicklung und Evaluation daten- getriebener Batterie-Alterungsmodelle für Hybridfahrzeuge**

Zur Erlangung des akademischen Grades eines

**Doktors der Ingenieurwissenschaften (Dr.-Ing.)**

von der KIT-Fakultät für Maschinenbau  
des Karlsruher Instituts für Technologie (KIT)

genehmigte

## **DISSERTATION**

von

**M. Sc. Dominik Herz**

geb. in Bad Mergentheim

Tag der mündlichen Prüfung: 25.06.2025

Hauptreferent: Prof. Dr. rer. nat. Frank Gauterin

Korreferent: Prof. Dipl.-Ing. Dr. techn. Peter Fischer



# Kurzfassung

Die Bestimmung des Gesundheitszustandes von Lithium-Ionen-Batterien ist zentraler Bestandteil der Lebensdauerauslegung moderner, elektrifizierter Fahrzeuge. Komplexe Wechselwirkungen innerhalb des Batteriesystems erfordern simulative Ansätze, um Degradationsmechanismen abbilden zu können. Historische Betriebsdaten erlauben die Berücksichtigung des Nutzungsverhaltens und deren Einfluss auf die finalen Alterungseffekte der Batterie. Durch die steigende Verfügbarkeit großer Mengen ausgelesener Diagnosedaten rücken zunehmend maschinelle Lernmethoden in den Fokus der Anwendung.

Als Datengrundlage werden üblicherweise Spannungs- sowie Stromprofile zyklisierter Zellen eingesetzt. Obwohl die Schätzung des Alterungsverhaltens umfassend untersucht wird, existieren im Ingenieurbereich wenige bis keine Untersuchungen zur Prädiktion auf Basis historischer Nutzungsdaten. Um bisherige Lücken in der Fachliteratur zu schließen, sollen vielversprechende Konzepte des maschinellen Lernens auf die zugrundeliegende Datenbasis übertragen sowie optimierte Methoden entwickelt werden, um das Degradationsverhalten der Batterie abbilden zu können.

Im ersten Schritt wird zunächst die zugrundeliegende Datenbasis systematisch ausgewertet und analysiert. Durch die Kombination von Ingenieursverständnis und statistischer Methoden werden relevante Alterungsparameter identifiziert sowie neue signifikante Bewertungsgrößen basierend auf Filteroperationen generiert.

Darauf aufbauend werden auf Basis jüngster Beiträge aus dem Automobilbereich und der Spracherkennung maschinelle Prädiktionsmodelle als Alternative zur physikalischen Modellierung des Alterungsverhaltens identifiziert. Im Rahmen des Methodenentwicklungsprozesses werden diese Modelle erweitert und generalisiert, um die Anwendbarkeit für die im Rahmen der Arbeit behandelten Problemstellung zu gewährleisten. Die Ergebnisse zeigen Stärken und Schwächen verschiedener Ansätze auf. Die größte Herausforderung stellt für alle Modellvarianten die Prädiktion geringer Zustandswerte dar. Nichtsdestotrotz wird gezeigt, dass der Ansatz eines Rekurrenten Neuronalen Netzes, eines Support Vector Regressors sowie eines xGBoost-Algorithmus durchaus zielführend ist, um das Alterungsverhalten eines komplex belasteten Batteriesystems – als Komponente eines dynamischen Gesamtsystems – mittels der zugrundeliegenden Datenbasis zu bewerten sowie zu prädizieren. In allen Validierungssequenzen bestätigt sich, dass die Ergebnisqualität durch die umgesetzten Maßnahmen sukzessive gesteigert werden kann.

Dies motiviert die Untersuchung der Modellgüte hinsichtlich ungewohnter Daten und somit die einer Zukunftsprädiktion. Basierend auf der Analyse von vier Beispielfahrzeugen wird gezeigt, dass speziell für niedrige  $SoH_C$ -Bereiche aufgrund von Datenqualitätsproblemen ungültige Prädiktionen auftreten können. Schließlich wird auf Basis dieser Erkenntnisse die Frage

behandelt, ob eine Übertragbarkeit der Methodik auf andere Fahrzeugsysteme gegeben ist. Die Ergebnisse legen dar, dass nur begrenzt grundlegende Zusammenhänge abgebildet werden können. Eine genauere Realitätsabbildung erfordert die Integration aussagekräftigerer Eingangsgrößen und eine spezifische Betrachtung der Zellchemie, weshalb eine separate Modellentwicklung für jedes Fahrzeugsystem empfohlen wird.



# Abstract

Assessing the health status of lithium-ion batteries is a critical aspect in the lifespan design of modern, electrified vehicles. The intricate interactions within the battery system necessitate simulation-based approaches to represent degradation mechanisms accurately. Historical operational data enable the incorporation of user behavior and its impact on the battery's final aging effects. With the increasing availability of large-scale diagnostic data, machine learning methods are becoming more central to this field.

Typically, voltage and current profiles of cycled cells are used as the data foundation. Despite extensive research on aging behavior estimation, there is a scarcity of studies in the engineering field focusing on predictions based on historical usage data. To address gaps in existing literature, this study aims to apply and optimize successful machine learning methods to this unique data set, enhancing our understanding of battery degradation behavior. The initial step involves a systematic evaluation and analysis of the underlying data set. Combining engineering insights with statistical methods, relevant aging parameters are identified, and new, meaningful metrics are generated through filtering operations.

Building on this, the study identifies machine learning prediction models, informed by recent contributions in the automotive and speech recognition fields, as alternatives to the electrical-physical modeling of aging behavior.

In the method development process, these models are refined and generalized to ensure applicability to the present problem. The results highlight the strengths and weaknesses of various approaches, with the greatest challenge for all model variants being the prediction of low state values. However, it is demonstrated that the approach of using Recurrent Neural Networks, Support Vector Regressors, and xGBoost algorithms is valid for estimating the aging behavior of a complexly stressed battery system. This estimation is viewed as a component of a dynamic overall system, using the underlying data set. Validation sequences confirm that the error measure can be progressively reduced through implemented measures.

This motivates an investigation into the model's accuracy with unseen data and its potential for future prediction. Using four example vehicles, it is shown that the approach, particularly for low  $SoH_C$  ranges, can lead to invalid predictions due to data quality issues. Finally, based on these findings, the study addresses whether the methodology is transferable to other vehicle systems. The results suggest that only limited fundamental correlations can be represented. A more accurate depiction of reality requires the integration of more significant input variables and a specific consideration of cell chemistry, hence recommending separate model development for each vehicle system.

# Inhaltsverzeichnis

<b>Kurzfassung</b> . . . . .	<b>i</b>
<b>Abstract</b> . . . . .	<b>v</b>
<b>Abkürzungen und Symbole</b> . . . . .	<b>xi</b>
<b>1 Einleitung</b> . . . . .	<b>1</b>
1.1 Motivation . . . . .	1
1.2 Zielsetzung . . . . .	2
1.3 Aufbau der Arbeit . . . . .	3
<b>2 Theorie und Grundlagen</b> . . . . .	<b>5</b>
2.1 Lithium-Ionen-Batterien . . . . .	5
2.1.1 Aufbau und Funktionsweise . . . . .	7
2.1.2 Batterie-Management-System . . . . .	8
2.1.3 Alterungssmechanismen . . . . .	11
2.2 Maschinelles Lernen . . . . .	18
2.2.1 Generalisierungsfähigkeit und Regularisierung . . . . .	23
2.2.2 Künstliche Neuronale Netze . . . . .	25
2.2.3 Support Vector Machines . . . . .	32
2.2.4 Extreme Gradient Boosting . . . . .	36
<b>3 Stand der Forschung</b> . . . . .	<b>41</b>
3.1 Neuronale Netze . . . . .	42
3.2 Support Vector Machines . . . . .	46
3.3 Extreme Gradient Boosting . . . . .	50
3.4 Zusammenfassung und Forschungslücke . . . . .	56

<b>4 Methodenentwicklung</b> . . . . .	<b>59</b>
4.1 Datenanalyse und Datenvorverarbeitung . . . . .	59
4.1.1 Datengrundlage für den Entwurf der Modelle . . . . .	60
4.1.2 Datenvorverarbeitung . . . . .	65
4.2 Eingangsdaten für die Modelle . . . . .	71
4.2.1 Feature Engineering . . . . .	71
4.2.2 Finaler Datenvorverarbeitungsprozess . . . . .	77
4.3 Modellierung des realen Systemverhaltens . . . . .	82
4.3.1 Auswahl und Vergleich der Modellarchitekturen . . . . .	82
<b>5 Ergebnisse und Diskussion</b> . . . . .	<b>99</b>
5.1 Inhalt und Auswertungsmethodik . . . . .	99
5.2 Ergebnisse der Modellentwicklung . . . . .	100
5.2.1 Tiefes Künstliches Neuronales Netzwerk . . . . .	100
5.2.2 Ergebnisse der Support Vector Machine . . . . .	106
5.2.3 Ergebnisse des xGBoost-Modells . . . . .	112
5.2.4 Vergleich der Modellvarianten und abgeleitete Maßnahmen . . . . .	116
5.3 Validierung der Modellergebnisse . . . . .	117
5.4 Einzelfahrzeuganalyse . . . . .	123
5.5 Übertragbarkeit der xGBoost-Methodik auf andere Baureihen .	127
<b>6 Zusammenfassung und Ausblick</b> . . . . .	<b>129</b>
6.1 Zusammenfassung . . . . .	129
6.1.1 Hauptergebnisse der Arbeit . . . . .	130
6.1.2 Abschließende Bewertung der Forschungshypothesen .	132
6.2 Ausblick . . . . .	133
<b>A Anhang</b> . . . . .	<b>137</b>
<b>Abbildungsverzeichnis</b> . . . . .	<b>141</b>
<b>Tabellenverzeichnis</b> . . . . .	<b>143</b>
<b>Literaturverzeichnis</b> . . . . .	<b>145</b>

<b>Eigene Publikationen</b> . . . . .	<b>167</b>
<b>Betreute studentische Abschlussarbeiten</b> . . . . .	<b>169</b>



# Abkürzungen und Symbole

## Abkürzungen

<b>BEV</b>	<u>B</u> attery <u>E</u> lectric <u>V</u> ehicle
<b>BMS</b>	<u>B</u> atterie- <u>M</u> anagement- <u>S</u> ystem
<b>BoL</b>	<u>B</u> egin <u>o</u> f <u>L</u> ife
<b>DoD</b>	<u>D</u> epth <u>o</u> f <u>D</u> ischarge
<b>EoL</b>	<u>E</u> nd <u>o</u> f <u>L</u> ife
<b>FAP</b>	<u>F</u> ahrzeug- <u>A</u> nalys <u>e</u> - <u>P</u> rotokoll
<b>FFNN</b>	<u>F</u> eed <u>F</u> orward <u>N</u> eural <u>N</u> etwork
<b>GKE</b>	<u>G</u> aussian- <u>K</u> ernel- <u>E</u> stimation
<b>HDM</b>	<u>H</u> istorien- <u>D</u> aten- <u>M</u> atrizen
<b>HEV</b>	<u>H</u> ybrid <u>E</u> lectric <u>V</u> ehicle
<b>HP</b>	<u>H</u> yper- <u>P</u> arameter
<b>KIT</b>	<u>K</u> arlsruher <u>I</u> nstitut für <u>T</u> echnologie
<b>KNN</b>	<u>K</u> ünstliches <u>N</u> euronales <u>N</u> etz
<b>LIB</b>	<u>L</u> ithium- <u>I</u> onen- <u>B</u> atterie

<b>MAE</b>	<u>M</u> ean <u>A</u> bsolute <u>E</u> rror
<b>MPE</b>	<u>M</u> ean <u>P</u> rediction <u>E</u> rror
<b>MSE</b>	<u>M</u> ean <u>S</u> quared <u>E</u> rror
<b>PAG</b>	Dr. Ing. h.c. F. <u>P</u> orsche <u>AG</u>
<b>PHEV</b>	<u>P</u> lug-In- <u>H</u> ybrid <u>E</u> lectric <u>V</u> ehicle
<b>RBF</b>	<u>R</u> adiale- <u>B</u> asis- <u>F</u> unktion
<b>ReLU</b>	<u>R</u> ectified <u>L</u> inear <u>Unit</u>
<b>RMSE</b>	<u>R</u> oot <u>M</u> ean <u>S</u> quared <u>E</u> rror
<b>RNN</b>	<u>R</u> ekurrentes <u>N</u> euronales <u>N</u> etz
<b>SEI</b>	<u>S</u> olid <u>E</u> lectrolyte <u>I</u> nterphase
<b>SoH<sub>C</sub></b>	<u>S</u> tate <u>o</u> f <u>H</u> ealth - <u>C</u> apacity
<b>SoH<sub>R</sub></b>	<u>S</u> tate <u>o</u> f <u>H</u> ealth - <u>R</u> esistance
<b>SVM</b>	<u>S</u> upport <u>V</u> ector <u>M</u> achine
<b>SVR</b>	<u>S</u> upport <u>V</u> ector <u>R</u> egressor
<b>xGBoost</b>	<u>E</u> xtreme <u>G</u> radient <u>B</u> oosting

### **Lateinische Symbole und Variablen (Großbuchstaben)**

<b><math>C^+</math></b>	Minderheitsklasse
<b><math>C^-</math></b>	Mehrheitsklasse
<b><math>C_{\text{Akt}}</math></b>	Aktuelle Batteriekapazität

$C_{\text{Nenn}}$	Nennkapazität der Batterie
$C$	Regularisierungsparameter
$\mathcal{D}_V$	Vergangenheitsdatensatz
$\mathcal{D}_Z$	Zukunftssdatensatz
$J$	Strafterm der Verlustfunktion
$K$	Kernelfunktion des Support Vector Regressors
$L_1, L_2$	Varianten der Straffunktion
$R_{\text{BoL}}$	Innenwiderstand zum Zeitpunkt des Begin of Life
$R_{\text{EoL}}$	Innenwiderstand zum Zeitpunkt des End of Life
$R_{\text{Akt}}$	Aktueller Innenwiderstand
$T_B$	Batterietemperatur
$T_L$	Lagertemperatur
$U_{\text{Zelle}}$	Zellspannung
$W$	Gewichtungsmatrix
$X$	Eingabematrix
$Y$	Referenzwert
$\hat{Y}$	Approximierter Ausgabewert

### **Lateinische Symbole und Variablen (Kleinbuchstaben)**

$a$	Aktion - Reinforcement Learning
-----	---------------------------------

$a$	Aktivierungsvektor
$b$	Bias-Vektor
$g$	Belohnung/Bestrafung - Reinforcement Learning
$i_B$	Batteriestrom
$i_{NR}$	Strom aus den Nebenreaktionen der Batterie
$n_E$	Anzahl an Entscheidungsbäumen
$t_L$	Lagerzeit
$r$	Normalisierungsintervall
$s$	Status - Reinforcement Learning
$w$	Gewichtungsvektor
$x$	Eingabevektor
$y$	Referenzwert
$\hat{y}$	Approximierter Ausgabewert
$z$	Innerer Zustand eines Neurons

### **Griechische Symbole und Variablen (Großbuchstaben)**

$\Delta$	Absolute oder relative Differenz zwischen zwei Größen
$\phi$	Aktivierungsfunktion

## Griechische Symbole und Variablen (Kleinbuchstaben)

$\alpha$	Relativer Wichtigkeitswert
$\beta$	Skalierungsfaktor der Verlustfunktion
$\gamma$	Kernel-Koeffizient der Support Vector Regressors
$\delta$	Dirac-Distribution
$\varepsilon$	Insensitives Fehlermaß
$\kappa$	Lernrate
$\lambda$	Weight Decay
$\mu$	Erwartungswert
$\xi$	Schlupfvariable
$\sigma$	Logistische Sigmoidfunktion
$\chi$	Fehlerwahrscheinlichkeitsdichte

## Operatoren und mathematische Symbole

$a$	Skalare Größe (Kleinbuchstaben, nicht fettgedruckt)
$a$	Vektorielle Größe (Kleinbuchstaben, fettgedruckt)
$A$	Matrix Größe (Großbuchstaben, fettgedruckt)



# 1 Einleitung

## 1.1 Motivation

Die Automobilindustrie befindet sich im Wandel, weg von klassischen Verbrennungsmotoren hin zu elektrisch angetriebenen Fahrzeugen. Dieser Wandel bringt eine Vielzahl neuer Potentiale und Chancen für die Fahrzeughersteller mit sich, gleichzeitig sieht sich dieser mit bislang unbekannten Herausforderungen konfrontiert. Eine der größten Herausforderungen ist die mittels Elektroantrieb erzielbare Reichweite elektrischer Fahrzeuge über das gesamte Fahrzeugleben hinweg, welche maßgeblich durch die Lebensdauer der Antriebskomponenten, insbesondere der Hochvolt-Batterie (im Folgenden auch Batterie) beeinflusst wird [1]. Das Einhalten gesetzlicher und politischer Vorgaben muss folglich durch eine entsprechende Entwicklung der Batterie sichergestellt werden.

Neben der primären Antriebsart ändert sich jedoch auch zunehmend die Digitalisierung, Vernetzung und Konnektivität moderner Fahrzeuge. Ausgestattet mit einer Vielzahl von Sensoren und Hochleistungssteuergeräten werden in gegenwärtigen Fahrzeugen enorme Datenmengen generiert. Getrieben durch neuartige Ansätze, sowie stetig steigende Rechenperformance, gewinnen vor allem Methoden des maschinellen Lernens und des Big-Data-Umfelds zunehmend an Bedeutung. Auch in der Batterieentwicklung

gibt es an vielen Stellen potentielle Einsatzmöglichkeiten für mathematische Modelle dieser Art.

Die Batterie fungiert als elektrochemischer Energiespeicher, in dem viele Prozesse parallel ablaufen und sich gegenseitig beeinflussen. Um einen sicheren Betrieb zu gewährleisten wird die Batterie von einer Elektronik überwacht, die gemessene und errechnete Zustände prüft und speichert. Diese gespeicherten Betriebsdaten, welche die Historie des Fahrzeugs wiedergeben, können genutzt werden, um Erkenntnisse über den Einfluss des Nutzungsverhaltens auf die Komponentenperformance für den gesamten Lebenszyklus zu erhalten. So kann im Zusammenhang der Alterungsbestimmung die Eignung von maschinellen Lernmethoden sowohl zur Extraktion von Alterungsparametern als auch zur Schätzung des Alterungsverhaltens der Batterie untersucht werden.

## 1.2 Zielsetzung

In der Fachliteratur erfolgt die Untersuchung und Entwicklung von datengetriebenen Modellen für die Bestimmung des Gesundheitszustandes der Batterie gesondert für den jeweiligen Anwendungsfall bzw. die zugrundeliegende Datenbasis. Es wird erwartet ein komplexes dynamisches Gesamtsystem durch vorhandene Historiendaten beschreiben und verstehen zu können und demnach Rückschlüsse auf Alterungsfaktoren zu ziehen. Ziel dieser Arbeit ist es demnach mit bereits vorhandenen historischen Betriebsdaten Rückschlüsse auf bisher unbekanntes Alterungsverhalten zu ziehen und den Gesundheitszustand der Batterie zu schätzen und somit

neue *virtuelle Sensoren*<sup>1</sup> zu generieren. Da in der Literatur für eine Problemstellung mit einer derartigen Datenbasis kaum Hinweise zu finden sind, sollen im Rahmen der vorliegenden Arbeit erste Schritte durchgeführt werden, um diese Lücke zu schließen. In Anbetracht dieser Herausforderungen werden zwei Forschungshypothesen abgeleitet:

**Hypothese H-1.** Die systematische Analyse der historischen Betriebsdaten erlaubt die Identifizierung von Parametern, welche die Alterung der Batterie signifikant beeinflussen, um letztlich eine Alterungsmodellierung zu ermöglichen.

**Hypothese H-2.** Der Gesundheitszustand der Batterie kann durch die Weiterentwicklung maschineller Lernmodelle auf Basis der historischen Betriebsdaten prädiziert werden.

## 1.3 Aufbau der Arbeit

Die vorliegende Arbeit gliedert sich in vier Inhaltskapitel. Im ersten Kapitel „Theorie und Grundlagen“ werden wichtige Termini für den weiteren Verlauf dieser Arbeit definiert sowie eine kompakte Erläuterung der

---

<sup>1</sup> **Definition:** Virtueller Sensor nach [2]: Modellgestützte Algorithmen, die Zielgrößen auf Basis von Eingangsgrößen und dem funktionellen Zusammenhang der Ein- und Ausgangsgrößen berechnen

zum Verständnis erforderlichen Grundlagen bereitgestellt. Darauf aufbauend beinhaltet Kapitel 3 einen Überblick über relevante Ergebnisse des aktuellen Standes der Forschung. Gefolgt von der genauen Analyse der zugegrundeliegenden Daten ist der systematische Entwurf verschiedener Modellarchitekturen zur Approximation des Gesundheitszustandes der Batterie Teil von Kapitel 4. Zusätzlich werden physikalisch und statistisch motiviert relevante Eingangsgrößen ermittelt sowie deren Einfluss auf die Modellperformance untersucht. Der Diskussion und Interpretation der Ergebnisse widmet sich Kapitel 5. Den Abschluss bildet die Zusammenfassung der Ergebnisse sowie ein Ausblick auf mögliche folgende Arbeitsinhalte.

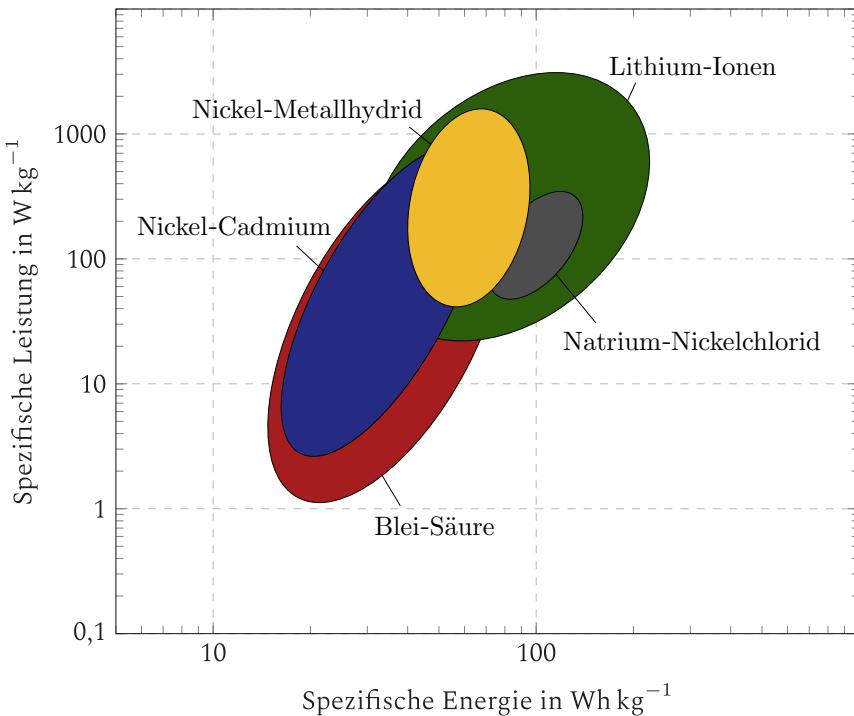
## 2 Theorie und Grundlagen

Im Folgenden wird in Abschnitt 2.1 eine Einführung in die Lithium-Ionen-Technologie gegeben. Der Aufbau und die Funktionsweise einer Lithium-Ionen-Batterie (kurz: LIB) werden beschrieben sowie ein Überblick über verfügbare Batterietechnologien gegeben. Die grundlegenden Alterungsmechanismen während des Betriebs sowie der Lagerung einer LIB werden vorgestellt und bekannte Herausforderungen im Zusammenhang mit dem Einsatz in Hybridfahrzeugen beleuchtet. Anschließend werden in Abschnitt 2.2 die Grundlagen des maschinellen Lernens aufgearbeitet sowie ausgesuchte Algorithmen näher erläutert.

### 2.1 Lithium-Ionen-Batterien

Die Lithium-Ionen-Technologie ist aus dem Automobilsektor nicht mehr wegzudenken. Obwohl die Fortschritte bei LIB seit deren Einführung beträchtlich sind, gibt es mehrere technische Herausforderungen, um die künftigen Anforderungen der Automobilindustrie zu erfüllen [3]. Im Vergleich zu anderen Branchen haben Automobilanwendungen höhere technische Anforderungen hinsichtlich kalendarischer Lebensdauer, Zykluslebensdauer und Systemkosten [4]. Besonders Ersteres rückt in den Fokus dieser Arbeit.

Im Wesentlichen wird zwischen Primärzellen bzw. -batterien und Sekundärzellen bzw. -batterien unterschieden [5]. So sind Primärbatterien nach der Energieentnahme nicht mehr nutzbar, wohingegen Sekundärbatterien wiederaufladbar und daher mehrfach nutzbar sind.



**Abbildung 2.1:** Ragone-Diagramm ausgewählter Batterietechnologien in eigener Darstellung nach [6].

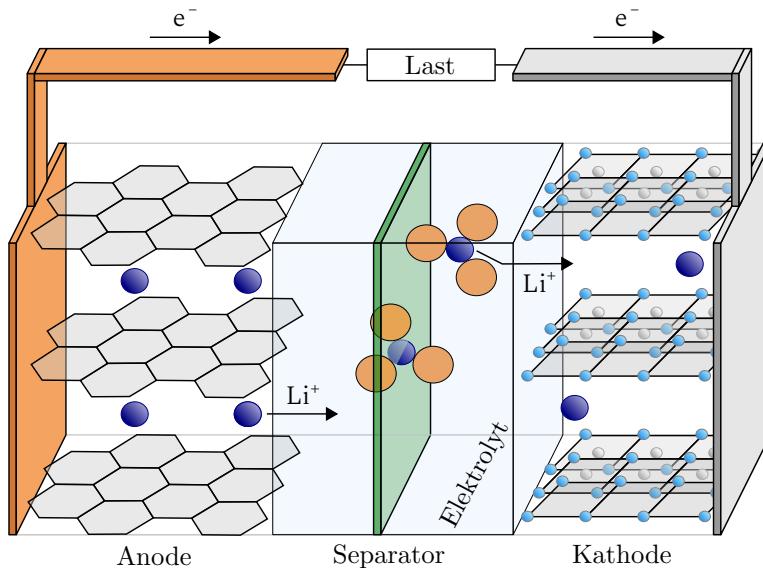
Unterschiedlichste Batterietypen können für Elektromobilitätskonzepte herangezogen werden. Hierzu zählen bspw. Nickel-Cadmium-, Blei-Säure-, Nickel-Metallhydrid-, Natrium-Nickelchlorid- sowie Lithium-Ionen-Batterien.

Die Eignung verschiedener Technologien für Mobilitätsanwendungen lässt sich mittels eines in Abbildung 2.1 dargestellten Ragone-Plots beurteilen. Im grün eingefärbten Bereich ist zu erkennen, dass Lithium-Ionen-Zellen - abhängig von der jeweiligen Zellchemie - sowohl hohe spezifische Energiedichten (Reichweite) als auch spezifische Leistungsdichten (Performance) erreichen können. Des Weiteren rechtfertigen langfristige Haltbarkeit, hohe Laderaten für schnelle Ladevorgänge, Skalierbarkeit und Umweltverträglichkeit den Einsatz der Lithium-Ionen-Technologie für automobile Anwendungen [7]. Auch bei den im Rahmen dieser Arbeit untersuchten Energiespeichern handelt es sich um LIB. Demnach soll speziell auf diese näher eingegangen werden. Für tiefergehende Informationen bezüglich der anderen Technologien sei an dieser Stelle auf [8] verwiesen.

### 2.1.1 Aufbau und Funktionsweise

Prinzipiell besteht jede Zelle einer Lithium-Ionen-Batterie aus einer negativen Elektrode (Anode) und einer positiven Elektrode (Kathode). Diese sind von einem Elektrolyten umgeben, der im Wesentlichen ionisch jedoch nicht elektrisch leitfähig ist. Die Vermeidung eines elektrischen Kontakts von Anode und Kathode wird bei Verwendung eines flüssigen Elektrolyts durch einen permeablen Separator gewährleistet. Ein typischer Aufbau der LIB ist in Abbildung 2.2 schematisch dargestellt. Damit nun beim Entladen der Batterie Strom fließen kann, erfolgt ein Ionenaustausch durch den Elektrolytraum von der Anode zur Kathode durch den Separator [9]. Werden zwei oder mehr Zellen elektrisch miteinander verbunden, ist von einem Batteriepack bzw. einer Batterie die Rede [10]. Das gesamte Batteriesystem im Fahrzeug besteht letztendlich aus mehreren verschalteten Packs, die als Module bezeichnet werden. Grundsätzlich führt eine Reihenschaltung der

Zellen in den Modulen zu einer Erhöhung der Spannung (Spannungsaddition). Eine Parallelschaltung der Module zum Batteriestack erhöht durch die Addition der Zellkapazitäten [9] die Gesamtkapazität des Batteriesystems sowie die maximale Leistung.



**Abbildung 2.2:** Schematischer Aufbau einer Lithium-Ionen-Zelle während eines Entladevorgangs in eigener Darstellung nach [11, 12].

## 2.1.2 Batterie-Management-System

Die Steuerung und Überwachung Batterie-interner Prozesse übernimmt das Batterie-Management-System (kurz: BMS). Die Hauptfunktion des BMS besteht darin, durch die kontinuierliche Überwachung der Zustände und

Prozesse innerhalb der Batterie eine optimale Steuerung zu gewährleisten, sodass die Lebensdauer maximiert und die funktionale Sicherheit über den gesamten Lebenszyklus sichergestellt wird. Alterungsparameter sowie -mechanismen werden in Unterabschnitt 2.1.3 genauer beleuchtet. Neben der Isolationsüberwachung kontrolliert das BMS bspw. die Zellspannungs- und Temperatursensorik, legt zulässige Lade- sowie Entladeströme fest und ist maßgeblich für ein geschicktes Thermomanagement verantwortlich. Optimale Temperaturen liegen bei  $20^{\circ}\text{C} \leq T_{\text{B}} \leq 60^{\circ}\text{C}$  [9].

Einerseits sorgt das BMS dafür, dass eine starke Entladung der Zellen durch eine Unterschreitung der Zellspannung  $U_{\text{Zelle}} < 2,5\text{ V}$  verhindert wird. Irreversiblen Schäden, welche mit einer beschleunigten Alterung sowie einer erhöhten Selbstentladung einhergehen, kann so Abhilfe geleistet werden. Andererseits wird beim Laden einer zu hohen Zellspannung von  $U_{\text{Zelle}} > 4,2\text{ V}$  entgegengewirkt, sodass die Gefahr einer spontanen Selbstzündung verhindert wird. Des Weiteren ermöglicht das BMS die Umsetzung einer intelligenten Betriebsstrategie, wodurch eine gewöhnliche Batterie bis zu 1000 Zyklen durchlaufen kann, ehe diese eine Restkapazität von 80 % ausgehend von der Nennkapazität erreicht [13]. Für diese Funktion bedarf es einer Berechnung von Gesundheits- und Ladezustand als notwendige Zustandgrößen während des Fahrbetriebs. Dies geschieht gegenwärtig durch mathematische Modelle, welche die Zustände aus überwachten Strom-, Spannungs- und Temperaturwerten berechnen. Bei einer Verwendung dieser Größen muss mit gewissen Fehlertoleranzen gerechnet werden. Mögliche Fehlermargen durch die auf dem BMS implementierten Algorithmen können bei bis zu 10 % liegen. In Unterabschnitt 4.1.2 wird dieses Thema eingehender diskutiert.

**Ladezustand** Der State of Charge (kurz: *SoC*) beschreibt den Ladezustand der Batterie. Die Berechnung erfolgt meist über das *Coulomb-Counting*. Hierbei werden Batteriestrommesswerte über den Nutzungszeitraum integriert. Für den Ladevorgang gilt: Ist der Ladezustand  $SoC_0$  zum Zeitpunkt  $t = 0$  bekannt und der Hauptreaktionsstrom  $i = i_B - i_{NR}$  als Differenz von Batteriestrom  $i_B$  und Strom in die Nebenreaktionen  $i_{NR}$  bekannt, so lässt sich der Ladezustand wie folgt bestimmen:

$$SoC := SoC(t) := SoC_0 + \frac{1}{C_{\text{Nenn}}} \int_0^t i(\tau) d\tau, \quad \tau \in [0, t]. \quad (2.1)$$

In Laderichtung wird der Batteriestrom  $i(\tau)$  als positiv und in Entladerichtung als negativ definiert. Somit nimmt der *SoC* bei positivem Stromfluss zu und verringert sich bei negativem Stromfluss. In der Regel liegt der Wertebereich des *SoC* zwischen 0 und 1:  $SoC \in [0, 1]$ . [14] schlägt eine Skalierung des Wertes in Prozent vor.

**Gesundheitszustand** Der State of Health (kurz: *SoH*) beschreibt den Gesundheitszustand der Batterie und kann mit Hilfe zweier Größen bewertet werden. Zum einen unter Berücksichtigung des Innenwiderstands (*State of Health - Resistance*, kurz:  $SoH_R$ ). Zum anderen durch eine kapazitive Betrachtungsweise (*State of Health - Capacity*, kurz:  $SoH_C$ ).

Zur Berechnung des  $SoH_R$  werden die Innenwiderstände zum Zeitpunkt des *Begin of Life*<sup>2</sup> (kurz: BoL)  $R_{\text{BoL}}$  sowie des *End of Life* (kurz: EoL)  $R_{\text{EoL}}$  der Batterie und der aktuelle Innenwiderstand  $R_{\text{Akt}}$  benötigt. Die Berechnungsvorschrift kann Gleichung 2.2 entnommen werden:

---

<sup>2</sup> **Definition:** Anfangszustand der Batterie bezüglich Innenwiderstand und Kapazität

$$SoH_R = \frac{R_{EoL} - R_{Akt}}{R_{EoL} - R_{BoL}} \cdot 100\% . \quad (2.2)$$

Der EoL einer Batterie ist erreicht, sobald bestimmte Grenzwerte der Alterungskenngrößen unterschritten werden [15]. Für Fahrzeuganwendungen ist dieser Wert auf eine Restkapazität von 80 % der Nennkapazität festgelegt [16]. Bei Unterschreiten dieses Grenzwertes wird dem Fahrzeugnutzer ein Auswechseln der Batterie dringend empfohlen, um Sicherheit, Effizienz und Leistungsfähigkeit des Fahrzeugs zu gewährleisten.

Eine Möglichkeit zur Beurteilung des Gesundheitszustandes der Batterie bietet der  $SoH_C$ , der auf der kapazitiven Alterung basiert. Diese Größe lässt sich aus dem Quotienten der aktuellen Batteriekapazität  $C_{Akt}$  und der Nennkapazität  $C_{Nenn}$  ermitteln:

$$SoH_C = \frac{C_{Akt}}{C_{Nenn}} \cdot 100\% . \quad (2.3)$$

Für die Berechnung im BMS gibt es verschiedene Verfahren. Neben elektrischen, physikalischen und elektrochemischen Modellen rücken aufgrund steigender Rechenperformance auch zunehmend datengetriebene Ansätze in den Fokus der Anwendung. Auch in dieser Arbeit liegt das Augenmerk ausschließlich auf rein datenbasierten Ansätzen. Auf eine tiefergehende Betrachtung bezüglich anderer Methodiken wird im Rahmen dieser Arbeit verzichtet und auf [17] verwiesen.

### 2.1.3 Alterungssmechanismen

Aufgrund normativer Forderungen müssen Batterien so ausgelegt werden, dass die spezifizierte Kapazität und der spezifizierte Innenwiderstand bis

zum Ende der Lebenszeit garantiert werden können [15]. Aufgrund der elektrochemischen Materialeigenschaften lässt sich während des Lebenszyklus der Zelle eine Degradation feststellen. Diese Alterung kann auf zwei Arten erfolgen. Es bietet sich an eine Unterscheidung zwischen kalendarischen und zyklischen Alterungsphänomenen vorzunehmen.

**Kalendarische Alterung** Die kalendarische Alterung ist der irreversible Anteil der verlorenen Kapazität während der Lagerung. Diese nutzungsabhängigen Alterungsprozesse können als die funktionsbedingte Degradation durch die Energiespeicherung bezeichnet werden [18]. Sie stehen in Zusammenhang mit einer Elektrolytreduktion und -oxidation sowie der Bildung von Oberflächenfilmen auf den Aktivmaterialien [19]. Bestimmte Einflussfaktoren, welche spezielle Zustände oder Belastungen herbeiführen, begünstigen diese Degradationsphänomene. [20] zeigen, dass die Alterung bei höheren Temperaturen schneller voranschreitet. Aus diesem Grund wird durch entsprechende Maßnahmen versucht, die Batterietemperatur  $T_{B_{max}}$  unter 60 °C zu halten. Der Einfluss der bereits genannten sowie weiterer Faktoren auf die Alterung wird in Tabelle 2.1 zusammenfassend dargestellt.

**Zyklische Alterung** Zyklische Alterungsprozesse sind Folge von dynamischen Prozessen innerhalb der Zelle. Die Komplexität jener Prozesse ist deutlich höher als die der kalendarischen Alterung. Sie treten vor allem im Zusammenhang mit Lade- bzw. Entladevorgängen der Batterie auf. Dies ist eine direkte Folge des *SoC*, der Nutzungsart, der Temperaturbedingungen und der Stromanforderung an die Batterie [21]. Eine weitere in der Literatur wiederkehrende Einflussgröße ist der  $\Delta SoC$ , der die Veränderung des

Ladezustands während des Zyklierens darstellt. [22] belegen in einer Studie, dass hohe  $\Delta SoC$ -Werte beim Entladen mit einem Verlust der Batterieperformance einhergehen, unabhängig von anderen Bedingungen. Dieses Phänomen ist hauptsächlich auf die Degradation der Kathode und die Entwicklung der in Abschnitt 2.1.3 erläuterten *Solid Electrolyte Interphase* (kurz: SEI) zurückzuführen.

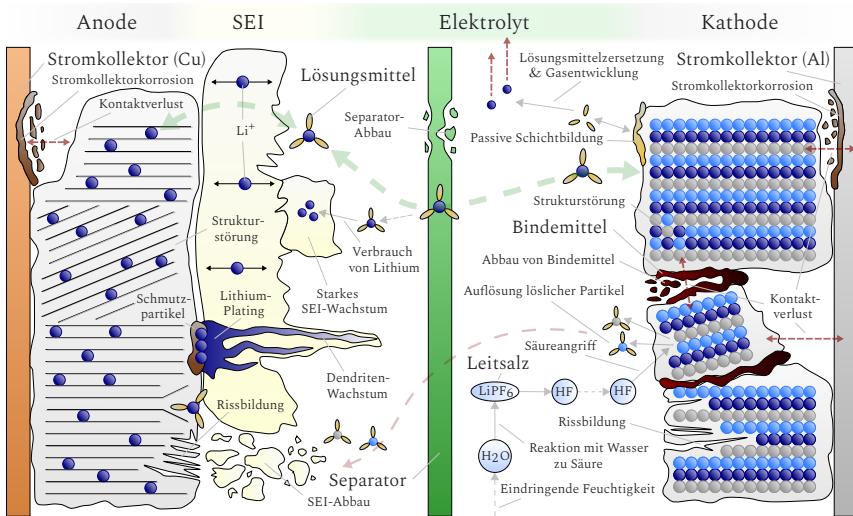
Ein weiterer Faktor ist der *Depth of Discharge* (dt.: Entladetiefe, kurz: *DoD*), welcher den Anteil der entnommenen Kapazität relativ zur Gesamtkapazität beschreibt. Er gibt demnach an, wie stark eine Batterie entladen wurde. Hohe DoD-Werte gehen mit einer beschleunigten Kapazitätsdegradation einher, da tiefere Entladungen höhere mechanische Belastungen auf die Elektrodenmaterialien ausüben sowie zu einer verstärkten SEI-Bildung führen [23]. Zudem belegen Simulationsergebnisse von [24], dass hohe DoD-Werte, insbesondere in Verbindung mit großen Entladeströmen, die Alterung der Batterie erheblich beschleunigen. Des Weiteren schlagen [25] vor, Lade- bzw. Entladeraten für Automobilanwendungen auf Werte kleiner  $2\text{ }C$  zu begrenzen, um die Alterung zu minimieren. Ebenso kann sich ein zu großes Betriebsspannungsfenster  $\Delta U$  zu einer Beschleunigung der Alterungsprozesse führen. In Abhängigkeit der Zellchemie wird generell empfohlen den Betrieb der Zelle im Bereich  $2,5\text{ V} \leq \Delta U \leq 4,2\text{ V}$  sicherzustellen [26].

Da jede Zyklierung der Zelle bzw. der Batterie auch kalendarische Alterung hervorruft, sei an dieser Stelle angemerkt, dass beide Alterungsformen parallel ablaufen und sich gegenseitig beeinflussen [27]. So werden auch im weiteren Verlauf der Arbeit beide Alterungstypen als überlagerter Prozess betrachtet.

**Tabelle 2.1:** Maßgebliche Einflussgrößen auf die kalendarische und zyklische Alterung mit:  
 $SoC(t_L)$  = Lager- $SoC$  in Abhängigkeit der Lagerzeit  $t_L$ ,  $DoD$  = Entladetiefe,  
 $I_L$  = Ladeströme,  $T_L$  = Lagertemperatur,  $T_B$  = Batteritemperatur,  $\Delta U$  =  
Betriebsspannungsfenster, C-Rate = Lade-/Entladerate. Große Werte werden  
mit  $\uparrow$ , kleine mit  $\downarrow$  gekennzeichnet.

Alterungsart	$\uparrow SoC(t_L)$	$\uparrow DoD$	$\uparrow I_L$	$\uparrow T_L$	$\downarrow T_B$	$\uparrow \Delta U$	$\uparrow$ C-Rate
Kalendarisch	•			•		•	
Zyklisch		•	•		•		•

Grundsätzlich wirken sich die beschriebenen Prozesse hinsichtlich der Zelleigenschaften auf zwei Arten aus. Einerseits steigt der Innenwiderstand, was in direktem Zusammenhang mit dem  $SoH_R$  steht. Andererseits vermindert sich die Kapazität der Zelle, was sich in einer Reduktion des  $SoH_C$  äußert. Einen Überblick der damit einhergehenden Degradationsmechanismen innerhalb einer Zelle ist im Folgenden durch Abbildung 2.3 illustriert. Auf die hervorgehobenen Mechanismen wird in den folgenden Abschnitten näher eingegangen. Neben der Korrosion an den Stromkollektoren sowie an den Elektroden lassen sich weitere Mechanismen identifizieren. Durch eine Volumenänderung des Graphits kann es zu Strukturstörungen und Rissbildungen im Material kommen. Des Weiteren können im Laufe der Batterienutzung Dendritenwachstum, SEI und eine Gasentwicklung auftreten. Auf zwei wichtige Verlustprozesse soll im Folgenden näher eingegangen werden.



**Abbildung 2.3:** Wesentliche Alterungsmechanismen innerhalb einer Li-Ionen-Zelle. Eigene Darstellung nach [27].

**Solid Electrolyte Interface** Während der Zyklierung der Zelle kommt es zu chemischen Zersetzungreaktionen des Elektrolyten an der Anodenoberfläche. Die thermodynamische Instabilität des Elektrolyts gegenüber lithiiertem Graphit führt zur einer Zersetzung der Lösungsmittelbestandteile des Elektrolyten. Es entsteht eine Passivierungsschicht auf der Oberfläche der Anode. Diese Schutzschicht wird als *Solid Electrolyte Interface* bezeichnet. Ihre Ausbildung führt dazu, dass sowohl Lade- als auch Entladeprozess während den ersten Zyklierungen bei Graphit nicht vollständig reversibel ablaufen, da besonders bei der ersten Reduktion des Graphits Elektrolytkomponenten reduziert werden und aktives Lithium verbraucht wird [28]. Der passivierte Bereich der Anodenoberfläche kann folglich nicht mehr vollständig an den elektrochemischen Prozessen der Batterie teilnehmen. Dadurch wird die Lebensdauer der Zelle signifikant beeinflusst.

Anzumerken sei an dieser Stelle, dass die SEI für Zyklierung zwingend erforderlich ist. Würde keine SEI entstehen, kann es zu einer ungehemmten Interkalation der Elektrolytbestandteile in die Graphitschichten kommen. Dies hätte eine erhöhte Volumenänderung der Zelle zur Folge, einhergehend mit einer mechanischen Zerstörung der Zelle bei zunehmender Zyklenzahl. Durch eine angepasste Zusammensetzung des Elektrolyts kann die Stabilität der Zelle, bei gezielter Beeinflussung der SEI, verbessert werden [19]. Ziel ist es eine Permeabilität der SEI für Lithium-Ionen, jedoch nicht für andere Elektrolytkomponenten oder Elektronen anzustreben [11]. So kann nicht nur die Elektrode vor Korrosion, sondern auch die Komponenten des Elektrolyts vor fortschreitender Zersetzung geschützt werden.

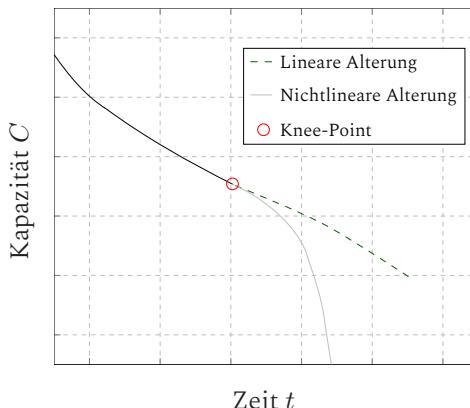
**Lithium-Plating** *Lithium-Plating* beschreibt den Prozess, bei welchem sich die im Elektrolyt gelösten  $\text{Li}^+$ -Ionen als metallisches Lithium an der Oberfläche des Aktivmaterials ablagern. Charakteristischerweise findet das Abscheiden durch eine Kombination aus hohen Ladeströmen und niedrigen Temperaturen statt [29]. Die Interkalationskinetik der  $\text{Li}^+$ -Ionen ins Anodenmaterial ist zu gering, wodurch eine Lithiumdeposition auf der Oberfläche des Graphits begünstigt wird [30, 31]. Ein kontinuierliches Wachstum dieser Ablagerungen und die Bildung von Dendriten können den Separator zerstören und schließlich einen internen Kurzschluss der Zelle verursachen. Häufig ist dies der Grund für einen *Thermal Runaway*<sup>3</sup>

Im Folgenden soll die Auswirkung der beschriebenen Degradationsmechanismen auf die Batterielebensdauer aufgezeigt werden. In Abbildung 2.4

---

<sup>3</sup> **Definition:** Unkontrollierter Temperaturanstieg, der durch eine sich selbst verstärkende exotherme Reaktion verursacht wird. Dies kann Schäden an der Batterie hervorrufen, die zu einem Brand oder einer Explosion führen [32].

sind zur Verdeutlichung zwei charakteristische Alterungsverläufe schematisch dargestellt. Diese entstammen unterschiedlich zyklierten Batteriezellen. In beiden Fällen nimmt die Kapazität mit steigender Zyklenzahl ab. Auffällig ist, dass sich die Funktionsverläufe ab einem bestimmten Zeitpunkt (*Knee-Point*) stark unterscheiden. Während die in grau dargestellte Zelle exponentiell zu altern beginnt und innerhalb weniger Zyklen stark an Kapazität verliert, degradiert die in grün dargestellte Zelle linear. Durch das Zusammenspiel der Alterungsprozesse und -mechanismen kann die Alterung nichtlinear verlaufen (vgl. Abbildung 2.4). Unter normalen Betriebsbedingungen stellt das SEI-Wachstum den Hauptalterungsmechanismus dar. Dieser resultiert in einem linearen Kapazitätsverlust im Laufe der Zeit [33]. Ein beschleunigter Kapazitätsabfall hingegen kann primär durch die Ablagerung von metallischem Lithium auf der Graphitanode beobachtet werden [25].



**Abbildung 2.4:** Charakteristischer Verlauf einer Alterungskennlinie in eigener Darstellung nach [34].

Wichtig zu verstehen ist, dass durch verschiedenste Bedingungen und Anspruchungen, wie bspw. dem Laden oder Entladen, Batteriezellen auf unterschiedliche Art und Weise altern können. Für ein tiefergehendes Verständnis hinsichtlich der Ursachen sowie zugrundeliegenden Mechanismen sei an dieser Stelle explizit auf [27] verwiesen.

## 2.2 Maschinelles Lernen

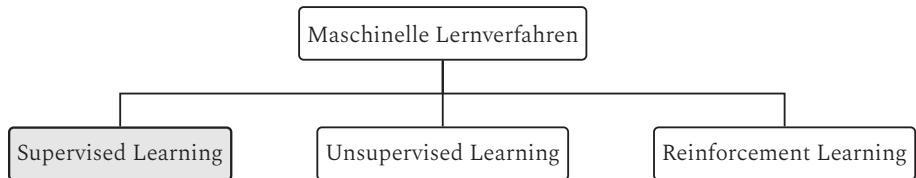
Der Bereich Machine Learning (dt.: Maschinelles Lernen, kurz: ML) ist ein Teilgebiet der Artificial Intelligence (dt.: künstliche Intelligenz) und befasst sich mit der Entwicklung selbstlernender Algorithmen. Diese sind dazu in der Lage, innerhalb großer Datenmengen, Zusammenhänge bzw. Muster zu erkennen und daraus zu lernen. Vereinfacht handelt es sich somit um eine künstliche Generierung von Wissen aus Erfahrung. Mitchell schlägt die Definition

„A computer program is said to learn from experience  $E$  with respect to some class of tasks  $T$  and performance measure  $P$ , if its performance at tasks in  $T$ , as measured by  $P$ , improves with experience  $E$ .“ [35]

vor. Für das weitere Verständnis wird nachfolgend auf die zuvor gewählten Formulierungen Erfahrung  $E$ , Aufgabe  $T$  und Leistungsmaß  $P$  eingegangen. Die folgenden Abschnitte orientieren sich dabei an [35] sowie [36].

**Erfahrung** Die Erfahrung  $E$  beschreibt die Datenbasis, aus der ein Algorithmus lernt. Sie umfasst die Menge an Eingabe- und Ausgabedaten, mit denen das Modell trainiert wird. Das Lernen selbst kann als eine Suche von

Zusammenhängen innerhalb dieser Datenbasis verstanden werden. Basierend auf der Art der verfügbaren Daten und den Zielsetzungen des Modells lässt sich eine Unterscheidung statistischer Lernverfahren wie folgt treffen:



**Abbildung 2.5:** Einteilung der maschinellen Lernverfahren.

Beim *Supervised Learning* (dt.: Überwachtes Lernen) wird ein Modell mittels gekennzeichneter Trainingsdaten gelernt. Dies bedeutet, dass jedem Datensample<sup>1</sup>  $\mathbf{x}$  (z. B. gemessene Eingangs-/Zustandsgrößen) ein Vergleichswert – oder auch *Label* – zugewiesen wird, das dem tatsächlichen Ergebnis  $y$  entspricht. Es wird demzufolge oft von *gelabelten* Daten gesprochen. Hauptziel dieses Lernverfahrens ist es nun die Differenz zwischen dem Referenzwert  $y$  und dem geschätzten Wert  $\hat{y}$  zu minimieren, um die bedingte Wahrscheinlichkeitsverteilung

$$\hat{Y} = P(Y \mid \mathbf{X}) \quad (2.4)$$

möglichst gut zu lernen. Das überwachte Lernen ist bis heute die am häufigsten verwendete Form der maschinellen Lernverfahren. Auch in dieser Arbeit findet die Herangehensweise an die vorliegende Problemstellung

<sup>1</sup> dt.: Eingabe – Ein Datenpunkt, der von einem Modell verarbeitet wird

mit dieser Methode statt. Für ein tiefergehendes Verständnis der zwei verbleibenden Methoden sei explizit auf [37] und [38] verwiesen.

**Aufgabe** Eine Aufgabe im Zusammenhang mit maschinellen Lernmethoden wird als eine bestimmte Problemstellung verstanden, welche sich mit lernenden Algorithmen lösen lässt. Typischerweise kann diese durch die Verarbeitung eines Eingangsvektors  $\mathbf{x} \in \mathbb{R}^n$  beschrieben werden, dessen Einträge  $x_i$  als Merkmale bezeichnet werden. Neben Klassifikations- und Regressionsproblemen existiert noch eine Vielzahl an anderen zu erlernenden Aufgaben, wobei an dieser Stelle auf eine detaillierte Unterteilung verzichtet und speziell auf die zwei zuvor genannten näher eingegangen wird.

Grundprinzip der Klassifikation ist es, Eingangsdaten einer entsprechenden Klasse  $k$  zuzuordnen. Für das Lösen dieser Aufgabe muss eine Funktion

$$f : \mathbb{R}^n \mapsto \{1, \dots, k\} \quad k \in \mathbb{N} \quad (2.5)$$

gefunden werden, deren Funktionswert  $\hat{y} = f(\mathbf{x})$  eine bestimmte Klasse beschreibt, die dem Eingabevektor zugewiesen werden kann. Ein klassisches Anwendungsbeispiel findet sich im Bereich des autonomen Fahrens wieder. Dabei steht die automatische Erkennung von Objekten wie Straßen- schildern, Fahrzeugen oder Fußgängern im Vordergrund.

Im Gegensatz dazu, wird bei Regressionsmodellen ein skalarer Wert anhand bestimmter Eingabegrößen prädiziert. Es wird folglich eine Funktion  $f : \mathbb{R}^n \mapsto \mathbb{R}$  als Ausgangsgröße gesucht, die den Eingangsvektor auf einen

kontinuierlichen Wertebereich abbildet. Neben zahlreichen, verschiedenartigen Ansätzen ist die lineare Regression die wohl bekannteste und zugleich einfachste Form dieser Aufgabenkategorie. Hier besteht die Zielsetzung darin, anhand eines Eingangsvektors  $\mathbf{x}$  der Dimension  $n$  einen skalaren Schätzwert, gegeben durch:

$$\hat{y} = \mathbf{w}^\top \mathbf{x} \quad (2.6)$$

zu prädizieren. Der Vektor  $\mathbf{w} \in \mathbb{R}^n$ , dessen Komponenten  $w_i$  als Gewichte beschrieben werden, sind durch eine Optimierung anzupassen, um aussagekräftige Ergebnisse gekennzeichnet durch eine hohe Prognosegüte zu ermitteln.

**Leistungsmaß** Um die Güte eines maschinellen Lernalgorithmus zu bewerten, muss eine Messgröße eingeführt werden, welche die Leistungsfähigkeit  $P$  des Modells beschreibt. Diese ist abhängig von der gewählten Aufgabe  $T$ . Bei Klassifikationsproblemen wird üblicherweise die *Accuracy* (dt.: Genauigkeit) gemessen. Diese gibt an, welche Eingaben das Modell richtig zuordnen kann. Für Regressionsprobleme hat sich neben dem *Mean Absolute Error* (dt.: mittlerer absoluter Fehler, kurz: MAE) und dem *Mean Squared Error* (dt.: mittlerer quadratischer Fehler, kurz: MSE) auch der *Root Mean Squared Error* (dt.: Wurzel der mittleren Fehlerquadratsumme, kurz: RMSE) etabliert. Diese skalaren Maßzahlen zur Beurteilung der Prognosegüte berechnen sich wie folgt:

$$MAE = \frac{1}{n} \sum_{i=1}^n |\hat{y} - y|_i \quad (2.7)$$

$$MSE = \frac{1}{n} \sum_{i=1}^n (\hat{y} - y)_i^2 \quad (2.8)$$

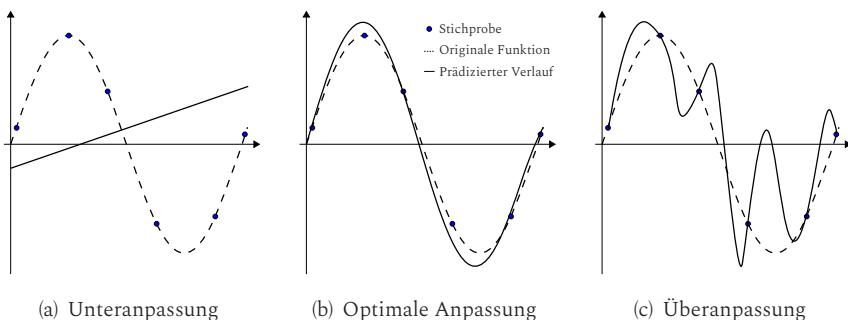
$$RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^n (\hat{y} - y)_i^2} . \quad (2.9)$$

Kleine Werte korrelieren dabei mit einer hohen Güte des Modells. Des Weiteren muss an dieser Stelle erwähnt werden, dass eine klare Unterteilung der zugrundeliegenden Daten in die folgenden drei Mengen erforderlich ist: Trainings-, Validierungs- und Testdaten. Anhand der Trainingsdaten wird der Algorithmus gelernt. Eine Optimierung erfolgt demnach allein auf Basis des Leistungsmaßes der Trainingsdaten. Diese werden auch als *Parameter* bezeichnet und müssen eindeutig von den sogenannten *Hyperparametern* unterschieden werden, welche vom Anwender vorgegeben werden. Für deren Abstimmung auf optimale Werte wird lediglich das Fehlermaß auf die Validierungsdaten herangezogen.

Durch die Testdaten wird schlussendlich evaluiert, wie der Algorithmus auf neue Daten reagiert, um beurteilen zu können wie gut das Modell generalisierbar ist. Ziel ist es, den Fehler sowohl für das Test- als auch das Trainingsset zu minimieren. Zwei Herausforderungen, die sich in diesem Zusammenhang ergeben, sind das *Underfitting* (dt.: Unteranpassung) und das *Overfitting* (dt.: Überanpassung). Genauer sollen diese Begriffe im Folgenden Unterabschnitt 2.2.1 erläutert werden.

## 2.2.1 Generalisierungsfähigkeit und Regularisierung

Die Generalisierungsfähigkeit beschreibt die Eignung, vorhandene Strukturen oder Muster in Daten zu erkennen und zu erlernen. Wie diese Zusammenhänge abgebildet werden, soll nachfolgend am Beispiel einer einfachen Sinusfunktion in Abbildung 2.6 verdeutlicht werden. In Abbildung 2.6 (a) ist erkennbar, dass der Algorithmus nicht in der Lage ist, den sinusförmigen Zusammenhang wiederzugeben. Bedingt durch einen zu einfachen Aufbau werden Relationen nicht erkannt und lediglich lineares Verhalten abgebildet. In diesem Fall wird von Unteranpassung gesprochen. Abhilfe schafft eine Steigerung der Modellkomplexität, weshalb sie kein grundlegendes Problem darstellt.



**Abbildung 2.6:** Schematische Darstellung der Generalisierungsfähigkeit. Ein optimal angepasstes Modell reproduziert die Trainingsdaten und kann auf neue Daten verallgemeinert werden, während ein überangepasstes Modell nur die Trainingsdaten reproduziert.

Abbildung 2.6 (b) zeigt das gewünschte Verhalten. Das Modell erkennt eindeutig die Funktionsstruktur und bildet sie ab. Für ein Gelingen muss die entwickelte Modellarchitektur über eine ausreichende Komplexität verfügen.

Ein schwer zu beherrschendes Problem stellt hingegen die Überanpassung dar. Anstatt grundlegende Zusammenhänge zu erkennen, passt sich das Modell zu stark an die Trainingsdaten an und lernt dabei auch spezifische, nicht repräsentative Eigenschaften. Das Modell lernt während des Trainingsprozesses Daten sozusagen auswendig, worunter die Generalisierungsfähigkeit leidet (siehe Abbildung 2.6(c)). Zudem steigt die Varianz der Vorhersage auf den Testdaten. Ob Überanpassung auftritt oder nicht, ist schwer abzuschätzen. Ein Faktor, der dieses Verhalten begünstigt, ist eine zu hohe Anzahl an Parametern für die zugrundeliegende Datenbasis. Die Lösung dieses Problems besteht beispielsweise in der Vergrößerung des Trainingsdatensatzes oder einer Vereinfachung der Modellstruktur.

Vor Trainingsbeginn werden die Gewichtungswerte zufällig mit kleinen reellen Werten initialisiert, da sonst aufgrund eines gleichbleibenden Fehler-sIGNALS kein Lernfortschritt eintreten kann [39]. [40] beschreiben dieses Problem als *Symmetry Breaking* (dt.: Brechen der Symmetrie). Die Initialisierung geschieht vorwiegend mit einer um Faktor 0,01 - 0,1 skalierten Standard-normalverteilung [41].

Um das spezielle Problem der Überanpassung zu vermeiden, sind laut [42] ebenfalls kleine Werte für die Gewichtungen anzustreben. Mittlerweile haben sich auch andere Methoden durchgesetzt, welche die Komplexität des Algorithmus beschränken. Wie stark die Werte dabei kleingehalten werden, beschreibt der empirisch anzupassende Hyperparameter  $\lambda$ , der als *Weight Decay* bezeichnet wird (s. Gleichung 2.10). Daraus ergibt sich eine gleichmäßige Verteilung der Gewichte, wodurch die Einführung von Nichtlinearitäten begrenzt wird. Diese Bestrafung hoher Gewichte wird als Regularisierung bezeichnet und durch Hinzufügen eines Strafterms  $J(\mathbf{W})$  zur Verlustfunktion umgesetzt. Das Leistungsmaß (hier: *MSE*) kann dem-nach folgendermaßen notiert werden:

$$P = \frac{1}{n} \sum_{i=1}^n (\hat{y} - y)_i^2 + \lambda J(\mathbf{W}) . \quad (2.10)$$

Die Gewichtungsmatrix  $\mathbf{W}$  umfasst dabei die Menge aller Gewichte. Für die Straffunktion sind zwei Varianten gebräuchlich, die L<sub>1</sub>- bzw. L<sub>2</sub>-Regularisierung. Erstere verhält sich proportional zum Absolutwert der Gewichtungskoeffizienten:  $J_{L_1}(\mathbf{W}) = \sum |w|$ . Zweitere proportional zum Quadrat der Werte der Gewichtungskoeffizienten:  $J_{L_2}(\mathbf{W}) = \sum |w^2|$ .

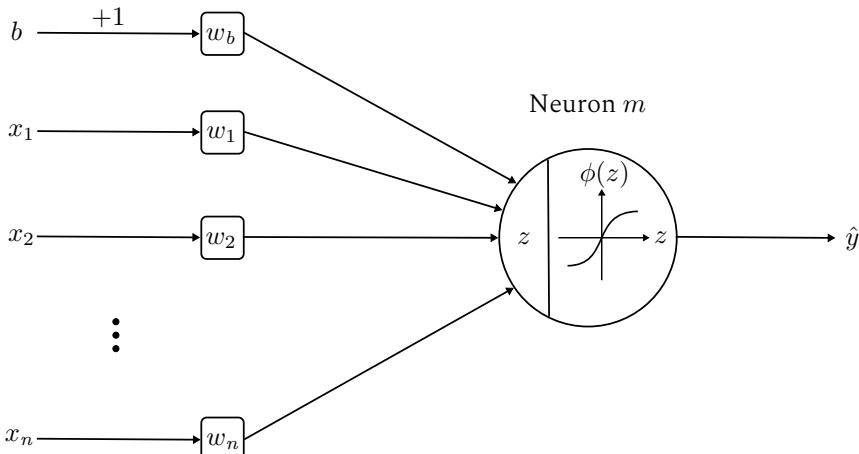
Im Rahmen dieser Arbeit erfolgt eine Herangehensweise an die Aufgabenstellung in Form von verschiedenen ML-Ansätzen. Dabei werden Künstliche Neuronale Netze, Support Vector Machines und xGBoost-Modelle gewählt, da sie unterschiedliche Stärken bieten: KNN für die Modellierung komplexer nichtlinearer Zusammenhänge, SVM für robuste Schätzungen in hochdimensionalen Räumen und xGBoost in Form von effizienten, leistungsstarken Entscheidungsbäumen. Die Eignung dieser Methoden hinsichtlich des Alterungsverhaltens von Lithium-Ionen-Batterien wird im Rahmen der Literaturrecherche (s. Kapitel 3) detailliert herausgearbeitet. In den folgenden Abschnitten wird nun gezielt auf diese Konzepte des maschinellen Lernens eingegangen.

## 2.2.2 Künstliche Neuronale Netze

Künstliche Neuronale Netze (kurz: KNN) können im Sinne der Graphentheorie als gerichtete Graphen aufgefasst werden, die aus einer bestimmten Menge an Neuronen und einer Vielzahl an gerichteten Kanten – den Verbindungen zwischen den neuronalen Einheiten – bestehen [43]. Diese Kernkomponenten bilden die Basis für nahezu alle Netzwerktypen. Hierbei

kann wiederum eine Unterteilung in statische und dynamische Elemente getroffen werden. Während erstgenannte Auskunft über die räumliche Struktur geben, beschreiben dynamische Komponenten die Verarbeitung von Informationen in KNNs [44]. Ziel ist es, durch mathematische Modelle die Informationsverarbeitung des menschlichen Gehirns so nachzubilden, dass komplexe Zusammenhänge durch ein Zusammenwirken vieler einfacher Einheiten erkennbar sind. Für das weitere Verständnis dieser Arbeit werden nachfolgend die wichtigsten Begrifflichkeiten eingeführt.

**Künstliches Neuron** Einzelne künstliche Neuronen bilden die Grundlage jedes Neuronalen Netzes. Deren einfachster Aufbau kann folgender Abbildung 2.7 entnommen werden.



**Abbildung 2.7:** Illustrierte Darstellung des Aufbaus eines künstlichen Neurons mit  $x_1 \dots n$  = Eingangsrößen,  $w_1 \dots n$  = Gewichte der zu verarbeitenden Daten,  $b$  = Bias-Wert,  $w_b$  = Bias-Gewicht,  $z$  = innerer Zustand des Neurons,  $\phi(z)$  = Aktivierungsfunktion,  $\hat{y}$  = geschätzter Ausgangswert.

Jedes Neuron  $m$  besitzt  $n$  Eingänge über welche es Eingabeinformationen anderer Neuronen erhält. Diese zu verarbeitenden Daten werden in einem  $n$ -dimensionalen Vektor  $\mathbf{x}$  zusammengefasst und mit einem Gewichtungsvektor  $\mathbf{w}$  der gleichen Dimension skalar multipliziert. Eine Regulierung der Stärke einzelner Eingaben wird über den jeweiligen Eintrag des Vektors  $\mathbf{w}$  beschrieben, welche als Gewichtungen bezeichnet werden. Anschließend wird ein sogenannter *Bias*  $b$  addiert, der einen konstanten Wert beinhaltet und sowohl positiv als auch negativ gewichtet sein kann. Grafisch erlaubt dieser Wert eine Verschiebung der Aktivierungsfunktion (s. Abschnitt 2.2.2) entlang der Abzisse. Um systematische Fehler auszugleichen und gleichzeitig erfolgreich zu lernen wird bei geringem Eingangssignal sichergestellt, dass das Neuron bei positivem Gewicht aktiv bleibt. Negative Gewichtungen hingegen sind gleichbedeutend mit einer Verharrung im inaktiven Zustand. Im Gegensatz zu gewöhnlichen Neuronen erhält das Bias-Neuron selbst kein Eingangssignal [45]. Um nun den Ausgangswert

$$\hat{y} = \phi(z) = \phi(\mathbf{w}^\top \mathbf{x} + b w_b) \quad (2.11)$$

zu erhalten, wird eine *Aktivierungsfunktion*  $\phi$  auf den inneren Zustand  $z$  des Neurons angewendet, wobei der Begriff Aktivierungsfunktion im Folgenden genauer erläutert wird.

**Aktivierungsfunktion** Für die Anwendung Neuronaler Netze ist vor allem die Art der Aktivierungsfunktion  $\phi$  entscheidend. Diese wird für gewöhnlich in einem 2D-Diagramm dargestellt, bei dem das Aktivitätslevel eines Neurons über den Netzeingang aufgetragen ist. Die Wahl von  $\phi$  trägt somit signifikant zum Leistungsmaß des Netzwerkes bei [46]. Die einzige Möglichkeit des Netzes, Nichtlinearitäten abzubilden, besteht durch die

Wahl der Aktivierungsfunktion. Jedoch sei an dieser Stelle darauf hingewiesen, dass im Hinblick auf die Optimierung, eine Differenzierung der Aktivitätsfunktionen einzelner Neuronen gewährleistet sein muss [47]. Drei typische Beispiele nichtlinearer Aktivierungsfunktionen – die auch im Rahmen dieser Arbeit Verwendung finden – werden im Folgenden kurz erläutert. Die nach [48] eingeführte *Rectified Linear Unit* (kurz: ReLU) ist die am häufigsten verwendete Funktion im Bezug auf Deep Learning Anwendungen [49] und folgt der Funktionsvorschrift:

$$\phi(z) = \text{ReLU}(z) = \max(0, z) . \quad (2.12)$$

Abbildung 2.8 zeigt, dass sie zwar  $\mathcal{C}^{(1)}$ -stetig, jedoch bei  $z = 0$  nicht differenzierbar ist. Zudem kann abschnittsweise ein konstanter, von Null abweichender Gradient nachgewiesen werden. Dies hat zum Vorteil, dass er bei aktivierte Neuronen rückpropagiert werden kann, wohingegen er bei nicht aktivierte Neuronen Null entspricht. [50] weist nach, dass die Möglichkeit besteht, tiefe Neuronale Netze mittels der ReLU-Funktion zu trainieren. Sowohl die logistische Sigmoidfunktion  $\sigma$ , definiert durch

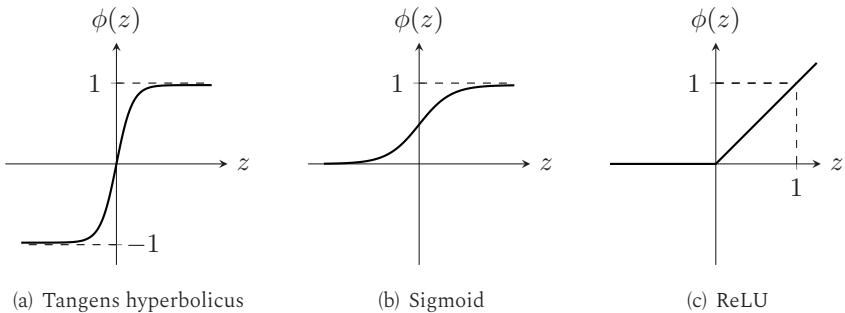
$$\phi(z) = \sigma(z) = \frac{1}{1 + e^{-z}} , \quad (2.13)$$

als auch der Tangens hyperbolicus

$$\phi(z) = \tanh(z) = \frac{e^z - e^{-z}}{e^z + e^{-z}} \quad (2.14)$$

zeigen für große Argumente der Ableitung asymptotisches Verhalten gegen Null. Dadurch wird die Dauer des Lernprozesses negativ beeinflusst.

Der qualitative Verlauf beider Funktionen kann nachfolgend Abbildung 2.8 entnommen werden.



**Abbildung 2.8:** Funktionsverläufe für nichtlineare Aktivierungsfunktionen.

Der Tangens hyperbolicus weist nach [51] hinsichtlich der Lernverfahren jedoch bessere Ergebnisse auf. Dies resultiert zum einen aus dem Wertebereich zwischen  $\{-1, 1\}$ , zum anderen aus dem größeren Gradienten. Je nach Aufgabentyp werden unterschiedliche Aktivierungen der letzten Schicht verwendet. Für den Fall einer Regression sind die Ausgangsneuronen ausgenommen von einer nichtlinearen Aktivierungsfunktion. Grund hierfür ist, dass es durch eine lineare Aktivierung möglich ist einen kontinuierlichen Wertebereich darzustellen [52].

**Vorwärtsgerichtete Neuronale Netze** Vorwärtsgerichtete Neuronale Netze (engl.: Feed Forward Neural Network, kurz: FFNN) lassen sich in ein- sowie mehrschichtige Netztopologien unterteilen, wobei der Fokus im Rahmen dieser Arbeit auf Architekturen mit mehreren Schichten liegt. Diese weisen neben einer Eingabe- und Ausgabeschicht zusätzlich verdeckte Schichten auf. Die Verknüpfung aller Neuronen einer vorherigen

mit der darauffolgenden Schicht ist charakteristisch für derartige Netzstrukturen. Die Eingabeschicht umfasst dabei die Menge aller Eingangsneuronen des Netzes, an welche einzelne Merkmale der unterschiedlichen Samples übergeben werden. Bezeichnend für die Neuronen dieser Schicht ist es, dass keine gerichteten Verbindungen zu diesen existiert. Aufbauend auf die Eingabeschicht folgen die verdeckten Schichten. Die Anzahl dieser Schichten ist neben der Menge an Neuronen variabel und muss entsprechend der Aufgabenstellung angepasst werden. [53] schlägt jedoch ein Verhältnis der Neuronenzahl von Eingabe- zu verdeckter Schicht von  $n_{\text{Eingang}}/n_{\text{verdeckt}} = 2/3$  vor.

Für ein besseres Verständnis ist in Abbildung 2.9 beispielhaft die Architektur eines dreischichtigen FFNN für eine Regression dargestellt. Für eine tiefergehende Beschreibung hinsichtlich der Netztopologie einer Klassifikation sei auf [54] verwiesen.

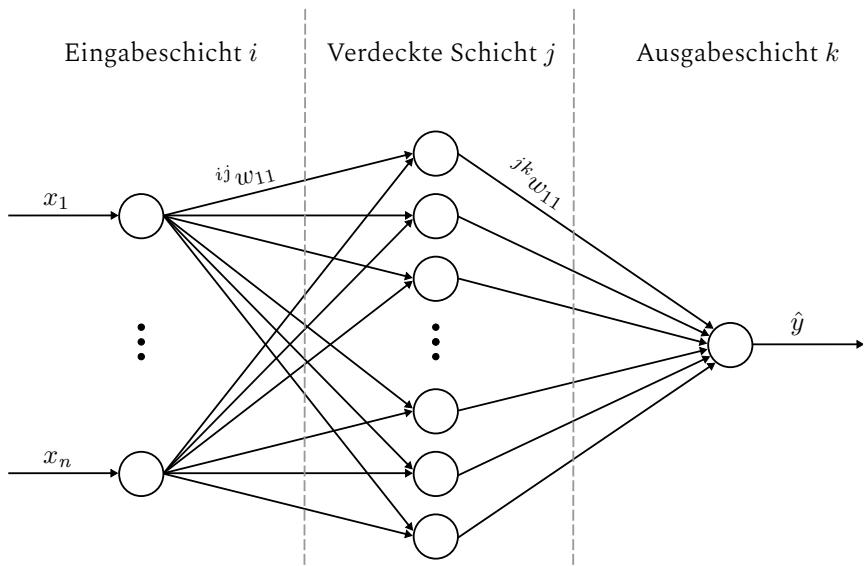
Die Aktivität jeder neuronalen Einheit (s. Abschnitt 2.2.2) wird ausgehend von der ersten verdeckten Schicht berechnet und über die Kantengewichtungen zur nächsthöheren weitergeleitet. Die Aktivierungen der einzelnen Neuronen der verdeckten Schicht werden in einem Vektor  $\mathbf{a}$  zusammengefasst:

$${}^j \mathbf{a} = \phi({}^j z) = \phi({}^{ij} \mathbf{W}^\top {}^i \mathbf{x} + {}^i \mathbf{b}) . \quad (2.15)$$

Im Sinne einer konsistenten Dokumentation im weiteren Verlauf dieser Arbeit soll die in Gleichung 2.15 eingeführte Indizierung Verwendung finden. Dabei wird durch die Gewichtsmatrizen  $\mathbf{W}$  die gewichtete Summe der Neuronenaktivierungen der vorherigen Schicht beschrieben. Die Eingabeschicht ist durch den Index  $i$ , die verdeckte Schicht durch  $j$  und die Ausgabeschicht durch  $k$  gekennzeichnet. Das Kantengewicht zwischen erstem

Neuron der Eingabeschicht  $i$  und erstem Neuron der verdeckten Schicht  $j$  wird beispielhaft durch  $^{ij}w_{11}$  beschrieben. Neuronen der Ausgabeschicht stellen mit ihren Ausgangswerten das Ergebnis der Verarbeitung dar.

Je nach Aufgabenstellung ist der Aufbau der letzten Schicht zu unterscheiden. Während für eine Klassifikation die Menge der Neuronen gleich der Anzahl der Labels entspricht, wird bei der Regression lediglich ein Ausgangsneuron verwendet.



**Abbildung 2.9:** Architektur eines dreischichtigen vorwärtsgerichteten Neuronalen Netzes für ein Regressionsproblem.

### 2.2.3 Support Vector Machines

*Support Vector Machines* (kurz: SVM) sind leistungsfähige ML-Algorithmen, die sowohl für Klassifizierungs- als auch für Regressionsprobleme verwendet werden. Sie sind den überwachten Lernmodellen zuzuordnen und zählen zu den am besten erforschten Algorithmen des maschinellen Lernens [55]. Ferner sind sie gegenüber anderen ML-Ansätzen in der Lage mit verhältnismäßig geringen Datenmengen zufriedenstellende Ergebnisse zu produzieren [56]. Grundgedanke einer SVM ist das Finden einer oder mehrerer „optimaler“ Hyperebenen im mehrdimensionalen Raum für die Lösung des gegebenen Lernproblems [57]. In Anlehnung an [56, 58, 59] soll im Folgenden die Funktionsweise einer SVM für den linearen Fall anhand eines Klassifikations- sowie Regressionsproblems mathematisch beschrieben werden.

**Klassifikation** Für den linearen Fall liegt die gesuchte Hyperebene auf dem Raum der Eingabedaten  $\mathbf{x}$ . So kann der Hypothesenraum als eine Teilmenge aller Hyperebenen der Form:

$$f(\mathbf{x}) = \langle \mathbf{w}, \mathbf{x} \rangle + b \quad (2.16)$$

beschrieben werden. Ausgehend von einem Datenset  $\mathcal{D}$  mit  $n$  Datenpunkten  $x_i$  und den entsprechenden Merkmalen  $y_i \in \{+1, -1\}$  ergibt sich die Minderheitsklasse  $\mathcal{C}^+$ , welche sich aus allen Datenpunkten mit positivem Merkmal  $|\mathcal{C}^+| = n^+$  zusammensetzt. Die restlichen Datenpunkte bilden eine entsprechende Mehrheitsklasse  $\mathcal{C}^-$  mit  $|\mathcal{C}^-| = n^-$  und  $n = n^+ + n^-$ . Jeder Trainingsdatenpunkt  $x_i$  kann als  $d$ -dimensionaler Vektor in  $\mathbb{R}^d$  interpretiert werden. Die SVM berechnet die dazu passenden Hyperebenen

der Dimension  $d - 1$ , welche beide Klassen voneinander trennen. Die optimale Trennebene ist diejenige mit dem größtmöglichen Abstand zwischen den beiden Klassen. Sie wird demnach auch als *Large-Margin-Klassifikator* bezeichnet. Formal wird dies durch die Lösung folgenden Optimierungsproblems erreicht:

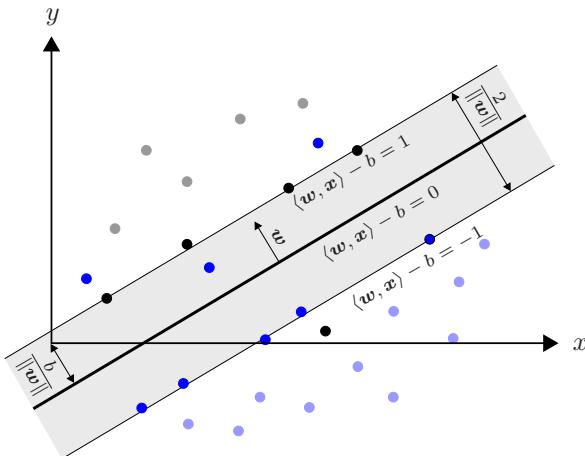
### SVM-Klassifikation

$$\text{minimiere} \quad \frac{1}{2} \|\mathbf{w}\|^2 + C \sum_{i=1}^n \xi_i \quad (2.17)$$

$$\text{unter Einhaltung von} \quad y_i(\langle \mathbf{w}, \mathbf{x}_i \rangle + b) \geq 1 - \xi_i, \quad \xi_i \geq 0. \quad (2.18)$$

Die Variablen  $\xi_i = \max(0, 1 - y_i(\langle \mathbf{w}, \mathbf{x}_i \rangle - b))$  werden auch als *Slackvariablen* (dt.: Schlupfvariablen) bezeichnet und erlauben es den Klassifikator flexibler zu gestalten. Die SVM kann dadurch auf bestimmte Fälle erweitert werden, in welchen die Daten nicht linear trennbar sind. Das heißt, dass keine Hyperebene existiert, die  $\mathbf{C}^+$  und  $\mathbf{C}^-$  voneinander trennt. So wird zwar eine Fehlklassifizierung zugelassen, gleichzeitig aber sichergestellt, dass jeder Datenpunkt  $x_i$  auf der richtigen Seite der Marge liegt. Die Marge bezeichnet dabei den Bereich zwischen den beiden Begrenzungsebenen, die parallel zur Trennebene verlaufen (s. Abbildung 2.10). Zugleich werden Fehlklassifikationen bestraft, da die Schlupfterme Teil der Minimierungsfunktion sind. Diese Erweiterung der SVM wird als *Soft Margin Formulation* bezeichnet. Der Parameter  $C > 0$  dient als Hyperparameter, welcher die Stärke der Regularisierung beim Optimieren der SVM-Verlustfunktion beeinflusst. Neue Datenpunkte werden dann mittels der Vorzeichenfunktion der Hyperebenengleichung  $h(\mathbf{x}) = \text{sign}(\langle \mathbf{w}, \mathbf{x} \rangle + b)$  zugeordnet. Vereinfacht ausgedrückt separiert die Hyperebene den Eingaberaum in zwei Teile und ordnet Datenpunkte oberhalb in  $\mathbf{C}^+$  und unterhalb in  $\mathbf{C}^-$  ein.

In Abbildung 2.10 sind für ein besseres Verständnis die beschriebenen Zusammenhänge für eine nicht-lineare Problemstellung illustriert. Insbesondere die Bedeutung der Schlupfvariablen wird an dieser Stelle deutlich. Diese ermöglichen es der SVM eine Trennebene für den nicht-linearen Fall zu finden. Zudem lässt sich  $\frac{2}{\|w\|}$  als geometrischer Abstand der Ränder interpretieren.



**Abbildung 2.10:** Eigene Darstellung der Prädiktion eines Support Vector Klassifikators nach [60].

Folglich muss  $\|w\|$  minimiert werden, um die bestmögliche Hyperebene zu ermitteln. Des Weiteren wird deutlich, dass die Hyperebene mit der maximalen Marge ausschließlich von denjenigen Datenpunkten bestimmt wird, die am nächsten an ihr liegen. Diese Datenpunkte werden auch als *Support Vectors* bezeichnet und haben den größten Einfluss auf die Bestimmung der Trennebene und damit auf die Vorhersage der SVM.

**Regression** Bei der Anwendung einer SVM für ein Regressionsproblem geht es um die Schätzung von reellwertigen Funktionen. Für eine Verallgemeinerung des SVM-Algorithmus auf den Regressionsfall, wird analog zur Klassifikation ein Soft-Margin erstellt. Für die Zielgrößen gilt nun  $y \in \mathbb{R}$ . Im Gegensatz zu Klassifikation bedarf es für den vorliegenden Fall einer Verlustfunktion die Fehler ignoriert, welche innerhalb eines bestimmten Abstandes zum tatsächlichen Wert liegen [57]. [61] führt dazu die sogenannte  $\varepsilon$ -insensitive loss function ein. Dadurch kann der Verlust, der durch die Vorhersage  $f(\mathbf{x})$  anstelle von  $y$  entsteht, berechnet werden:

$$|y - f(\mathbf{x})|_{\varepsilon} := \max(0, |y - f(\mathbf{x})| - \varepsilon). \quad (2.19)$$

Der qualitative Verlauf von  $\varepsilon$  kann nachfolgend Abbildung 2.11 entnommen werden. Um nun eine lineare Regression der Funktionsvorschrift aus Gleichung 2.16 zu prädizieren, muss folgendes beschränktes Optimierungsproblem betrachtet werden:

### SVM<sub>Regression</sub>

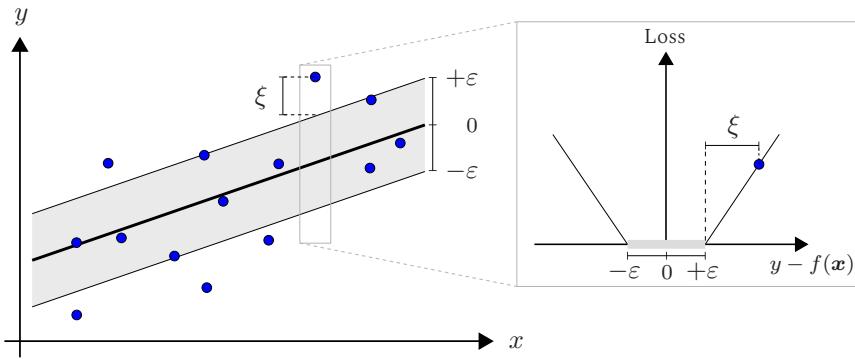
$$\text{minimiere} \quad \frac{1}{2} \|\mathbf{w}\|^2 + C \sum_{i=1}^n \overbrace{|y_i - f(\mathbf{x}_i)|_{\varepsilon}}^{(\xi_i + \hat{\xi}_i)} \quad (2.20)$$

$$\text{unter Einhaltung von} \quad f(\mathbf{x}_i) - y_i \leq \varepsilon + \xi_i \quad (2.21)$$

$$y_i - f(\mathbf{x}_i) \leq \varepsilon + \hat{\xi}_i, \quad \xi_i, \hat{\xi}_i \geq 0. \quad (2.22)$$

Es sei anzumerken, dass die Spektralnorm in Form des Terms  $\|\mathbf{w}\|^2$  gegenüber einer Klassifikation unverändert bleibt. Für den vorliegenden Regressionsfall müssen zwei Schlupfvariablen für die Fälle  $f(\mathbf{x}_i) - y_i > \varepsilon$  und  $y_i - f(\mathbf{x}_i) > \varepsilon$  eingeführt werden. Ersterer wird durch  $\xi_i$ , Letzterer mit  $\hat{\xi}_i$

beschrieben. Es ist zu beachten, dass gemäß Gleichung 2.21 und 2.22 jeder Fehler kleiner als  $\varepsilon$  kein  $\xi_i$  bzw.  $\hat{\xi}_i$  ungleich null erfordert. Somit geht dieser auch nicht in die Zielfunktion ein. Abbildung 2.11 veranschaulicht die beschriebenen Zusammenhänge. Für ein tiefergehendes mathematisches Verständnis sei an dieser Stelle explizit auf [58] verwiesen.



**Abbildung 2.11:** Eigene Darstellung der Prädiktion eines Support Vector Regressors innerhalb des definierten Bereichs  $[-\varepsilon, +\varepsilon]$  nach [58]. Ein Kompromiss zwischen Modellkomplexität und Datenpunkten außerhalb des Toleranzbandes (mit positiven Schlupfvariablen  $\xi$ ) wird durch die Minimierung von Gleichung 2.20 bestimmt (links). Lineare  $\varepsilon$ -sensitive Verlustfunktion (rechts).

## 2.2.4 Extreme Gradient Boosting

Der von [62] entwickelte *Extreme Gradient Boosting*-Algorithmus (kurz: xG-Boost) ist ein leistungsfähiges ML-Modell, welches auf dem sogenannten *Boosting*-Verfahren basiert [63]. Anwendung findet dieser sowohl bei Regressions- als auch bei Klassifikationsproblemen. Die wichtigsten Begrifflichkeiten sollen für ein besseres Verständnis nachfolgend erläutert werden.

Die Grundlage bilden hierarchische Strukturen in Form von sogenannten *Decision Trees* (dt.: Entscheidungsbäumen). Im mathematischen Sinne können diese als ein azyklischer Graph mit einer festen Wurzel verstanden werden. Ein Knoten beschreibt ein Attribut in den Daten. Die Kanten stellen eine Entscheidung auf Grundlage des Attributs dar. In der Anwendungsphase eines trainierten Modells durchläuft ein Datentupel, ausgehend von seiner Wurzel, die Baumstruktur. An den Knoten folgt die Dateninformation je nach Wert eines bestimmten Merkmalen im Datentupel einer der Kanten chronologisch zum nächsten Knoten. Angekommen am letzten Knotenpunkt kann dann der Ausgabewert prädiziert werden. Für eine detailliertere Betrachtung wird [64] empfohlen. Entscheidungsbäume weisen jedoch eine Anfälligkeit für Überanpassung auf, insbesondere bei zu tiefen Architekturen [65]. Aus diesem Grund werden sogenannte *Ensemble*-Methoden angewendet, um Robustheit sowie Performance zu steigern. Nachfolgend wird in Kürze auf die verschiedenen Methoden und Werkzeuge eingegangen.

**Ensemble-Methode** Die Grundkonzeption von Ensemble-Methoden ist es mehrere Prädikatoren miteinander zu kombinieren. Dadurch kann die Qualität der Vorhersage auf Kosten der Interpretierbarkeit gegenüber Schätzern bestehend aus nur einem Modell gesteigert werden. Der Aufbau der Prädikatoren erfolgt dabei auf unerschöpfliche Art und Weise. Das finale Ergebnis setzt sich aus der Kombination der individuellen Schätzungen der Teilmodelle zusammen. Ensemble-Modelle sind demnach in der Lage durch die Diversität ihrer Einzelmodelle eine exaktere Abbildung der Abhängigkeiten von Eingangs- und Ausgangsgrößen darzustellen [66]. Für die Konstruktion von Ensemble-Modellen müssen bestimmte Regeln definiert werden. Diese legen fest, wie hinreichend genaue sowie hinreichend verschiedene Modelle erstellt und auf welche Weise diese zur finalen

Schätzung kombiniert werden. Hierfür gibt es verschiedene Methoden und Werkzeuge, die [67] zusammenfasst. Eine der Methoden ist das bereits erwähnte Boosting bzw. *Gradient Boosting* welche nachfolgend näher erläutert werden.

**Boosting** Das Boosting-Verfahren beschreibt eine Methode, die mittels Kombination mehrerer schwacher Lerner ein leistungsfähiges Ensemble-Modell erstellt, wobei jedes Modell versucht, die Fehler des vorherigen Modells zu korrigieren. Der Algorithmus arbeitet iterativ und passt sich so an die Trainingsdaten an. Während des Trainings werden die Trainingsdaten durch gezielte Manipulation der Gewichtungen einzelner Trainingsdatentupel variiert [68]. Dadurch kann die Diversität der unterschiedlichen Ensemble-Modelle gewährleistet werden. Für eine umfassendere Betrachtung hinsichtlich des Boosting-Verfahrens sei an dieser Stelle auf [69] verwiesen.

**Gradient Boosted Trees** Die *Gradient Boosted Trees* sind ein Mitglied der sogenannten *Tree Ensembles*. Vereinfacht kann von einem Wald bestehend aus mehreren Bäumen gesprochen werden, der durch die Gradient-Boosting-Methode erstellt wird. Der Grundgedanke des Algorithmus besteht – wie beim Boosting – darin nacheinander neue Regressionsbäume anzupassen, um eine genauere Schätzung zu erhalten. Der Unterschied zum konventionellen Boosting-Verfahren besteht in der Art und Weise, wie die Gewichtungen während des Trainings sowie die Vorhersagefehler behandelt werden. Eine Anpassung bzw. Konstruktion des Ensembles wird durch das Gradientenabstiegsverfahren erreicht. Eine maximale Korrelation des Gradienten der Verlustfunktion wird angestrebt [70]. Generell resultiert durch die Anwendung der Gradient-Boosting-Methode ein Vorteil

auf die Modellperformance hinsichtlich ungleichverteilten Datensätzen. Mittels der Gewichtung einzelner Datentupel während des Trainings kann der Algorithmus auf unterrepräsentierte Klassen bzw. Bereiche fokussiert werden. Dadurch ist dieser in der Lage relevante Zusammenhänge in den Daten zu erlernen [71].

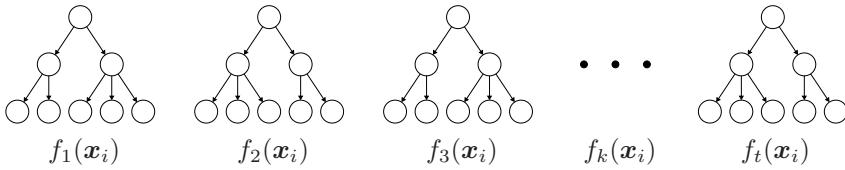
**Extreme Gradient Boosting** Der xGBoost-Algorithmus stellt eine Weiterentwicklung der Gradient Boosted Trees dar. Modelle dieser Art liefern State-of-the-Art-Ergebnisse für eine Vielzahl an ML-Problemstellungen (s. Abschnitt 3.3). Der Hauptunterschied zu Gradient Boosted Trees besteht in einem unabhängigen Aufbau der einzelnen Entscheidungsbäume [72]. Für eine hohe Effizienz des Algorithmus werden zudem mehrere performante Techniken eingeführt. xGBoost erweitert die eigentliche Verlustfunktion im Optimierungsprozess auf die sogenannte *Taylor*-Erweiterung zweiter Ordnung. Auf diese Weise kann unter der Voraussetzung einer zweifachen Differenzierbarkeit sowie Konvexität nahezu jegliche Verlustfunktion implementiert werden. Des Weiteren wird zur Verringerung der Modellkomplexität sowie zur Verhinderung einer Überanpassung ein Regularisierungsterm in die Verlustfunktion mitaufgenommen. Durch Heranziehen eines Datensets  $\mathcal{D} = \{(\mathbf{x}_i, y_i)\}$  kann die Prädiktion des finalen xGBoost-Baummodells folgendermaßen notiert werden:

$$\hat{y}_i^{(K)} = \sum_{k=1}^K f_k(\mathbf{x}_i) = \hat{y}_i^{(K-1)} + f_K(\mathbf{x}_i) . \quad (2.23)$$

mit der Gesamtzahl an Baummodellen  $K$ , der Schätzung des zuletzt erzeugten Modells  $\hat{y}_i^{(K-1)}$  und dem neusten Baummodell  $f_K(\mathbf{x}_i)$ . Für ein besseres

Verständnis ist in Abbildung 2.12 eine vereinfachte Darstellung des Berechnungsprozesses eines xGBoost-Algorithmus illustriert. Auf eine detailliertere Betrachtung der mathematischen Zusammenhänge wird explizit auf [62, 73] verwiesen.

Generieren neuer Entscheidungsbäume in Richtung des negativen Gradienten der Verlustfunktion  
Minimierung des Fehlers mit steigender Modellanzahl



**Abbildung 2.12:** Schematischer Aufbau eines xGBoost-Modells nach [74].

## 3 Stand der Forschung

Im Bereich der modernen Datenanalyse spielen maschinelle Lernmethoden eine zentrale Rolle bei der Vorhersage, Analyse und Modellierung von Zeitreihendaten sowie von Alterungsprozessen der Batterie in automotiven Anwendungen. Heutige State-of-the-Art-Ansätze nutzen eine Vielzahl von maschinellen Lernmodellen, um Muster in Daten zu erkennen und zukünftige Werte präzise vorherzusagen. [17] und [75] belegen dies jeweils in verschiedenen Studien. Die Untersuchungen zeigen, dass statistische Methoden – neben elektrochemischen und physikalischen Ansätzen – eine begründete Relevanz in der Modellierung des Alterungsprozesses von Lithium-Ionen-Batterien aufweisen. Die vorliegende Arbeit konzentriert sich jedoch primär auf die Exploration von Künstlichen Neuronalen Netzen, Support Vector Machines und xGBoost-Modellen. Die folgenden Abschnitte geben Aufschluss über die Eignung der verschiedenen Modellvarianten zur Schätzung von Zeitreihen und zur Vorhersage des Degradationsverhaltens der Batterie.

### 3.1 Neuronale Netze

Die Anwendung von Neuronalen Netzen zur Modellierung und Prognose von Zeitreihendaten hat in der Forschung signifikant an Relevanz gewonnen. In diversen Studien haben sich Künstliche Neuronale Netze als effektive Alternative zu traditionellen Zeitreihenmodellen etabliert [76]. Ein klassisches Beispiel Vorwärtsgerichteter Neuronale Netze ist das Multi-Layer-Perceptron (kurz: MLP), welches sich aus mehreren verdeckten Schichten zusammensetzt. Solche universellen Netzarchitekturen eignen sich für eine Vielzahl von Anwendungen. Insbesondere im Bereich des autonomen Fahrens konnten vielversprechende Ergebnisse erzielt werden, wie [77] darlegen. Aber auch im Bereich des sequentiellen Lernens finden FFNN Anwendung. [78] untersucht im Rahmen seiner Arbeit die Eignung von tiefen Vorwärtsgerichteten Netzen im Vergleich zu bisherigen Modellen für die Zeitreihenprognose. Er zeigt, dass auch MLPs in der Lage sind Zeitreihen vorherzusagen. Jedoch haben auf dem Gebiet des sequentiellen Lernens vor allem Rekurrente Neuronale Netze (kurz: RNN) zu erfolgsversprechenden Ergebnissen geführt. Aus diesem Grund soll nachstehend ausführlicher auf die Anwendung und Qualifikation von RNNs auf diesem Bereich eingegangen werden.

Rekurrente Neuronale Netze haben in der Vergangenheit speziell auf Anwendungsbereichen wie der Bildverarbeitung und Mustererkennung zu erfolgsversprechenden Ergebnissen geführt, ähnlich wie FFNNs. Insbesondere im Bereich der natürlichen Sprachverarbeitung weisen RNNs eine Vielzahl an Durchbrüchen auf [79]. Erfolge werden vor allem in den Teilbereichen der automatischen Beantwortung von Fragen, der Spracherkennung und des maschinellen Übersetzens erzielt. Darüberhinaus haben sich RNNs auch bei Anwendungen auf dem Gebiet des sequentiellen Lernens

als vielversprechend erwiesen. In diesem Zusammenhang vergleichen [80] verschiedene RNN-Architekturen für Problemstellungen mit langen zeitlichen Abhängigkeiten. Die Bewertung hebt hervor, dass RNNs besonders geeignet für Aufgaben dieser Art sind.

Eine Schwierigkeit bei der Anwendung von einfachen rekurrenten Einheiten stellt das von [81] beschriebene Problem der verschwindenden Gradien-ten dar. Erfolgreiche Anwendungen von Rekurrenten Neuronalen Netzen für sequenzielles Lernen sind bisher auf komplexe rekurrente Einheiten zurückzuführen, wie z. B. Long Short-Term Memory-Zellen (kurz: LSTM). Eine umfassende Analyse anhand unterschiedlicher Experimente von [82] zeigt, dass durch die Verwendung von LSTM-Einheiten Langzeitabhängigkeiten deutlich besser erlernt werden können als durch andere Variationen Rekurrenter Neuronaler Netze. Am Beispiel der Handschrifterkennung demonstrieren [83], dass LSTM-Netze zu weit besseren Resultaten geführt haben als bisherige Ergebnisse auf Basis dynamischer Modelle, um komplexe Sequenzen mit weitreichender Struktur zu erzeugen. Angesichts der nachgewiesenen Effektivität von Long Short-Term Memory-Netzwerken in der Sequenzmodellierung erscheint ihre Anwendung in spezifischen und praktischen Kontexten, wie der Modellierung der Batteriealterung, besonders relevant. Im Folgenden wird erörtert, warum LSTM-Netzwerke speziell für vorliegende Problemstellung geeignet sind, indem ihre Fähigkeit zur präzisen Erfassung und Analyse der zeitlich veränderlichen Eigenschaften von Batterien herausgearbeitet wird.

Der umfassende Überblick von [84] unterstreicht zunächst die weite Verbreitung von Künstlicher Intelligenz in Batteriemanagementsystemen elektrifizierter Fahrzeuge, unter anderem mit tiefen Rekurrenten Netzen. Diese haben entscheidend dazu beigetragen, die Vorhersage der Batterielebensdauer zu verbessern und das Energiemanagement zu optimieren. Im

Zusammenhang mit den Anforderungen an die Trainingsdaten stellen [85] ein flexibles *SoH*-Vorhersageschema vor, das einen LSTM-Ansatz verwendet und den Bedarf an umfangreichen Trainingsdaten effektiv reduziert. In ähnlicher Weise verwendet [86] gekoppelte LSTM-Netzwerke für eine präzise Prädiktion und rückt die Datenvorverarbeitung für verbesserte Modellperformance in den Vordergrund. Damit einhergehend bestätigen [87] in deren Arbeit die Bedeutung des Datenvorverarbeitungsprozesses bei der Vorhersage des *SoH* mittels LSTM-Architekturen.

[88] heben die Effektivität von LSTM-Methoden zur Abschätzung des Gesundheitszustands von Batterien in Elektrofahrzeugen hervor. Die Bedeutung von Parametern, die aus dem Lade-/Entladeprozess abgeleitet werden, wird explizit unterstrichen. Ergänzend dazu zeigt [89], dass durch eine integrierte Schätzung des Ladezustands sowie Gesundheitszustands eine effektivere Erfassung und Analyse der zeitlichen Batteriedaten durch rekurrente Einheiten möglich ist. [90] fügen hinzu, dass die Modellierung der Batteriekapazität als Funktion des *SoH* es ermöglicht, komplexe Muster der Kapazitätsdegeneration über die Zeit hinweg genau zu verfolgen und vorherzusagen.

[91] eröffnen eine neue Perspektive in der Anwendung von LSTM-Modellen, indem diese für kurzfristige elektrische Lastprognosen angepasst und die Herausforderungen bei der Auswahl von Hyperparametern angegangen werden. Dennoch grenzt sich die Problemstellung im Rahmen dieser Arbeit ab. Es werden keine elektrischen Lasten im herkömmlichen Sinne betrachtet, sondern die Umsetzung angestrebt, unbekannte Beanspruchungen einer komplex belasteten Batterie innerhalb eines hochdynamischen Systems mittels Diagnosedaten zu beschreiben. Die erste Orientierung hinsichtlich der Wahl der Hyperparameter soll nichtsdestotrotz an der von [91] untersuchten Studie erfolgen.

In einem ähnlichen Kontext, jedoch mit einem anderen Fokus, untersuchen [92] eine datengetriebene LSTM-Methode, die auf eine gleichzeitige Schätzung verschiedener Batteriezustände ausgerichtet ist. Diese Methode erzielt unter verschiedenen Bedingungen eine hohe Genauigkeit, was die Vielseitigkeit von LSTM-Modellen im Umgang mit mehreren Zustands schätzungen unterstreicht. Ergänzend dazu demonstrieren [93] eine weitere Facette dieser Vielseitigkeit, indem ein RNN mit traditionellen modellba sierten Ansätzen kombiniert wird, um die Lebensdauer von Batterien vor herzusagen. Adaptive Ansätze zur Vorhersage des Batteriezustands werden in der Studie von [94] veranschaulicht, mit Fokus auf der dynamischen Zu standsvorhersage von Lithium-Ionen-Batterien. In diesem Kontext schlägt die Studie von [95] eine Untersuchung zur Auswahl von Merkmalen sowie einen LSTM-Algorithmus für eine verbesserte Lebensdauervorhersage un ter Verwendung von Teillade-/Entladedaten zur Erhöhung der Genauigkeit vor. Diese Studien zusammen veranschaulichen das breite Anwendungs spektrum und die Anpassungsfähigkeit von LSTM-Modellen in verschie denen Bereichen der Prognostik.

Des Weiteren zeigen [96] anhand synthetisierter Daten, dass die Verwen dung von Long Short-Term Memory-Netzwerken durchaus geeignet ist, um ein reales, komplexes sowie dynamisches Batteriesystem mit seinen Alte rungsmechanismen zu beschreiben. Die Fortschritte bei LSTM-basierten Techniken im Bezug auf das Degradationsverhalten von Lithium-Ionen Batterien werden von [97] vertieft. Im Rahmen der Arbeit wird gezeigt, dass es die Anpassung des Algorithmus über mehrere Ladezyklen hin weg ermöglicht, komplexere Muster in den Daten zu erkennen und zu lernen. Dadurch wird eine präzisere Vorhersage des Gesundheitszustan des erreicht. Abschließend zeigt [98] im Rahmen einer Untersuchung, dass LSTM-Ansätze durchaus als Alternative zu erweiterten Kalman-Filttern für

die Prädiktion des Alterungsverhalten auf Basis von Bordnetzströmen darstellen.

Basierend auf bisherigen Erfahrungen bei der Anwendung von Rekurrenten Neuronalen Netzen stellt die Verwendung komplexer rekurrenter Einheiten einen erfolgsversprechenden Ansatz im Hinblick auf Effektivität von LSTM-Einheiten für die Vorhersage der Batteriealterung dar. Der aktuelle Stand der Forschung zeigt die vielfältigen Anwendungen dieser Algorithmen und demonstriert die Anpassungsfähigkeit und Genauigkeit von LSTM-Topologien im Bereich der Batterietechnologie. Aus diesem Grund wird im weiteren Verlauf der Arbeit der Einsatz von LSTM-Zellen vorgeschlagen, um den Gesundheitszustand auf Basis der Diagnosedaten der Batterie zu prädizieren.

## 3.2 Support Vector Machines

Support Vector Machines haben sich als leistungsfähiges Werkzeug im Bereich der Sequenzmodellierung und der Vorhersage von Zeitreihen erwiesen [99]. Aufgrund ihrer Fähigkeit, mittelgroße Datensätze mit geringer Überanpassung zu verarbeiten sowie hoher Recheneffizienz und Robustheit gegenüber Datenrauschen werden im Rahmen dieser Arbeit SVM-Ansätze als praktikable und vielversprechende Methodik untersucht [100].

Im Kontext der fortgeschrittenen Forschung zur Sequenzmodellierung und der prädiktiven Analyse von Zeitreihendaten haben sich SVMs bereits als effiziente und flexible Instrumente erwiesen. Die Arbeit von [101] stellt einen bedeutenden Vorstoß in den Bereich der Finanzzeitreihenvorhersage dar. SVMs zeigen eine deutliche Überlegenheit gegenüber herkömmlichen Vorhersagemodellen. [102] beschäftigen sich mit der Verarbeitung

natürlicher Sprache. Sie verdeutlichen die Leistungsfähigkeit von SVMs bei der Vorhersage komplex strukturierter Einheiten. Auch im Bereich des autonomen Fahrens haben [103] die Weiterentwicklung eines SVM-Algorithmus angestrebt, welcher in der Lage ist auf Basis historischer Daten, menschenähnliche Entscheidungsprozesse in vielfältigen Verkehrsszenarien nachzubilden. Unter Berücksichtigung eines ähnlichen Kontextes, aber mit anderem Schwerpunkt haben [104] eine SVM-Methodik für die Prognose von Zeitreihen des Verkehrsflusses entwickelt. Die Autoren heben die beschleunigte Konvergenz und verbesserte Verallgemeinerungsfähigkeiten hervor.

[105] führen im Rahmen ihrer Arbeit ein SVM-Konzept in Form einer fortgeschrittenen Modellarchitektur ein, welches zu einer signifikanten Steigerung der Generalisierungsleistung führt. Diese trägt demnach entscheidend zum Erweiterungsprozess des theoretischen und praktischen Verständnisses in der Methodologie der Zeitreihenprognose bei. In Bezug auf die Analyse von Sensordaten haben [106] ein SVM-Regressionsmodell entwickelt, das die Komplexitäten, die mit fluktuierenden und verrauschten Datensätzen verbunden sind, wirksam abschwächt und die Thematik um die Datenqualität in den Vordergrund stellt. Zuletzt bestätigen [107] die Anwendbarkeit von SVMs bei der Prädiktion von Sequenzen in Form von Geschwindigkeitsprofilen. Die erzielten Forschungsergebnisse demonstrieren eine überlegene Leistungsfähigkeit gegenüber KNNs, was die Bedeutung von SVMs in der Analyse von historischen Datensätzen untermauert.

Die Erfolge von SVMs in verschiedenen Disziplinen ebnet auf natürliche Weise den Weg für deren Anwendung im spezialisierten Feld der Batteriealterung, einem Bereich, in dem präzise Vorhersagen und Modellierungen

von entscheidender Bedeutung sind. Angesichts ihrer bewährten Wirksamkeit im Umgang mit komplexen Datensätzen und der Prognose zeitabhängiger Phänomene treten SVMs als potenzielles Analyseinstrument zur Bewältigung der Komplexitäten der Batteriedegradation und Lebenszyklusanalyse hervor. Aus diesem Grund soll nachstehend ausführlicher auf die Anwendung und Qualifikation von SVMs in diesem Bereich eingegangen werden.

Die Studien von [108] und [109] veranschaulichen spezifische Anwendungen von SVMs in der *SoH*-Schätzung. [108] implementieren einen SVM-Algorithmus, der sich auf Batterie-Konstantstrom-Entladespannungsprofile stützt. [109] gründen ihre Analyse auf Datensätze der nichtlinearen Frequenzgangmessung. Beide Ansätze zeigen eindrucksvoll die Fähigkeit von SVMs, die Batteriealterung präzise zu überwachen und abzubilden. Dabei wird die Anzahl der benötigten Stützvektoren und die Messzeit signifikant reduziert. Ebenso leisten [110] und [111] signifikante Beiträge zur Erweiterung des wissenschaftlichen Verständnisses im Bereich der Restnutzungsdauer-Prognose. Erstgenannte implementieren Alterungstestdaten von Lithium-Ionen-Batterien zur Validierung ihres SVM-Verfahrens. Zweitgenannte ziehen Ladungs- und Entladungsparameter für verschiedene Zellpacks heran, um Entwicklung und Überprüfung des umgesetzten SVM-Modells zu gewährleisten. Beide Untersuchungen heben die außergewöhnliche Präzision von SVMs in der Restlebensdauerprognose hervor. Die Resultate von [112] und [113] unterstreichen vor allem die Anpassungsfähigkeit von SVMs an unterschiedliche Batterie-Datentypen und -Bedingungen. Die Entwicklung der Algorithmen für eine Vorhersage des Ladezustands basiert auf experimentellen Ladedaten. Die Arbeiten betonen speziell die Flexibilität der SVM bei der Verarbeitung unterschiedlicher Dateneingänge und zeigen gleichzeitig eine hohe Vorhersagegenauigkeit und Stabilität auf.

Die jüngsten Entwicklungen in der Effizienzsteigerung bei der Prüfung von Batterien manifestieren sich eindrucksvoll in den folgenden Forschungsarbeiten. [114] haben eine innovative Methode unter Anwendung von SVM zur Klassifikation ausgedienter Lithium-Ionen-Zellen entwickelt, welche auf Daten aus demontierten Batteriemodulen basiert. Dieser methodische Ansatz führt zu einer signifikanten Steigerung der Effizienz beim Identifizieren wiederverwendbarer Zellen. [115] hingegen implementierten eine performante Screening-Methodik für ausgemusterte LiFePO<sub>4</sub>-Batterien, wobei die Präzision und Geschwindigkeit der SVMs bei der Klassifizierung und Evaluation von Batterien hervorgehoben wird, was maßgeblich die Wege für eine effektive Sekundärnutzung ebnet.

Parallel dazu rückt die Optimierung von Parametern und Modellen innerhalb der SVM-Technologie in den Fokus mehrerer Studien. Bedeutende Beiträge in diesem Bereich werden von [116] und [117] geleistet. [116] setzen einen kombinierten SVM-Algorithmus zur Prognose des *SoH* ein. Dieser hybride Ansatz trägt nicht nur zur Verbesserung der Prognosegenauigkeit bei, sondern demonstriert zudem die Anpassungsfähigkeit der SVM-Technologie in der Integration mit anderen algorithmischen Methoden. Im Gegensatz dazu wählen [117] einen differenzierten Ansatz, indem sie SVMs mit einem diskreten Batteriealterungsmodell kombinieren, um die Kapazität von Batterien zu schätzen. Diese methodische Verschmelzung illustriert, wie SVMs maßgeschneidert eingesetzt werden können, um die Präzision der Vorhersagen unter realen Einsatzbedingungen zu optimieren.

Parallel dazu führen [118] sowie [119] eine vergleichende Analyse verschiedener maschineller Lernalgorithmen durch, einschließlich SVM, mit Schwerpunkt auf der Schätzung des Ladezustands sowie Gesundheitszustandes. Ihre Ergebnisse unterstreichen aufs Neue die überlegene Leistung von

SVM im Vergleich zu alternativen Methoden. In einer separaten Arbeit tragen [29] einen neuartigen Ansatz bei, indem sie Alterungsmechanismen in Lithium-Ionen-Zellen quantifizieren. Dies wird durch die Anwendung einer Differenzialspannungsanalyse in Kombination mit SVM-Techniken erreicht. Die Implementierung der Methode an einem Datensatz von Automobil-Pouch-Zellen verdeutlicht ausgeprägte Unterschiede im Alterungsverhalten und in den Homogenitätszuständen der Zellen. Abschließend zeigt [120] im Rahmen seiner Forschungsarbeit eine beispielhafte Implementierung einer SVM in Form eines *SoHC*-Schätzalgorithmus.

Auf Basis der umfassenden Literaturrecherche und den bisherigen Erfahrungen in der Anwendung von SVMs im Bereich der Batterietechnologie, stellt die Verwendung dieser Methodik einen vielversprechenden Ansatz dar. Die Forschungsergebnisse unterstreichen die Eignung von SVMs für die präzise Vorhersage der Batteriegesundheit. Daher wird im weiteren Verlauf der Arbeit der Einsatz von SVMs vorgeschlagen und vertieft.

### 3.3 Extreme Gradient Boosting

xGBoost-Verfahren haben sich zu einem herausragenden Verfahren für die Sequenz- sowie Zeitreihenmodellierung entwickelt. Die Anwendungen erstrecken sich über verschiedene Bereiche wie die Verarbeitung natürlicher Sprache, autonomes Fahren und weitere verschiedenste Anwendungen im Automobilsektor. Die Vielseitigkeit und Effizienz von xGBoost bei der Verarbeitung komplexer Datensätze und der prädiktiven Modellierung werden im Folgenden veranschaulicht.

[121] demonstrieren am Beispiel der Routenplanung die Effektivität von xGBoost bei der Vorhersage von Reisezeiten auf Autobahnen. Die Resultate

zeigen dabei eine klare Überlegenheit gegenüber traditionellen Modellen um komplexe Sequenzen mit ausgeprägten strukturellen Eigenschaften zu erzeugen. In ähnlichem Kontext demonstrieren [122] im Bereich der autonomen Fahrzeuge die Schlüsselrolle von xGBoost. Die Fähigkeit von xGBoost, komplexe Muster in den Trajektoriendaten von Fahrzeugen zu identifizieren und vorherzusagen, zeigt seine Anwendbarkeit in hochdynamischen und unvorhersehbaren Umgebungen. Durch eine umfassende Analyse anhand unterschiedlicher Experimente zeigen [123], dass durch Verwenden von xGBoost zur Extraktion von Merkmalen für die Vorhersage multivariater Zeitreihen, Zeitabhängigkeit deutlich besser erlernt werden können. Dies veranschaulicht das Potenzial des Modellansatzes in der industriellen Datenanalyse. [124] bewerten das Verhalten von xGBoost für die Aufgabe einer Klassifikation auf Basis zeitlich abhängiger Daten im Vergleich zu anderen Ansätzen, einschließlich KNNs. Die Resultate zeigen die klare Überlegenheit von xGBoost. Darüber hinaus belegen [125] im Rahmen einer umfangreichen Untersuchung, dass xGBoost-Algorithmen leicht verbesserte Ergebnisse liefern als LSTM-Ansätze, was die Wirksamkeit des Modells im Umgang mit vielfältigen und langzeitabhängigen Dateneigenschaften demonstriert.

[126] entwickeln im Rahmen einer Untersuchung ein xGBoost-Hybridmodell zur Vorhersage von Aktienkursen. Zwar illustriert dieses Modell, dass hybridisierte Modelle nicht-lineare Beziehungen auf Basis von Zeitreihendaten lernen können, gleichzeitig liefern alleinstehende xGBoost-Algorithmen im Vergleich ähnlich Resultate. Dies bestätigen [127] am Beispiel der Erkennung von Stromprofilen. Die Ergebnisse zeigen, wie xGBoosting-Verfahren effektiv eingesetzt werden, um dynamische und oft rauschbehaftete Datenströme zu schätzen. Zuletzt arbeiten [128] anhand verschiedener Experimente die Effizienz eines xGBoost-Algorithmus bei der Vorhersage von historischen Daten heraus. Besonders hervorgehoben

wird die Bedeutung des Datenvorverarbeitungsprozesses, indem relevante Größen selektiert werden. Infolgedessen kann sowohl die Prognosegüte des Modells gesteigert als auch die Rechenzeit signifikant gesenkt werden.

In Anbetracht der demonstrierten Effektivität des xGBoost-Modells hinsichtlich Sequenz- und Zeitreihenmodellierung soll nun die Eignung speziell für die Vorhersage der Batteriealterung untersucht werden.

[129] und [130] haben bedeutende Fortschritte im Bereich der *SoH*-Schätzung sowie Lebensdauervorhersage von LIB mit xGBoost erzielt. [129] führen einen xGBoost-Ansatz ein, der die multiplen Alterungscharakteristiken auf Zellebene abbildet. Darauf aufbauend entwickeln [130] einen ähnlichen Ansatz, mit dem Unterschied, dass sich der Fokus der Arbeit auf die Interpretierbarkeit des Algorithmus konzentriert und somit zum Verständnis der zugrundeliegenden Alterungsmechanismen von LIB beiträgt. Diese Methode erweist sich auf Basis einer quantitativen Analyse als durchaus wirksam bei der Prädiktion der Batterielebensdauer und ergänzt den Ansatz von [129].

Parallel dazu verwenden [131] im Rahmen einer aufwendigen Studie xGBoost-Algorithmik für die Vorhersage der kalendarischen Alterung in verschiedenen Zellchemien. Die Ergebnisse unterstreichen die Überlegenheit des Algorithmus bei den Fehlerraten und der Anpassungsfähigkeit unabhängig untersuchten Zellchemie, insbesondere im Vergleich zu anderen Algorithmen des maschinellen Lernens wie KNNs. Die Arbeit liefert wertvolle Einblicke in die relativen Stärken von xGBoost im Zusammenhang mit der Batteriealterung.

[132] schlagen einen anwendungsspezifischen Ansatz vor, um den *SoH* in Hybridefahrzeugen zu schätzen. Die Datengrundlage der Untersuchungen stellen Strom- sowie Spannungsprofile zyklisierter Zellen aus Lade-

bzw. Entladevorgängen dar. Zusammenfassend wird gezeigt, dass Boosting-Methoden, darunter xGBoost, einen vielversprechenden Ansatz bei der Alterungsmodellierung darstellen. Für die vorliegende Thesis gelten im Hinblick auf die Schätzgröße ähnliche Voraussetzungen. Dennoch grenzt sich die Problemstellung im Rahmen dieser Arbeit deutlich ab. Es werden eindeutigen Spannungs- sowie Stromprofile einzelner Zellen betrachtet, sondern die Umsetzung angestrebt, durch Flotten-Diagnosedaten das Kunderverhalten zu analysieren und auf Basis dessen das Degradationsverhalten der komplex belasteten Hochvoltbatterien innerhalb eines hochdynamischen Systems zu beschreiben.

Auch [133] und [134] leisten wichtige Beiträge in diesem Bereich. Erstere entwickeln einen auf xGBoost basierenden Ansatz, um die Restnutzungsdauer von LiB vorherzusagen. Besonders die Bedeutung der Merkmalsauswahl für die Verbesserung der Vorhersageleistung des Modells wird hervorgehoben. [134] zeigen eine signifikante Steigerung der Genauigkeit der *SoH*-Schätzung durch den Einsatz von xGBoost. Genauer führt die vorgeschlagene Methode zu einer Steigerung der Prognosegüte von 10-20 % im Vergleich zu SVMs sowie KNNs.

Sowohl [135] als auch [136] betonen die rechnerischen Vorteile von xGBoost. [135] konzentrieren sich auf die Effizienz und Skalierbarkeit von xGBoost für die Vorhersage des *SoH* in LiB. [136] stellen hingegen eine schnelle Online-Methode zur *SoH*-Schätzung auf der Grundlage xGBoost vor und unterstreichen die Fähigkeit des Algorithmus, schnelle und genaue Vorhersagen mit minimalen Fehlern zu liefern. [137] entwerfen und bewerten im Rahmen einer experimentellen Studie von einer Vielzahl an unterschiedlich zyklisierten Zellen einen xGBoost-Algorithmus. Das Training erfolgt anhand einer Datenbasis von Ladespannungs- sowie Stromkurven,

welche nach einem Schnellladeprotokoll gemessen wurden. Die erfolgsversprechenden Ergebnisse zeigen auch hier klar auf, dass xGBoost-Ansätze durchaus in der Lage sind, mittels Spannungs-/Stromprofilen den Gesundheitszustand zu prädizieren. Daran anknüpfend entwickelt [138] eine Methode unter Einsatz des xGBoost-Algorithmus zur Schätzung der Lebensdauer ausgedienter Energiespeicherbatterien. Diese Arbeit erweitert die Einsatzmöglichkeiten von xGBoost auf verschiedene Phasen im Lebenszyklus einer Batterie. Die hohe Vorhersagegenauigkeit trägt zur Entwicklung nachhaltiger Batteriemanagementstrategien bei.

In der Auseinandersetzung mit Herausforderungen bezüglich der Datenverfügbarkeit demonstrieren [139] die Effektivität des xGBoost-Algorithmus in Szenarien mit geringen Datensätzen. Es wird gezeigt, dass derartige Modellansätze in der Lage sind, aussagekräftige Muster zu identifizieren, was für die Vorhersage der Batterieleistung und -alterung von entscheidender Bedeutung ist. Die Ergebnisse sind insbesondere für die Entwicklung von zuverlässigen und effizienten Batteriemanagementsystemen relevant, da diese aufzeigen, wie mit begrenzten Daten maximale Erkenntnisse gewonnen werden können.

[140] und [141] tragen beide signifikant zur Schätzung des Gesundheitszustands von Lithium-Ionen-Batterien bei, allerdings mit unterschiedlichen methodischen Ansätzen. [140] entwickeln ein innovatives Modell unter Verwendung von xGBoosting in Kombination mit einem verbesserten Merkmal-Extraktionsalgorithmus. Dies führt zu einem robusten Modell mit hoher Genauigkeit und niedrigen Fehlerquoten. Im Gegensatz dazu

kombiniert [141] Boosting-Maschinen mit einer gewichteten Quantilsregression, um eine effiziente Schätzung zu erreichen. Beide Studien demonstrieren erneut das Potenzial innovativer xGBoost-Ansätze in der präzisen *SoH*-Vorhersage und deren praktische Anwendbarkeit in Energiespeichersystemen und Elektrofahrzeugen.

Die Arbeit von [142] konzentriert sich speziell auf die Restnutzungsdauer von Lithium-Ionen-Batterien in Batteriespeichersystemen. Es wird explizit der Einfluss der Temperatur als entscheidende Variable für eine begünstigte Alterung der Zelle erwähnt. Des Weiteren steht die Durchführung einer Hyperparameter-Studie im Vordergrund, um die bestmöglichen Einstellungen des Modells für Datensätze zyklisierter Zellen zu finden. Die erste Orientierung hinsichtlich der Wahl der Hyperparameter soll an der von [142] untersuchten Studie erfolgen. Abschließend empfehlen [143] die Verwendung von xGBoost und unterstreichen die Effizienz des Algorithmus im Umgang mit nichtlinearen Beziehungen und komplexen Mustern in Alterungsdaten von Lithium-Ionen-Batterien.

Zusammenfassend unterstreichen diese Studien gemeinsam die Vielseitigkeit und Effektivität des xGBoost-Algorithmus für Sequenz- bzw. Zeitreihenmodellierung sowie bei der Vorhersage des Gesundheitszustandes von LIB. Sie heben speziell die Anpassungsfähigkeit des Algorithmus über verschiedene Anwendungen und Kontexte hinweg hervor. Aus diesen Gründen wird der Einsatz des xGBoost-Algorithmus im Rahmen dieser Arbeit vorgeschlagen, um die Vorteile seiner fortschrittlichen Analysefähigkeiten und seiner Anpassungsfähigkeit an verschiedene Batteriedaten und -szenarien voll auszuschöpfen.

### 3.4 Zusammenfassung und Forschungslücke

Die vorgestellten Studien unterstreichen die Wirksamkeit und Vielseitigkeit von KNN, insbesondere von LSTM-Netzwerken, für die Modellierung komplexer Zeitreihendaten und die Vorhersage des Gesundheitszustands von LIB. Ebenso zeigen SVMs sowie xGBoost eine beachtliche Effizienz in der Verarbeitung und Analyse von Batteriedaten, insbesondere in Bezug auf die präzise Vorhersage des *SoH*.

Bei der Betrachtung aller aufgeführten Ansätze fällt auf, dass diese zwar in verschiedenen Bereichen der Zeitreihenmodellierung und Vorhersageanwendungen effektiv sind, aber die spezifische Anwendung auf die Vorhersage der Batteriealterung und die Bewertung des *SoH*, insbesondere im Kontext der Diagnosedaten von Fahrzeugflotten, weniger bis gar nicht erforscht ist. Die Anwendung der Ansätze bietet mehrere Vorteile, wie beispielsweise die effiziente Verarbeitung großer Datensätze, hohe Anpassungsfähigkeit an verschiedene Datentypen und die Fähigkeit, komplexe nichtlineare Beziehungen in den Daten zu erfassen. Angesichts dieser Merkmale scheinen alle Algorithmen geeignet zu sein, um die komplexen und dynamischen Muster, die in den Diagnosedaten von Fahrzeugflotten vorhanden sind, zu modellieren und zu analysieren.

Die Einzigartigkeit der vorliegenden Thesis liegt in der Anwendung sowie Anpassung verschiedener maschineller Lernalgorithmen auf einen spezifischen Datentyp - die Diagnosedaten von Fahrzeugflotten - welche sich in ihrer Struktur und Dynamik von anderen Datentypen unterscheiden, die üblicherweise in der Batteriealterungsforschung verwendet werden. Diese spezifische Anwendung könnte neue Einsichten in die Muster und Beziehungen innerhalb der Diagnosedaten liefern und so zu genaueren Vorhersagen des Batteriegesundheitszustands beitragen.

Abschließend lässt sich sagen, dass die Anwendung bzw. Anpassung von KNNs, SVMs sowie xGBoost-Algorithmen auf die Vorhersage der Batteriealterung unter Verwendung von historischen Diagnosedaten aus Fahrzeugflotten, einen innovativen und vielversprechenden Forschungsansatz darstellt. Um bisherige Forschungslücken in der Fachliteratur zu schließen, sollen diese erfolgreichen Konzepte des maschinellen Lernens auf diesen Anwendungsfall übertragen werden. Da zudem in der Literatur im Hinblick auf die einzigartige Datenstruktur und den einzigartigen Informationsgehalt kaum Hinweise zu finden sind, sollen im Rahmen der vorliegenden Arbeit erste Schritte durchgeführt werden, um diese Lücke zu schließen.



# 4 Methodenentwicklung

Das folgende Kapitel beschäftigt sich mit Datenverarbeitungstechniken für historische Betriebsdaten sowie der Anpassung verschiedener maschinel- ler Lernmodelle an die vorliegende Problemstellung. Die Datengrundlage wird zunächst in Abschnitt 4.1 formell eingeführt und untersucht. Im An- schluss werden in Abschnitt 4.2 Eingangsgrößen für das Modelltraining selektiert, die maßgeblichen Einfluss auf die Batteriealterung haben. Dar- an anknüpfend wird ein geeigenter Datenvorverarbeitungsprozess vorge- schlagen. Schließlich gibt Abschnitt 4.3 einen Überblick über ausgearbei- tete Prädiktionsmodelle, welche mit unterschiedlichen Methoden das De- gradationsverhalten des dynamischen Batteriesystems abbilden.

## 4.1 Datenanalyse und Datenvorverarbeitung

Eine systematische Analyse und Vorverarbeitung der Daten ist im Hinblick auf Machine-Learning-Anwendungen unabdinglich. Die Extraktion wich- tiger Informationen dient dazu ein besseres Verständnis über die vorlie- genden Daten zu erlangen. Es besteht die Möglichkeit charakteristische Phänomene zu entdecken und somit Rückschlüsse auf systembeschreiben- de Effekte zu ziehen, die als Grundlage für aussagekräftige Eingangsgrößen des Modells dienen. Daran anknüpfend lässt die Vorverarbeitung der Ein- und Ausgangsdaten eine gezielte Anpassung des Optimierungsproblems

zu. In den folgenden Abschnitten sollen zunächst die zur Verfügung stehenden Daten näher analysiert werden. Anschließend wird explizit auf den Datenvor- und aufbereitungsprozess eingegangen, da dieser einen wichtigen Bestandteil der vorliegenden Arbeit darstellt.

#### **4.1.1 Datengrundlage für den Entwurf der Modelle**

Der Entwurf einer Modellarchitektur für das Schätzen der Batterielebensdauer erfolgt auf Basis von Datensätzen vieler verschiedener Fahrzeuge. Genauer handelt es sich um Plug-In-Hybridfahrzeuge einer bestimmten Baureihe des Sportwagenherstellers Porsche, welche sich seit dem Jahr 2017 auf dem Markt befinden. Es kann demnach von einem Felddatensatz gesprochen werden mit allen sich auf dem Markt befindlichen Kundenfahrzeugen. Durch den Informationsgehalt der Daten können Rückschlüsse auf das reale Nutzungsverhalten, die tatsächlich auftretenden Betriebszustände und die daraus resultierenden Mechanismen innerhalb der Batterie gezogen werden.

Für den überwachten Lernansatz müssen die entsprechenden Referenzdaten bereitgestellt werden. Die Beschaffung erfolgt bei Werkstatt-Serviceterminen sowie bei außerplanmäßigen Werkstattbesuchen. Daraus ergibt sich ein Datenset, das im Laufe eines Fahrzeuglebens mehrere sogenannter Fahrzeug-Analyse-Protokolle (kurz: FAPs) enthält, welche im Folgenden näher erläutert werden.

**Fahrzeug-Analyse-Protokolle** Die FAPs bilden die Datengrundlage, auf deren Basis das Prädiktionsmodell der Batterie entworfen wird. Mit Hilfe der Fahrzeug-Identifikations-Nummer (kurz: FIN) kann ein FAP eindeutig einem bestimmten Fahrzeug anonymisiert zugeordnet werden. Es werden

jedoch nicht alle Informationen des Protokolls für die Modellentwicklung in Betracht gezogen. Bei der Vielzahl an Steuergeräten in heutigen Serienfahrzeugen liegt der Fokus der vorliegenden Arbeit ausschließlich auf den Auslesedaten des Steuergeräts der Hochvoltbatterie. Wichtiger Bestandteil sind hierbei die sogenannten Historiendaten, welche das Steuergerät mitaufzeichnet. Diese sollen nachstehend detaillierter beschrieben werden.

**Historiendaten** Die historischen Betriebsdaten geben Aufschluss über die Vergangenheit eines Fahrzeugs. Genauer werden dabei Informationen über einen bestimmten Betriebsmodus oder -zustand des Fahrzeugs dargelegt und in verschiedenen Intervallen in Form einer Matrixstruktur abgespeichert. Im weiteren Verlauf der Arbeit werden diese als Historien-Daten-Matrizen (kurz: HDMs) bezeichnet. Insgesamt stehen 20 dieser HDMs zur Verfügung. Für ein besseres Verständnis ist in Abbildung 4.1 exemplarisch die HDM 1 illustriert.

**HDM 1** Die HDM 1 ist ein Zeitzähler. Diese zeichnet den Zeitraum auf, in der sich das BMS im sogenannten „Schlafmodus“ befindet. Das bedeutet, dass die Batterie in diesem Modus vom Hochvoltsystem getrennt wird, um Nebenverbraucher abzukoppeln. Die kumulierte Zeit wird gezählt und verschiedenen *SoC*- bzw. Batterie-Temperaturintervallen zugeordnet. Das heißt, die HDM 1 speichert die Gesamtzeitdauern seit Kilometerstand von  $s_{\text{ges}} = 0$ . Folglich ergeben sich 48 Merkmale für HDM 1. Ein jedes kann als separates Eingangssignal für die Modellbildung betrachtet werden. Diese enthalten wichtige Informationen, welche zur Bestimmung der kalendari- schen Alterung genutzt werden können. Die Strukturen und Inhalte der übrigen HDMs sind Anhang A zu entnehmen.

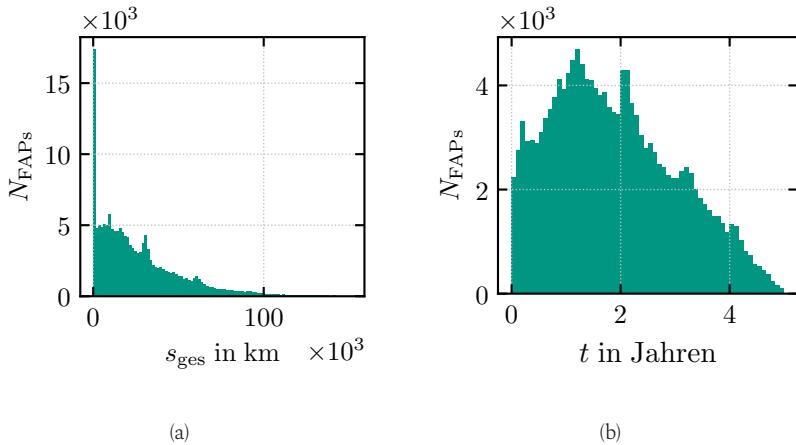
Kumulierte Zeit in s		SoC-Bereich in %					
		[ $SoC_1, SoC_2]$	[ $SoC_2, SoC_3]$	[ $SoC_3, SoC_4]$	[ $SoC_4, SoC_5]$	[ $SoC_5, SoC_6]$	[ $SoC_6, SoC_7]$
Temperaturbereich in °C	[ $-T_\infty, T_1]$	1	2	3	4	5	6
	[ $T_1, T_2]$	7	8	9	10	11	12
	[ $T_2, T_3]$	13	14	15	16	17	18
	[ $T_3, T_4]$	19	20	21	22	23	24
	[ $T_4, T_5]$	25	26	27	28	29	30
	[ $T_5, T_6]$	31	32	33	34	35	36
	[ $T_6, T_7]$	37	38	39	40	41	42
	[ $T_7, +T_\infty]$	43	44	45	46	47	48

**Abbildung 4.1:** Schematische Darstellung des HDM 1 mit den 48 in blau eingefärbten Merkmalen. Die blauen Zahlen in der Matrix stellen lediglich eine Nummerierung der Merkmale dar und keine beispielhaften Werte der Verweilzeit. Inhalt jedes Merkmals sind kumulierte Aufenthaltszeiten in den jeweiligen Temperatur-SoC-Bereichen.

**Verteilung des Datensatzes** Für eine erste Einschätzung soll nun ein Überblick über die Rohdaten gegeben werden, welche die Grundlage für die Datenanalyse und anknüpfende Alterungsmodellierung darstellen. Genauer soll auf die Häufigkeitsverteilung aller vorliegenden FAPs und auf die der Zielgröße – dem  $SoH_C$  – eingegangen werden.

Insgesamt liegen 156 463 Protokolle von 32 358 verschiedenen Fahrzeugen vor. Für die nachstehenden Darstellungen werden aus Gründen der Übersichtlichkeit Ausreißer mit Werten über 120 % und wenige FAPs mit einer Laufleistung über 300 000 km ignoriert. Abbildung 4.2 (a) stellt die Protokollanzahl  $N_{FAPs}$  in Abhängigkeit des Gesamtkilometerstands  $s_{ges}$  dar. Im Histogramm ist ein Peak bei sehr geringen Laufleistungen zu erkennen. Dies ist auf die Fahrzeugauslieferung zurückzuführen, bei der das erste Protokoll vom Händler ausgelesen wird. Zu diesem Zeitpunkt weisen alle Fahrzeuge einen nahezu identischen Kilometerstand auf. Abgesehen davon

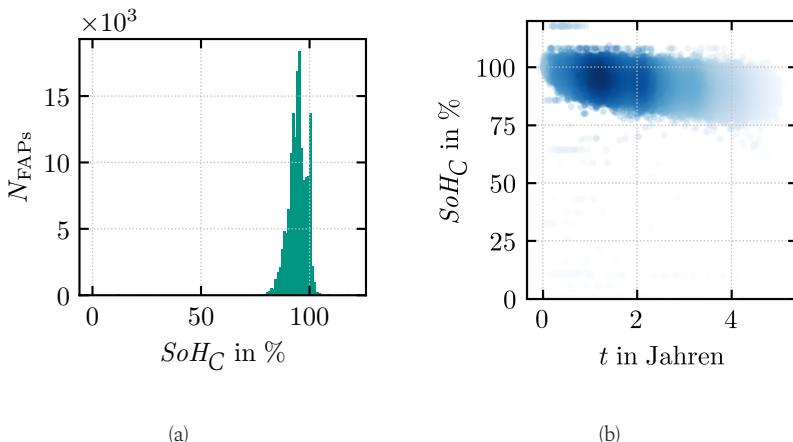
können die meisten Daten im Intervall von 0 km bis 20 000 km festgestellt werden. Des Weiteren macht eine Darstellung in Abhängigkeit des Batteriealters mittels Abbildung 4.2 (b) deutlich, dass im untersuchten Datensatz die Fahrzeugbatterien meist ein Alter von 1 bis 2 Jahren aufweisen. Wohingegen die höchste Nutzungsdauer einer Batterie bei 5 Jahren liegt.



**Abbildung 4.2:** Verteilung der Protokollanzahl  $N_{\text{FAPs}}$  über dem Gesamtkilometerstand  $s_{\text{ges}}$  der Rohdaten (a) und Verteilung der Protokollanzahl in Abhängigkeit des Batteriealters  $t$  der Rohdaten (b).

In jedem Fahzeuganalyseprotokoll lässt sich neben den HDMs auch eine Größe auslesen, die in direktem Zusammenhang mit der Alterung der Batterie steht: den  $SoH_C$  (s. Abschnitt 2.1.2). Diese wird als Zielgröße für den überwachten Lernansatz verwendet. Bei diesem Wert handelt es sich um eine im BMS berechnete Größe, die nicht direkt aus Sensormessungen stammt, sondern mittels eines komplexeren Modells bestimmt wird. Der im FAP enthaltene  $SoH_C$ -Wert stellt somit eine modellgestützte Größe dar, die als Ground-Truth für das Training der Vorhersagemodelle dient. Für den

Rohdatensatz ergibt sich gemäß des Histogramms in Abbildung 4.3 (a) eine nahezu normalverteilte Distribution der Daten um den Mittelwert  $SoH_C \approx 94,5\%$ . Auffallend ist, dass nur wenige Daten in niedrigen  $SoH_C$ -Bereichen vorhanden sind, was für die anschließende Modellbildung ein Problem darstellt. Eine Begründung hierfür liegt in der Tatsache, dass die Hochvoltbatterien gemäß der Garantievorschrift ab einer Restkapazität von 80 % als defekt gelten und sie somit nicht in den Felddaten vorzufinden sind.



**Abbildung 4.3:** Rohdatenverteilung der Protokollanzahl  $N_{FAPs}$  in Abhängigkeit des  $SoH_C$  (a) und  $SoH_C$ -Verteilung über dem Batteriealter  $t$  der Rohdaten (b). Dunkle Bereiche korrelieren mit einer hohen Datenanhäufung.

Zusätzlich zeigt die rechte Darstellung in Abbildung 4.3 die Abhängigkeit des  $SoH_C$ -Wertes vom Batteriealter, welche weitere Aussagen über die Verteilung des Rohdatensatzes erlaubt. Es resultiert eine in verschiedenen Blautönen abgebildete Datenwolke. Die Deckkraft der Farbe korreliert dabei mit der Häufigkeit der Datenpunkte. Es ist zu erkennen, dass die Mehrheit der Daten im Bereich um  $SoH_C \approx 90\%$  bei einem Alter von wenigen

Monaten zu finden sind. Der dunkle Schweif der Punktewolke zeigt dabei, dass die meisten Fahrzeuge linear mit steigenden Batteriealter in Richtung 90 %  $SoH_C$  altern. Hinzu kommt, dass die meisten Daten im Bereich des Batteriealters von ein bis zwei Jahren angesiedelt sind und zugleich unter 60 000 km Laufleistung aufweisen.

### 4.1.2 Datenvorverarbeitung

Durch Vorverarbeiten der Ein- und Ausgangsdaten besteht die Möglichkeit, das Optimierungsproblem gezielt anzupassen. Infolgedessen kann sowohl die Konvergenzgeschwindigkeit als auch die Güte des Modells beeinflusst werden. Es handelt sich hierbei um einen iterativen Prozess. Im Rahmen der Vorverarbeitung wird dies gemäß den Erkenntnissen von [144] durch die Schritte Datentupelselektion, Merkmalsextraktion sowie Merkmalselektion beschrieben.

**Datentupelselektion** Zu Beginn sollen die Rohdaten hinsichtlich deren Qualität untersucht und gefiltert werden. Ziel ist es, einen qualitativ hochwertigen und bereinigten Datensatz zu generieren, sodass das Modell während des Trainingsprozesses keine fehlerhaften Zusammenhänge erlernt. Die Datenanalyse zeigt, dass durch Softwareinkompatibilitäten vermehrt nicht verwertbare bzw. redundante Daten generiert werden. Um den Aufwand der Datenver- und aufbereitung möglichst gering zu halten, wird die Automatisierung des gesamten Preprocessings angestrebt.

Zunächst werden alle FAPs entfernt, welchen ein inkompatibler Softwarestand zugewiesen werden kann. Dem anknüpfend wird eine Plausibilitätsprüfung durchgeführt und in Form von diversen Filtern in den Vorverarbeitungsprozess implementiert. Darunter fallen neben den Filterungen hinsichtlich inkompatibler Softwarestände, fehlerhafte HDMs in spezifischen Softwareständen, inkonsistente Identifikationsnummern einzelner FAPs sowie fehlerbehaftete Standardwerte in den HDMs bedingt durch Konvertierungsfehler.

**Plausibilitätsprüfung** [145] zeigt in seiner Arbeit, dass durch Relaxationsprozesse innerhalb der Batterie eine Zunahme des  $SoH_C$  von bis zu 7 % nach zwei Tagen als plausibel gilt. Aus diesem Grund wird eine obere Grenze von  $SoH_C = 107\%$  zugelassen. Dementsprechend wird auch eine Änderung zwischen zwei aufeinanderfolgenden Protokollen innerhalb der zwei Tage von höchstens  $\Delta SoH_C = 7\%$  als plausibel angesehen. Übersteigt die Änderung diesen Wert, wird dieser als nicht plausibel interpretiert und von der weiteren Analyse ausgeschlossen. Des Weiteren wird gewährleistet, dass die Werte der HDMs zwischen benachbarten Protokollen für die Gesamtaufleistung über Zeit monoton steigend sind. FAPs mit Einträgen im HDM 8, welches die Anzahl der Schnellladevorgänge (DC-Laden) dokumentiert, werden ebenfalls verworfen. Grund hierfür ist, dass die betrachteten Hybridfahrzeuge nicht in der Lage sind DC zu laden. Aber auch hier kommt es durch ungleiche Software zur Entstehung von fehlerhaften Daten. Aufgrund der kontinuierlichen Akkumulation von Zeit oder dem Auftreten spezifischer Ereignisse in den HDMs zeigen sich innerhalb weniger Tage nur geringfügige Änderungen in den Daten. Die Umsetzung eines zeitlichen Mindestabstands von zehn Tagen zwischen zwei FAPs eines Fahrzeugs während eines Reparatur- oder Serviceaufenthalts – bei dem mehrere Protokolle aus dem Steuergerät geladen werden – stellt sicher, dass

keine redundanten Daten in das Modell eingehen. Vielmehr würden ohne diesen Prozessschritt Widersprüche bei ähnlichen HDM-Werten aber unterschiedlicher  $SoH_C$  ins Modell fließen.

Die Umsetzung der beschriebenen Maßnahmen reduziert das Datenset drastisch. Lediglich 41 % der Protokolle von rund 51 % verschiedener Fahrzeuge verbleiben für die anknüpfenden Prozessschritte. Dennoch verbleibt die resultierende Verteilung der Protokollanzahl über dem  $SoH_C$  in einer annähernd normalverteilten Struktur. Auch eine Verschmälerung des Streubereichs aus Abbildung 4.3 (b) wird erreicht. Darüber hinaus werden übrige signifikante Ausreißer mit  $SoH_C < 50\%$  entfernt, da eine Detailanalyse deren Plausibilität in Frage stellt.

**Merkmalsextraktion** Im zweiten Schritt werden die verbleibenden Merkmale weiter für das ML-Modell verarbeitet. Durch die Charakteristik der HDMs besitzen Fahrzeuge mit hohem Alter sowie hoher Laufleistung aufgrund kumulierter Werte automatisch größere Matrizeninträge als junge Fahrzeuge. Aus diesem Grund kann eine direkte Nutzung der kumulierten Werte zu Verzerrungen führen. Daher wird eine Normalisierung der Daten vorgenommen, um Skalierungsunterschiede auszugleichen und eine einheitliche Vergleichbarkeit durch eine Entkopplung des Fahrzeugalters aus den HDMs herzustellen. Um die Altersinformation dennoch zu erhalten, wird das Alter der Batterie als separates Merkmal in die Modellbildung explizit mit eingebunden. Zudem wird im Rahmen der Arbeit eine Unterteilung in *statische*, *dynamische* sowie *historische* Merkmale eingeführt, die im Folgenden näher erläutert werden.

**Statische Eingangsgrößen** Als statische Merkmale werden diejenigen Einträge innerhalb eines HDMs bezeichnet, bei welchen sich nach einer bestimmten Zeit je nach Fahr- bzw. Ladeverhalten sowie klimatischen Bedingungen, eine Verteilung innerhalb einer HDM eines Fahrzeugs einstellt. Mathematisch kann das durch die Normierung

$$\bar{x}_{ij} = \frac{x_{ij}}{\sum_{k=1}^n \sum_{l=1}^m x_{kl}} \quad (4.1)$$

beschrieben werden.  $x_{ij}$  bezeichnet das Merkmal in der  $i$ -ten Zeile und  $j$ -ten Spalte. Die Indizes  $k$  und  $l$  sind die Summationsindizes über alle Matrixelemente. Dadurch kann die Batterie bzw. das Nutzungsverhalten charakterisiert werden. Einerseits ist diese Normierung essentiell für die Übergabe ins Modell. Andererseits entsteht ein gewisser Informationsverlust. Zwar sind die Anteile der Intervalle bekannt, jedoch ist keine Größe mehr vorhanden, welche eine Aussage über Gesamtzeiten oder Gesamtzahl der Ereignisse zulässt. Demnach wird eine zusätzliche Eingangsgröße generiert, welche die Summe aller Intervalle in Form eines dynamischen Merkmals einführt. Eine Erläuterung ist nachfolgend gegeben.

**Dynamische Eingangsgrößen** Bei den dynamischen Merkmalen wird die Annahme getroffen, dass diese mit steigendem Batteriealter nahezu linear ansteigen. Zu diesen Größen gehören bspw. der Gesamtkilometerstand, der Energieumsatz der Batterie in Form des HDM 18 (s. Abschnitt A) sowie jene HDMs, bei welchen ein Informationsverlust entsteht. Als Beispiel kann hier das statische Merkmal HDM 7 genannt werden, welches AC-Ladevorgänge mit Start- und Stopp-*SoC* dokumentiert. Durch die Normierung kann die explizite Anzahl der Ladevorgänge nicht mehr direkt aus

den Daten entnommen werden. Während HDM 18 als dynamisches Merkmal den Energieumsatz durch das Laden beschreibt, liefert HDM 7 dennoch eine zusätzliche Information über das Muster der Ladehübe. Da das Alterungsverhalten nicht nur vom Energieumsatz, sondern auch von der Verteilung der Ladehübe beeinflusst wird, ergänzen sich beide Merkmale für eine differenzierte Modellierung.

Für die individuellen Zellbalancingzeiten (HDM 9, s. Abschnitt A) wird eine Normierung gemäß Gleichung 4.2 eingeführt:

$$\bar{x}_i = x_i \left( \sum_{j=1}^{108} x_j \right)^{-1}. \quad (4.2)$$

$x_i$  beschreibt die kumulierte Balancing-Zeitdauer der Zelle  $i$ . Der Index  $j$  ist der Summationsindex über alle 108 Merkmale. Es können somit jene Zellen identifiziert werden, die über- bzw. unterdurchschnittlich im jeweiligen Protokoll gebalanced werden. Beim Zellbalancing gleicht das BMS Unterschiede im Ladezustand einzelner Zellen aus. Da sich Zellen fertigungsbedingt unterschiedlich entladen, entstehen Spannungsdifferenzen. Das Balancing führt die Zellen auf ein ähnliches Spannungsniveau zurück, indem überschüssige Energie abgeführt oder schwächer geladene Zellen bevorzugt geladen werden. Dadurch wird die nutzbare Kapazität optimiert, während gleichzeitig kritische Ladezustände einzelner Zellen vermieden werden. Es wird erwartet, dass die Balancingzeit in direktem Zusammenhang mit der Alterung der Zelle steht. Die Kompensation des Informationsverlusts über die Zeit, welcher durch die Normierung entsteht, wird durch das Hinzufügen der Gasamtbalancingzeit als synthetische Größe erreicht. Durch den kontinuierlichen Anstieg der Zeit- und Eventzähler in den FAPs

seit BoL ist in jedem aktuellen Protokoll implizit eine Information über vergangene FAPs enthalten. Dennoch sind sie aufgrund geringer Abtastraten nur schwach miteinander verknüpft. Aus diesem Grund wird nachstehend eine weitere Art an Eingangsgrößen eingeführt.

**Historische Eingangsgrößen** Um Zusammenhänge sowie zeitliche Abhängigkeiten zwischen Protokollen eines spezifischen Fahrzeugs zu lernen und somit Informationen vorangehender Protokolle in das Modell miteinzubeziehen, werden zusätzliche, sogenannte historische Merkmale eingeführt. Basierend auf den vorherigen FAPs werden der Mittelwert, Standardabweichung sowie Maximal- und Minimalwert jedes Merkmals berechnet. Diese Metriken beinhalten weitere wichtige Informationen hinsichtlich Verteilung und Charakteristik der Daten und begünstigen somit den Lernprozess. Diese Kennwerte werden als abgeleitete Größen zum aktuellsten Protokoll ergänzt. Der beschriebene Vorgang erfolgt für jedes FAP des gesamten Datensatzes.

Einer weiteren Möglichkeit zeitliche Abhängigkeiten während der Trainingsphase zu erlernen, widmet sich die nächste Maßnahme. Für die Umsetzung wird jedem Protokoll eines Fahrzeugs der  $SoH_C$  des vorangehenden Protokolls  $SoH_C(n_{FAP} - 1)$  hinzugefügt. Jeder zurückliegende  $SoH_C$  kann dabei als eigenständiges Merkmal interpretiert werden. Es ist jedoch wichtig zu beachten, dass die Erstellung historischer Eingangsgrößen nur bei normierten statischen Größen sinnvoll ist. Die Betrachtung der genannten Metriken liefert nur für normierte Werte sinnvolle Ergebnisse.

**Merkmalsselektion** Im Rahmen der Merkmalsselektion werden nun die Merkmale, welche in die ML-Modelle eingehen, ausgewählt sowie deren Informationsgehalt qualitativ bewertet. So ist es in Anbetracht der starken

Zunahme der Merkmalsanzahl durch die Extraktion historischer Eingangsgrößen sinnvoll die Gesamtheit der Merkmale nun wieder zu reduzieren. Ziel ist es, mit möglichst wenigen, dafür aussagekräftigen Eingangsgrößen dem Algorithmus maximalen Informationsgehalt bereitzustellen. Alle dazugehörigen Maßnahmen werden in den nächsten Abschnitten näher erläutert.

## 4.2 Eingangsdaten für die Modelle

Wie bereits erwähnt, dienen den Modellen ausnahmslos Signale als Eingangsdaten, welche im Fahrzeug auf den Steuergeräten vorliegen. Dem Algorithmus sollen zunächst alle HDM-Daten als Eingangsgrößen übergeben werden. Daran anknüpfend können mittels statistischen Methoden sowie physikalischem Wissen jene Größen herausgefiltert werden, die maßgeblichen Einfluss auf die Modellgüte vorweisen.

### 4.2.1 Feature Engineering

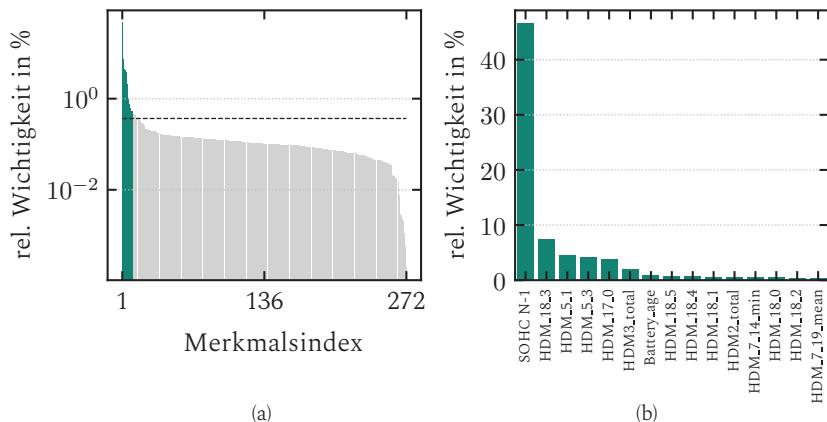
Die standardgemäße Anpassung verschiedener Modellparameter, wie beispielsweise die Einstellung der verdeckten Schichten oder der Hyperparameter, sind essentiell für das erfolgreiche Lernen. Ein weiterer vielversprechender Ansatz stellt der Prozess des Feature-Engineerings dar. [146] schlägt dazu folgende Definition vor:

„Feature Engineering is the process of transforming raw data into features that better represent the underlying problem to the predictive models, resulting in improved model accuracy on unseen data.“

Daran anknüpfend können, mithilfe von domänenübergreifendem Wissen durch Datenmodellierung und -untersuchung, relevante von weniger relevanten Merkmalen getrennt oder sogar neu generiert werden. Darunter fallen bereits die durchgeführten Schritte in Abschnitt 4.1.2. Insgesamt liegen aus der Summe aller verwendbaren HDMs inklusive der Erweiterung des Datensets durch die Fahrzeughistorie 1467 Eingangsgrößen vor. Durch das Entfernen von Nullspalten, deren Großteil durch die synthetisch erzeugten historischen Merkmale ins Datenset eingebracht werden, wird die Datenmenge deutlich verringert. Zudem sollen quasi-konstante Merkmale ebenfalls eliminiert werden. Damit sind jene Einträge gemeint, welche sich zwar nicht zeitlich konstant über die Datentupel verhalten, jedoch kohärent zueinander sind. So kann verhindert werden, dass redundante Informationen ins Modell gelangen. In der abschließenden Phase des groben Filterungsprozesses wird ein Schwellenwert festgelegt, um Merkmale zu eliminieren, die eine zeitliche Konstanz von 99,99 % aufweisen. Durch diese Selektionsschwelle wird die Anzahl an Merkmalen für eine detailliertere, feingranulare Analyse auf 272 reduziert.

**Korrelationsanalyse** Im Rahmen der folgenden Untersuchungen wird keine Korrelationsanalyse im herkömmlichen Sinne verwendet. Der Grund hierfür liegt in der speziellen Struktur sowie den physikalischen Informationen des Datensatzes begründet. Nichtlineare Beziehungen sowie eine Vielzahl von Datenverteilungen können von Standard-Korrelationsmetriken nicht angemessen erfasst werden. Daher werden alternative Analysemethoden als geeigneter erachtet, um die tieferen und komplexeren Muster innerhalb der Daten aufzudecken. Dies ermöglicht ein genaueres und umfassenderes Verständnis der zugrundeliegenden Phänomene. Demnach wird im Folgenden ein Random-Forest-Regressionsmodell entwickelt. Dieses

basiert auf einer Vielzahl an Entscheidungsbäumen und wird unter Verwendung des gefilterten Datensatzes trainiert. Ein signifikanter Vorteil des Random Forest Ansatzes ist seine Fähigkeit, eine Rangfolge der Merkmale gemäß ihrer relativen Bedeutung für die endgültige Vorhersage zu generieren. Die „relative Wichtigkeit“ gibt Aufschluss über den Einfluss jedes einzelnen Merkmals auf das Vorhersageergebnis. Abbildung 4.4 (a) zeigt in diesem Zusammenhang eine grafische Darstellung. Dabei wird die relative Wichtigkeit der 272 Merkmale auf der Ordinate in logarithmischer Skalierung dargestellt, während die Abszisse die Merkmale in absteigender Reihenfolge ihrer Bedeutung auflistet. Aus Gründen der Übersichtlichkeit wird auf die Benennung jedes einzelnen Merkmals verzichtet und lediglich ein numerischer Wert angegeben.



**Abbildung 4.4:** Relative Wichtigkeit je Eingangsgröße mit dem Gleichverteilungswert (schwarze Linie) (a) und quantitative Darstellung der relativen Wichtigkeiten der fünfzehn essenziellen Merkmale im Kontext der Korrelationsanalyse (b).

Die schwarz-gestrichelte horizontale Linie illustriert einen hypothetischen Gleichverteilungswert von 0,36 % pro Merkmal. Diese Wert ergibt sich, wenn jedes Merkmal die gleiche Bedeutung hätte. Die grafische Analyse zeigt, dass bestimmte Merkmale für das Vorhersagemodell besonders relevant sind. In den folgenden Abschnitten werden diese relevanten Merkmale detailliert betrachtet. In Abbildung 4.4 (b) wird eine detailliertere Darstellung der 15 relevantesten Merkmale präsentiert. Auf der Ordinate ist die Bedeutung jedes Merkmals in Prozentwerten dargestellt, während die Abszisse die spezifischen Merkmale auflistet. Im Folgenden sollen die Größen näher erläutert werden. Hierzu sind zur Verdeutlichung in Tabelle 4.1 die wichtigsten Merkmale aufgelistet.

An oberster Stelle steht die Größe  $SoH_C(n_{FAP} - 1)$ , das mit einer Relevanz von nahezu 47 % hervorsticht, begründet damit, dass es in direktem Zusammenhang zur Schätzgröße steht. Zudem weist der Wert des  $SoH_C$  zwischen verschiedenen Protokollen innerhalb eines einzelnen Fahrzeugs nur marginale Unterschiede auf. Somit stellt der  $SoH_C(n_{FAP} - 1)$  einen validen Ausgangspunkt für das entwickelte Modell dar. Dieses Modell nutzt vor allem diese Größe als Basis, um die fortschreitende Degradation der Batteriekapazität präzise zu prognostizieren. Die Merkmale in HDM 5 quantifizieren die zeitlichen Anteile innerhalb definierter Ladezustandsintervalle. Die Analyse erfasst speziell Zeitanteile in höheren  $SoC$ -Intervallen. Ein höherer Ladezustand korreliert mit einer erhöhten Klemmenspannung, was auf ein verringertes Anodenpotential und ein erhöhtes Kathodenpotential hindeutet. Dies führt besonders bei einer Graphitanode mit niedrigrem Potential zu einer Beschleunigung der Seitenreaktionen, insbesondere der Verdickung der SEI (s. Abschnitt 2.1.3). Des Weiteren kann mittels der durchgeführten Analyse der Energie- bzw. Ladungsdurchsatz der Batterie (HDM 18) als dynamisches Merkmal identifiziert werden, um die Lastzyklen der Batterie über der Lebensdauer zu quantifizieren.

**Tabelle 4.1:** Detaillierte Übersicht der Merkmalsrelevanz, einschließlich quantitativer Wichtigkeitswerte  $\alpha$  und deskriptiver Erläuterungen, systematisch geordnet nach absteigender Bedeutung der einzelnen Merkmale.

Merkmal	$\alpha$ in %	Beschreibung
$SoH_C(n_{FAP} - 1)$	46, 62	$SoH_C$ des vorangehenden Protokolls
HDM_18_2	7, 38	Energiedurchsatz im Fahrmodus (Entladen)
HDM_5_4	4, 42	Zeitanteil im 4. $SoC$ -Intervall im Fahrmodus
HDM_5_3	4, 06	Zeitanteil im 3. $SoC$ -Intervall im Fahrmodus
HDM_17_3	3, 76	Anzahl der Hauptschützöffnungen
total_time_HDM_3	2, 05	Gesamtentladezeit im Fahrmodus
battery_age	0, 94	Gesamalter der Batterie
HDM_18_5	0, 74	Energiedurchsatz der Batterie (Laden)
HDM_18_4	0, 61	Ladungsdurchsatz der Batterie (Laden)
HDM_18_1	0, 54	Energiedurchsatz der Batterie (Rekuperation)
total_time_HDM_2	0, 51	Gesamtlaudezeit (Laden und Rekuperation)
HDM_7_14_min	0, 46	Historischer Minimalwert Anteil $\Delta SoC$ (Laden)
HDM_18_0	0, 43	Ladungsdurchsatz im Fahrmodus (Laden)
HDM_18_2	0, 42	Ladungsdurchsatz im Fahrmodus (Entladen)
HDM_7_19_mean	0, 37	Historischer Mittelwert $\Delta SoC$ (Laden)

Bei Hybridfahrzeugen korreliert die Nutzung der Batterie nicht zwangsläufig linear mit der Fahrzeuglaufleistung, da die Antriebsleistung je nach Fahrbedingungen und Betriebsmodus variabel zwischen dem Verbrennungskraftmotor und dem elektrischen Antriebssystem aufgeteilt wird. Insbesondere bei Langstreckenfahrten oder in Fällen, in denen das Fahrzeug

nicht extern aufgeladen wird, dominiert der Verbrennungsmotor, was zu einer verminderten Abhängigkeit der Fahrzeugperformance von der Batteriekapazität und folglich zu einer modifizierten Nutzungs dynamik der Batterie führt. Zusätzlich werden drei kumulative, zeitabhängige Größen als wesentlich identifiziert: Die Gesamtentladezeit während des Fahrens, die Gesamtladezeit, welche sowohl das externe Laden als auch die Rekuperation umfasst, sowie das Gesamtalter der Batterie. Die Gesamtentladezeit ist ein Indikator für die Nutzungsdauer und -intensität der Batterie, wobei längere Entladezeiten zu beschleunigter Alterung führen können [147]. Speziell die Gesamtladezeit externer Ladevorgänge, einschließlich der Rekuperation, übt einen wesentlichen Einfluss auf die Lebensdauer von Lithium-Ionen-Batterien (LIB) aus, was anhand der durchgeführten Analyse herausgearbeitet wird. Besonders die Bildung und Veränderung der SEI, die strukturellen Veränderungen der Elektrodenmaterialien unter mechanischer Belastung und die thermische Beanspruchung der Batteriezellen werden als Schlüsselfaktoren ausgemacht. Schließlich stellt das Gesamtalter der Batterie einen Indikator dar, der sowohl die chronologische Alterung als auch den Einfluss von Betriebsbedingungen berücksichtigt.

Im HDM 17 wird die Anzahl der vom Fahrzeug- oder Batteriesteu ergerät in sicherheitsrelevanten Situationen veranlassten Öffnungen der Hauptschütze dokumentiert. Die Hauptschütze unterbrechen in diesem Fall schlagartig die Stromleitung zwischen Hochvoltbatterie und Hochvoltverbrauchern. Dabei können hohe Strom spitzen auftreten, welche über einen Event-Zähler dokumentiert und gespeichert werden. Jede schlagartige Öffnung der Hauptschütze unter Last oder ohne Abbau der im Zwischenkreiskondensator gespeicherten Energie führt zu einer erhöhten induzierten Wärme einwirkung auf die Zelle, was wiederum zu einer beschleunigten Alterung führen kann.

Mit Hilfe der durchgeföhrten Korrelationsanalyse kann eine Reihe von Schlüsseleinflussgrößen und Parametern für die Batteriealterung in Hybride Fahrzeugen herausgearbeitet werden, die sowohl aus ingenieurtechnischer als auch aus physikalischer Perspektive sinnvoll erscheinen. Die so ermittelten relevanten Größen bilden die Basis für ein Batteriealterungsmodell. Während des Modelltrainings stellt sich jedoch heraus, dass die ausschließliche Nutzung dieser 15 Merkmale nicht ausreicht, um die Modellgüte zu verbessern. Dies deutet darauf hin, dass die verbleibenden Merkmale zusätzliche Informationen enthalten, die den Lernprozess unterstützen und zur Genauigkeit der Vorhersagen beitragen. Daher werden im Training, sofern nicht explizit darauf hingewiesen, weiterhin alle 272 Eingangsdaten verwendet.

## 4.2.2 Finaler Datenvorverarbeitungsprozess

Im vorherigen Abschnitt werden zunächst Datentupel ausgewählt, um inkompatible FAPs aus dem Datensatz zu eliminieren. Anschließend werden relevante Merkmale ausgewählt, um ausschließlich Merkmale für das Modell zu berücksichtigen, die einen signifikanten Informationsgehalt aufweisen. Nach diesen Vorverarbeitungsschritten werden nun maschinelle Lernmodelle entwickelt. Diese Algorithmen sind als Regressionsmodelle konzipiert und haben die Aufgabe, den  $SoH_C$  basierend auf den definierten Eingangsgrößen zu prognostizieren.

Abbildung 4.5 zeigt den Prozess des Modellaufbaus in Form eines Flussdiagramms. In der ersten Phase der Datenverarbeitung werden nur Protokolle ausgewählt, bei denen die Fahrzeuge eine Mindestlaufleistung von 1000 km oder einen  $SoH_C > 50\%$  aufweisen. Die Protokolle der Fahrzeuge im Auslieferungszustand werden nicht berücksichtigt, da sich die Fahrzeuge in

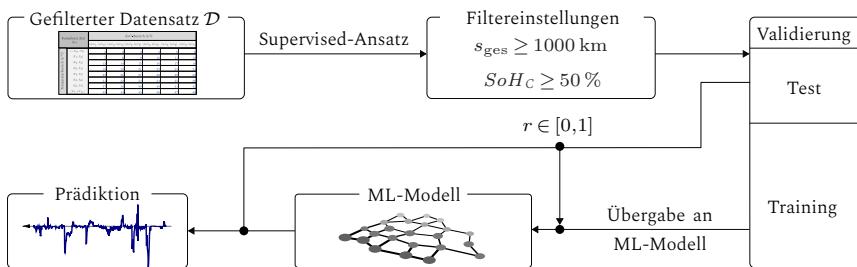
diesem Zustand noch nicht im realen Kundenbetrieb befinden, fehlen wesentliche Einflussfaktoren wie individuelle Nutzungsprofile, Ladeverhalten oder zyklische Belastungen der Batterie. Ein weiteres Problem dieser Protokolle besteht darin, dass sie überwiegend identische oder sehr ähnliche Werte für viele Merkmale aufweisen. Da alle Fahrzeuge mit einem nahezu identischen Anfangszustand ausgeliefert werden, tragen diese Daten wenig zur Variabilität des Trainingsdatensatzes bei und können die Lernfähigkeit der Modelle negativ beeinflussen. Zudem kann das BMS im Neuzustand aufgrund fehlender Langzeiterfahrungen noch keine präzisen Schätzwerte für die Degradation berechnen. Durch den Fokus auf Protokolle mit einer Mindestlaufleistung von 1000 km und einem  $SoH_C > 50\%$  wird sichergestellt, dass nur Datensätze mit aussagekräftigen Alterungsinformationen in das Modell einfließen. Dies erhöht die Relevanz der Trainingsdaten und verhindert eine Verzerrung des Modells durch unrepräsentative Anfangszustände.

Nach dieser initialen Datenselektion erfolgt die Segmentierung des Datensatzes während des Trainings in verschiedene Teilmengen. 70 % der Daten werden als Trainingsdaten zugewiesen, 20 % als Testdaten und 10 % als Validierungsdaten. Anschließend wird im Hinblick auf die Datenbeschaffenheit, den Wertebereich der Schätzgröße sowie die verwendeten Regularisierungstechniken die Varianznormierung:

$$x_{i,\text{norm}} = \frac{x_i - \bar{x}_i}{\sigma(x_i)} \quad \text{mit } \sigma = \text{Standardabweichung} \quad (4.3)$$

auf alle Eingangsgrößen angewendet.  $\bar{x}_i$  beschreibt hierbei den Mittelwert jedes Merkmals über alle Datentupel. Der Index  $i$  bezeichnet in diesem Zusammenhang das jeweilige FAP. Grundgedanke ist es eine Skalenvarianz

zu erreichen. Durch die Normalisierung wird der Einfluss unterschiedlicher Wertebereiche der Eingangsgrößen neutralisiert. Diese Notwendigkeit ergibt sich zudem aus dem Umstand, dass die in Unterabschnitt 2.2.1 beschriebene L<sub>2</sub>-Regularisierung auf dem Vergleich der absoluten Werte verschiedener Eingangsgrößen basiert.



**Abbildung 4.5:** Grafische Illustration des Datenaufbereitungsprozesses.

Konkret bedeutet dies, dass im Rahmen dieser Regularisierungsart größere Werte aufgrund des quadratischen Strafterms stärker gewichtet werden als kleinere Werte. Wenn Merkmale, die in unterschiedlichen Einheiten gemessen werden, ohne entsprechende Normalisierung in das Modell einfließen, führt dies zu einer unverhältnismäßigen Bestrafung und kann die internen Mechanismen des Modells verzerren. Diese Maßnahme bildet den Abschluss des gesamten Prozesses, sodass die aufbereiteten Daten nun an das jeweilige Modell übergeben werden können.

## Zukunftsprädiktion

Nachdem der vorangehende Abschnitt die Aufbereitung der Daten vor Beginn des Trainings behandelt, soll nun eine Vorgehensweise nähergebracht

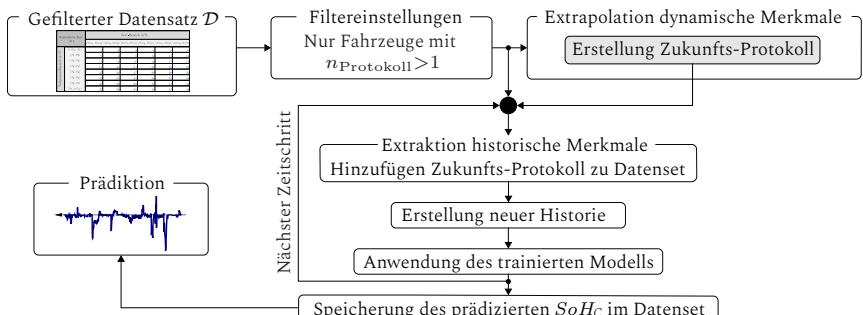
werden, um mit einem bereits trainierten Modell eine Vorhersage der Batteriegesundheit zu erstellen. Für diese Prognose ist es erforderlich, das Modell auf Daten anzuwenden, die zukünftige Zustände abbilden. Diese prognostischen Daten müssen jedoch zunächst generiert werden. Die Methodik zur Erstellung dieser prognostischen Daten und die Anwendungsweise des Modells in diesem spezifischen Kontext werden im nachfolgenden Abschnitt detailliert erörtert.

**Extrapolation zum Zukunfts-Protokoll** Für die Generierung prognostischer Daten wird eine Methodik angewendet, die auf der Verwendung dynamischer Merkmale basiert (s. Abschnitt 4.1.2). Es wird postuliert, dass diese Merkmale eine lineare Zunahme in Korrelation mit dem fortschreitenden Alter der Batterie aufweisen. Neben dem Gesamtkilometerstand, der Anzahl der Schützöffnungen (HDM 17) und des Ladung-/Energiedurchsatzes der Batterie (HDM 18) werden für die anstehende Untersuchung zudem die Gesamtzeiten aus HDM 2, HDM 3 sowie HDM 4 betrachtet. Im Erstellungsprozess des Zukunftsprotokolls erfolgt eine lineare Extrapolation der dynamischen Merkmale in Abhängigkeit vom Alter der Batterie. Konkret bedeutet dies, dass ein spezifischer Zeitrahmen definiert wird, innerhalb dessen das Modell Vorhersagen treffen soll. Basierend auf diesem Zeitrahmen werden die dynamischen Merkmale entsprechend einer linearen Trajektorie extrapoliert, um eine konsistente und zukunftsorientierte Datenbasis für die Modellprognosen zu schaffen.

In der Konstruktion des Zukunftsprotokolls werden die statischen Merkmale als konstant angenommen. Diese Annahme basiert auf der Prämisse, dass das Nutzungsverhalten des Fahrzeugbesitzers bezüglich gefahrener

Strecken, externem Laden, elektrischem Fahren sowie dem Energiedurchsatz der Batterie sich gleichbleibend verhält. Dies ist eine notwendige Bedingung für die Erstellung von Zukunftsdaten. Des Weiteren gilt es im weiteren Verlauf der Arbeit zu diskutieren, ob sich die Anzahl der Öffnungen der Batterieschütze im Laufe der Zeit linear extrapoliieren lässt.

**Anwendung auf das Zukunftsprotokoll** Die Generierung des Zukunftsprotokolls stellt lediglich eine initiale Phase im Prozess dar. Wie durch [148] und [144] vorgestellt, ist für eine effektive Verarbeitung durch das bereits trainierte Modell eine weitere Aufbereitung dieser Daten erforderlich. Abbildung 4.6 illustriert diesen Prozess in seiner Gesamtheit, beginnend bei der Datengenerierung bis hin zur finalen Zukunftsprognose. Diese Darstellung bietet einen detaillierten Überblick über die sequenziellen Schritte, die zur Transformation von Rohdaten in modellkonforme Prognosedaten erforderlich sind.



**Abbildung 4.6:** Ablaufdiagramm der Zukunftsextrapolation und Anwendung des trainierten Modells.

Zu Beginn wird der gefilterte Datensatz von Fahrzeugen bereinigt, die ausschließlich über ein einzelnes Protokoll verfügen. Diese Eliminierung ist

notwendig, da bei diesen Fahrzeugen die Voraussetzung für eine lineare Extrapolation aufgrund fehlender, adäquater Stützstellen nicht erfüllt ist. Diese Daten aus der Vergangenheit werden dann wie zuvor beschrieben zum Zukunfts-Protokoll extrapoliert. Infolge der historischen Erweiterung jedes Protokolls, welche Informationen aus der Vergangenheit einschließt, erfolgt zunächst eine Integration der Zukunftsdaten. Die Zusammenführung ist von essentieller Bedeutung, um daraufhin eine präzise Berechnung der historischen Daten für die prognostizierten Zukunftsdaten vorzunehmen. Durch den Prozess wird gewährleistet, dass die Zukunftsprognosen nicht nur auf aktuellen, sondern auch auf historisch relevanten Daten basieren, wodurch die Genauigkeit und Zuverlässigkeit der zukünftigen Vorhersagen verbessert wird.

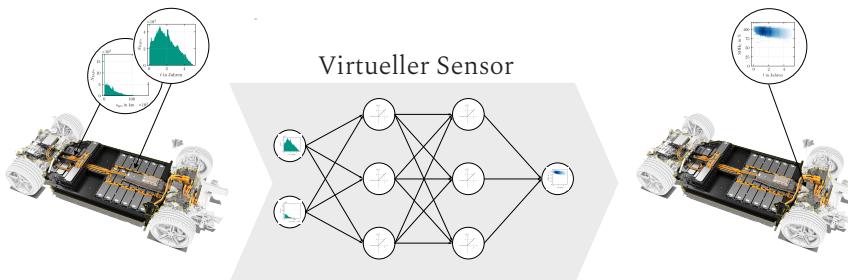
Hierbei wird zudem der  $SoH_C(n_{FAP}-1)$  aus dem vorangehenden Fahrzeug-Anwendungsprotokoll integriert. Nach Abschluss dieser Phase ist das trainierte Modell einsatzbereit, um eine Vorhersage für die neu aufbereiteten Daten zu generieren. Diese prognostizierte Information wird anschließend im Datensatz gespeichert, woraufhin der Prozess zum nächsten zukünftigen Zeitintervall übergeht. Hier beginnt erneut die Extrapolation zum Zukunfts-FAP, und das Verfahren wiederholt sich in einer iterativen Schleife. Dieser Algorithmus ermöglicht es, eine beliebige Anzahl von zukünftigen Zeitschritten zu simulieren und zu analysieren.

## 4.3 Modellierung des realen Systemverhaltens

### 4.3.1 Auswahl und Vergleich der Modellarchitekturen

Ziel dieser Arbeit ist der Entwurf verschiedener maschineller Lernmodelle, welche eine Approximation des Alterungsverhaltens der Hochvoltbatterie

ermöglichen. Die grundsätzliche Struktur für diese Problemstellung ist in Abbildung 4.7 verdeutlicht.



**Abbildung 4.7:** Schematische Darstellung eines maschinellen Lernmodells für den Einsatz als virtueller Sensor.

**Künstliche Neuronale Netze** Wie bereits im Rahmen des Standes der Forschung diskutiert, sollen für die Schätzung des Gesundheitszustandes LSTM-Zellen verwendet werden, um zeitliche Abhängigkeiten aus den Messdaten erlernen zu können. Um reproduzierbare Ergebnisse zu erhalten, ist es notwendig, die bereits erwähnte Anfangssymmetrie der Gewichte vor Trainingsbeginn zu brechen. Dieser Effekt wird durch die zufällige Initialisierung mittels einer skalierten Standardnormalverteilung erreicht. Neben den Gewichten muss auch der interne Zustand der rekurrenten Zellen initialisiert werden. [51] schlägt vor, den internen Zustand analog zu den Gewichten des Netzes zufällig zu wählen. Alternativ kann der initiale Zustand – neben den Gewichten – als zusätzliche Variable neben durch den Optimierer gelernt werden [149]. Für diese Methode und die folgenden Entwicklungsschritte des Modells wird die zufällige Initialisierung gewählt, da sich dadurch ein Abbauen des Fehlers gezeigt hat und das Modell so zu einer robusteren Schätzung gelangt.

Ein weiterer Punkt, der während des Trainings beachtet werden muss, ist die Wahl der Regularisierungsart. Um von Beginn an einer Überanpassung entgegenzuwirken und somit die Generalisierungsfähigkeit zu verbessern, soll bereits an dieser Stelle die Art der Regularisierungstechnik festgelegt werden. [150] zeigt, dass Dropout als effektiver Regularisierungstyp auf die Eingangs- und Ausgangsgewichte, jedoch nicht auf die Gewichtungen der rekurrenten Kanten angewendet werden sollte.

**Erste Abschätzung** Für eine erste Abschätzung der Approximationsqualität durch ein rekurrentes Netz mit LSTM-Zellen wird zunächst eine einfache Modellarchitektur gewählt. Üblicherweise wird als Aktivierungsfunktion der vorwärtsgerichteten Kante die ReLU-Funktion gewählt, für die der rekurrenten Kante die logistische Sigmoid-Funktion. Diese Kombination ergab auch im Rahmen einer Parameterstudie die vielversprechendsten Ergebnisse. Von dieser Schicht aus werden alle internen Zustände anschließend direkt auf das einzelne Neuron der Ausgabeschicht mit einer linearen Aktivierungsfunktion projiziert, welches die Funktion der wertkontinuierlichen Ausgabe des Modells übernimmt.

In [151] wird vorgeschlagen die Erweiterung des Eingangs und Ausgangs rekurrenter Einheiten durch zusätzliche vorwärtsgerichtete Schichten vorzunehmen. Für die Prädiktion soll demnach durch Addition einer zusätzlichen vorwärtsgerichteten Schicht zwischen der Eingabeschicht und der rekurrenten Schicht versucht werden, die Modellgüte zu verbessern. Diese in die Netzarchitektur ergänzte Schicht wird im Folgenden als Projektionsschicht bezeichnet. Aufgrund der Normalisierung der Daten zwischen  $[0, 1]$  wird für die Aktivierung dieser Schicht bewusst die ReLU-Funktion verwendet. Ausschlaggebend hierfür ist der Wertebereich dieser Funktion, welcher analog zu den skalierten Daten für das Intervall  $\text{ReLU}(z) : \mathbb{R} \mapsto$

$[0, \infty)$  definiert ist. Es wird davon ausgegangen, dass das Netz dadurch abstraktere Zusammenhänge in den Eingangsdaten erkennt.

**Anpassung der Zielfunktion** Um das Leistungsmaß des Neuronalen Netzes bewerten zu können, wird für die bisher durchgeführten Modellbildungsschritte der Mean Squared Error verwendet. Dieser bildet die quadratische Summe des Abstands jedes geschätzten Wertes  $\hat{y}$  zum Referenzwert  $y$ . Da die Ergebnisse der bereits abgeschlossenen Maßnahmen gemein haben, dass Ausreißer sowie Fahrzeuge mit hoher Laufleistung und geringem  $SoH_{mathrm{C}}$  schwer vorhergesagt werden, wird anschließend die Modifikation der Verlustfunktion des Netzes angestrebt. Die Implementierung des Codes in Python ist nachfolgend im Quellcode 4.1 aufgeführt.

```
1 import tensorflow as tf
2
3 def custom_loss(y_true, y_pred, scale=500.0, min_scale
4     ↪ = 1.0):
5     """
6         This function computes a user-defined loss metric
7         on the given problem
8
9         Parameters
10        -----
11        y_true: TensorFlow Tensor Object
12            the observed time series
13        y_pred: TensorFlow Tensor Object
14            the predicted time series
15        scale: float
16            scale factor
17            default: 500.0
18        min_scale: float
19            cut-off value
20            default: 1.0
21
22        Returns
```

```
22 -----
23 MPE: float
24 user-defined loss
25 """
26     absolute_difference = tf.math.square(y_true -
27         ↪ y_pred)
28     scaled_difference = tf.abs(scale * y_true)
29     clipped_scaled_difference = tf.clip_by_value(
30         ↪ scaled_difference, min_scale, scale * 2)
31
32     return tf.reduce_mean(absolute_difference /
33         ↪ clipped_scaled_difference, axis=-1)
```

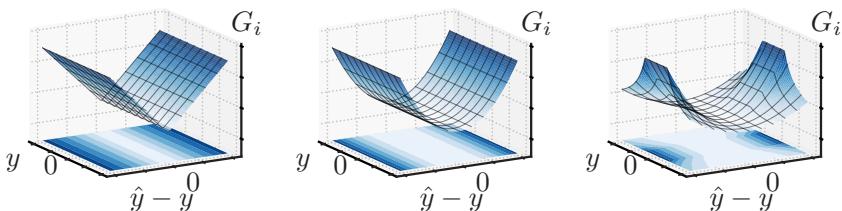
**Quellcode 4.1:** Implementierung der Verlustfunktion in Python

Grundgedanke dieser Funktion ist, Abweichungen bei geringeren absoluten Werten zu bestrafen. Dazu wird die Gewichtung des Datenpunkts abhängig von der Größe des tatsächlichen Zeitpunkts gewählt. Der Skalierungsfaktor `scale` gibt ein Maß für die Stärke der Bestrafung an. Um zu verhindern, dass der Einfluss der Werte, die nahe am Mittelwert sind, vernachlässigt werden, kann zusätzlich eine untere Schranke, die als Minimalgewicht interpretierbar ist, angegeben werden. Es sei explizit darauf hingewiesen, dass diese Funktion eine Erweiterung des Mean-Squared-Errors ist und nicht mit den vorimplementierten  $L^N$ -Normen verwechselt werden darf, die hohe Abweichungen bestraft. Für den vorliegenden Fall werden Abweichungen bei niedrigen Werten bestraft.

Durch die beschriebene Anpassung der Verlustfunktion wird erwartet, dass kleine Werte ( $SoH_C < 80\%$ ) innerhalb der Datenwolke besser prädiziert werden können, obwohl diese nur selten in den Daten vorzufinden sind. Im Umkehrschluss kann dies auf Kosten der Schätzgüte erfolgen. Die Performance des Algorithmus soll im weiteren Verlauf der Arbeit durch das eigens implementierte mittlere Fehlermaß

$$MPE = \frac{1}{n} \sum_{i=1}^n (\hat{y} - y)_i^2 \cdot (\max(|\beta \cdot y_i|, G_{\min}))^{-1} \quad (4.4)$$

erfolgen, für welches die Abkürzung *MPE* verwendet wird. Die Größe  $\beta$  beschreibt den Skalierungsfaktor, welcher die Stärke der Gewichtung an absoluten Funktionswerten bestimmt. Dieser wird auf Basis einer Versuchsreihe zu  $\beta = 500$  gewählt. Die Größe  $G_{\min}$  stellt eine untere Schranke dar, sodass Werte  $< G_{\min}$  dennoch mit dem festgelegten Wert von  $G_{\min} = 1$  berücksichtigt werden. Die Graphen in Abbildung 4.8 sollen den beschriebenen Sachverhalt veranschaulichen. Hier lässt sich – anders als bei der *MPE*-Metrik – für die *MAE*- und *MSE*-Metrik keine Abhängigkeit der Gewichtung  $G_i$  vom absoluten Funktionswert  $y$  erkennen.



**Abbildung 4.8:** Grafische Darstellung verschiedener Zielfunktionen: *MAE* (links), *MSE/RMSE* (mittig) und *MPE* (rechts) nach [152].

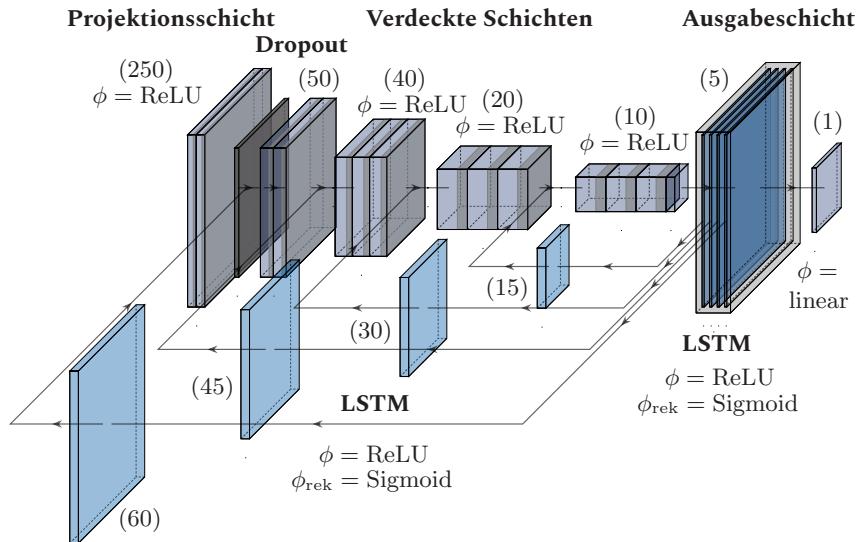
Die Stärke der Gewichtung kann zudem durch die Projektion der Graphenfläche in die  $y$ -( $\hat{y} - y$ )-Ebene verdeutlicht werden. Dunkelblaue Bereiche korrelieren hierbei mit einer höheren Gewichtung der Abweichungen.

**Einfluss ausgewählter Daten** Eine weitere Möglichkeit zur Steigerung der Modellgüte besteht darin, die vorhandenen Messdaten zu analysieren, um charakteristische Phänomene zu identifizieren (s. Unterabschnitt 4.1.2). Richtig gedeutet, können dadurch Rückschlüsse auf systembeschreibende Effekte gezogen werden, die als Grundlage für aussagekräftige Eingangsgrößen dienen. Die Selektierung informationsreicher Signale soll einerseits durch Ingenieursverständnis, andererseits durch das Heranziehen der Korrelationsuntersuchung aus Abschnitt 4.2.1 erfolgen. Für die zweitgenannte Maßnahme sollen lediglich Signale berücksichtigt werden, welche die Bedingung  $\alpha > 0,01$  erfüllen. Dadurch reduziert sich das Parameterset für den Trainingsvorgang auf 208 Merkmale. Dieses Datenset wird im weiteren Verlauf der Arbeit mit  $\mathcal{D}_V$  bezeichnet. Durch das Separieren nicht aussagekräftiger Merkmale soll das Modell in der Lage sein, Zusammenhänge in den Daten zu erkennen und dementsprechend seine Güte steigern. Dieser methodische Ansatz soll nun umgesetzt, ausgewertet und für einen Vergleich mit den bisherigen Ergebnissen herangezogen werden.

**Steigerung der Modelltiefe** Abschließend soll dieser Abschnitt Aufschluss über die endgültige Wahl der Netzkonfiguration geben und so die Methodenentwicklung des Neuronalen Netzes durch Erweiterung des bisherigen Modells um weitere verdeckte Schichten abrunden. Aus theoretischer Perspektive soll das dynamische Gesamtsystem durch die Erhöhung der Schichtenzahl zu einer noch komplexeren bzw. höheren Abstraktionsfähigkeit des Modells führen, da die Anzahl der zu berechnenden Interaktionen zwischen den Eingangsgrößen zunimmt. In Abbildung 4.9 ist die finale Netzarchitektur aufgeführt, welche im Folgenden beschrieben wird.

Die Neuronenanzahl der Projektionsschicht wird zu 250 gewählt. Daran anschließend wird eine Dropout-Schicht integriert, um die Generalisierungsfähigkeit des Modells zu verbessern. Zudem wird die Kapazität der LSTM-Ebenen auf fünf erhöht, da sich gezeigt hat, dass dadurch die Güte des Modells weiter verbessert werden kann. Es folgt eine sequenzielle Architektur, bestehend aus fünf LSTM-Schichten mit abnehmender Anzahl rekurrenter Einheiten von 60, gefolgt von 45, 30, 15 und schließlich 5 LSTM-Einheiten. Zwischen den rekurrenten Schichten sind jeweils vorwärtsgerichtete Schichten eingefügt. Die Neuronenanzahl in diesen Ebenen reduziert sich sukzessive von 50 über 40, 20 bis hin zu 10. Die kontinuierliche Reduktion der Neuronen- bzw. LSTM-Einheitenzahl pro Schicht mit Zunahme der Netztiefe erweist sich für die vorliegende Problemstellung als eine durchaus sinnvolle Herangehensweise. Dies gilt sowohl für die vorwärtsgerichteten Schichten als auch für die rekurrenten Schichten, bis hin zur wertkontinuierlichen Ausgabe des Einzelneurons. Die Grundidee stellt das stetige Herunterbrechen des Abstraktionsgrades bis zum skalaren Ausgabewert des Ausgangsneurons dar.

Zudem hat sich gezeigt, dass die Verwendung der ReLU-Aktivierungsfunktion in den vorwärtsgerichteten Schichten die besten Ergebnisse liefert. Aufgrund der Kapazitätssteigerung muss zusätzlich zum Dropout die L1- und L2-Regularisierung verwendet werden, da sich sonst eine Überanpassung des Modells nicht vermeiden lässt.



**Abbildung 4.9:** Abstrakte Darstellung der finalen rekurrenten Netzarchitektur mit den abgeleiteten Parametern. Die Anzahl der neuronalen Einheiten wird in runden Klammern angegeben.

**Support Vector Machine** Im Kontext der vorliegenden Thesis soll für die Schätzung der Batterielebensdauer ein spezifischer Ansatz verfolgt werden, welcher sich mit der Anwendung eines Support Vector Regressors (kurz: SVR) befasst. Dieser wird auf die zugrundeliegende Problemstellung hin optimiert, um Effektivität und Genauigkeit dieser Methode zu maximieren. Anders als bei KNNs, basiert die SVR auf der Prinzipien der SVMs, angepasst für das spezifische Regressionsprobleme. Die SVR zielt darauf ab eine Funktion zu finden, die eine kontinuierliche Ausgabe vorhersagt und dabei den Fehler innerhalb des Fehlertoleranzmaßes  $\varepsilon$  (s. Abschnitt 2.2.3) minimiert.

Zur Erzielung reproduzierbarer und verlässlicher Ergebnisse ist eine sorgfältige Initialisierung und Konfiguration der SVR unerlässlich. Während

bei KNNs die Initialisierung der Gewichte und des internen Zustandes von zentraler Bedeutung ist, konzentriert sich die SVR auf andere Schlüsselaspekte:

**Kernelfunktion K** Möglichkeit des Algorithmus in einem höherdimensionalen Raum zu arbeiten, ohne die Dimensionalität der Daten explizit zu erhöhen

**Kernel-Koeffizient  $\gamma$**  Sensitivität des Algorithmus auf die spezifischen Eigenschaften der Trainingsdaten

**Regularisierungsparameter C** Ausmaß der Regularisierung

**Fehlertoleranzmaß  $\epsilon$**  Festlegung der Toleranz des Algorithmus gegenüber Vorhersagefehlern

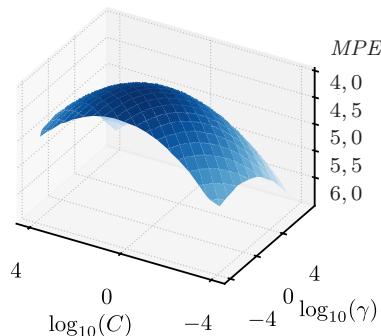
Neben der dargelegten Methoden der Datenvorverarbeitung aus Unterabschnitt 4.1.2 ist zur Erzielung eines robusten Modells eine Optimierung der Hyperparameter des Algorithmus in Bezug auf die spezifische Problemstellung von entscheidender Bedeutung. In diesem Kontext zeigen [153] im Rahmen einer umfangreichen Studie, dass die Umsetzung einer Hyperparameter-Optimierung eine entscheidende Rolle spielt. Basierend auf den Erkenntnissen werden in dieser Arbeit die Hyperparameter gemäß Tabelle 4.2 gewählt. Der Trainingsprozess erfolgt analog zu den KNNs mit der in Abschnitt 4.3.1 eingeführten Datenbasis  $\mathcal{D}_V$ .

**Tabelle 4.2:** Übersicht der Wertebereiche des jeweiligen Hyperparameters (hier: HP) im Rahmen der Parameterstudie mit RBF=Radiale-Basis-Funktion.

HP	Wertebereich	HP <sub>ideal</sub>
$K$	{Polynomial; RBF; Sigmoid}	RBF
$\gamma$	$(10^x, x = \{-4; -3,9; -3,8; \dots; 4\})$	0, 630
$C$	$(10^x, x = \{-4; -3,9; -3,8; \dots; 4\})$	0, 126
$\varepsilon$	$(10^x, x = \{-3; -2,5; -2; \dots; -0,5\})$	0, 100

Im Folgenden wird ein systematischer Ansatz verfolgt, bei dem für jede mögliche Kombination von Hyperparametern innerhalb der vorgegebenen Wertebereiche ein Modell trainiert und evaluiert wird. Zunächst wird aufgrund der Normalisierung der Daten bewusst die Radial-Basis-Funktion (kurz: RBF) verwendet (s. [154]). Ausschlaggebend hierfür ist der Wertebereich dieser Funktion, welcher analog zu den skalierten Daten für das Intervall  $RBF(z) : \mathbb{R} \mapsto [0, 1]$  definiert ist. Die Annahme besteht darin, dass das Modell dadurch in der Lage ist, abstraktere Korrelationen innerhalb der Eingangsdaten zu identifizieren. Diese Annahme kann auch im Rahmen der Studie bestätigt werden. Für die vorliegende Datengrundlage und die spezifische Problemstellung erweist sich im Rahmen der Untersuchung ein Wert des Fehlertoleranzmaßes  $\varepsilon = 0,1$  als besonders effektiv. Da dieser Parameter die Breite des Bereichs um die Regressionsfunktion definiert, in dem Abweichungen zwischen prädizierten und tatsächlichen Werten nicht bestraft werden (s. Abschnitt 2.2.3), ist seine Festlegung von wesentlicher Bedeutung. Die Wahl von  $\varepsilon$  scheint demnach aufgrund der spezifischen Datencharakteristik und der mit Unsicherheiten behafteten Schätzgröße besonders sinnvoll.

Abbildung 4.10 visualisiert die Ergebnisse für den Regularisierungsparameter  $C$  und den Kernel-Parameter  $\gamma$ . Im Sinne einer konsistenten Dokumentation und um eine Vergleichbarkeit der Modellgüte zu gewährleisten, erfolgt die Festlegung des Bewertungsmaßes anhand der in Abschnitt 4.3.1 eingeführten Leistungsmetrik  $MPE$  (z-Achse).



**Abbildung 4.10:** Verlauf des Fehlermaßes  $MPE$  als Funktion des Regularisierungsparameters  $C$  und des Kernel-Koeffizienten  $\gamma$  mit der optimalen Parameterkonfiguration im globalen Minimum.

Es resultiert eine in verschiedenen Blautönen abgebildete Zielfunktionsfläche, welche durch die diversen  $\{C, \gamma\}$ -Kombinationen aufgespannt wird. Die Deckkraft der Farbe korreliert mit dem Fehlermaß. Dunkelblaue Bereiche kennzeichnen eine gute Prognosegüte des Modells und dennoch einen kleinen  $MPE$ -Wert. Aus Gründen der Übersichtlichkeit läuft das Fehlermaß in positiver z-Richtung zu kleineren Werten hin. Das im Diagramm erkennbare globale Minimum repräsentiert die Kombinationen von  $C$  und  $\gamma$ , welche zur Minimierung des Vorhersagefehlers führen. Diese Kombination ist demzufolge als optimales Hyperparameterdupel der Support Vector

Machine in dem spezifischen Anwendungskontext anzusehen. Die optimalen Werte sind durch  $\log_{10}(C) = -0,9$  und  $\log_{10}(\gamma) = -0,2$  gekennzeichnet. Abschließend deutet diese spezielle Kombination auf eine sorgfältige Abstimmung des Gleichgewichts im Modell hin. Das abgeleitete Parameterset kann Tabelle 4.2 entnommen werden.

**Extreme Gradient Boosting** Bei der Bewältigung der Komplexitäten des prädiktiven Modellierens des Degradationsverhaltens innerhalb der verschiedenenartigen HDM-Matrizen, orientiert sich die folgende Methodik an der Entwicklung eines xGBoost-Schätzers für die zugrundeliegende Problemstellung. Wie bereits im Rahmen der Literaturrecherche herausgearbeitet, ermöglicht diese Methode die Erzielung einer hohen Robustheit bei gleichzeitiger Effizienz sowie Vermeidung von Überanpassungen und scheint demnach den vielversprechendsten der drei untersuchten Ansätze darzustellen.

**Parameterstudie der modellinternen Parameter** In einer mehrstufigen Raster-Suche werden verschiedene Parameterkombinationen getestet, um eine optimale Parametrierung des Algorithmus auf die vorliegende Fragestellung zu finden. Die Untersuchung von [155] zeigt die Auswirkung verschiedener Hyperparameter auf die Modellleistung innerhalb der xGBoost-Algorithmen. Insbesondere die Lernrate  $\kappa$  und die Anzahl der Entscheidungsbäume  $n_E$  beeinflussen signifikant die Modellperformance [155]. Der Fokus der Untersuchung soll demnach auf der Optimierung des Parametersets  $\{\kappa, n_E\}$  liegen. Des Weiteren soll zusätzlich noch die maximale Tiefe

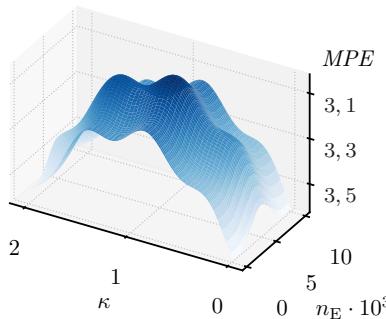
eines jeden Baumes  $d_{\max}$  untersucht werden. Eine Übersicht mit den dazugehörigen Variationsbereichen ist Tabelle 4.3 zu entnehmen. Zusätzliche Analysen weiterer interner Parameter zeigen, dass diese keinen signifikanten Mehrwert für die Optimierung der Modellqualität bieten. Lediglich die L1-Regularisierung wird für das Sicherstellen von Diversität sowie zum Entgegenwirken einer Überanpassung abgestimmt. Generell wird aufgrund der geringfügigen Einflüsse und unter Berücksichtigung der Beschränkungen in Bezug auf Rechenressourcen und -zeit, auf eine tiefergehenden Untersuchung dieser Parameter verzichtet.

**Tabelle 4.3:** Übersicht der Wertebereiche der jeweiligen Hyperparameter (hier: HP) im Rahmen der Parameterstudie mit dem optimalen Wert  $HP_{\text{ideal}}$ .

HP	Wertebereich	$HP_{\text{ideal}}$
$\kappa$	$\{0,1; 0,2; 0,3; \dots; 2\}$	0,792
$n_E$	$\{100; 200; 300; \dots; 10\,000\}$	5100
$d_{\max}$	$\{1; 2; 3; \dots; 500\}$	208

Auf Basis der zuvor beschriebenen Überlegungen wird die Optimierung mit einem Ensemble aus  $n_E = 100$  Entscheidungsbäumen und einer initialen Lernrate von  $\kappa = 0,1$  durchgeführt. Dies soll zunächst dazu dienen, die Trainingsdauer für nachfolgende Kombinationen von Hyperparametern zu minimieren. Die Lernrate muss sorgfältig in Relation zur Anzahl der Entscheidungsbäume im Ensemble kalibriert werden. Es wird beobachtet, dass umfangreichere Ensembles von Entscheidungsbäumen tendenziell niedrigere Lernraten erfordern, während bei kleineren Ensembles höhere Lernraten angemessen sein können. Dies wird in Abbildung 4.11 grafisch ver deutlicht. Die Illustration zeigt die Ergebnisse der Parameterstudie. Dabei

wird die Optimierungslandschaft durch  $\kappa$  und  $n_E$  aufgespannt. Analog zu der Untersuchung aus Abschnitt 4.3.1 nehmen die Werte für das Fehlermaß  $MPE$  aus Gründen der Anschaulichkeit in positiver z-Richtung ab.



**Abbildung 4.11:** Grafische Darstellung der Parameterlandschaft als Ergebnis der Parameterstudie mit geringstem Fehler  $MPE$  im globalen Minimum für das Parameterset aus  $\kappa$  und  $n_E$ .

Die Resultate der Studie zeigen, dass ein optimales Parameterset  $\{\kappa; n_E\}$  für  $\{0, 792; 5100\}$  gefunden werden kann. Dunkle Bereiche repräsentieren Modelleinstellungen mit hoher Prognosegüte. Die Koordinaten entsprechen dem globalen Minimum in Abbildung 4.11. Der zugehörige Wert des Fehlermaßes kann durch  $MPE = 2,99$  gefunden werden.

Für die Bestimmung der maximalen Baumtiefe  $d_{\max}$  des xGBoost-Algorithmus wird basierend auf den Ergebnissen einer an die Rastersuche vorangestellte Analyse empfohlen,  $d_{\max}$  entsprechend der Anzahl der Eingangsgrößen zu wählen. Diese Empfehlung stützt sich auf die in Abschnitt 4.3.1 diskutierten Maßnahmen, welche signifikante Erfolge in der Modellleistung zeigen.

Unter Berücksichtigung dieser Erkenntnisse und der spezifischen Eigen-  
schaften des vorliegenden Datensatzes wird  $d_{\max}$  auf 208 festgelegt. Diese  
Entscheidung reflektiert eine Optimierungsbalance zwischen Modellkom-  
plexität und Vermeidung von Überanpassung, basierend auf der gegebe-  
nen Anzahl der Merkmale und den in der genannten Analyse erzielten  
Erkenntnissen. Es wird erwartet, dass die spezifische Übereinstimmung  
von Baumtiefe und Merkmalsanzahl dazu führt, dass das Modell potenziell  
in der Lage ist, die umfassenden Zusammenhänge und den spezifischen  
Informationsgehalt der Daten zu extrahieren, diese in den Kontext der Bat-  
teriealterung zu setzen und letztlich zu lernen.

Die endgültige Parametrisierung wird mittels einer zehnfachen Kreuzvali-  
dierung überprüft. Dies dient dem Zweck, die Generalisierbarkeit und Ro-  
bustheit der Parameter über verschiedene Trainings- und Testdatensätze  
hinweg sicherzustellen, insbesondere im Hinblick auf die Heterogenität  
des zugrundeliegenden Datensatzes  $\mathcal{D}_V$ .



# 5 Ergebnisse und Diskussion

Kapitel 5 beginnt mit der formalen Einführung der Auswertungsmethodik in Abschnitt 5.1. In Abschnitt 5.2 werden Konsequenzen der umgesetzten methodischen Vorgehensweise für die jeweiligen Modellarchitekturen bewertet, quantifiziert und für einen Vergleich herangezogen. Darauf aufbauend wird in Abschnitt 5.3 die Validierung der Ergebnisse diskutiert. Daran anknüpfend wird in Abschnitt 5.4 die performanteste Modellvariante im Rahmen einer Einzelfahrzeuganalyse hinsichtlich einer Zukunftsprädiktion untersucht. Schließlich wird in Abschnitt 5.5 die Eignung sowie Übertragbarkeit der Methodik von PHEVs auf BEVs demonstriert.

## 5.1 Inhalt und Auswertungsmethodik

Im nachfolgenden Abschnitt werden zunächst die Ergebnisse der in Kapitel 4 vorgestellten statistischen Methoden dargestellt, klassifiziert und diskutiert. Die Auswertung erfolgt dabei auf der Gesamtheit der vorliegenden Datenbasis. Übergeordnet werden die durch die Modelle prädizierten Werte  $\hat{y}$  quantitativ mit den als Ground Truth deklarierten Realdaten  $y$  verglichen. Für die Bewertung der Modellgüte über den gesamten Definitionsbereich werden die in Abschnitt 2.2 und Abschnitt 4.3.1 definierten Fehlermaße herangezogen. Aufgrund der ungleichmäßigen Verteilung der

Realdaten über den Definitionsbereich der statistischen Methoden, ist zusätzlich zu untersuchen, ob und in welcher Weise die Vorhersagegüte vom Wertebereich der zu schätzenden Alterungsmetrik abhängig ist. Zu diesem Zweck erfolgt – nach abgeschlossenem Modelltraining – eine zusätzliche Klassifizierung der Daten in Intervalle  $\mathcal{I}^{(i)} = [SoH_C^{(i)} \pm 1\%]$  mit  $SoH_C^{(i)} = [86.0\%, 88.0\%, \dots 100\%]$ .

Anschließend wird die Fehlerverteilung für jedes dieser Intervalle durch ein *Gaussian-Kernel-Estimation*-Verfahren (kurz: GKE) kontinuierlich approximiert. Dadurch ergibt sich die Fehlerwahrscheinlichkeitsdichte  $\chi$  mit der integralen Normierungsbedingung

$$\int_{-\infty}^{\infty} \chi d(\hat{y} - y) = 1. \quad (5.1)$$

Als erste Qualitätsmetrik der Schätzung kann die Varianz der Fehlerwahrscheinlichkeitsdichte herangezogen werden. Ein exaktes Modell entspricht der Fehlerwahrscheinlichkeitsdichte  $\chi_{\text{ideal}} = \delta(y - \hat{y})$  mit der Dirac-Distribution  $\delta$ . Ein fehlerbehaftetes Modell, das jedoch keine systematische Verzerrung aufweist, liefert einen verschwindenden Erwartungswert  $\mu$  als erstes Moment der Verteilung

$$\mu := \int_{-\infty}^{\infty} \chi(\hat{y} - y) d(\hat{y} - y) = 0. \quad (5.2)$$

## 5.2 Ergebnisse der Modellentwicklung

### 5.2.1 Tiefes Künstliches Neuronales Netzwerk

**Erste Abschätzung** Für die im Rahmen des ersten Modellschritts vereinfachte Netzarchitektur wird ein Fehler für die Trainingsdaten von

$MSE_{Training} = 15,94$  und  $MSE_{Test} = 19,75$  für die Testdaten erreicht. Damit zeigt das Modell für diesen vereinfachten Ansatz keine ausgeprägte Generalisierungsfähigkeit. Die im Stand der Forschung postulierte, erfolgsversprechende Eignung von LSTM für die Schätzung des Degradationsverhaltens kann für diesen Ansatz noch nicht vollständig bestätigt werden. Zudem wird klar, dass die Verwendung weniger LSTM-Einheiten nicht geeignet ist, um das dynamische Systemverhalten des stark nichtlinearen Systems zu approximieren.

Die Ergebnisse zu Beginn des Kapitels zeigen deutlich die Grenzen einer zu niedrig gewählten Modellkomplexität auf. Zwar kann die Charakteristik des Systems bereits annähernd beschrieben werden, eine Schätzung des  $SoH_C$  für ältere Batterien kann jedoch nicht mittels dieses Ansatzes erfolgen. Zurückzuführen ist dies darauf, dass derartige Ereignisse vergleichsweise selten in den Trainingsdaten vorzufinden sind. Aus diesem Grund kann das Neuronale Netz in seiner einfachen Struktur für solche Sonderereignisse auf keine grundlegenden Zusammenhänge in den Daten schließen. Aus Sicht der Batterie ist es jedoch von großer Bedeutung, eben diese  $SoH_C$ -Bereiche adäquat vorauszusagen.

Die beschriebenen Ergebnisse des ersten Modellschritts zeigen, dass die Verwendung weniger LSTM-Einheiten zwar noch keinen universellen Approximator für die Schätzung des  $SoH_C$  eines dynamischen Systems darstellt. Dennoch kann der prinzipielle Verlauf, mit Ausnahme niedriger  $SoH_C$ -Werte, durch das Modell abgebildet werden.

**Steigerung der Netztiefe & Anpassung der Zielfunktion** Durch die Erweiterung des Modells um eine Projektionsschicht sowie 39 weitere LSTM-Einheiten der rekurrenten Schicht zeigt sich erwartungsgemäß, dass die Schätzgüte für die Testdaten ansteigt. Der Wert des Fehlers im

Bezug auf die Trainingsdaten halbiert sich nahezu auf  $MSE_{Training} = 8,44$  und im Bezug auf die Testdaten auf  $MSE_{Test} = 10,53$ . Die Abnahme des Fehlers zeigt auf, dass das erweiterte Modell durch die zusätzliche vorwärtsgerichtete Schicht abstraktere und für die rekurrenten Zellen aussagekräftigere Repräsentationen der Eingangsdaten erlernen kann. Des Weiteren wird die Substitution des  $MSE$  durch die in Abschnitt 4.3.1 eingeführte Verlustfunktion umgesetzt. Die erhöhte Netztiefe und Anpassung der Zielfunktion bewirkt eine bessere Schätzung des  $SoH_C$  auch bei niedrigen  $SoH_C$ -Zuständen. Dies ist eindeutig auf die angepasste Verlustfunktion zurückzuführen und bestätigt damit diese Herangehensweise. Der Fehler auf den Trainingsdaten ist durch  $MPE_{Training} = 7,37$  gegeben. Demgegenüber steht eine deutlich verbesserte Güte auf den Testdaten von  $MPE_{Test} = 10,23$ . Obwohl sich der bereits erwähnte Vergleich der vorangehenden Modellschritte anhand des Fehlermaßes nicht treffen lässt, haben Detailanalysen einzelner Fahrzeuge gezeigt, dass der prädizierte  $SoH_C$ -Wert eine deutliche Verbesserung auf die Testdaten aufweist. Das in diesem Arbeitsschritt weiterentwickelte Modell ermöglicht nun eine breitere Anwendung auch auf Daten mit geringen  $SoH_C$ -Werten bei verbesserter Prognosegüte auch für neue Datensätze.

Zusammenfassend kann gezeigt werden, dass die deutliche Tendenz zur Überschätzung der  $SoH_C$ -Werte aufgrund der eingeführten Fehlerfunktion reduziert werden kann, was von einer exakteren Abbildung der nichtlinearen Eigenschaften des Batteriesystems zeugt.

**Einfluss ausgewählter Daten** Nachdem nur ausgesuchte Eingangsgrößen gemäß Abschnitt 4.2.1 in Betracht gezogen werden, lässt sich eine deutliche Verbesserung der Ergebnisse festhalten. Mit der beschriebenen Vorauswahl lässt sich die Prognosegüte weiter steigern ( $MPE_{Training} =$

5, 76,  $MPE_{Test} = 8, 98$ ). Dies legt nahe, dass das Modell trotz der reduzierten Datenbasis enthaltene Zusammenhänge besser gelernt hat und somit die These der Leistungssteigerung des Modells durch Feature-Engineering stützt. Vor allem in der starken Korrelation des  $SoH_C(n_{FAP}-1)$  des vorangehenden Protokolls lässt sich hierfür eine Erklärung finden. Sowohl ein hoher Energiedurchsatz während des Entladevorgangs (HDM\_18\_2) als auch eine kurze Gesamtentladezeit haben eine hohe thermische Belastung der Batterie zur Folge, sodass chemische Reaktionen beschleunigt und letztlich Alterungsprozesse wie Lithium-Plating begünstigt wird (s. Unterabschnitt 2.1.3). Auch die Zeitanteile in mittleren bis hohen  $SoC$ -Bereichen während des Fahrens stehen damit in direktem Zusammenhang. Mechanische und elektrische Belastungen, die das Resultat der Hauptschützöffnungen (HDM\_17\_3) sein können, erweisen sich ebenfalls als einflussreiches Merkmal historisch des Alterungsverhaltens. Letztlich kann mittels des Gesamalters der Batterie ein Indikator für kumulative Alterungseffekte gefunden werden, da sich im Laufe der Zeit mikroskopische Schäden und chemische Veränderungen akkumulieren und diese Vorgänge die Kapazität der Batterie reduzieren.

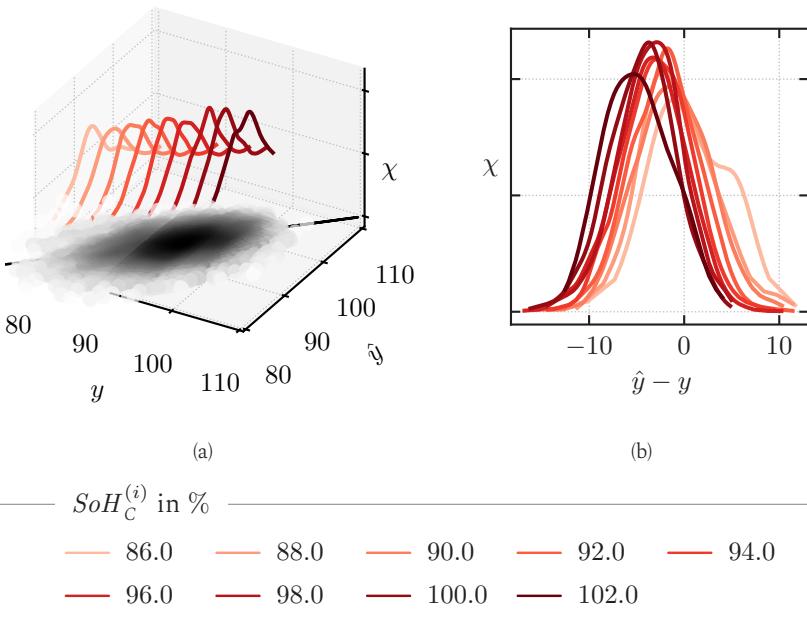
Dennoch weist das Modell weiterhin eine Neigung zur Unterschätzung des  $SoH_C$  auf. Dieses Phänomen lässt sich einerseits durch den Datenmagazin für geringe  $SoH_C$ -Werte interpretieren. Anderererseits liegt die Vermutung nahe, dass die rekurrente LSTM-Konfiguration des Netzes aufgrund der Häufigkeitsverteilung der Daten zu niedrigeren Werten von  $SoH_C \approx 90\%$  hin, diese Bereiche unzureichend präsentiert.

**Steigerung der Modelltiefe** Die Maßnahmen, welche in diesem Schritt des Modellbildungsprozesses umgesetzt werden, führen zu einer weiteren geringfügigen Verbesserung der Ergebnisse. Das durchgeführte Training

und die anschließende Anwendung auf das Testdatenset ergeben, dass die Werte des Trainingsfehlers bei  $MPE_{\text{Training}} = 5,57$  und des Testdatenfehlers bei  $MPE_{\text{Test}} = 8,72$  liegen. Die minimale Reduzierung des Fehlermaßes kann auf die tiefere Netzstruktur zurückgeführt werden. Es wird klar, dass die Wahl weiterer integrierter LSTM-Schichten zulässig ist, um abstraktere, zeitliche Zusammenhänge in den Daten zu identifizieren und zu lernen. Doch selbst das komplexere Modell weist Schwächen in der Prädiktion geringer  $SoH_C$ -Werte auf. Mögliche Erklärungen lassen sich wiederholt zu den Hypothesen aus dem vorangehenden Modellschritt finden. Global betrachtet liefert das RNN über die Länge der Testdaten noch keine zufriedenstellenden Ergebnisse. Jedoch lässt die ungenauere Vorhersage zu tieferen  $SoH_C$  vermuten, dass der Informationsgehalt sowie Qualität und Menge der Daten nicht ausreicht, um Gesetzmäßigkeiten für Alterungsprozesse dieser Art zu lernen. Es können somit eindeutig Grenzen von Neuronalen Netzen für eineartig einzigartige Datenstruktur aufgezeigt werden.

**Ergebnisbewertung der finalen RNN-Architektur** Die Ergebnisse der Schätzung mit dem in Abschnitt 4.3.1 spezifizierten Künstlichen Neuronalen Netzwerks sind nachfolgend in Abbildung 5.1 dargestellt. Dabei ist linkstehend in Abbildung 5.1 (a) die Verteilung aller geschätzten Werte  $\hat{y}$  über den Realdaten  $y$  dargestellt. Die Deckkraft der Datenpunkte korreliert dabei mit deren Auftrittshäufigkeit. Bei ca. 90 % lässt sich hier eine Anhäufung der Messdaten erkennen. Die durchgezogene schwarze Winkelhalbierende entspricht dabei einer idealen Modellantwort, d. h. Schätzung

und Messung sind identisch. Die auf der dritten Achse aufgetragene Fehlerwahrscheinlichkeitsdichte  $\chi$  ist hierbei, wie zuvor erläutert, für die verschiedenen Wertintervalle in Rottönen dargestellt. Zur besseren Sichtbarkeit sind die Werte zusätzlich in Abbildung 5.1 (b) über dem Schätzfehler  $\hat{y} - y$  aufgetragen.



**Abbildung 5.1:** Verteilung aller geschätzten Werte  $\hat{y}$  über den Realdaten  $y$  mit den in schwarz dargestellten Winkelhalbierenden für das RNN (a). Auf der vertikalen Achse ist die Fehlerwahrscheinlichkeitsdichte  $\chi$  für die Wertintervalle  $\mathcal{I}^{(i)} = [SoH_C^{(i)} \pm 1\%]$  mit  $SoH_C^{(i)} = [86.0\%, 88.0\%, \dots 102.0\%]$  gegeben. Darstellung von  $\chi$  über dem Schätzfehler  $\hat{y} - y$  (b).

Die Ergebnisse zeigen, dass ein starker Zusammenhang zwischen dem betrachteten Intervall  $\mathcal{I}^{(i)}$  und der sich ergebenden Dichtefunktion  $\chi$  besteht. Der Erwartungswert, bzw. das Maxima verschiebt sich mit zunehmendem  $SoH_C$  nach links. Diese Charakteristik entspricht einer systematischen Verzerrung des Modells, d.h. das KNN prädiziert überproportional häufig zu niedrige Werte. In guter Näherung korreliert die Varianz mit der Häufigkeit der Werte im betrachteten Wertebereich. Besonders für niedrige  $SoH_C$ -Werte lässt sich eine höhere Streuung der Fehlerwerte aufzeigen. Diese Varianz nimmt mit steigendem  $SoH_C$  zunächst ab, erreicht bei 93 % ihr Minimum und steigt dann wieder an.

Die Ergebnisse legen dar, dass das KNN durch die Aufnahme der statistischen Maßzahlen aus Abschnitt 4.1.2 in den Eingangsgrößenvektor sowie das parallele Eliminieren irrelevanter Merkmale in der Lage ist, den  $SoH_C$  zu schätzen. Zusammenfassend kann schließlich gezeigt werden, dass eine Kombination aus physikalischen Verständnis und statistischen Analysemethoden enormes Potential hinsichtlich einer Steigerung der Modellperformance birgt.

## 5.2.2 Ergebnisse der Support Vector Machine

**Erste Abschätzung** Ausgehend von einer simplen Modellarchitektur wird die SVM im Kontext der Schätzung des Degradationsverhaltens von Batterien zunächst ohne spezifische Anpassungen vorgenommen. Die verbesserten Werte der Fehlermaße im Vergleich zum KNN weisen darauf hin, dass die SVM bereits in ihrer einfachen Form eine gewisse Präzision in der Vorhersagefähigkeit aufweist, was ihre Eignung für die gegebenen Problemstellungen untermauert. In Relation zum vereinfachten Ansatz des

RNN aus Abschnitt 5.2.1 kann die Ergebnisqualität bereits um ca. 20 % verbessert werden.

Die erzielten Ergebnisse legen nahe, dass die SVM eine robuste Generalisierungsfähigkeit besitzt. Dies steht im Einklang mit der allgemeinen Forschung in Abschnitt 3.2, die SVMs als effektive Werkzeuge im Umgang mit komplexen Datensätzen hervorhebt. Ein kritischer Aspekt ist die Berücksichtigung der Seltenheit von Ereignissen mit stärker gealterten Batterien in den Trainingsdaten, d. h. mit niedrigerem  $SoH_C$ . Trotz dieser Datenbeschränkungen können bereits adäquatere Ergebnisse im Vergleich zu einer vereinfachten Modellarchitektur der KNNs erzielt werden. Dies ist ein Hinweis darauf, dass die SVM in der Lage ist, tiefergehende Muster und Zusammenhänge innerhalb der Daten zu identifizieren und zu nutzen, selbst wenn bestimmte Ereignisse nur selten auftreten. Besonders bemerkenswert ist, dass trotz der inhärenten Herausforderungen, die mit der Vorhersage in einem stark nichtlinearen System hervorgehen, die SVM in der aktuellen Konfiguration bereits eine vielversprechende Performance zeigt.

Die Ergebnisse aus der Anwendung der SVM im initialen Schritt des Modellierungsprozesses verdeutlichen, dass die SVM als alleinstehendes Werkzeug zwar nicht als universeller Schätzer des  $SoH_C$  eines dynamischen Systems fungieren kann, aber dennoch in der Lage ist, den grundsätzlichen Alterungszustand abzubilden.

**Anpassung Zielfunktion & Einfluss ausgewählter Daten** Analog zu den Resultaten aus Abschnitt 5.2.1 wird die Anpassung der Zielfunktion ebenfalls für den Support-Vector-Ansatz untersucht. Wie erwartet erweist sich diese Änderung als besonders effektiv bei der Schätzung von niedrigeren  $SoH_C$ -Werten. Die Umsetzung der modifizierten Verlustfunktion führt zu einem  $MPE_{Training} = 5,12$  auf den Trainingsdaten. Zudem kann

durch das Fehlermaß auf den Testdaten ebenfalls eine signifikante Steigerung der Modellgenauigkeit mit einem  $MPE_{Test} = 7,34$  verzeichnet werden. Die Anpassung der Verlustfunktion trägt auch hier maßgeblich dazu bei, die Generalisierungsfähigkeit des SVM-Modells zu verbessern und dessen Leistung auf unbekannten Daten erheblich zu steigern, ohne dabei die Komplexität des Modells zu erhöhen.

Darauf aufbauend führt die sorgfältige Auswahl und die damit einhergehende Reduktion an Merkmalen in den Eingangsgrößenvektor des Algorithmus gemäß der Untersuchungen bezüglich des Einflusses der ausgewählten Daten gemäß Abschnitt 4.3.1 für eine weitere signifikante Verbesserung der Ergebnisse. Die gezielte Vorgehensweise führt zu einer Abnahme des Fehler auf  $MPE_{Training} = 3,92$  im Bezug auf die Trainingsdaten sowie auf  $MPE_{Test} = 4,94$  im Bezug auf die Testdaten. Diese Ergebnisse zeigen, dass das SVM-Modell durch die optimierte Datenbasis zugrundeliegende Alterungszusammenhänge effektiver lernt. Dies unterstützt wiederholt die Annahme, dass eine gezielte Auswahl und Aufbereitung der Merkmale wesentlich zur Leistungssteigerung des Modells beiträgt.

Trotz einer allgemeinen Verbesserung der Vorhersagegenauigkeit zeigt das SVM-Modell eine gewisse Tendenz zur Unterschätzung des  $SoH_C$ . Diese Neigung kann teilweise durch den Mangel an Daten für niedrige  $SoH_C$ -Werte erklärt werden. Im Gegensatz zu Modellen mit rekurrenten LSTM-Konfigurationen ist eine Unterrepräsentation dieser Bereiche im SVM-Modell weniger signifikant für die Prognosegüte. Dies lässt sich darauf zurückführen, dass die SVM ihre Entscheidungsgrenze auf die relevanten Randdatenpunkte stützt und dabei durch die Maximierung der Margin eine robuste und generalisierbare Trennlinie schafft, die weniger anfällig für Verzerrungen ist. Es zeigt sich, dass die Architektur und der Mechanismus

der SVM eine effizientere Anpassung und Interpretation dieser spezifischen Datenbereiche ermöglichen.

**Ergebnisse der Parameterstudie** Die durchgeführte Hyperparameterstudie hat nicht nur die Relevanz von Hyperparameteroptimierungen in der prädiktiven Modellierung unterstrichen, sondern auch zu konkreten, quantitativ messbaren Verbesserungen geführt. Die Ergebnisse dieser Studie zeigen, dass sich der Fehler auf den Trainingsdaten um 8 % auf  $MPE_{\text{Training}} = 3,61$  und der Fehler auf den Testdaten um 11 % auf  $MPE_{\text{Test}} = 4,43$  im Vergleich zu den vorangehenden Schritten reduziert hat. Diese signifikante Verbesserung in der Fehlerreduktion ist ein direktes Resultat der sorgfältigen Abstimmung und Anpassung der Hyperparameter des SVM-Modells.

Insbesondere die Auswahl und Feinabstimmung der Kernelfunktion  $K$ , des Kernel-Koeffizienten  $\gamma$ , des Regularisierungsparameters  $C$  sowie des Fehlertoleranzmaßes  $\varepsilon$  haben entscheidend zur Verbesserung der Modellgenauigkeit beigetragen. Die Annahme in höherdimensionalen Räumen zu arbeiten und somit abstraktere Alterungszusammenhänge in den Daten zu lernen kann durch die Wahl der RBF bestätigt werden. Zudem kann durch das ermittelte Gleichgewicht zwischen  $C$  und  $\gamma$  ein optimaler Kompromiss hinsichtlich Modellgenauigkeit und der Fähigkeit zur Generalisierung auf unbekannte Daten gefunden werden.

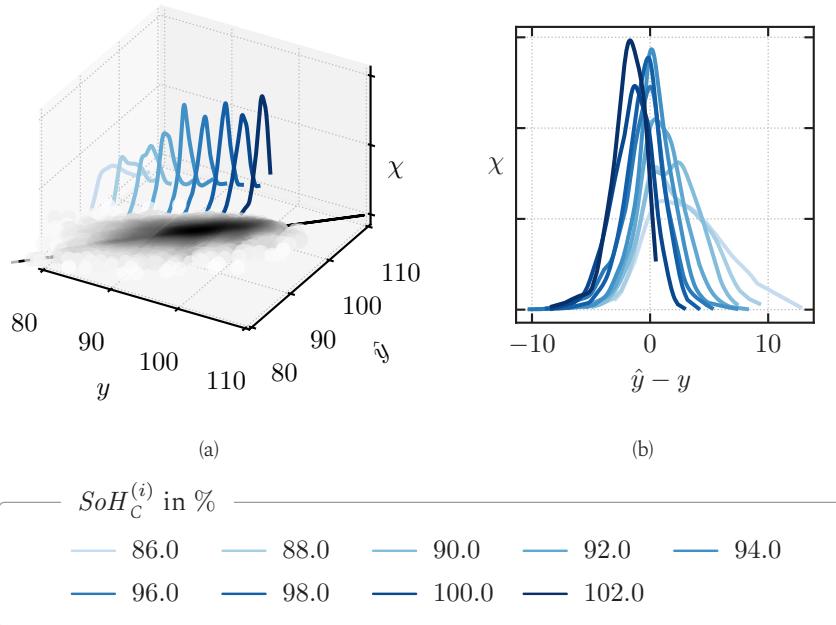
Zusammenfassend führt die systematische Auswertung verschiedener Kombinationen von Hyperparametern zur Identifizierung der optimalen Konfiguration des Modells auf die vorliegende Problemstellung. Weiter unterstreicht die Reduzierung der Fehlermaße die Effektivität der Hyperparameteroptimierung für die präzise und zuverlässige Schätzung des  $SoH_C$ .

Zudem bestätigt diese Studie die kritische Rolle einer gezielten Hyperparameteroptimierung bei der Entwicklung von SVM-basierten prädiktiven Modellen. Durch die präzise Anpassung der Hyperparameter kann eine signifikante Verbesserung der Modellperformance erreicht werden, was die SVM als ein robustes und effizientes Werkzeug für die Schätzung des Batteriezustands etabliert.

**Ergebnisbewertung der finalen SVM-Architektur** Die Ergebnisse der  $SoH_C$ -Schätzung des in Abschnitt 4.3.1 optimierten Support-Vector-Regressors sind im Folgenden durch Abbildung 5.2 dargestellt. Dabei lässt in der linken 3D-Darstellung in Abbildung 5.2 (a) eine Messdatenanhäufung bei ca. 92 % beobachten. Die engere Konzentration der Datenpunkte entlang der Winkelhalbierenden deutet auf eine generell höhere Vorhersagegüte verglichen mit dem KNN hin. Kritisch ist auch hier die Prädiktion kleiner  $SoH_C$ -Werte anzusehen. Eine Asymmetrie der Schätzung ist offensichtlich. Die Referenzdaten, welche für die Trainingsphase des Modells herangezogen werden, weisen eine deutlich höhere Anzahl an hohen  $SoH_C$ -Werten verglichen mit kleinen Werten auf. Diese quantitativen Abweichungen stellen die Hauptursache für die Approximationsunterschiede dar. Weiter zeigt eine detaillierte Betrachtung der 2D-Darstellung in Abbildung 5.2 (b), dass die Fehlerwerte der SVM eine breitgefächerte Verteilung aufweisen, die sich auf das Intervall  $(\hat{y} - y) \in [-10, 10]$  erstreckt.

Die Verschiebung des Erwartungswertes nach links mit zunehmendem  $SoH_C$  kann im direkten Vergleich zum KNN deutlich reduziert werden. Diese Charakteristik entspricht zwar immer noch einer systematischen Verzerrung des Modells, jedoch deutlich schwächer ausgeprägt. In guter Näherung korreliert die Varianz mit der Häufigkeit der Werte im betrachteten Werteintervall. Die Streuung der Fehlerwerte deutet darauf hin, dass

die SVM eine ausgewogene Mischung aus Unter- und Überschätzungen erzielt, was eine geringere Tendenz zu systematischen Fehlern impliziert. Dies geht mit einer robusteren und vielseitigeren Güte in der Vorhersagefähigkeit des Modells einher. Dennoch weisen speziell niedrige  $SoH_C$ -Werte eine höhere Streuung der Fehlerwerte auf.



**Abbildung 5.2:** Verteilung aller geschätzten Werte  $\hat{y}$  über den Realdaten  $y$  mit den in schwarz dargestellten Winkelhalbierenden für die SVM. Auf der vertikalen Achse ist die Fehlerwahrscheinlichkeitsdichte  $\chi$  für die Wertintervalle  $\mathcal{I}^{(i)} = [SoH_C^{(i)} \pm 1\%]$  mit  $SoH_C^{(i)} = [86.0\%, 88.0\%, \dots, 102.0\%]$  gegeben. Darstellung von  $\chi$  über dem Schätzfehler  $\hat{y} - y$  (b).

### 5.2.3 Ergebnisse des xGBoost-Modells

**Erste Abschätzung** Bei der Untersuchung eines nicht auf die Problemstellung angepassten xGBoost-Modells werden wie bisher die Fehlerwerte ermittelt. Im Unterschied zum simplifizierten RNN kann der Fehler auf die Testdaten um nahezu 28 % und zur SVM um ca. 7 % reduziert werden. Diese Ergebnisse lassen vermuten, dass das Modell in seiner vereinfachten Form im Vergleich zu den bisher entwickelten Algorithmen eine deutlich höhere Fähigkeit zur Generalisierung aufweist. Dies bestätigt bereits in der frühen Phase der Methodenentwicklung die Eignung der xGBoost-Algorithmik hinsichtlich der vorliegenden Problematik und unterstreicht die Wirksamkeit, die in der aktuellen Forschungsliteratur hervorgehoben wird. In diesem spezifischen Kontext bieten die erzielten Ergebnisse vielversprechende Anhaltspunkte für die nachfolgenden Untersuchungen. Obwohl die Resultate noch keine definitive Bestätigung für die Effektivität bei der Modellierung der komplexen Dynamiken des stark nichtlinearen Systems liefern, legen sie dennoch eine solide Grundlage für Weiterentwicklungen.

**Einfluss der Zielfunktion und des Feature-Engineerings** Die Analyse der Ergebnisse aus Abschnitt 5.2.1 führt zur Untersuchung der Zielfunktionsanpassung des xGBoost-Modells. Entgegen der Erwartungen zeigt sich, dass diese Änderung nur einen begrenzten Einfluss auf die Ergebnisqualität hat. Diese Beobachtung kann durch die Integration fortgeschrittenener Gradient-Boosting-Techniken, die durch iteratives Optimieren von Entscheidungsbäumen eine gesteigerte Vorhersageleistung ermöglicht, erklärt werden. Die Mechanismen weisen eine hohe Fehlerresistenz auf und ermöglichen es, auch ohne umfangreiche Modifikationen der Zielfunktion präzise Vorhersagen zu generieren.

Im Gegensatz dazu führt die Auswahl und Anpassung der Eingangsgrößen zu einer erheblichen Steigerung der Leistung des xGBoost-Modells. Durch das gezielte Feature-Engineering kann der Fehler auf den Trainingsdaten auf  $MPE_{\text{Training}} = 2,77$  und auf den Testdaten auf  $MPE_{\text{Test}} = 3,80$  reduziert werden. Dies deutet darauf hin, dass xGBoost besonders effektiv darin ist, komplexe Beziehungen und Muster in den Daten zu erkennen. Die Fähigkeit von xGBoost, durch iterative Verfeinerung jedes Baumes in der Ensemble-Lernmethode effektiv aus den Daten zu lernen, erklärt die hohe Wirksamkeit des Feature-Engineerings in diesem Kontext.

**Ergebnisse der Parameterstudie** Durch die Optimierung kann eine abgestimmte Hyperparameter-Konfiguration von  $\kappa = 0,792$  und  $n_E = 5100$  gefunden werden. Diese Einstellungen markieren einen signifikanten Unterschied zu den anfänglichen Parametern und führen zu einer deutlich präziseren Modellleistung. Insbesondere die Kombination einer reduzierten Lernrate mit einer erhöhten Anzahl von Entscheidungsbäumen unterstützt eine effektivere Datenmodellierung. Dies steht im Einklang mit der Erkenntnis, dass eine langsamere Anpassung an Trainingsdaten in umfangreicherer Ensembles zu einer präziseren Vorhersage führt.

Die Festlegung der maximalen Baumtiefe zu  $d_{\text{max}}$  auf 208 erweist sich ebenfalls sinnvoll. Indem  $d_{\text{max}}$  genau auf die Anzahl der Merkmale abgestimmt wird, kann das Modell komplexe Interaktionen und nichtlineare Beziehungen innerhalb des Datensatzes effektiver abbilden. Dadurch wird eine umfassendere und genauere Erfassung sowie Wiedergabe der zugrundeliegenden Datenstrukturen und -dynamiken ermöglicht. Diese Entscheidung stellt somit eine strategische Maßnahme dar, um das Potenzial des Modells zur Erkennung relevanter Beziehungen voll auszuschöpfen, ohne dabei die

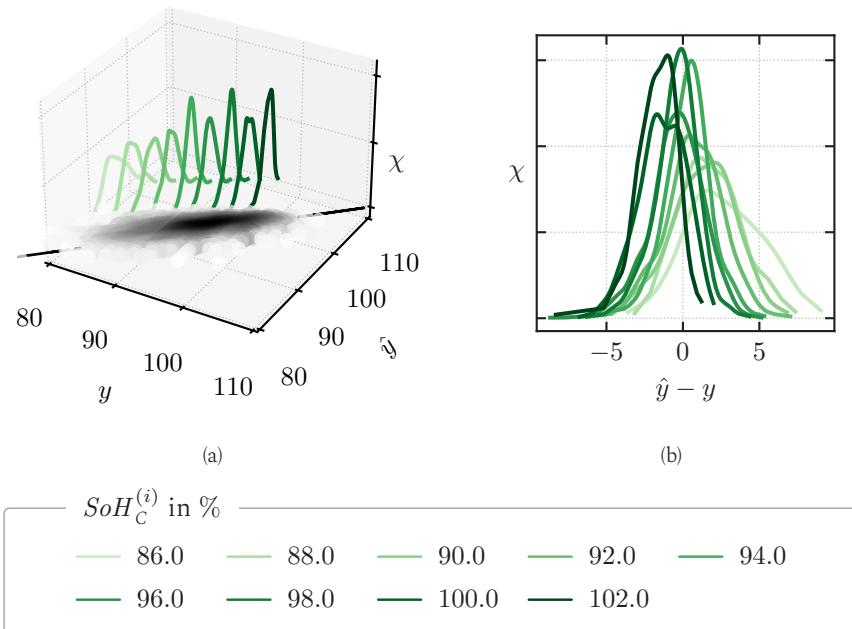
Gefahr der Überanpassung signifikant zu erhöhen. Dies spiegelt sich ebenfalls durch die quantifizierten Fehlermaßzahlen  $MPE_{\text{Training}} = 2,28$  und  $MPE_{\text{Test}} = 2,99$  wider.

Die Ergebnisse belegen eindrucksvoll die Effektivität der spezifischen Hyperparameter-Anpassung, vor allem unter Berücksichtigung der Heterogenität des Datensatzes. Zusammenfassend zeigt sich, dass eine gezielte und detaillierte Abstimmung der Hyperparameter wesentlich zur Steigerung der Leistungsfähigkeit des xGBoost-Modells beiträgt.

**Ergebnisbewertung der finalen xGBoost-Architektur** Die Ergebnisse der Schätzung des xGBoost-Modells sind in Abbildung 5.3 dargestellt. Auffällig ist die Konzentration der Datenpunkte entlang dieser ersten Winkelhalbierenden in Abbildung 5.3 (a) was auf eine hohe Genauigkeit des Modells in einem breiten Bereich von  $SoH_C$ -Werten schließen lässt.

Weiterhin zeigt Abbildung 5.3 (b) im Vergleich zu den vorherigen Modellen eine deutlich engere Verteilung der Fehlerwerte, begrenzt auf das Intervall  $(\hat{y} - y) \in [-5, 5]$ , während sich die Varianz der SVM im direkten Vergleich auf den Bereich  $(\hat{y} - y) \in [-10, 10]$  erstreckt. Diese kompaktere Verteilung spricht für eine geringere Streuung und damit für eine konsistenter Leistung des Modells über den gesamten Bereich des  $SoH_C$ . Während das KNN eine systematische Verzerrung mit einer Tendenz zu niedrigeren Schätzungen aufzeigt und die SVM zwar eine geringere, aber immer noch merkliche Verzerrung zeigt, liefert das xGBoost-Modell eine deutlich ausgeglichene und zuverlässigere Schätzung. Dies wird durch die symmetrischere Verteilung um die Winkelhalbierende und die reduzierte Varianz der Fehlerwerte belegt.

Die höhere Streuung niedrigerer  $SoH_C$ -Werte muss dennoch auch hier festgestellt werden. Die fehlenden Ereignisse in den Trainingsdaten ermöglichen es dem Modell nur bedingt diese Zustände abzubilden. Zwar kann die Ergebnisqualität auch für geringere  $SoH_C$ -Bereiche durch die umgesetzten Maßnahmen gesteigert werden, jedoch bleibt eine gänzlich zufriedenstellende Prädiktion aus.



**Abbildung 5.3:** Verteilung aller geschätzten Werte  $\hat{y}$  über den Realdaten  $y$  mit den in schwarz dargestellten Winkelhalbierenden für das xGBoost-Modell. Auf der vertikalen Achse ist die Fehlerwahrscheinlichkeitsdichte  $\chi$  für die Werteintervalle  $\mathcal{I}^{(i)} = [SoH_C^{(i)} \pm 1\%]$  mit  $SoH_C^{(i)} = [86.0\%, 88.0\%, \dots 102.0\%]$  gegeben. Darstellung von  $\chi$  über dem Schätzfehler  $\hat{y} - y$  (b).

## 5.2.4 Vergleich der Modellvarianten und abgeleitete Maßnahmen

Im Rahmen des Methodenentwicklungprozesses wird die Ergebnisqualität durch die sukzessiv angepassten Modellkonfigurationen kontinuierlich gesteigert. Sowohl die Implementierung einer individuell an die Problemstellung angepassten Verlustfunktion als auch die physikalisch-statistisch motivierte Datenauswahl stellen dabei den größten Benefit dar.

Die Resultate des RNN offenbaren eine systematische Tendenz zur Unterschätzung, insbesondere bei höheren  $SoH_C$ -Werten. Dieses Phänomen kann teilweise auf die Komplexität des Modells und eine daraus resultierende Überanpassung an die Trainingsdaten zurückgeführt werden, was eine adäquate Generalisierung erschwert. Zudem zeigt sich, dass die Performance des RNN stark von den einbezogenen Merkmalen beeinflusst wird, was in einem unausgewogenen Trainingsdatensatz zu Verzerrungen führen kann.

Im Gegensatz dazu weist die SVM zwar eine verbesserte Genauigkeit auf, offenbart jedoch ebenfalls eine Neigung zur Unterschätzung, wenn auch in geringerem Maße als das RNN. Diese Verzerrung der SVMs kann mit der Wahl der Kernel-Funktion sowie mit Herausforderungen in der Einstellung der Margen und der Regularisierung zusammenhängen. Beide Modelle, KNN und SVM, zeigen eine erhöhte Fehlerstreuung, besonders bei niedrigeren  $SoH_C$ -Werten.

Im Vergleich dazu demonstriert die xGBoost-Algorithmik seine Überlegenheit für diesen Anwendungsfall in Form der geringsten systematischen

Verzerrung über einen breiten Bereich des  $SoH_C$ . Diese robuste Performance wird durch eine gleichmäßige Fehlerverteilung und eine effizientere Handhabung von Unausgewogenheiten sowie komplexen, nichtlinearen Beziehungen im Datensatz unterstützt.

Insgesamt deutet der Vergleich darauf hin, dass xGBoost effektiver mit den Herausforderungen der Schätzungen umgeht, insbesondere im Hinblick auf niedrigere  $SoH_C$ -Werte. Die Analyse offenbart somit nicht nur die spezifischen Stärken und Schwächen jedes Modells, sondern vor allem die Bedeutung einer ausgewogenen Datengrundlage und sorgfältigen Merkmalsauswahl für die optimale Modellkonstruktion und -leistung. Die Resultate der finalen xGBoost-Modellarchitektur zeigen, dass ein Modell entwickelt werden kann, das durchaus geeignet ist, um die charakteristischen Alterungsmechanismen der Batterie zu lernen und zu schätzen. Es sei jedoch angemerkt, dass Ereignisse, welche die Eigenschaft niedriger  $SoH_C$ -Werte aufweisen, enorm an Schätzqualität einbüßen. Bedingt durch eine stark reduzierte Datenbasis aufgrund mangelnder Datenqualität.

## 5.3 Validierung der Modellergebnisse

In einem nächsten Schritt werden die vorgestellten Ergebnisse aus Abschnitt 5.2.3 auf Validität untersucht, um die Generalisierbarkeit und Robustheit des xGBoost-Modells bewerten zu können. Ausgehend von dem in Abschnitt 4.2.2 vorgeschlagenen Vorgehen werden für die Validierung des Modells vor dem Modelltraining erneut Testdaten aus dem Trainings- set entfernt, die der Validierung der Zukunfts-Prädiktion dienen. Für die Generierung dieser spezifischen Validierungsdaten werden ausschließlich

Fahrzeuge selektiert, die mehr als sieben Fahrzeuganalyseprotokolle aufweisen. Diese Selektion basiert auf der Prämisse, dass für eine valide Extrapolation mindestens vier FAPs als Stützstellen notwendig sind, während die verbleibenden Protokolle zur Validierung der Prognosen dienen. Um eine adäquate Vergleichbarkeit zwischen den prognostizierten Daten und der Ground Truth zu gewährleisten, wird das letzte verfügbare FAP zur Extrapolation in die Zukunft herangezogen, wobei derselbe Zeitpunkt berücksichtigt wird, der in den vorhandenen Ausgangsdaten verzeichnet ist. Diese methodische Vorgehensweise ermöglicht es, die Vorhersagen auf denselben Zeitpunkt zu beziehen, zu dem die Ground Truth-Protokolle vom Fahrzeug erfasst werden, und somit einen direkten Vergleich der Prognoségüte zu ermöglichen. Aufbauend auf dem in Unterabschnitt 4.2.2, Abbildung 4.6 beschriebenen Prozess wird das Vorgehen für die Validierung der Zukunfts-Prädiktion in Abbildung 5.4 veranschaulicht.

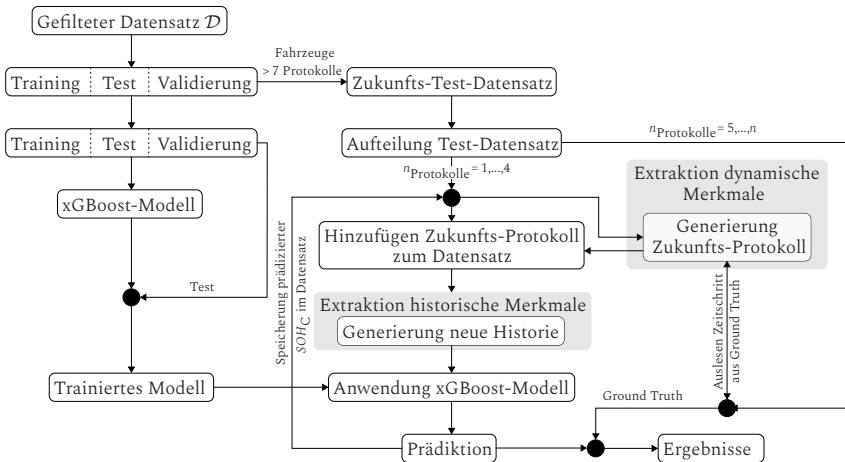


Abbildung 5.4: Prozess der Zukunftsprädiktion.

Im linken Teil des Schemas wird das Training der xGBoost-Pipeline gezeigt, ergänzt durch die zuvor beschriebene Maßnahme, in der Fahrzeuge mit mehr als sieben Protokollen aus dem Datensatz entnommen werden. Vier dieser FAPs dienen dann – wie beschrieben – als Grundlage für die Extrapolation und die darauf basierende Prädiktion. Die verbleibenden Protokolle, die das vierte FAP überschreiten, werden gemäß des rechten Pfads des Schemas als die empirischen Referenzdaten behandelt. In diesem Kontext wird der relevante Zeitschritt erfasst. Anschließend erfolgt eine Extrapolation der dynamischen Merkmale auf Basis des ermittelten Zeitschritts.

Das extrapolierte Zukunfts-Protokoll wird mit historischen Protokollen vereint, um die historischen Merkmale zu berechnen, die folglich für die Modellanwendung herangezogen werden können. Das bereits trainierte Modell kann in Form der gespeicherten Pipeline nun auf die zukünftigen Protokolle angewendet werden. Die resultierenden Prognosen werden einerseits mit der Ground Truth verglichen, um die Validierungsergebnisse zu erzielen, andererseits werden sie als  $SoH_C$  in das zukünftige Protokoll integriert. Dies ermöglicht die Aufnahme des aktualisierten Protokolls in das Gesamtdatenset und die Prognose des nachfolgenden Zeitschritts. Dieser Prozess wird für jedes Fahrzeug wiederholt, bis zu jedem Ground Truth-Protokoll eine entsprechende Prognose erstellt wird. Dieses Verfahren wird auf das gesamte Testset angewendet, wobei die daraus resultierenden Ergebnisse im Folgenden dargelegt werden.

Auf Basis der beschriebenen Methodik werden für die Validierung der Zukunftsprädiktion rund 25 % der für den Trainingsprozess zur Verfügung stehenden Protokolle entfernt. Für die eigentliche Bewertung verbleiben schließlich ca. 10 000 Ground-Truth-Protokolle ( $\mathcal{D}_Z$ ). Die Zukunftsprädiktion wird nun auf den mit  $\mathcal{D}_V$  trainierten Algorithmus (s. Abschnitt 4.3.1

- Einfluss ausgewählter Daten) angewendet. Ein Vergleich der Modellperformance zwischen den Realdaten aus der Vergangenheit ( $\mathcal{D}_V$ ) und den Zukunftsdaten erfolgt mittels der bereits beschriebenen Kriterien zur Bewertung der Prognosegüte (s. Tabelle 5.1).

**Tabelle 5.1:** Gegenüberstellung der Vorhersagequalität auf den Realdaten aus der Vergangenheit sowie den Zukunftsdaten mittels verschiedener Fehlermetriken.  $\Delta_{\text{rel}}$  gibt die relative Abweichung in Prozent an.

	<i>MAE</i>	<i>MSE</i>	<i>RMSE</i>	<i>MPE</i>
$\mathcal{D}_V$	1,17	2,82	1,68	2,99
$\mathcal{D}_Z$	1,35	2,63	1,91	3,52
$\Delta_{\text{rel}}$	15,4 %	28,7 %	19,1 %	17,8 %

In der Gesamtbetrachtung ist eine leichte Erhöhung der Fehlermaße zu beobachten. Genauer erhöht sich der *MPE* um 17,8 % auf *MPE* = 3,52. Dies legt nahe, dass das optimierte xGBoost-Modell aufgrund der reduzierten Datenbasis zwar alterungsspezifische Zusammenhänge wie erwartet weniger effizient erlernt, jedoch immer noch von einem universelles Prognosmodell für die Schätzung des  $SoH_C$  eines hochdynamischen Systems gesprochen werden kann.

Die Diskrepanz zwischen *MAE* und *MSE* deutet darauf hin, dass nicht nur der durchschnittliche Fehler, sondern auch die Anzahl größerer Fehler im Modell zugenommen haben. Nichtsdestotrotz deutet die moderate Verschlechterung in den Fehlermaßen darauf hin, dass der Extrapolationsansatz dynamischer Merkmale sowie der Beibehaltung statischer Merkmale eine valide Methodik darstellt. Abschließend kann gesagt werden, dass die

Eignung des xGBoost-Ansatzes für die vorliegende Problemstellung gegeben ist. Zur detaillierten Analyse der Vorhersagequalität erfolgt nachfolgend in Abschnitt 5.4 eine separate Untersuchung von vier Validierungsbeispielen auf Fahrzeugebene. Zudem fasst abschließend Abbildung 5.5 die Ergebnisse der entwickelten Modelle zusammen.

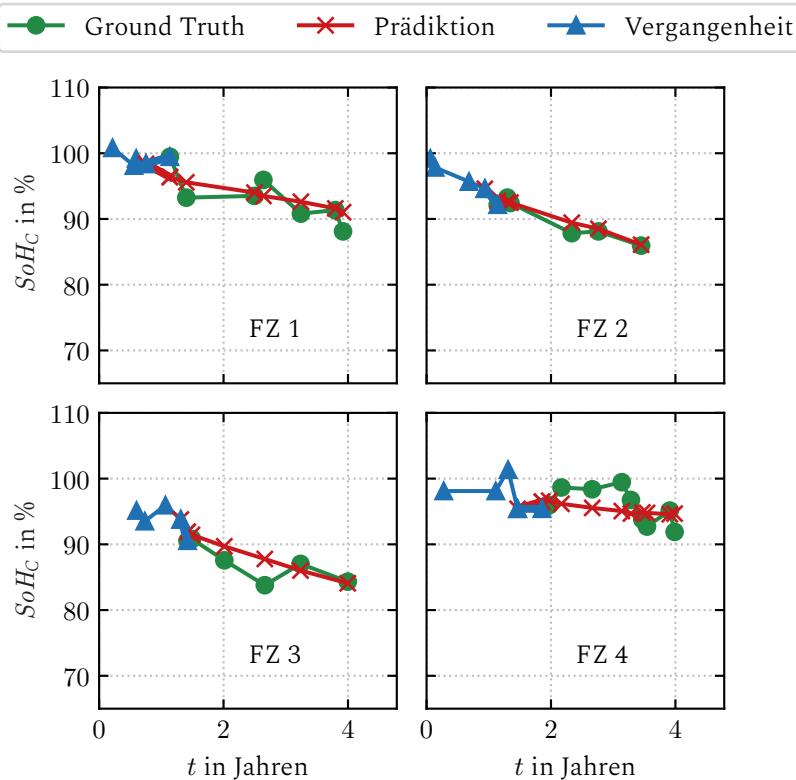
**Abbildung 5.5:** Übersicht der hergeleiteten Modellvarianten mit Angabe der verschiedenen Fehlermetriken, Hyperparametern sowie relevanten Bewertungsplots. Für das KNN können die Hyperparameter der abstrakten Darstellung des Neuronalen Netzes entnommen werden.

## 5.4 Einzelfahrzeuganalyse

Die tiefergehende Analyse einzelner Fahrzeuge ist entscheidend, um ein umfassendes Verständnis für die spezifischen Charakteristika und Verhaltensweisen auf Fahrzeugebene zu entwickeln. Die isolierte Betrachtung einzelner Fahrzeuge ermöglicht es, Abweichungen innerhalb des Modells und den Daten aufzudecken und Erkenntnisse über die Vorhersagequalität zu gewinnen. Hierzu sind in Abbildung 5.6 die Prädiktionsergebnisse von vier Beispiefahrzeugen dargestellt. Auf der Abszisse ist das Batteriealter in Jahren angegeben, während die Ordinate den  $SoH_C$  in Prozent repräsentiert. Die ersten vier Messprotokolle, dargestellt in blau, dienen als Stützstellen für die anschließende Extrapolation. An diese initialen Daten schließen sich die tatsächlich gemessenen Ground-Truth-Werte in grün sowie die korrespondierende Prädiktion des Modells in rot an.

Beginnend mit dem ersten Beispiefahrzeug wird eine Zukunftsprognose der  $SoH_C$  über einen Zeitraum von drei Jahren dargestellt (s. Abbildung 5.6 oben links). Die Extrapolation basiert auf vier Protokollen, die alle innerhalb des ersten Jahres erfasst werden. Der Verlauf der realen Messdaten, weist eine starke Variabilität mit mehreren Ausschlägen von bis zu 6 % im  $SoH_C$  auf, gefolgt von Erholungsphasen. Im Gegensatz dazu prognostiziert das Modell einen stetigen linearen Abfall. Während die Abweichung zwischen den prognostizierten Werten und den realen Messdaten augenscheinlich ist, muss die Genauigkeit der Ground Truth-Werte hinterfragt werden. Zwar sind wie bereits in Abschnitt 4.1.2 beschrieben, Relaxationsprozesse von bis zu 7 % durchaus plausibel, sollten jedoch insbesondere angesichts der bekannten möglichen Fehlermarge des Batteriemanagementsystems von bis zu 10 % bei der Bestimmung des  $SoH_C$  kritisch betrachtet werden. Daher lässt diese Diskrepanz keine abschließende Bewertung der

Modellqualität zu, unterstreicht jedoch die Herausforderungen bei der Verwendung des BMS- $SoH_C$  als Referenzgröße.



**Abbildung 5.6:** Funktionsverläufe des  $SoH_C$  für vier Beispelfahrzeuge in Abhängigkeit des Batteriealters  $t$  mit dem prädizierten Verlauf (rot), der Ground Truth (grün) und der Vergangenheit (blau).

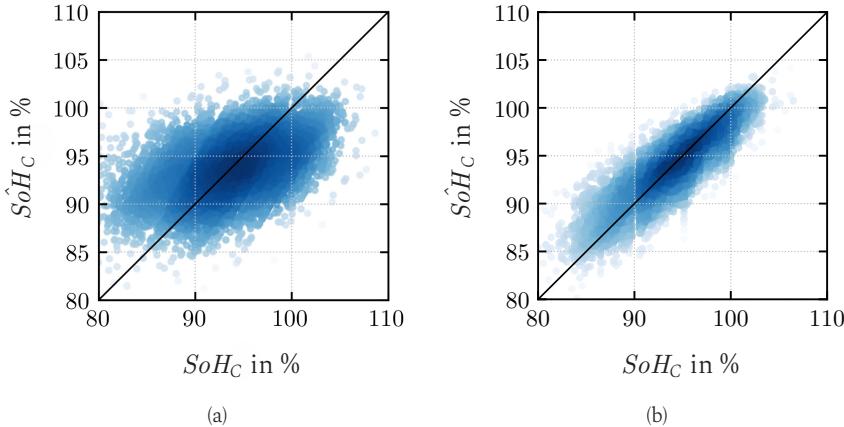
Die obere rechte Darstellung in Abbildung 5.6 zeigt ein Fahrzeug (FZ 2), in dem die extrapolierten Datenpunkte eine Reduktion des  $SoH_C$  von knapp unter 100 % auf annähernd 95 % über den Zeitraum eines Jahres zeigen.

Ab diesem Punkt schließt die Vorhersage an, die sechs weitere Protokolle für einen Zeithorizont von zusätzlichen 2,5 Jahren umfasst. Die tatsächlichen Messwerte und die Vorhersagen des Modells weisen einen nahezu deckungsgleichen Verlauf mit nur geringfügigen Abweichungen auf. Dies beweist eine hohe Präzision des Modells und zeigt, dass das Modell in der Lage ist bei schwach alternierenden  $SoH_C$ -Ground-Truth-Werten durchaus Degradationsprozesse zu lernen.

Das dritte Fahrzeug (FZ 3) weist, analog zu FZ 1, signifikante Regenerationserscheinungen des  $SoH_C$  auf. Die Regeneration ist in den Ground-Truth-Daten durch einen Anstieg der Kurve erkennbar. Diesen Verlauf bildet die Prädiktion jedoch nicht ab. Die Vermutung liegt nahe, dass dies zum einen an zu wenigen dieser charakteristischen Ereignisse in den Daten liegt und zum anderen an der unzureichenden Datenqualität. Beides in Kombination ermöglicht es dem Algorithmus nur in bestimmtem Maße, spezifische Alterungseffekte adäquat zu modellieren. Positiv anzumerken ist jedoch, dass das Modell trotz der Modellierungsdivergenz des  $SoH_C$  den letzten Wertepunkt nach ca. 4 Jahren zufriedenstellend voraussagt.

Abbildung 5.6 (FZ 4) zeigt erneut das Problem der temporären Regeneration der Batteriekapazität, diesmal über nahezu 1,5 Jahre. Daran anknüpfend ist ein nahezu linearer Abfall erkennbar, der den üblichen Degradationsprozess widerspiegelt. Die Häufung der Prädiktionswerte um den dominierenden  $SoH_C$ -Bereich von ca. 92 % des Lerndatensatzes legt nahe, dass das Modell tendenziell Ergebnisse liefert, die diesen häufig vertretenen Zustand reflektieren. Dennoch zeigt dieses Fahrzeug durch den leichten Anstieg bei ca. 1,75 Jahren anschaulich, dass durchaus auch Relaxationsprozesse vom Modell abgebildet werden können.

Zur Verdeutlichung sind abschließend die Ergebnisse der Prädiktion auf den Testdatensatz aus Abschnitt 5.2.3 sowie der Zukunftsprädiktion in zeitunabhängigen Bewertungsplots illustriert (s. Abbildung 5.7).



**Abbildung 5.7:** Vergleich der Prädiktionsergebnisse ( $\hat{SoH}_C$ ) für das Zukunftsdatenset  $\mathcal{D}_Z$  (a) und das Vergangenheitsdatenset  $\mathcal{D}_V$  (b) jeweils über den tatsächlichen Werten ( $SoH_C$ ).

Die Auswertung der Zukunftsprädiktion (Abbildung 5.7 (a)) veranschaulicht eine signifikante Ausweitung der Punktwolke um die Winkelhalbierende, was eine reduzierte Präzision der Vorhersagen im Vergleich zu dem Ergebnis aus Abbildung 5.7 (b) zeigt. Da über den  $SoH_C$ -Wert ungleich verteilte Datensätze zur Parametrierung des Prognosemodells zur Verfügung stehen und viele Daten für den Bereich  $90\% \leq SoH_C \leq 95\%$  vorliegen, nimmt die Prognosegüte mit sinkendem  $SoH_C$ -Wert ab. Dies zeigt die höhere Streuung der prognostizierten  $SoH_C$ -Werte im unteren  $SoH_C$ -Bereich (s. Abbildung 5.7 (a)). Insbesondere wird deutlich, dass  $SoH_C$ -Werte unterhalb 85 %

vom Modell systematisch überschätzt werden. Einerseits kann dieses Muster auf die in Unterabschnitt 2.1.3 beschriebene nichtlineare Degradationsdynamik zurückzuführen sein. Andererseits auf die Zukunftsextrapolation aufgrund des Fehlens entsprechender Trainingsdaten im unteren  $SoH_C$ -Spektrum.

## 5.5 Übertragbarkeit der xGBoost-Methodik auf andere Baureihen

Im weiteren Verlauf der Arbeit soll die Übertragbarkeit der in Kapitel 4 hergeleiteten Methodik basierend auf dem xGBoost-Algorithmus auf andere Fahrzeugderivate diskutiert werden. Hierzu wird die Anwendung des Prognoseverfahrens für PHEV und BEV verglichen. Es sei an dieser Stelle darauf hingewiesen, dass es sich im Folgenden um Fahrzeuge mit größer dimensionierten Batteriesystemen sowie verschiedenartigen Zellchemien handelt. Zudem dienen – analog zu den PHEVs – für das Training des BEV-Prognosemodells entsprechende BEV-Betriebsdaten.

Das Ergebnis, welches mit der für die PHEVs entworfenen Modellkonfiguration erzielt wird, weist erhebliche Unterschiede in der Schätzqualität auf. Niedrige  $SoH_C$ -Bereiche zeigen analog zu den PHEVs merkliche Probleme auf. Eine systematische Verzerrung zu größeren Werten zeigt auf, dass das xGBoost-Modell nur in gewissen Bereichen ausreichenden Ergebnisqualität führt. Das Modell ist aufgrund fehlender Daten nicht in der Lage, abstrakte Zusammenhänge für die beschriebenen, relevanten Bereiche zu finden. Auch die Erhöhung der Streuung muss konstatiert werden. Für eine quantitative Vergleichbarkeit kann das Fehlermaß für den Testdatensatz

$MPE_{\text{Test}}$  herangezogen werden. Im Vergleich zur PHEV-Flotte erhöht sich der Fehler um nahezu 120 % von  $MPE_{\text{PHEV}} = 2,99$  auf  $MPE_{\text{BEV}} = 6,66$ .

In der Betrachtung der Übertragbarkeit dieser Methodik spielen neben den divergierenden Nutzungsprofilen und den variierenden BMS-Strategien insbesondere die Unterschiede in der Zellchemie eine entscheidende Rolle. Die chemische Zusammensetzung der Zellen ist ein wesentlicher Faktor, der die Eigenschaften und das Verhalten des Batteriesystems grundlegend beeinflusst. Unterschiedliche Materialien in den Zellen führen zu variierenden Anfälligen für Degradationsmechanismen. Darüber hinaus haben sie einen signifikanten Einfluss auf die Lade- und Entladecharakteristika der Batterie, was wiederum für die Bestimmung des *SoH* von zentraler Bedeutung ist.

Zusammenfassend kann festgehalten werden, dass die für die Degradationsschätzung der PHEVs entwickelte Modellarchitektur keinen universellen Approximator für die Übertragung auf BEVs darstellt. Auf Basis der Ergebnisse wird angenommen, dass das Modell durch die Auswahl spezifischer Eingangsgrößen aus Abschnitt 4.2.1 nur in geringem Maße grundlegenden Zusammenhänge bildet. Aufgrund der Abweichungen des Modells müssten zur exakteren Abbildung der Realität aussagekräftigere Eingangsgrößen in das Modell mit aufgenommen werden. Zudem ist eine differenzierte Betrachtung der Zellchemie unerlässlich, um die Übertragbarkeit und Anwendbarkeit der Methodik auf unterschiedliche Fahrzeugkonzepte adäquat beurteilen zu können. Es wird demnach empfohlen, die Modellentwicklung gesondert für das jeweilige Fahrzeugsystem durchzuführen, um eine gesteigerte Ergebnisqualität zu erzielen.

# 6 Zusammenfassung und Ausblick

In Abschnitt 6.1 werden die wichtigsten Ergebnisse und originären methodischen Beiträge der vorangegangenen inhaltlichen Kapitel aufgeführt. Auf der Grundlage dieser Zusammenfassung werden die ursprünglich definierten Forschungshypothesen kritisch bewertet. Der letzte Abschnitt dieser Arbeit ist Vorschlägen für weitere Untersuchungen und Erweiterungen gewidmet.

## 6.1 Zusammenfassung

Motiviert durch die Einbeziehung historischer Betriebsdaten in die simulationsgestützte Prädiktion des Degradationsverhaltens von Lithium-Ionen-Batterien, identifiziert eine umfassende Datenauf- und verarbeitung den Einfluss bestimmter alterungsabhängiger Parameter als entscheidendes, bisher kaum untersuchtes Forschungsgebiet. Der Zielsetzung folgend wird ein Regressionsproblem für das Alterungsverhalten in Lithium-Ionen-Batterien modelliert und explizite Lösungen als Grundlage für nachfolgende Analysen geliefert.

### 6.1.1 Hauptergebnisse der Arbeit

Degradationsprozesse von Lithium-Ionen-Batterien beschreiben mechanische und chemische Vorgänge, welche abhängig von verschiedenen Einflussfaktoren ablaufen. Im Zusammenhang mit datengetriebener Alterungsmodellierung mittels maschineller Lernmethoden, werden in der Fachliteratur bisher fast ausschließlich Spannungs- und Stromprofile zyklisierter Zellen herangezogen. Im Rahmen der vorliegenden Arbeit findet der Methodenentwicklungsprozess jedoch auf Basis anderer einzigartiger Datenstrukturen statt. Genauer handelt es sich um historische Betriebsdaten, welche verschiedenste Zustände der Batterie abbilden. Die vorliegende Arbeit zeigt, dass sowohl Rekurrente Neuronale Netze, Support Vector Machines und xGBoost-Methoden als eine mögliche Form maschineller Lernmethoden die Approximation des komplexen Alterungsverhaltens der Batterie auf Basis der Betriebsdaten in dynamischen Systemen prinzipiell ermöglichen. Anhand des Degradationsparameters ( $SoH_C$ ) Plug-In-basierter Fahrzeugsysteme werden im Rahmen eines umfangreichen Modellentwicklungsprozesses unterschiedliche an die Komplexität der Schätzaufgabe angepasste Modellarchitekturen entworfen. Beginnend mit dem Rekkurenten Netz lässt sich aufführen, dass die Prädiktion geringer  $SoH_C$ -Werte eine Schwierigkeit darstellt. Um eine möglichst realitätsnahe Abbildung des Gesundheitszustandes zu erhalten, wird die Anpassung der zu minimierenden Zielfunktion des Netzes vorgenommen. Durch die stärkere Gewichtung von Abweichungen bei niedrigen Werten kann somit eine eindeutig verbesserte Approximationsgüte festgestellt werden. Anschließend werden im Rahmen eines aufwendigen Datenaufbereitungsprozesses einerseits irrelevante Größen aussortiert sowie neue aussagekräftige Merkmale generiert. Die Umsetzung beider Maßnahmen zeigt, dass der Algorithmus auf diese Weise zu deutlich präziseren Schätzungen kommt. Die Prädiktion

der finalen Netzarchitektur bestätigen zwar die Eignung von Rekurrenten Neuronalen Netzen zur Schätzung des Alterungszustandes, dennoch werden geringe  $SoH_C$  systematisch unterrepräsentiert, worauf ein robusterer Support-Vector-Regressor sowie ein xGBoost-Algorithmus untersucht werden. Für beide werden die vielversprechenden methodischen Maßnahmen des Feature-Engineerings sowie der Zielfunktionsanpassung ebenfalls umgesetzt. Die letztlich finale Feinabstimmung der modellinternen Parameter anhand einer umfassenden Parameterstudie rundet den Methodenentwicklungsprozess beider Ansätze ab. Die Ergebnisse zeigen, dass beide entwickelten Varianten zu einer deutlich präziseren Schätzung in Kombination mit gesteigerter Rechenperformance während des Trainingsprozesses gelangen. Die Verzerrung der SVM-Resultate kann im direkten Vergleich zum KNN deutlich reduziert werden und bestätigt diese Herangehensweisen. Die verbesserten Ergebnisse der gefundenen xGBoost-Architektur zeigen, dass die Eignung dieser Modelle zur Prädiktion des Gesundheitszustandes der Batterie durchaus bestätigt werden können. Im Anschluss an die Modellentwicklung wird die Übertragbarkeit der entworfenen xGBoost-Methodik auf rein elektrische Fahrzeugsysteme untersucht. Die Ergebnisse offenbaren typische Schwächen und Stärken maschineller Lernmethoden. So zeigt sich, dass eine einfache Transformation auf ein neues, aber ähnliches Problem nicht mehr möglich ist, wenn sich Struktur, Wertebereich und Korrelation der Daten ändern und die Erkennung von Ursachen durch die fehlende Berücksichtigung chemischer Zusammenhänge erheblich erschwert ist. Auf der anderen Seite kann ein dynamisches System, das klassischerweise durch erheblichen Modellierungsaufwand in einen komplexen Satz nichtlinearer Differentialgleichungen überführt wird, durch eine vergleichsweise einfache informative Struktur beschrieben werden ohne im Vorfeld Vereinfachungen und Annahmen vornehmen zu müssen, die eine hohe Expertise des Anwenders erfordern.

## 6.1.2 Abschließende Bewertung der Forschungshypothesen

Die Gesamtheit der Ergebnisse ermöglicht es, die in Abschnitt 1.2 aufgestellten Forschungshypothesen erneut heranzuziehen und zu bewerten:

**Hypothese H-1.** Die systematische Analyse der historischen Betriebsdaten erlaubt die Identifizierung von Parametern, welche die Alterung der Batterie signifikant beeinflussen, um letztlich eine Alterungsmodellierung zu ermöglichen.

---

✓ | Durch die systematische Analyse spezifischer historischer Betriebsdaten von Batterien lassen sich Schlüsselparameter identifizieren, die die Batteriealterung signifikant beeinflussen. Dazu gehören neben der Gesamtladezeit für externes Laden und Rekuperation, den Entladezeiten im Fahrmodus auch die Aufenthaltszeiten bei hohen *SoC*-Ständen während des Fahrens, eine hohe Anzahl von Hauptschützöffnungen, hohe Ladungsdurchsätze im Fahrmodus beim Laden und Entladen sowie hohe Ladungs-/Energiedurchsätze während des externen Ladens und im Rekuperationsbetrieb. Diese Methode gilt sowohl für Standard-Betriebsbedingungen als auch für spezifische Nutzungsszenarien und ermöglicht die Entwicklung präziser und komplexer Alterungsmodelle.

**Hypothese H-2.** Der Gesundheitszustand der Batterie kann durch die Weiterentwicklung maschineller Lernmodelle auf Basis der historischen Betriebsdaten prädiziert werden.

✓ | Nach dem derzeitigen Stand der Forschung basieren Alterungsvorhersagen nahezu ausschließlich auf Strom- und Spannungsprofilen. Die entwickelte Methodik ermöglicht es, maschinelle Lernalgorithmen anhand von Betriebsdaten zu optimieren, um den Gesundheitszustand der Batterie mit höherer Vorhersagegenauigkeit und geringeren systematischen Fehlern zu prädizieren. Im Zuge der Modellentwicklung werden mehrere Verbesserungen umgesetzt:

- Feature Engineering: Entwicklung eines geeigneten Datenvorverarbeitungsprozesses sowie Auswahl relevanter Eingangsgrößen
- Zielfunktion: Implementierung einer Zielfunktion, die Abweichungen bei geringen absoluten  $SoH_C$ -Werten stärker bestraft
- Hyperparameter: Optimierung der modellinternen Parameter zur Identifikation eines optimalen Parametersatzes

## 6.2 Ausblick

Die Forschung auf dem Gebiet der datengetriebenen Alterungsmodellierung mittels fahrzeugspezifischen Betriebsdaten steht noch am Anfang. Die verfügbare Literatur weist für diesen Anwendungsfall nur wenige Quellen auf. Hier ist es ratsam, die neugewonnenen Informationen zur Struktur der

Modellvarianten und der Auswahl sowie Generierung aussagekräftiger Eingangsgrößen zu nutzen. Für niedrige  $SoH_C$ -Bereiche, verbunden mit fortgeschrittenen Alterung, resultieren die Schätzungen in höheren quantitativen Abweichungen. Um dieser Tatsache entgegenzuwirken, wird empfohlen, Maßnahmen zur Optimierung der Datenbasis und -qualität zu ergreifen sowie neue Ansätze in Form von *Transformers* in Betracht zu ziehen:

**Datengrundlage und -qualität** In zukünftigen Forschungsarbeiten kann eine Verbesserung der Ergebnisgenauigkeit durch die Integration einer angepassten Datenquelle erreicht werden. Spezifisch betrachtet wäre der Übergang von der Verwendung sporadischer Werkstattprotokolle zu einer Datenquelle, die zeitlich höher aufgelöste Daten liefert, ein entscheidender Schritt. Diese Modifikation würde die Ergebnisqualität, speziell die der Long-Short-Term-Memory-Netze, merklich steigern. Die Verwendung höher abgetasteter und aufgelöster Betriebsdaten kann sich als besonders vorteilhaft herausstellen. Um dieses Potenzial auszuschöpfen, kann die im Rahmen dieser Arbeit implementierte Netzarchitektur als Grundlage dienen. Dabei ist zu beachten, dass die Qualität der Daten weiterhin von entscheidender Bedeutung ist, wobei der Schwerpunkt auf der Qualität gegenüber der Quantität liegen sollte, um die Validität und Zuverlässigkeit zukünftiger Forschungsergebnisse zu gewährleisten.

**Transformer-Methode** In jüngerer Vergangenheit haben signifikante methodische Fortschritte im Bereich des Deep Learning, insbesondere durch die Einführung von Transformer-Methoden, neue Möglichkeiten eröffnet [156] [157]. Diese Fortschritte bieten leistungsfähige Werkzeuge zur Handhabung von komplexen Herausforderungen, die bei der Vorhersage der Batteriedegradation entstehen. Transformers, ursprünglich konzipiert

für die Anforderungen der natürlichen Sprachverarbeitung, nutzen *Self-Attention*-Mechanismen, um die Bedeutung verschiedener Punkte innerhalb einer Zeitreihe differenziert zu gewichten. Dies hat erhebliche Vorteile gegenüber traditionellen Zeitreihenmodellen, welche zwar Effektivität in der Bearbeitung sequenzieller Daten zeigen, aber oft an ihre Grenzen stoßen, wenn es sich um langfristige Abhängigkeiten und komplexe nicht-lineare Muster handelt. Die Anwendung derartiger Methoden auf die Alterungsprozesse von Hochvoltbatterien in automotiven Anwendungen sollte es ermöglichen, kritische Zeitabschnitte und Muster zu identifizieren, die signifikante Zustandsveränderungen der Batterie anzeigen. Dies verbessert nicht nur die Vorhersagegenauigkeit, sondern auch die Robustheit der Modelle gegenüber den vielfältigen Betriebsbedingungen. Da Transformer-Methoden eine höhere Anforderung an die Datenquantität sowie -qualität besitzen, rücken diese Methoden – aufgrund steigender Datenmengen und Verfügbarkeiten – für zukünftige Untersuchungen in den Fokus.

Die Herausforderung der Batteriealterung wird maßgeblich durch die Nichtlinearitäten der Degradationsprozesse bestimmt. Dies umfasst sowohl plötzliche Anomalien als auch unregelmäßige Degradationsmuster, die durch extreme Betriebsbedingungen oder Fertigungsinkonsistenzen hervorgerufen werden. Durch die Integration globaler *Self-Attention*-Mechanismen, wie sie in der Arbeit von [158] detailliert beschrieben werden, kann die Sensitivität für derartige Anomalien deutlich gesteigert werden. Die Forschung zeigt, wie diese Mechanismen effektiv eingesetzt werden können, um Anomalien in multivariaten Zeitreihendaten zu identifizieren. Ein Ansatz, der direkt auf die Überwachung der Batteriegesundheit anwendbar sein könnte. Durch die stärkere Gewichtung von Datenpunkten, die von der erwarteten Norm abweichen, bieten Modelle, die diese Techniken nutzen, die Möglichkeit, frühzeitig auf potenzielle Probleme hinzuweisen. Durch

die Anwendung globaler Self-Attention-Mechanismen kann effektiver vorgegangen werden, um die Komplexität und Dynamik der Batteriealterung zu adressieren und letztlich die Betriebssicherheit und Wirtschaftlichkeit von Elektrofahrzeugen zu steigern.

# A Anhang

## Historische Matrizen

### HDM 2

Temperaturintervalle	kumulierte Zeit in h	Stromintervalle							
		A	B	C	D	E	F	G	H
1									
2									
3									
4									
5									
6									
7									
8									

Abbildung A.1: Schematischer Aufbau HDM 2 - Kumulierte Rekuperationszeit.

### HDM 3

Temperaturintervalle	kumulierte Zeit in h	Stromintervalle							
		A	B	C	D	E	F	G	H
1									
2									
3									
4									
5									
6									
7									
8									

Abbildung A.2: Schematischer Aufbau HDM 3 - Kumulierte Entladezeit während des Fahrens.

## HDM 4

Temperaturintervalle	Stromintervalle							
	A	B	C	D	E	F	G	H
1								
2								
3								
4								
5								
6								
7								
8								

**Abbildung A.3:** Schematischer Aufbau HDM 4 - Kumulierte Ladezeit bei externem Laden.

## HDM 5

kumulierte Zeit in h	SOC-Intervalle					
	A	B	C	D	E	F

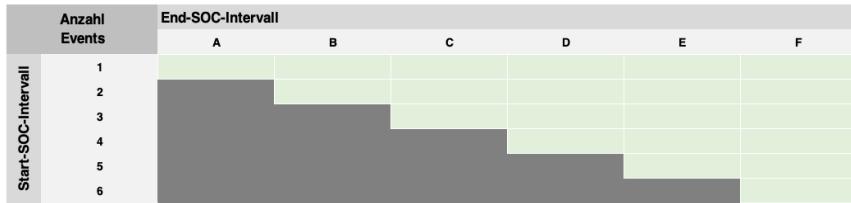
**Abbildung A.4:** Schematischer Aufbau HDM 5 - Kumulierte Aufenthaltszeit in verschiedenen SOC-Intervallen während des Fahrens.

## HDM 6

Anzahl Events	Lade- bzw. Entladehub-Intervall									
	A	B	C	D	E	F	G	H	I	J
	Anzahl Entladezyklen									

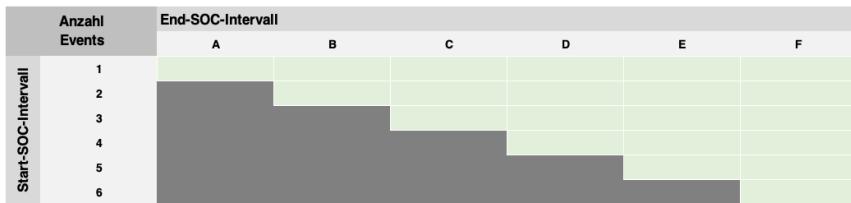
**Abbildung A.5:** Schematischer Aufbau HDM 6 - Eventzähler für Lade- und Entladeimpulse, unterschieden nach Lade- bzw. Entladehüben.

## HDM 7



**Abbildung A.6:** Schematischer Aufbau HDM 7 - Eventzähler Ladevorgänge bei externem AC-Laden unterschieden nach SOC Ladebeginn und SOC Ladeende.

## HDM 8



**Abbildung A.7:** Schematischer Aufbau HDM 8 - Eventzähler Ladevorgänge bei externem DC-Laden unterschieden nach SOC Ladebeginn und SOC Ladeende.

## HDM 9

Balancingzeit												
Zelle 1	Zelle 2	Zelle 3	Zelle 4	Zelle 5	Zelle 6	Zelle 7	Zelle 8	Zelle 9	Zelle 10	Zelle 11	Zelle 12	Zelle 13
Zelle 14	Zelle 15	Zelle 16	Zelle 17	Zelle 18	Zelle 19	Zelle 20	Zelle 21	Zelle 22	Zelle 23	Zelle 24	Zelle 25	Zelle 26
Zelle 27	Zelle 28	Zelle 29	Zelle 30	Zelle 31	Zelle 32	Zelle 33	Zelle 34	Zelle 35	Zelle 36	Zelle 37	Zelle 38	Zelle 39
Zelle 40	Zelle 41	Zelle 42	Zelle 43	Zelle 44	Zelle 45	Zelle 46	Zelle 47	Zelle 48	Zelle 49	Zelle 50	Zelle 51	Zelle 52
Zelle 53	Zelle 54	Zelle 55	Zelle 56	Zelle 57	Zelle 58	Zelle 59	Zelle 60	Zelle 61	Zelle 62	Zelle 63	Zelle 64	Zelle 65
Zelle 66	Zelle 67	Zelle 68	Zelle 69	Zelle 70	Zelle 71	Zelle 72	Zelle 73	Zelle 74	Zelle 75	Zelle 76	Zelle 77	Zelle 78
Zelle 79	Zelle 80	Zelle 81	Zelle 82	Zelle 83	Zelle 84	Zelle 85	Zelle 86	Zelle 87	Zelle 88	Zelle 89	Zelle 90	Zelle 91
Zelle 92	Zelle 93	Zelle 94	Zelle 95	Zelle 96	Zelle 97	Zelle 98	Zelle 99	Zelle 100	Zelle 101	Zelle 102	Zelle 103	Zelle 104
Zelle 105	Zelle 106	Zelle 107	Zelle 108									

MIN:  0,0 MAX:  0,0

**Abbildung A.8:** Schematischer Aufbau HDM 9 - Kumulierte, zell(gruppen)-individuelle Balancingzeit.

## HDM 10

kumulierte Zeit	Maximale SOC-Zelldifferenz	A	B	C	D	E	F	G	H	I	J
kumulierte Zeit											

**Abbildung A.9:** Schematischer Aufbau HDM 10 - Kumulierte Zeit der anliegenden maximalen SOC-Differenz zwischen Zellen oder Zellgruppen.

## HDM 18

kumulierte Werte	Drive Mode	Drive Mode	Charge	Charge
	Laden	Entladen	AC Laden	DC Laden
<b>Ladung (in Ah)</b>				
<b>Energie (in kWh)</b>				

**Abbildung A.10:** Schematischer Aufbau HDM 18 - Kumulierte Ladungs- und Energiemenge, unterschieden nach Betriebsmodus.

# Abbildungsverzeichnis

2.1	Ragone-Diagramm ausgewählter Batterietechnologien. . . . .	6
2.2	Schematischer Aufbau einer Lithium-Ionen-Zelle. . . . .	8
2.3	Übersicht der wesentlichen Alterungsmechanismen . . . . .	15
2.4	Alterungskennlinie einer Lithium-Ionen-Batterie. . . . .	17
2.5	Einteilung maschinelner Lernverfahren. . . . .	19
2.6	Schematische Darstellung der Generalisierungsfähigkeit. . . . .	23
2.7	Illustrierte Darstellung des Aufbaus eines künstlichen Neurons. .	26
2.8	Funktionsverläufe für nichtlineare Aktivierungsfunktionen. . .	29
2.9	Architektur eines vorwärtsgerichteten Neuronalen Netzes. . .	31
2.10	Darstellung der Prädiktion eines Support Vector Klassifikators. .	34
2.11	Darstellung der Prädiktion eines Support Vector Regressors. .	36
2.12	Schematischer Aufbau eines xGBoost-Modells. . . . .	40
4.1	Schematische Darstellung des HDM 1. . . . .	62
4.2	Rohdatenverteilung der Fahrzeuganalyseprotokolle. . . . .	63
4.3	Rohdatenverteilung des <i>SoHc</i> . . . . .	64
4.4	Relative Wichtigkeit der Eingangsgrößen. . . . .	73
4.5	Übersicht des Datenaufbereitungsprozesses. . . . .	79
4.6	Ablauf der Zukunftsextrapolation. . . . .	81
4.7	Einsatz des Modells als virtueller Sensor. . . . .	83
4.8	Grafische Darstellung verschiedener Zielfunktionen. . . . .	87
4.9	Abstrakte Darstellung der finalen rekurrenten Netzarchitektur. .	90
4.10	Parameteroptimierung des Support Vector Regressors. . . . .	93
4.11	Parameteroptimierung des xGBoost-Algorithmus. . . . .	96
5.1	Prädiktionsergebnisse des Rekurrenten Neuronalen Netzes. .	105
5.2	Prädiktionsergebnisse des Support Vector Regressors. . . . .	111
5.3	Prädiktionsergebnisse des xGBoost-Algorithmus. . . . .	115

5.4	Prozess der Zukunftsprädiktion . . . . .	118
5.5	Übersicht der hergeleiteten Modellvarianten. . . . .	122
5.6	Ergebnisse der Einzelfahrzeuganalyse. . . . .	124
5.7	Vergleich der Zukunfts- und Vergangenheitsprädiktion. . . . .	126
A.1	Schematischer Aufbau HDM 2. . . . .	137
A.2	Schematischer Aufbau HDM 3. . . . .	137
A.3	Schematischer Aufbau HDM 4. . . . .	138
A.4	Schematischer Aufbau HDM 5. . . . .	138
A.5	Schematischer Aufbau HDM 6. . . . .	138
A.6	Schematischer Aufbau HDM 7. . . . .	139
A.7	Schematischer Aufbau HDM 8. . . . .	139
A.8	Schematischer Aufbau HDM 9. . . . .	140
A.9	Schematischer Aufbau HDM 10. . . . .	140
A.10	Schematischer Aufbau HDM 18. . . . .	140

# Tabellenverzeichnis

2.1	Einflussgrößen auf die kalendarische und zyklische Alterung. . .	14
4.1	Übersicht der Eingangsgrößen als Resultat der Korrelationsanalyse.	75
4.2	Hyperparameterstudie des Support Vector Regressors. . . . .	92
4.3	Hyperparameterstudie xGBoost-Regressor . . . . .	95
5.1	Bewertung der Vorhersagequalität anhand der Fehlermetriken. .	120



# Literaturverzeichnis

- [1] F. Sun, R. Xiong und H. He, „A systematic state-of-charge estimation framework for multi-cell battery pack in electric vehicles using bias correction technique,” *Applied Energy*, Vol. 162, S. 1399–1409, 2016.
- [2] L. Fortuna, S. Graziani, A. Rizzo und M. G. Xibilia, *Soft sensors for monitoring and control of industrial processes*. London, UK: Springer-Verlag, 2007.
- [3] B. Liaw und G. Pistoia, *Behaviour of lithium-ion batteries in electric vehicles*. Cham, Schweiz: Springer-Verlag, 2018.
- [4] A. Masias, J. Marcicki und W. A. Paxton, „Opportunities and challenges of lithium ion batteries in automotive applications,” *ACS Energy Letters*, Vol. 6(2), S. 621–630, 2021.
- [5] K. W. Beard, *Linden’s handbook of batteries*. New York City, NY, USA: McGraw-Hill, 2019.
- [6] J.-M. Tarascon und M. Armand, „Issues and challenges facing rechargeable lithium batteries,” *Nature*, Vol. 414, S. 359–367, 2001.
- [7] S. S. Rangarajan, S. P. Sunddararaj, A. Sudhakar, C. K. Shiva, U. Subramaniam, E. R. Collins und T. Senju, „Lithium-ion batteries – The crux of electric vehicles with opportunities and challenges,” *Clean Technologies*, Vol. 4(4), S. 908–930, 2022.

- [8] P. Van den Bossche, F. Vergels, J. Van Mierlo, J. Matheys und W. Van Autenboer, „SUBAT: An assessment of sustainable battery technology,” *Journal of Power Sources*, Vol. 162(2), S. 913–919, 2006.
- [9] M. Doppelbauer, *Grundlagen der Elektromobilität: Technik, Praxis, Energie und Umwelt*. Wiesbaden, Deutschland: Springer-Verlag, 2020.
- [10] Z. Zhang und S. S. Zhang, *Rechargeable batteries : Materials, technologies and new trends*. Cham, Schweiz: Springer-Verlag, 2015.
- [11] M. Lang, „Degradationsmechanismen großformatiger Lithium-Ionen-Batterien für Elektrofahrzeuge,” Dissertation, Karlsruher Institut für Technologie, Karlsruhe, Deutschland, 2018.
- [12] G.-A. Nazri und G. Pistoia, *Lithium batteries: Science and technology*. New York City, NY, USA: Springer-Verlag, 2003.
- [13] R. Wittman, Y. Preger, A. Fresquez und B. Chalamala, „Systematic cycle and calendar aging of commercial 18650 LFP lithium-ion cells,” Präsentation auf dem 241. Electrochemical Society Meeting, Vancouver, Kanada (virtuell), 2022.
- [14] P. Kurzweil, *Angewandte Elektrochemie: Grundlagen, Messtechnik, Elektroanalytik, Energiewandlung, technische Verfahren*. Wiesbaden, Deutschland: Springer-Verlag, 2020.
- [15] R. Korthauer, *Handbuch Lithium-Ionen-Batterien*. Berlin, Heidelberg, Deutschland: Springer-Verlag, 2013.
- [16] L. Canals Casals, M. Rodríguez, C. Corchero und R. E. Carrillo, „Evaluation of the end-of-life of electric vehicle batteries according to the state-of-health,” *World Electric Vehicle Journal*, Vol. 10(4), Art. 63, 2019.

- [17] C. Lin, A. Tang und W. Wang, „A review of SOH estimation methods in lithium-ion batteries for electric vehicle applications,” *Energy Procedia*, Vol. 75, S. 1920–1925, 2015.
- [18] G. Sarre, P. Blanchard und M. Broussely, „Aging of lithium-ion batteries,” *Journal of Power Sources*, Vol. 127(1-2), S. 65–71, 2004.
- [19] P. Keil, „Aging of lithium-ion batteries in electric vehicles,” Dissertation, Technische Universität München, München, Deutschland, 2017.
- [20] F. Leng, C. M. Tan und M. Pecht, „Effect of temperature on the aging rate of Li ion battery operating above room temperature,” *Scientific reports*, Vol. 5, Art. 12967, 2015.
- [21] A. Barré, B. Deguilhem, S. Grolleau, M. Gérard, F. Suard und D. Riu, „A review on lithium-ion battery ageing mechanisms and estimations for automotive applications,” *Journal of Power Sources*, Vol. 241, S. 680–689, 2013.
- [22] I. Bloom, B. Cole, J. Sohn, S. A. Jones, E. G. Polzin, V. S. Battaglia, G. L. Henriksen, C. Motloch, R. Richardson, T. Unkelhaeuser et al., „An accelerated calendar and cycle life study of Li-ion cells,” *Journal of Power Sources*, Vol. 101(2), S. 238–247, 2001.
- [23] G. Luo, Y. Zhang und A. Tang, „Capacity degradation and aging mechanisms evolution of lithium-ion batteries under different operation conditions,” *Energies*, Vol. 16(10), Art. 4232, 2023.
- [24] A. Degla, M. Chikh, S. Boumecheta, M. B. Danoune und Y. Rehouma, „Aging effect simulation for lithium-ion battery under operating conditions,” in *Proceedings of the 14th International Renewable Energy Congress*, Art. 10389381, Sousse, Tunesien, 2023.

- [25] S. F. Schuster, T. Bach, E. Fleder, J. Müller, M. Brand, G. Sextl und A. Jossen, „Nonlinear aging characteristics of lithium-ion cells under different operational conditions,” *Journal of Energy Storage*, Vol. 1, S. 44–53, 2015.
- [26] X. Zhang, L. Zou, Y. Xu et al., „Advanced electrolytes for fast-charging high-voltage lithium-ion batteries in wide-temperature range,” *Advanced Energy Materials*, Vol. 10(22), Art. 2000368, 2020.
- [27] C. Schlasza, P. Ostertag, D. Chrenko, R. Kriesten und D. Bouquain, „Review on the aging mechanisms in li-ion batteries for electric vehicles based on the FMEA method,” in *Proceedings of the 3rd IEEE Transportation Electrification Conference and Expo*, S. 1–6, Dearborn, MI, USA, 2014.
- [28] C. Daniel und J. Besenhard, *Handbook of Battery Materials*. Weinheim, Deutschland: Wiley-VCH, 2012.
- [29] J. P. Fath, D. Dragicevic, L. Bittel, A. Nuhic, J. Sieg, S. Hahn, L. Alsheimer, B. Spier und T. Wetzel, „Quantification of aging mechanisms and inhomogeneity in cycled lithium-ion cells by differential voltage analysis,” *Journal of Energy Storage*, Vol. 25, Art. 100813, 2019.
- [30] V. Agubra und J. Fergus, „Lithium ion battery anode aging mechanisms,” *Materials*, Vol. 6(4), S. 1310–1325, 2013.
- [31] J. Li, J. Zhang, X. Zhang, C. Yang, N. Xu und B. Xia, „Study of the storage performance of a Li-ion cell at elevated temperature,” *Electrochimica Acta*, Vol. 55(3), S. 927–934, 2010.
- [32] D. Lisbona und T. Snee, „A review of hazards associated with primary lithium and lithium-ion batteries,” *Process Safety and Environmental Protection*, Vol. 89(6), S. 434–442, 2011.

[33] M. Ecker, N. Nieto, S. Käbitz, J. Schmalstieg, H. Blanke, A. Warnecke und D. U. Sauer, „Calendar and cycle life study of Li(NiMnCo) O<sub>2</sub>-based 18650 lithium-ion batteries,” *Journal of Power Sources*, Vol. 248, S. 839–851, 2014.

[34] S. Atalay, M. Sheikh, A. Mariani, Y. Merla, E. Bower und W. D. Widanage, „Theory of battery ageing in a lithium-ion battery: Capacity fade, nonlinear ageing and lifetime prediction,” *Journal of Power Sources*, Vol. 478, Art. 229026, 2020.

[35] T. M. Mitchell, *Machine Learning*. New York City, NY, USA: McGraw-Hill, 1997.

[36] I. Goodfellow, Y. Bengio und A. Courville, *Deep Learning*. Cambridge, MA, USA: MIT Press, 2016.

[37] S. Raschka und V. Mirjalili, *Python machine learning: Machine learning and deep learning with Python, scikit-learn, and TensorFlow*. Birmingham, UK: Packt Publishing, 2017.

[38] A. Nandy und M. Biswas, *Reinforcement Learning: With Open AI, TensorFlow and Keras Using Python*. Berkeley, CA, USA: Apress, 2018.

[39] D. D. Nauck, F. Klawonn und R. Kruse, *Neuronale Netze und Fuzzy-Systeme - Grundlagen des Konnektionismus, neuronaler Fuzzy-Systeme und der Kopplung mit wissensbasierten Methoden*. Wiesbaden, Deutschland: Springer-Verlag, 1996.

[40] D. E. Rumelhart, J. L. McClelland und P. R. Group, *Parallel distributed processing, volume 1: Explorations in the microstructure of cognition: Foundations*. Cambridge, MA, USA: MIT Press, 1986.

- [41] I. Sutskever, J. Martens, G. Dahl und G. Hinton, „On the importance of initialization and momentum in deep learning,” in *Proceedings of the 30th International Conference on Machine Learning*, S. 1139–1147, Atlanta, GA, USA, 2013.
- [42] A. Krogh und J. Hertz, „A simple weight decay can improve generalization,” in *Proceedings of the 4th International Conference on Neural Information Processing Systems*, S. 950–957, Denver, CO, USA, 1991.
- [43] R. Kruse, C. Borgelt, S. Braune, M. Steinbrecher, F. Klawonn und C. Moewes, *Computational Intelligence – Eine methodische Einführung in Künstliche Neuronale Netze, Evolutionäre Algorithmen, Fuzzy-Systeme und Bayes-Netze*. Wiesbaden, Deutschland: Springer-Verlag, 2015.
- [44] A. Scherer, *Neuronale Netze: Grundlagen und Anwendungen*. Wiesbaden, Deutschland: Vieweg+Teubner Verlag, 1997.
- [45] G. D. Rey und K. F. Wender, *Neuronale Netze - Eine Einführung in die Grundlagen, Anwendungen und Datenauswertung*. Bern, Schweiz: Hogrefe-Verlag, 2018.
- [46] B. Xu, R. Huang und M. Li, „Revise saturated activation functions,” in *Proceedings of the 4th International Conference on Learning Representations*, San Juan, Puerto Rico, 2016.
- [47] R. Rojas, *Neural Networks - A Systematic Introduction*. Berlin, Heidelberg, Deutschland: Springer-Verlag, 1996.
- [48] R. Hahnloser, R. Sarpeshkar, M. Mahowald, R. Douglas und H. Seung, „Digital selection and analogue amplification coexist in a cortex-inspired silicon circuit,” *Nature*, Vol. 405, S. 947–951, 2000.

[49] X. Glorot, A. Bordes und Y. Bengio, „Deep sparse rectifier neural networks,” in *Proceedings of the 14th international conference on artificial intelligence and statistics*, S. 315–323, Fort Lauderdale, FL, USA, 2011.

[50] Y. LeCun, L. Bottou, Y. Bengio und P. Haffner, „Gradient-based learning applied to document recognition,” *Proceedings of the IEEE*, Vol. 86(11), S. 2278–2324, 1998.

[51] G. Montavon, G. Orr und K.-R. Müller, *Neural networks: tricks of the trade*. Berlin, Heidelberg, Deutschland: Springer-Verlag, 2012.

[52] J. Kivinen und M. K. Warmuth, „Relative loss bounds for multidimensional regression problems,” in *Proceedings of the 10th International Conference on Neural Information Processing Systems*, S. 288–293, Denver, CO, USA, 1997.

[53] Z. Boger und H. Guterman, „Knowledge extraction from artificial neural network models,” in *Proceedings of the IEEE International Conference on Systems, Man, and Cybernetics*, S. 3030 – 3035, Orlando, FL, USA, 1997.

[54] S. Dörn, *Programmieren für Ingenieure und Naturwissenschaftler*. Berlin, Heidelberg, Deutschland: Springer-Verlag, 2018.

[55] E. Sadrifaridpour, T. Razzaghi und I. Safro, „Engineering fast multilevel support vector machines,” *Machine Learning*, Vol. 108, S. 1879–1917, 2019.

[56] L. Wang, *Support Vector Machines: Theory and Applications*. Berlin, Heidelberg, Deutschland: Springer-Verlag, 2005.

- [57] N. Cristianini und J. Shawe-Taylor, *An introduction to support vector machines and other kernel-based learning methods.* Cambridge, UK: Cambridge University Press, 2006.
- [58] B. Schölkopf und A. J. Smola, *Learning with kernels: support vector machines, regularization, optimization, and beyond.* Cambridge, MA, USA: MIT Press, 2002.
- [59] C. Cortes und V. Vapnik, „Support-vector networks,” *Machine Learning*, Vol. 20, S. 273–297, 1995.
- [60] M. Schmitt, „Support vector machines via multilevel label propagation,” Bachelorarbeit (betreut durch C. Schulz), Karlsruher Institut für Technologie, Institut für Theoretische Informatik (KIT-ITI), Karlsruhe, September 2018.
- [61] V. N. Vapnik, *The nature of statistical learning theory.* New York City, NY, USA: Springer-Verlag, 2010.
- [62] T. Chen und C. Guestrin, „XGBoost: A scalable tree boosting system,” in *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, S. 785–794, 2016.
- [63] M. R. Berthold, C. Borgelt, F. Höppner und F. Klawonn, *Guide to intelligent data analysis: how to intelligently make sense of real data.* London, UK: Springer-Verlag, 2010.
- [64] L. Breiman, J. Friedman, R. A. Olshen und C. J. Stone, *Classification and regression trees.* Boca Raton, FL, USA: Chapman & Hall/CRC Press, 1998.

- [65] A. Amro, M. Al-Akhras, K. E. Hindi, M. Habib und B. A. Shawar, „Instance reduction for avoiding overfitting in decision trees,” *Journal of Intelligent Systems*, Vol. 30(1), S. 438–459, 2021.
- [66] T. G. Dietterich, „Ensemble methods in machine learning,” in *Proceedings of the 1st International workshop on multiple classifier systems*, S. 1–15, Cagliari, Italien, 2000.
- [67] Z.-H. Zhou, *Ensemble Methods: Foundations and Algorithms*. Boca Raton, FL, USA: Chapman & Hall/CRC Press, 2012.
- [68] R. Mikut, *Data Mining in der Medizin und Medizintechnik*. Karlsruhe, Deutschland: KIT Scientific Publishing, 2008.
- [69] R. Schapire und Y. Freund, *Boosting: Foundations and Algorithms*. Cambridge, MA, USA: MIT Press, 2012.
- [70] J. H. Friedman, „Greedy function approximation: A gradient boosting machine,” *Annals of Statistics*, Vol. 29(5), S. 1189–1232, 2001.
- [71] A. Natekin und A. Knoll, „Gradient boosting machines, a tutorial,” *Frontiers in Neurorobotics*, Vol. 7, Art. 21, 2013.
- [72] B. Pan, „Application of XGBoost algorithm in hourly PM2. 5 concentration prediction,” in *Proceedings of the 3rd International Conference on Advances in Energy Resources and Environment Engineering*, Art. 012127, Harbin, China, 2018.
- [73] Y. Liu, H. Wang, Y. Fei, Y. Liu, L. Shen, Z. Zhuang und X. Zhang, „Research on the prediction of green plum acidity based on improved XG-Boost,” *Sensors*, Vol. 21(3), Art. 930, 2021.

- [74] H. Mo, H. Sun, J. Liu und S. Wei, „Developing window behavior models for residential buildings using XGBoost algorithm,” *Energy and Buildings*, Vol. 205, Art. 109564, 2019.
- [75] M. Zhang, D. Yang, J. Du, H. Sun, L. Li, L. Wang und K. Wang, „A review of SOH prediction of Li-ion batteries based on data-driven algorithms,” *Energies*, Vol. 16(7), Art. 3167, 2023.
- [76] G. Zhang, B. E. Patuwo und M. Y. Hu, „Forecasting with artificial neural networks: The state of the art,” *International Journal of Forecasting*, Vol. 14(1), S. 35–62, 1998.
- [77] Z. Shou, Z. Wang, K. Han, Y. Liu, P. Tiwari und X. Di, „Long-term prediction of lane change maneuver through a multilayer perceptron,” in *Proceedings of the IEEE Intelligent Vehicles Symposium*, S. 246–252, Las Vegas, NV, USA, 2020.
- [78] D. Vengertsev, „Deep learning architecture for univariate time series forecasting,” Tech. Rep., CS229, Stanford University, Stanford, CA, USA, 2014.
- [79] Y. LeCun, Y. Bengio und G. Hinton, „Deep learning,” *Nature*, Vol. 521, S. 436–444, 2015.
- [80] Z. C. Lipton, J. Berkowitz und C. Elkan, „A critical review of recurrent neural networks for sequence learning,” *arXiv preprint*, Art. 1506.00019, 2015.
- [81] S. Hochreiter, „Untersuchungen zu dynamischen neuronalen Netzen,” Dissertation, Technische Universität München, München, Deutschland, 1991.

[82] J. F. Kolen und S. C. Kremer, *A field guide to dynamical recurrent networks*. New York City, NY, USA: Wiley-IEEE Press, 2001.

[83] A. Graves, „Generating sequences with recurrent neural networks,” *arXiv preprint*, Art. 1308.0850, 2013.

[84] J. Oladimeji und O. Ogunniyi, „A review of artificial intelligence-based prognostic and health management systems for lithium-in batteries in electric vehicles,” *International Journal of Science and Research*, Vol. 12(12), S. 345–355, 2023.

[85] X. Shu, J. Shen, G. Li, Y. Zhang, Z. Chen und Y. Liu, „A flexible state-of-health prediction scheme for lithium-ion battery packs with long short-term memory network and transfer learning,” *IEEE Transactions on Transportation Electrification*, Vol. 7(4), S. 2238–2248, 2021.

[86] J. Qu, F. Liu, Y. Ma und J. Fan, „A neural-network-based method for RUL prediction and SOH monitoring of lithium-ion battery,” *IEEE Access*, Vol. 7, S. 87 178–87 191, 2019.

[87] S. Pepe und F. Ciucci, „Long-range battery state-of-health and end-of-life prediction with neural networks and feature engineering,” *Applied Energy*, Vol. 350, Art. 121761, 2023.

[88] Y. Ma, C. Shan, J. Gao und H. Chen, „A novel method for state of health estimation of lithium-ion batteries based on improved LSTM and health indicators extraction,” *Energy*, Vol. 251, Art. 123973, 2022.

[89] H. Huang, C. Bian, M. Wu, D. An und S. Yang, „A novel integrated SOC-SOH estimation framework for whole-life-cycle lithium-ion batteries,” *Energy*, Vol. 288, Art. 129801, 2024.

- [90] L. Zhang, T. Ji, S. Yu und G. Liu, „Accurate prediction approach of SOH for lithium-ion batteries based on LSTM method,” *Batteries*, Vol. 9(3), Art. 177, 2023.
- [91] S. Bouktif, A. Fiaz, A. Ouni und M. A. Serhani, „Multi-sequence LSTM-RNN deep learning and metaheuristics for electric load forecasting,” *Energies*, Vol. 13(2), Art. 391, 2020.
- [92] L. Ma, C. Hu und F. Cheng, „State of charge and state of energy estimation for lithium-ion batteries based on a long short-term memory neural network,” *Journal of Energy Storage*, Vol. 37, Art. 102440, 2021.
- [93] M. Catelani, L. Ciani, R. Fantacci, G. Patrizi und B. Picano, „Remaining useful life estimation for prognostics of lithium-ion batteries based on recurrent neural network,” *IEEE Transactions on Instrumentation and Measurement*, Vol. 70, Art. 3524611, 2021.
- [94] J. Liu, A. Saxena, K. Goebel, B. Saha und W. Wang, „An adaptive recurrent neural network for remaining useful life prediction of lithium-ion batteries,” in *Proceedings of the Annual Conference of the Prognostics and Health Management Society*, S. 1–9, Portland, OR, USA, 2010.
- [95] B. Chinomona, C. Chung, L.-K. Chang, W.-C. Su und M.-C. Tsai, „Long short-term memory approach to estimate battery remaining useful life using partial data,” *IEEE Access*, Vol. 8, S. 165 419–165 431, 2020.
- [96] H. Chun, J. Kim, J. Yu und S. Han, „Real-time parameter estimation of an electrochemical lithium-ion battery model using a long short-term memory network,” *IEEE Access*, Vol. 8, S. 81 789–81 799, 2020.
- [97] K. Park, Y. Choi, W. J. Choi, H.-Y. Ryu und H. Kim, „LSTM-based battery remaining useful life prediction with multi-channel charging profiles,” *IEEE Access*, Vol. 8, S. 20 786–20 798, 2020.

[98] E. Schimanko, „Alterungsprädiktion von Lithium-Ionen-Batterien auf Basis von Bordnetzströmen,” Masterarbeit (betreut durch D. Herz), Karlsruher Institut für Technologie, Institut für Fahrzeugtechnik (KIT-FAST), Karlsruhe, Juni 2022.

[99] G. Mahalakshmi, S. Sridevi und S. Rajaram, „A survey on forecasting of time series data,” in *Proceedings of the 1st IEEE International Conference on Computing Technologies and Intelligent Data Engineering*, Art. 7725358, Kovilpatti, Indien, 2016.

[100] D. Anguita, A. Ghio, N. Greco, L. Oneto und S. Ridella, „Model selection for support vector machines: Advantages and disadvantages of the machine learning theory,” in *Proceedings of the IEEE International Joint Conference on Neural Networks*, Art. 5596450, Barcelona, Spanien, 2010.

[101] K.-j. Kim, „Financial time series forecasting using support vector machines,” *Neurocomputing*, Vol. 55(1-2), S. 307–319, 2003.

[102] T. Joachims, T. Hofmann, Y. Yue und C.-N. Yu, „Predicting structured objects with support vector machines,” *Communications of the Association for Computing Machinery*, Vol. 52(11), S. 97–104, 2009.

[103] Y. Jeong, K. Yi und S. Park, „SVM based intention inference and motion planning at uncontrolled intersection,” *IFAC-PapersOnLine*, Vol. 52(8), S. 356–361, 2019.

[104] A. Ding, X. Zhao und L. Jiao, „Traffic flow time series prediction based on statistics learning theory,” in *Proceedings of the 5th IEEE International Conference on Intelligent Transportation Systems*, S. 727–730, Singapur, 2002.

[105] L. Cao, „Support vector machines experts for time series forecasting,” *Neurocomputing*, Vol. 51, S. 321–339, 2003.

- [106] Z. Li, N. Wang, Y. Li, X. Sun, M. Huo und H. Zhang, „Collective efficacy of support vector regression with smoothness priority in marine sensor data prediction,” *IEEE Access*, Vol. 7, S. 10 308–10 317, 2019.
- [107] M. A. Mohandes, T. O. Halawani, S. Rehman und A. A. Hussain, „Support vector machines for wind speed prediction,” *Renewable Energy*, Vol. 29(6), S. 939–947, 2004.
- [108] Z. Chen, X. Xia, M. Sun, J. Shen und R. Xiao, „State of health estimation of lithium-ion batteries based on fixed size LS-SVM,” in *Proceedings of the IEEE Vehicle Power and Propulsion Conference*, Art. 8605041, Chicago, IL, USA, 2018.
- [109] N. Harting, R. Schenkendorf, N. Wolff und U. Krewer, „State-of-health identification of lithium-ion batteries based on nonlinear frequency response analysis: First steps with machine learning,” *Applied Sciences*, Vol. 8(5), Art. 821, 2018.
- [110] M. U. Ali, A. Zafar, S. H. Nengroo, S. Hussain, G.-S. Park und H.-J. Kim, „Online remaining useful life prediction for lithium-ion batteries using partial discharge data features,” *Energies*, Vol. 12(22), Art. 4366, 2019.
- [111] J. Lin, Z. Liu und Y. Yang, „A remaining useful life prediction approach for Li-ion batteries based on variational mode decomposition and SVM,” in *Proceedings of the 6th International Conference on Energy, Environment and Materials Science*, Hulun Buir, China, Art. 012154, 2020.
- [112] R. Li, S. Xu, S. Li, Y. Zhou, K. Zhou, X. Liu und J. Yao, „State of charge prediction algorithm of lithium-ion battery based on PSO-SVR cross validation,” *IEEE Access*, Vol. 8, S. 10 234–10 242, 2020.
- [113] X. Feng, C. Weng, X. He, X. Han, L. Lu, D. Ren und M. Ouyang, „Online state-of-health estimation for Li-ion battery using partial charging

segment based on support vector machine,” *IEEE Transactions on Vehicular Technology*, Vol. 68(9), S. 8583–8592, 2019.

[114] Z. Zhou, B. Duan, Y. Kang, Y. Shang, N. Cui, L. Chang und C. Zhang, „An efficient screening method for retired lithium-ion batteries based on support vector machine,” *Journal of Cleaner Production*, Vol. 267, Art. 121882, 2020.

[115] Y. Zhang, Z. Zhou, Y. Kang, C. Zhang und B. Duan, „A quick screening approach based on fuzzy C-means algorithm for the second usage of retired lithium-ion batteries,” *IEEE Transactions on Transportation Electrification*, Vol. 7(2), S. 474–484, 2021.

[116] T. Wu, Y. Huang, Y. Xu, J. Jiang, S. Liu und Z. Li, „SOH prediction for lithium-ion battery based on improved support vector regression,” *International Journal of Green Energy*, Vol. 20(3), S. 227–236, 2023.

[117] T. Sun, R. Wu, Y. Cui und Y. Zheng, „Sequent extended Kalman filter capacity estimation method for lithium-ion batteries based on discrete battery aging model and support vector machine,” *Journal of Energy Storage*, Vol. 39, Art. 102594, 2021.

[118] V. Chandran, C. K. Patil, A. Karthick, D. Ganeshaperumal, R. Rahim und A. Ghosh, „State of charge estimation of lithium-ion battery for electric vehicles using machine learning algorithms,” *World Electric Vehicle Journal*, Vol. 12(1), Art. 38, 2021.

[119] K. Das, R. Kumar und A. Krishna, „Analyzing electric vehicle battery health performance using supervised machine learning,” *Renewable and Sustainable Energy Reviews*, Vol. 189, Art. 113967, 2024.

- [120] A. Siskos, „Alterungsmodellierung von hochvoltbatterien anhand von diagnosedaten,” Bachelorarbeit (betreut durch D. Herz), Karlsruher Institut für Technologie, Institut für Fahrzeugtechnik (KIT-FAST), Karlsruhe, September 2020.
- [121] Z. Chen und W. Fan, „A freeway travel time prediction method based on an XGBoost model,” *Sustainability*, Vol. 13(15), Art. 8577, 2021.
- [122] Y. Zhang, X. Shi, S. Zhang und A. Abraham, „A XGBoost-based lane change prediction on time series data using feature engineering for autopilot vehicles,” *IEEE Transactions on Intelligent Transportation Systems*, Vol. 23(10), S. 19 187–19 200, 2022.
- [123] N. Zhai, P. Yao und X. Zhou, „Multivariate time series forecast in industrial process based on XGBoost and GRU,” in *Proceedings of the 9th IEEE Joint International Information Technology and Artificial Intelligence Conference*, S. 1397–1400, Chongqing, China, 2020.
- [124] L. Zhong, L. Hu und H. Zhou, „Deep learning based multi-temporal crop classification,” *Remote Sensing of Environment*, Vol. 221, S. 430–443, 2019.
- [125] J. Luo, Z. Zhang, Y. Fu und F. Rao, „Time series prediction of COVID-19 transmission in america using LSTM and XGBoost algorithms,” *Results in Physics*, Vol. 27, Art. 104462, 2021.
- [126] Z. Shi, Y. Hu, G. Mo und J. Wu, „Attention-based CNN-LSTM and XGBoost hybrid model for stock prediction,” *arXiv preprint*, Art. 2204.02623, 2022.
- [127] L. Yang, Y. Li und C. Di, „Application of XGBoost in identification of power quality disturbance source of steady-state disturbance events,”

in *Proceedings of the 9th IEEE International Conference on Electronics Information and Emergency Communication*, S. 716–721, Peking, China, 2019.

[128] Z. Shilong und X. Dairu, „Machine learning model for sales forecasting by using XGBoost,” in *Proceedings of the IEEE International Conference on Consumer Electronics and Computer Engineering*, S. 480–483, Guangzhou, China, 2021.

[129] S. Xu, F.-L. Zha, B.-W. Huang, B. Yu, H.-B. Huang, T. Zhou, W.-Q. Mao, J.-J. Wu, J.-Q. Wei, S.-K. Gong et al., „Research on the state of health estimation of lithium-ion batteries for energy storage based on XGB-AKF method,” *Frontiers in Energy Research*, Vol. 10, Art. 999676, 2023.

[130] F. Jiang, Y. He, D. Gao, Y. Zhou, W. Liu, L. Yan und J. Peng, „An accurate and interpretable lifetime prediction method for batteries using extreme gradient boosting tree and TreeExplainer,” in *Proceedings of the 23rd IEEE International Conference on High Performance Computing & Communications*, S. 1042–1048, Haikou, China, 2021.

[131] B. Celen, M. B. Ozcelik, F. M. Turgut, C. Aras, T. Sivaraman, Y. Kotak, C. Geisbauer und H.-G. Schweiger, „Calendar ageing modelling using machine learning: an experimental investigation on lithium ion battery chemistries,” *Open Research Europe*, Vol. 2, Art. 96, 2023.

[132] S. Jafari, Z. Shahbazi und Y.-C. Byun, „Lithium-ion battery health prediction on hybrid vehicles using machine learning approach,” *Energies*, Vol. 15(13), Art. 4753, 2022.

[133] Z. Ma, M. Chen, J. Cao, Y. Yao und Y. Liu, „Suitable feature selection for prediction of lithium-ion batteries remaining useful life,” in *Proceedings of the 7th International Conference on Information, Cybernetics and Computational Social Systems*, S. 728–732, Guangzhou, China, 2020.

- [134] S. Song, C. Fei und H. Xia, „Lithium-ion battery SOH estimation based on XGBoost algorithm with accuracy correction,” *Energies*, Vol. 13(4), Art. 812, 2020.
- [135] K. O’Boyle, J. Park und D. Ajiboye, „Data-driven state of health (SOH) prediction for lithium-ion batteries,” in *Proceedings of the 242nd Electrochemical Society Meeting*, Atlanta, GA, USA, 2022.
- [136] J.-q. Bai und Y.-h. Xian, „Health state prediction of lithium-ion batteries based on XGBoost algorithm,” in *Proceedings of the 4th International Conference on Wireless Communications and Smart Grid*, Art. 012056, Hulun Buir, China, 2022.
- [137] D. Roman, S. Saxena, V. Robu, M. Pecht und D. Flynn, „Machine learning pipeline for battery state-of-health estimation,” *Nature Machine Intelligence*, Vol. 3, S. 447–456, 2021.
- [138] X. Chen, J. Hua, C. Yin, Z. Chen, Y. Chen und Z. Li, „A precise life estimation method for retired energy storage batteries based on energy storage batteries attenuation characteristics and XGBoost algorithm,” *IEEE Access*, Vol. 11, S. 135 968–135 978, 2023.
- [139] N. Naaz und J. Channegowda, „XGBoost based synthetic battery parameter generation to overcome limited battery dataset challenges,” in *Proceedings of the IEEE International Conference on Power Electronics, Smart Grid and Renewable Energy*, Art. 9715814, Thiruvananthapuram, Indien, 2022.
- [140] M. Zhang, W. Chen, J. Yin und T. Feng, „Lithium battery health factor extraction based on improved Douglas–Peucker algorithm and SOH prediction based on XGBoost,” *Energies*, Vol. 15(16), Art. 5981, 2022.

[141] H. Qin, X. Fan, Y. Fan, R. Wang, Q. Shang und D. Zhang, „A computationally efficient approach for the state-of-health estimation of lithium-ion batteries,” *Energies*, Vol. 16(14), Art. 5414, 2023.

[142] C. H. B. Apribowo, S. P. Hadi, F. D. Wijaya, M. I. B. Setyonegoro et al., „Early prediction of battery degradation in grid-scale battery energy storage system using extreme gradient boosting algorithm,” *Results in Engineering*, Vol. 21, Art. 101709, 2024.

[143] Y. Hou, Z. Zhang, P. Liu, C. Song und Z. Wang, „Research on a novel data-driven aging estimation method for battery systems in real-world electric vehicles,” *Advances in Mechanical Engineering*, Vol. 13(7), Art. 16878140211027735, 2021.

[144] M. Grüner, „Alterungsprädiktion von Lithium-Ionen Batterien mittels Methoden des Machine Learning,” Masterarbeit (betreut durch D. Herz), Karlsruher Institut für Technologie, Institut für Fahrzeugtechnik (KIT-FAST), Karlsruhe, Juli 2022.

[145] B. Epding, B. Rumberg, H. Jahnke, I. Stradtmann und A. Kwade, „Investigation of significant capacity recovery effects due to long rest periods during high current cyclic aging tests in automotive lithium ion cells and their influence on lifetime,” *Journal of Energy Storage*, Vol. 22, S. 249–256, 2019.

[146] J. Brownlee. “Discover feature engineering, how to engineer features and how to get good at it.” [Online Ressource] <https://machinelearningmastery.com/discover-feature-engineering-how-to-engineer-features-and-how-to-get-good-at-it/> Zugriff am 21. März 2025.

- [147] S. Saxena, D. Roman, V. Robu, D. Flynn und M. Pecht, „Battery stress factor ranking for accelerated degradation test planning using machine learning,” *Energies*, Vol. 14(3), Art. 723, 2021.
- [148] D. Herz, M. Gießler, S. Kehl und F. Gauterin, „A competitive benchmark of data driven approaches for battery aging prediction,” Präsentation auf dem WKM-Symposium für Bordnetze, CAN-Architektur und KI-Verfahren, Graz, Österreich, 2022.
- [149] F. A. Gers, N. N. Schraudolph und J. Schmidhuber, „Learning precise timing with LSTM recurrent networks,” *Journal of Machine Learning Research*, Vol. 3, S. 115–143, 2002.
- [150] W. Zaremba, I. Sutskever und O. Vinyals, „Recurrent neural network regularization,” *arXiv preprint*, Art. 1409.2329, 2014.
- [151] R. Pascanu, C. Gulcehre, K. Cho und Y. Bengio, „How to construct deep recurrent neural networks,” in *Proceedings of the 2nd International Conference on Learning Representations*, S. 1–8, Banff National Park, Canada, 2014.
- [152] D. Herz, C. Krauß, C. Zimmerling, B. Grupp und F. Gauterin, „Estimation of load-time curves using recurrent neural networks based on can bus signals,” in *Proceedings of the 14th World Congress on Computational Mechanics-WCCM & ECCOMAS Congress*, Art. 1700, Paris, Frankreich (virtuell), 2021.
- [153] L. Villalobos-Arias, C. Quesada-López, J. Guevara-Coto, A. Martínez und M. Jenkins, „Evaluating hyper-parameter tuning using random search in support vector machines for software effort estimation,” in *Proceedings of the 16th ACM International Conference on Predictive Models*

*and Data Analytics in Software Engineering*, S. 31–40, New York City, NY, USA (virtuell), 2020.

[154] X. Ding, J. Liu, F. Yang und J. Cao, „Random radial basis function kernel-based support vector machine,” *Journal of the Franklin Institute*, Vol. 358(18), S. 10 121–10 140, 2021.

[155] O.-E. Ørebæk und M. Geitle, „Exploring the hyperparameters of XG-Boost through 3D visualizations,” in *Proceedings of the AAAI Spring Symposium: Combining Machine Learning with Knowledge Engineering*, Palo Alto, CA, USA (virtuell), 2021.

[156] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. u. Kaiser und I. Polosukhin, „Attention is all you need,” in *Proceedings of 31st Conference on Neural Information Processing Systems*, Long Beach, CA, USA, 2017.

[157] H. Zhou, S. Zhang, J. Peng, S. Zhang, J. Li, H. Xiong und W. Zhang, „Informer: Beyond efficient transformer for long sequence time-series forecasting,” in *Proceedings of the 35th AAAI Conference on Artificial Intelligence*, S. 11 106–11 115, 2021.

[158] M. R. Fauzi, N. Yudistira und W. F. Mahmudy, „State-of-health prediction of lithium-ion batteries using exponential smoothing transformer with seasonal and growth embedding,” *IEEE Access*, Vol. 12, S. 14 659–14 670, 2024.



# Eigene Publikationen

Dieser Abschnitt enthält ein vollständiges Verzeichnis der eigenen Veröffentlichungen. In Publikation [152] wird der Ansatz einer angepassten Zielfunktion für Rekurrente Neuronale Netze entwickelt und auf die vorliegende Problemstellung übertragen. Die methodische Erweiterung ermöglicht eine verbesserte Modellierung nichtlinearer Zusammenhänge, insbesondere bei geringen  $SoH_C$ -Werten. Die Erkenntnisse aus dieser Arbeit fließen direkt in die Entwicklung der hier vorgestellten Modelle ein und tragen zur Optimierung der Prädiktionsgüte bei. Publikation [148] analysiert die Degradation von Batteriesystemen mit Fokus auf dem  $SoH_C$  und vergleicht Methoden zur Erkennung und Bewertung von Ausreißern hinsichtlich Effizienz und Qualität.

[148] D. Herz, M. Gießler, S. Kehl und F. Gauterin, „A Competitive Benchmark of Data-Driven Approaches for Battery Aging Prediction“, Präsentation auf dem WKM-Symposium für Bordnetze, CAN-Architektur und KI-Verfahren, Graz, Österreich, 2022.

Beiträge der Autoren

*Dominik Herz*: Konzept, Methodik, Implementierung, Datenaggregation, Validierung *Martin Gießler*: Fachliche Beratung *Steffen Kehl*: Projektleitung, Konzept, Bereitstellung von Ressourcen *Frank Gauterin*: Betreuung

[152] D. Herz, C. Krauß, C. Zimmerling, B. Grupp und F. Gauterin, „Estimation of load-time curves using recurrent neural networks based on can bus signals“, in *Proceedings of the 14th World Congress on Computational Mechanics-WCCM & ECCOMAS Congress*, Art. 1700, Paris, Frankreich (virtuell), 2021.

Beiträge der Autoren

*Dominik Herz*: Konzept, Methodik, Implementierung, Datenaggregation, Validierung, Schreiben des Manuskripts *Constantin Krauß*: Fachliche Beratung, Korrektur des Manuskripts *Clemens Zimmerling*: Fachliche Beratung, Korrektur des Manuskripts *Benedikt Grupp*: Projektleitung, Konzept, Bereitstellung von Ressourcen *Frank Gauterin*: Betreuung, Bereitstellung von Ressourcen

# Betreute studentische Arbeiten

Dieser Abschnitt enthält eine Übersicht der studentischen Arbeiten, die während meiner Zeit als Doktorand am Institut für Fahrzeugsystemtechnik des Karlsruher Instituts für Technologie (KIT) betreut wurden. Inhalte aus Abschlussarbeiten sind im Fließtext durch entsprechende Verweise gekennzeichnet.

[98] Erik Schimanko „Alterungsprädiktion von Lithium-Ionen-Batterien auf Basis von Bordnetzströmen“, Masterarbeit (betreut durch D. Herz), Karlsruher Institut für Technologie, Institut für Fahrzeugtechnik (KIT-FAST), Karlsruhe, Juni 2022

Beiträge der Autoren

*Erik Schimanko:* Literaturrecherche, Implementierung KNN, Datenaggregation, Schreiben des Manuskripts *Dominik Herz:* Konzept, Methodik, Fachliche Beratung, Korrektur des Manuskripts

[120] Alexandros Siskos „Alterungsmodellierung von Hochvoltbatterien anhand von Diagnosedaten“, Bachelorarbeit (betreut durch D. Herz), Karlsruher Institut für Technologie, Institut für Fahrzeugtechnik (KIT-FAST), Karlsruhe, September 2020

Beiträge der Autoren

*Alexandros Siskos*: Literaturrecherche, Implementierung SVM, Schreiben des Manuskripts *Dominik Herz*: Konzept, Datenaggregation, Fachliche Beratung, Korrektur des Manuskripts

[144] Marco Grüner „Alterungsprädiktion von Lithium-Ionen-Batterien mittels Methoden des Machine Learning“, Masterarbeit (betreut durch D. Herz), Karlsruher Institut für Technologie, Institut für Fahrzeugtechnik (KIT-FAST), Karlsruhe, Juli 2022

Beiträge der Autoren

*Marco Grüner*: Literaturrecherche, Konzept, Implementierung xG-Boost, Datenaggregation, Validierung, Schreiben des Manuskripts *Dominik Herz*: Konzept, Fachliche Beratung, Korrektur des Manuskripts